DDCA
CO-3 TERMINAL QUESTIONS & ANSWERS

Discuss the data transfer and arithmetic logicinstruction sets with examples.

Data Transfer Instructions:

- Move (Transfer) :Transfer word or block from source to destination

  Example: mov A, 09h

  mov AX, BX
- Store:Transfer word from processor to memory

  Example: STR T

  STR [R1], R3
- Load (fetch): Transfer word from memory to processor.

  Example: Load A, [R1]

  Load C

Arithmetic Iinstruction:

- Add: Compute sum of two operands

  Example: add al, 07h

  add ax, bx
- Subtract: Compute difference of two operands

  Example: sub ah, 05h

  sub ah, al
- Multiply: Compute product of two operands

  Example: mov ax,1234h

  mov bx,100h

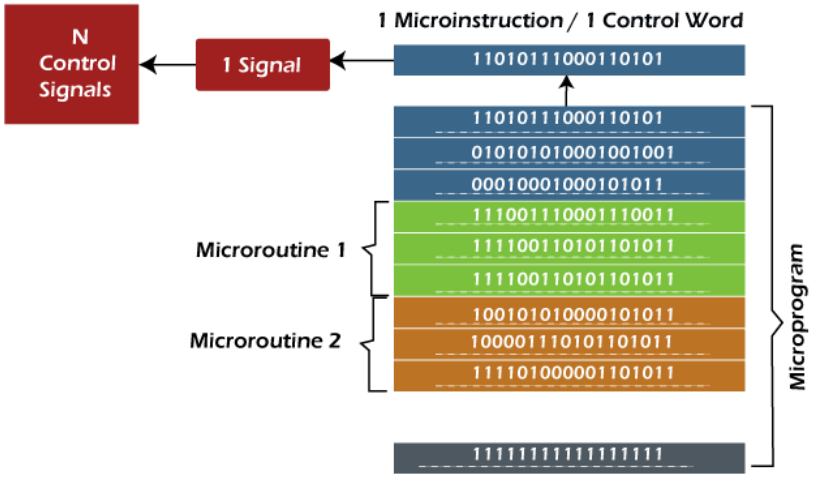
  mul bx
- Divide: Compute quotient of two operands

  Example: mov ax,8003h

  mov cx,100h

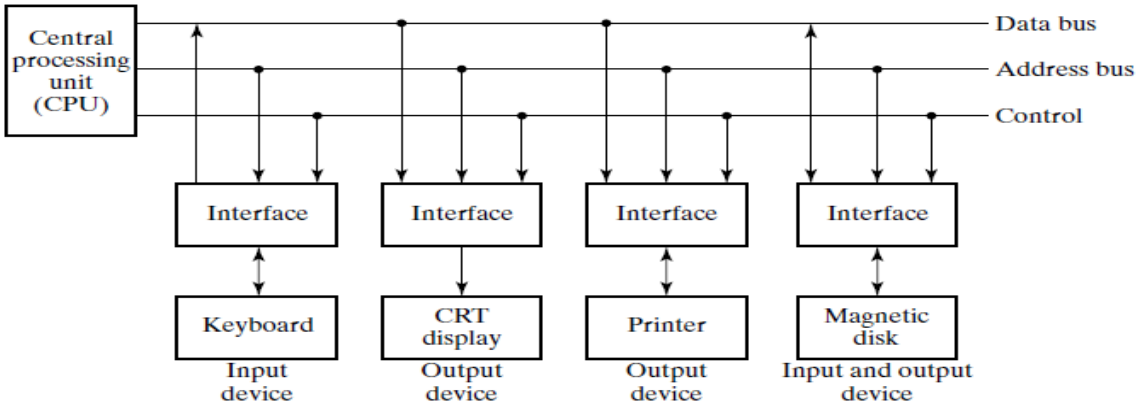
  div cx

Illustrate the micro-programmed realization in CPUdesign, detailing its architecture.



1. Micro-programmed control units operate as basic logic circuits.
2. They execute instructions by generating control signals and sequencing through microinstructions.
3. Microprograms stored in fast memory, termed control store or control memory, dictate the sequence of control signals for each instruction, facilitating efficient execution.

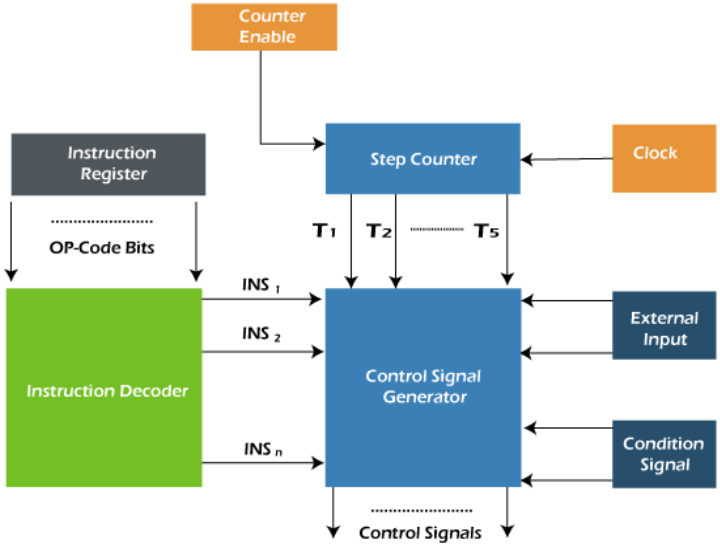
Analyze the role and significance of IO devices incomputer systems with examples.



1. **Keyboard:** Facilitates textual input for tasks such as typing documents or entering commands.

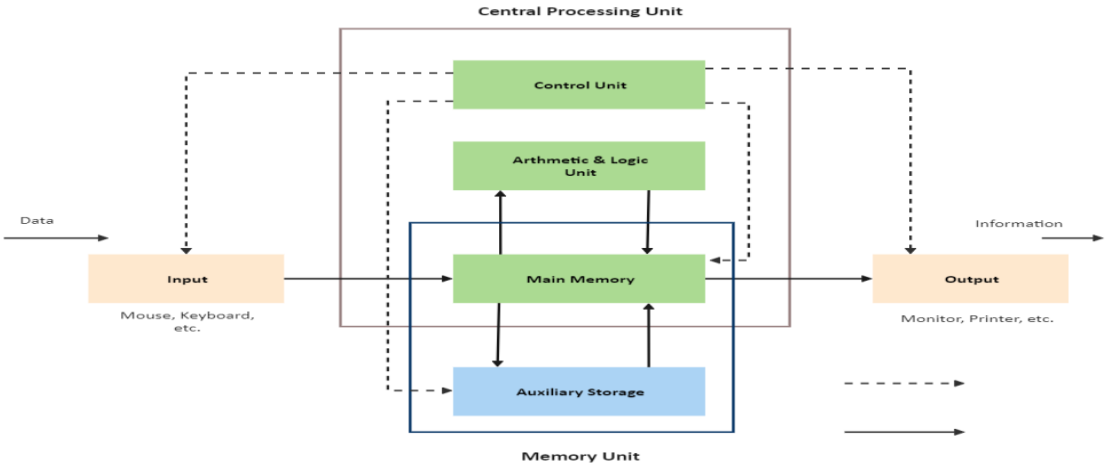
2. **Monitor:** Displays visual output, including text, images, videos, and graphical user interfaces.
3. **Printer:** Produces hard copies of digital documents or images for physical distribution or archival.
4. **Magnetic Disk:** Provides high-capacity, non-volatile storage for operating systems, applications, and user data, allowing quick access to stored information.

Illustrate the hardwired realization in CPU design,detailing its architecture.



1. Hardwired control units execute instructions by generating control signals at the right time and sequence.
2. Faster than micro-programmed units, they use PLA circuit and state counter to generate control signals.
3. Hardware-based, they employ circuitry to produce control signals needed by the CPU for operation.

Illustrate the architecture of a CPU and its constituent blocks, elaborating on their functions.



1. **Control Unit (CU):** Fetches and decodes instructions, generates control signals for CPU operations.
2. **Arithmetic Logic Unit (ALU):** Performs arithmetic and logical operations on data as directed by the control unit.
3. **Memory Unit (MU):** Stores data and instructions, comprising internal registers for fast, temporary storage, and main memory for larger capacity storage.

DDCA
CO-3 TERMINAL QUESTIONS & ANSWERS

Differentiate hardwired realization and micro- programmed realization in the design of control units within a CPU.

Hardwired	Micro-programmed
With the help of a hardware circuit, we can implement the hardwired control unit.	While with the help of programming, we can implement the micro-programmed control unit.
The hardwired control unit uses the logic circuit so that it can generate the control signals, which are required for the processor.	The micro-programmed CU uses microinstruction so that it can generate the control signals. Usually, control memory is used to store these microinstructions.
In the form of logic gates, everything has to be realized in the hardwired control unit. That's why this CU is more costly as compared to the micro-programmed control unit.	The micro-programmed control unit is less costly as compared to the hardwired CU because this control unit only requires the microinstruction to generate the control signals.

Examine the concepts of register, register indirect, displacement addressing modes in computer architecture.

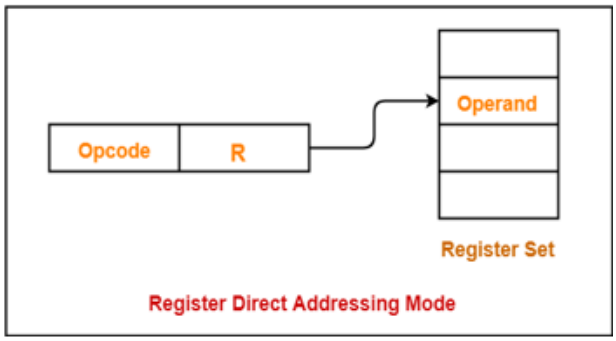
1. **Register Addressing mode:**

   Eg: MOV AX, BX

   ADD R1, R2

   XOR AX, DX

   MUL AL, BL

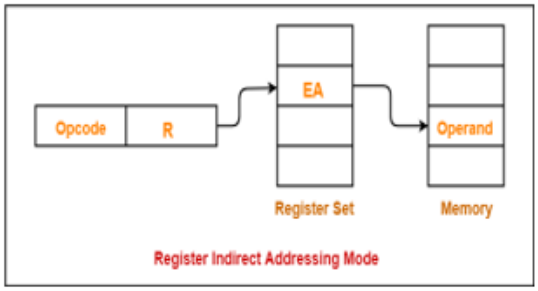


2. **Register Indirect addressing mode:**

   Eg: ADD AL, [BX]

   MOV AX, [BX]

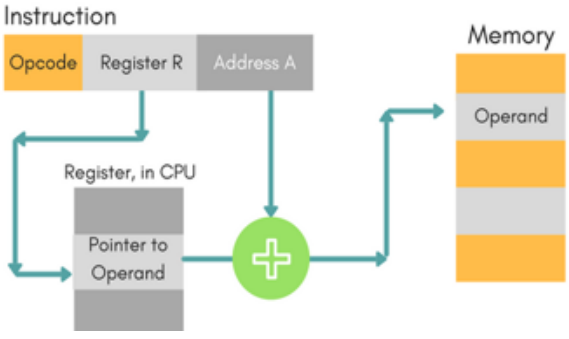
   MOV AX, [DI]



3. **Displacement addressing mode:**

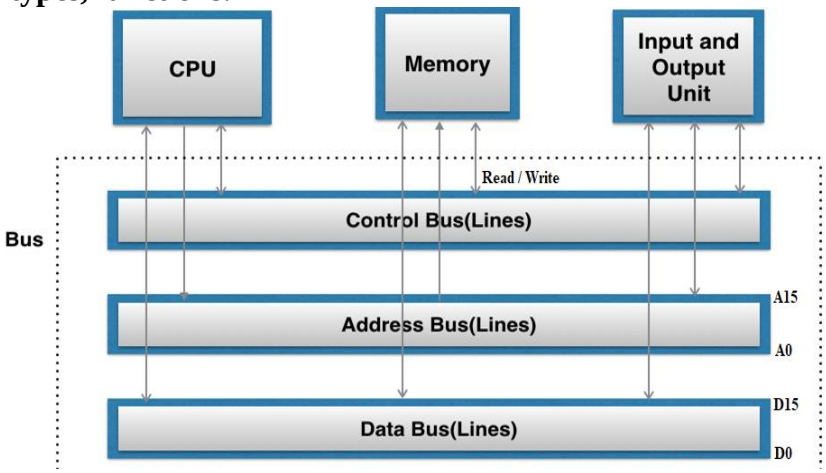
   Eg: MOV AX, [SI+DISP]

   MOV AX, [BP+0500]



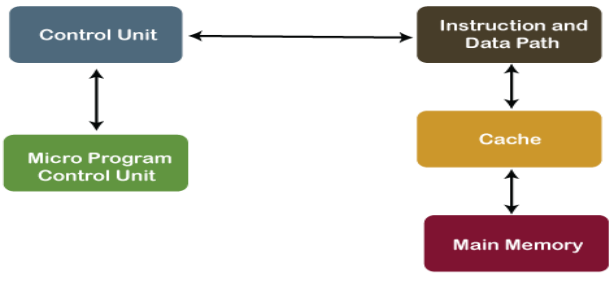
DDCA
CO-3 TERMINAL QUESTIONS & ANSWERS

Make use of various buses in microcomputer architecture to investigate on data transfer, detailingtheir types, functions.



- **Data Bus:** A bidirectional pathway that carries data between the CPU, memory, and other devices. It transfers binary information in parallel, typically in groups of 8, 16, 32, or 64 bits, depending on the architecture.
- **Address Bus:** A unidirectional pathway used by the CPU to specify memory addresses when accessing data or instructions stored in memory. It carries binary addresses that identify the location of data or instructions in the memory hierarchy.
- **Control Bus:** A bidirectional or unidirectional pathway that carries control signals between the CPU, memory, and I/O devices. It includes various signals such as read, write, clock, interrupt, and bus control signals.

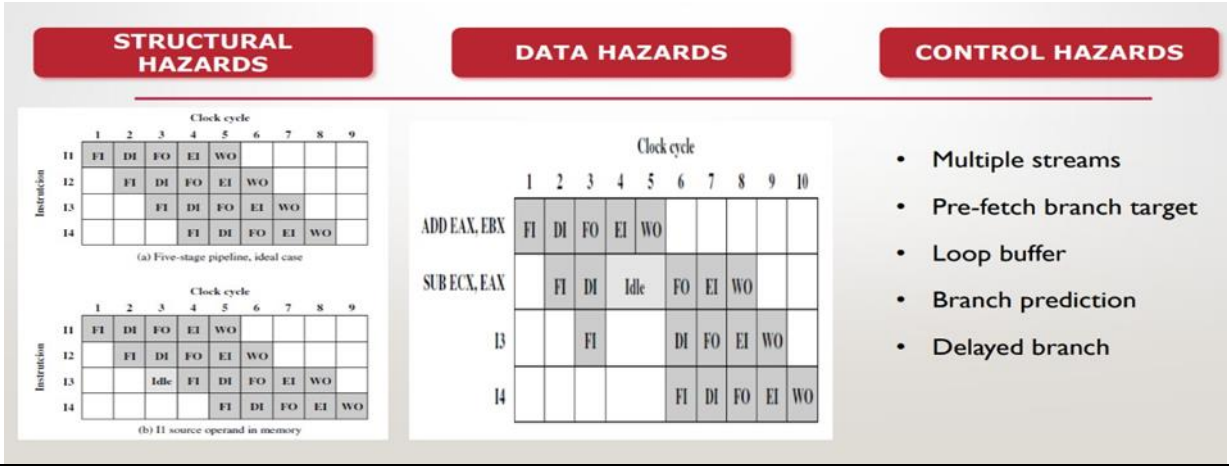
Develop the model of CISC architecture, detailing itsdesign philosophy, key features.



CISC Architecture

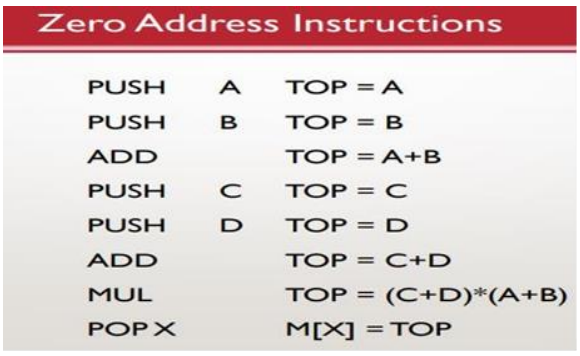
- CISC architectures feature a large set of complex instructions.
- Designed to accomplish tasks in fewer lines of code, with instructions capable of performing multiple operations.
- Aims to reduce the software complexity by providing instructions that can execute high-level tasks directly.

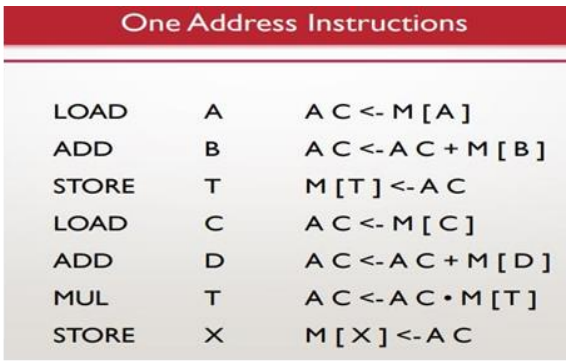
Identify the various pipelining hazards in processordesign, detailing the types of hazards, their causes.

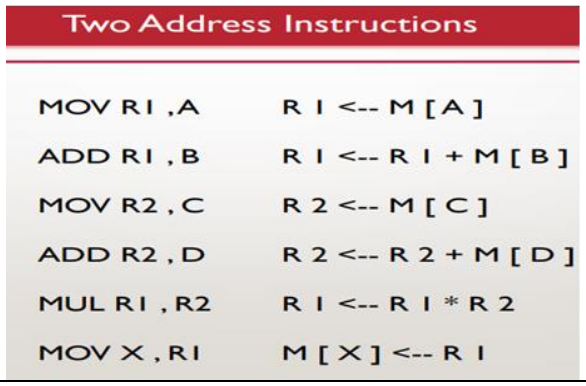


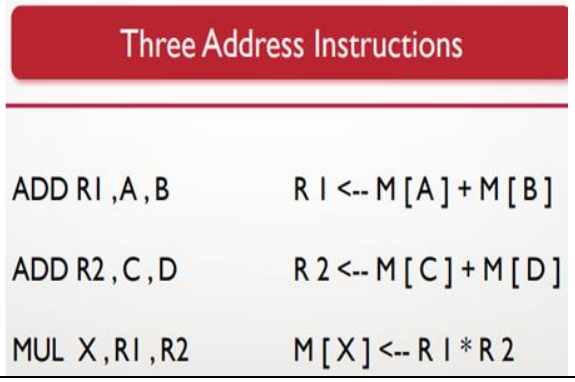
DDCA
CO-3 TERMINAL QUESTIONS & ANSWERS

Model the structures of different instruction formatsand provide illustrative examples for each.









Consider a scenario where you are developing a simple combinational circuit of basic logical operations. Make use of the related instruction setsand accomplish the task.

Let us consider the expression $F = A.B + C'.D$ to develop using the Logical instruction set.

Load input values

LOAD R0, A ; Load input A into register R0

LOAD R1, B ; Load input B into register R1

LOAD R2, C ; Load input C into register R2

LOAD R3, D ; Load input D into register R3

Perform logical operations

AND R5, R0, R1 ; Compute A AND B and store the result in R5

NOT R6, R2 ; Compute the complement of C and store it in R6

AND R7, R6, R3 ; Compute (NOT C) AND D and store the result in R7

OR R4, R5, R7 ; Compute (A AND B) OR ((NOT C) AND D) and store the result in R4

In order to determine the city with the most consistently warm temperatures from a large weatherdataset, which parallel processing architecture (SISD,SIMD, MISD, MIMD) would be best suited for efficient analysis and why?

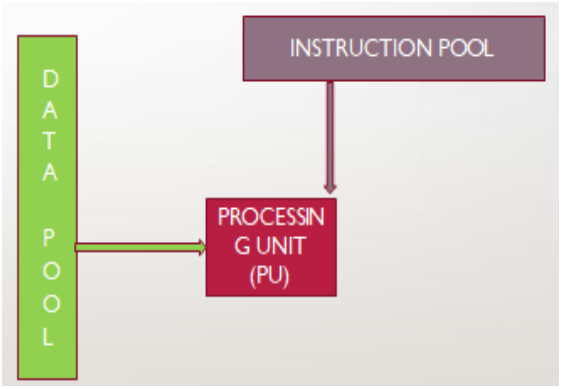
For determine the city with the most consistently warm temperatures from a large weather dataset, most appropriate architecture would be MIMD (Multiple Instruction, Multiple Data) due to following reasons.

- ✓ Independent Processing

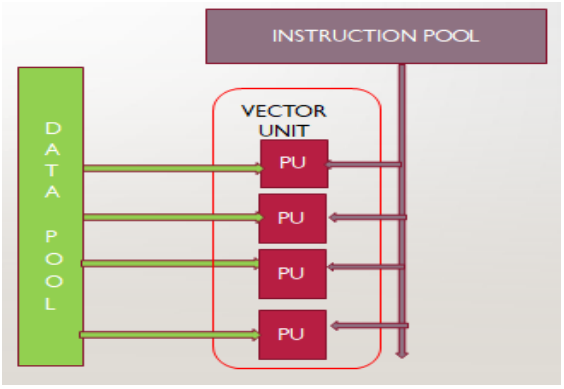
DDCA
CO-3 TERMINAL QUESTIONS & ANSWERS

- ✓ Flexibility and Scalability
- ✓ Complexity of Analysis
- ✓ Data Distribution and Communication

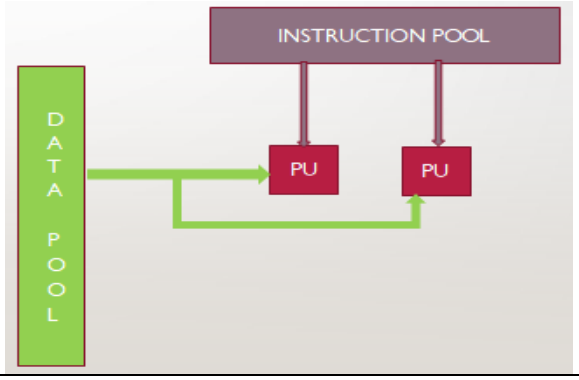
SISD



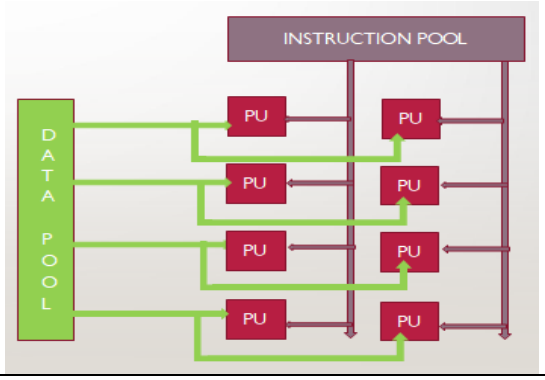
SIMD



MISD



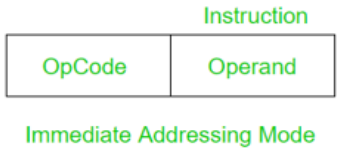
MIMD



Examine the concepts of immediate, direct, and indirect addressing modes in computer architecture.

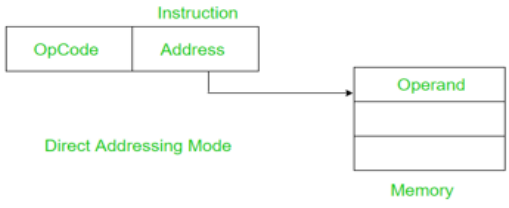
1. **Immediate Addressing mode:**

   Eg: MOV AX, 2000H
   ADD AL, 45H



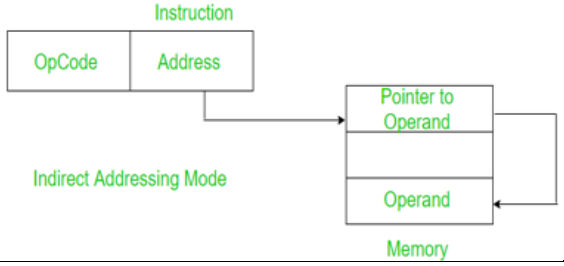
2. **Direct Addressing mode:**

   Eg: MOV AX, [1592H]
   MOV BL, [03H]



3. **Indirect Addressing mode:**

   Eg: MOV AX, @2005H
   LOAD R1, (1345H)



DDCA
CO-3 TERMINAL QUESTIONS & ANSWERS

Consider a scenario where you are developing a simple calculator application for a mobile device. You want to implement basic arithmetic operations. Make use of the related instruction sets and accomplish the task.

- **Add: Compute sum of two operands**

  Example: add al, 07h

  add ax, bx
- **Subtract: Compute difference of two operands**

  Example: sub ah, 05h

  sub ah, al
- **Multiply: Compute product of two operands**

  Example: mov ax, 1234h

  mov bx, 100h

  mul bx
- **Divide: Compute quotient of two operands**

  Example: mov ax,8003h

  mov cx,100h

  div cx
- **Negate: Change sign of operand**

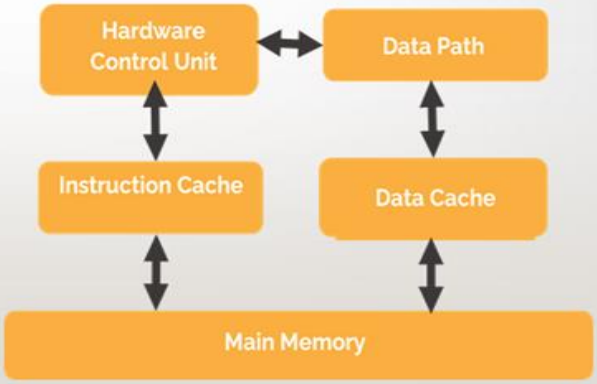
  Example: neg RT, RA
- **Increment : Add 1 to operand**

  Example: inc A
- **Decrement: Subtract 1 from operand**

  Example: dec A

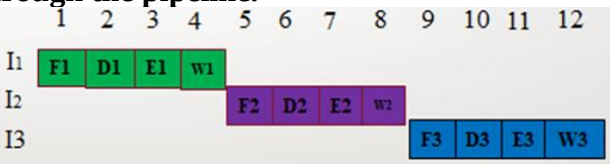
Develop the model of RISC architecture, detailing its design philosophy, key features.

- RISC architectures are designed around a small set of simple instructions.
- The goal is to execute instructions at a high speed, with most instructions completing in one clock cycle.
- Emphasizes simplicity and efficiency at the hardware level, allowing for faster instruction execution through pipelining.



Design the operational flow for a pipelined processor with four stages, outlining each stage's function and how instructions progress through the pipeline.

Conventional Sequential Execution



Pipelined Execution

F: Fetch, Read the instruction from the memory

D: Decode, decode the instruction and fetch the source operand (S)

E: Operate, perform the operation

W: Write, store the result in the destination location.

Step	1	2	3	4	5	6	7	8	9
1	F1	D1	E1	W1					
2		F2	D2	E2	W2				
3			F3	D3	E3	W3			
4				F4	D4	E4	W4		
5					F5	D5	E5	W5	
6						F6	D6	E6	W6