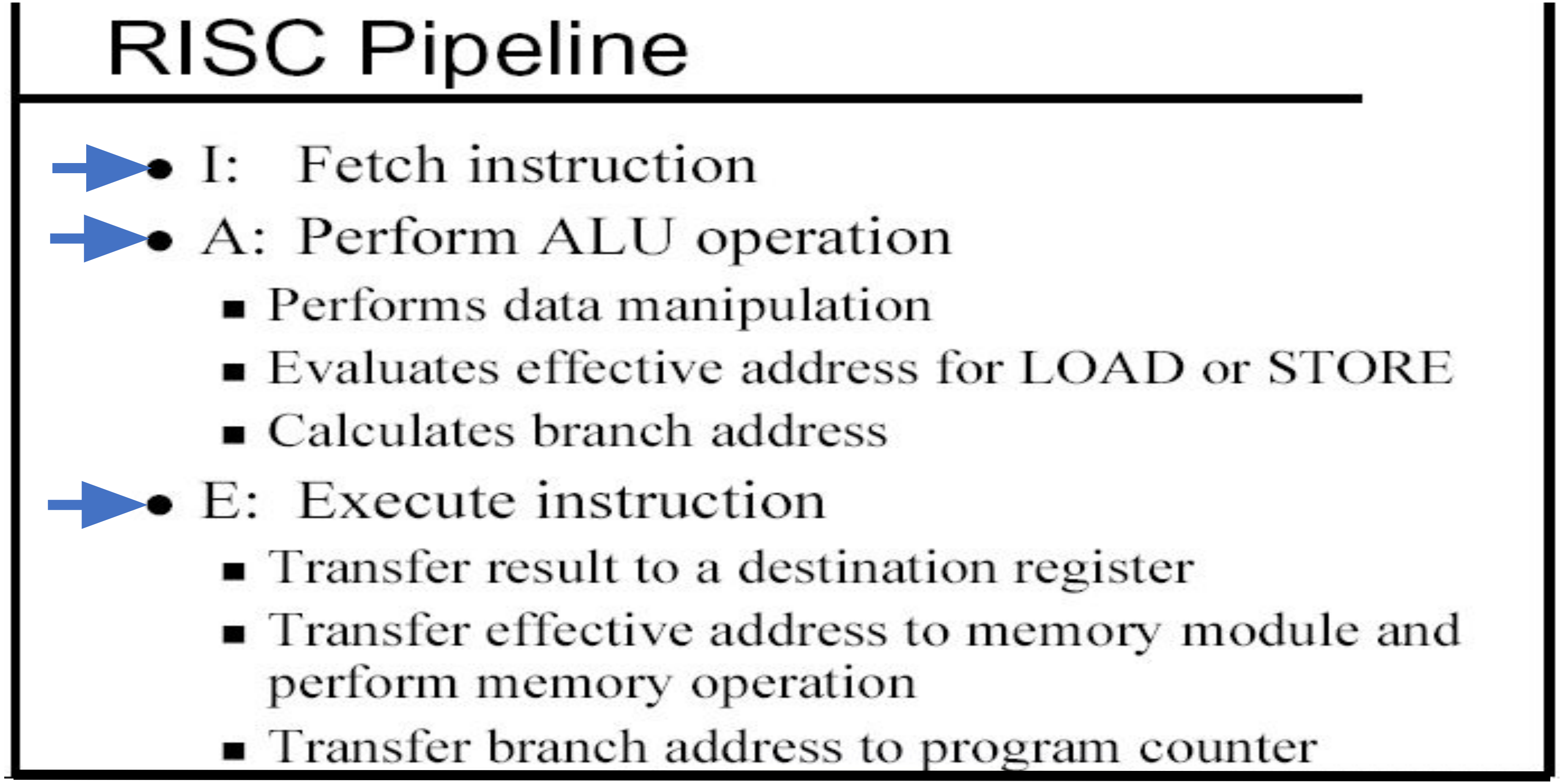
Pipeline Applications

- Pipeline organization is applicable to
  - Arithmetic pipeline
    - It divides an arithmetic operation into suboperations for execution in the pipeline segments
  - Instruction pipeline
    - It operates on a stream of instructions by overlapping the fetch, decode, and execute phases of the instruction cycle



RISC Pipeline

- I: Fetch instruction
- A: Perform ALU operation
  - ▪ Performs data manipulation
  - ▪ Evaluates effective address for LOAD or STORE
  - ▪ Calculates branch address
- E: Execute instruction
  - ▪ Transfer result to a destination register
  - ▪ Transfer effective address to memory module and perform memory operation
  - ▪ Transfer branch address to program counter

▯ The instruction cycle can be divided into three suboperations and implemented in three segments.

♦I: Instruction fetch

♦A: ALU Operations

♦E: Execute instruction

RISC PIPELINE

RISC

- Machine with a very fast clock cycle that executes at the rate of one instruction per cycle

<- Simple Instruction Set
Fixed Length Instruction Format
Register-to-Register Operations

Instruction Cycles of Three-Stage Instruction Pipeline

Data Manipulation Instructions

I: Instruction Fetch
A: Decode, Read Registers, ALU Operations
E: Write a Register

Load and Store Instructions

I: Instruction Fetch
A: Decode, Evaluate Effective Address
E: Register-to-Memory or Memory-to-Register

Program Control Instructions

I: Instruction Fetch
A: Decode, Evaluate Branch Address
E: Write Register(PC)

RISC

- **Reduced Set Instruction Set Architecture (RISC) –** The main idea behind is to make hardware simpler by using an instruction set composed of a few basic steps for loading, evaluating and storing operations just like a load command will load data, store command will store the data.
- **Characteristic of RISC –**

1. Simpler instruction, hence simple instruction decoding.
2. Instruction come under size of one word.
3. Instruction take single clock cycle to get executed.
4. More number of general purpose register.
5. Simple Addressing Modes.
6. Less Data types.
7. Pipeling can be achieved.

CISC

- **Complex Instruction Set Architecture (CISC) –** The main idea is that a single instruction will do all loading, evaluating and storing operations just like a multiplication command will do stuff like loading data, evaluating and storing it, hence it’s complex.
- **Characteristic of CISC –**

1. Complex instruction, hence complex instruction decoding.
2. Instruction are larger than one word size.
3. Instruction may take more than single clock cycle to get executed.
4. Less number of general purpose register as operation get performed in memory itself.
5. Complex Addressing Modes.
6. More Data types.

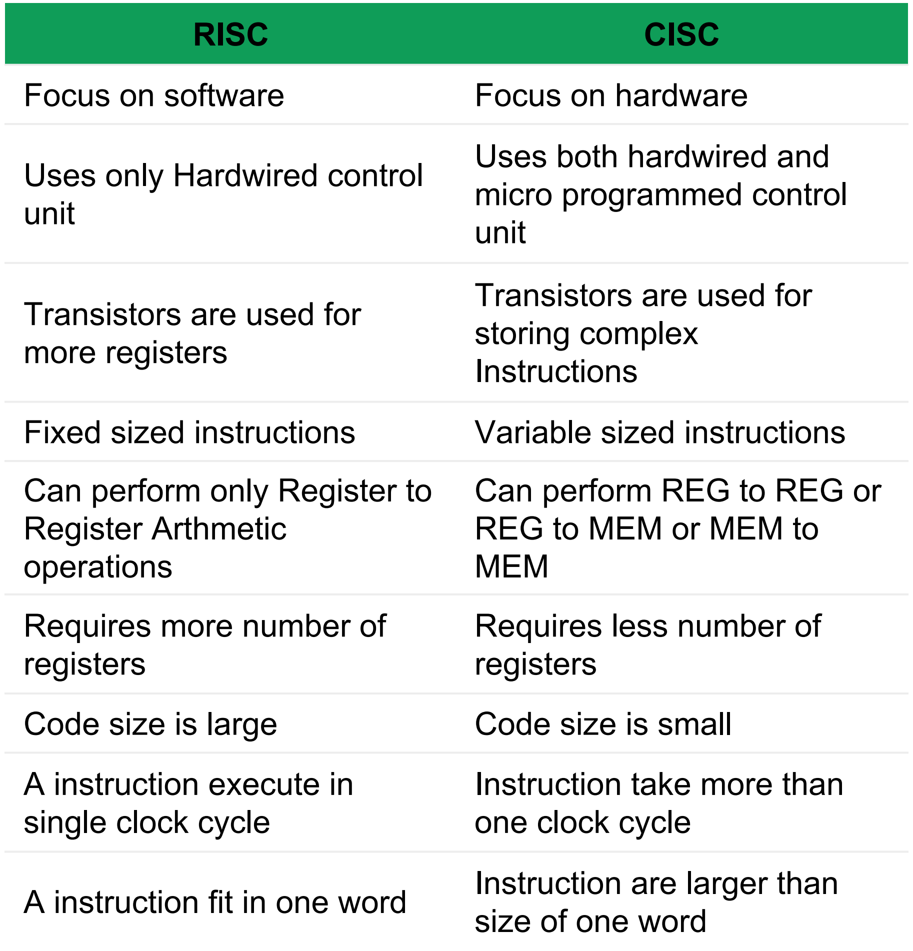
- Both approaches try to increase the CPU performance
- **RISC:** Reduce the cycles per instruction at the cost of the number of instructions per program.
- **CISC:** The CISC approach attempts to minimize the number of instructions per program but at the cost of increase in number of cycles per instruction.

$$CPU\ Time = \frac{Seconds}{Program} = \frac{Instructions}{Program} X \frac{Cycles}{Instructions} X \frac{Seconds}{Cycle}$$

EXAMPLE

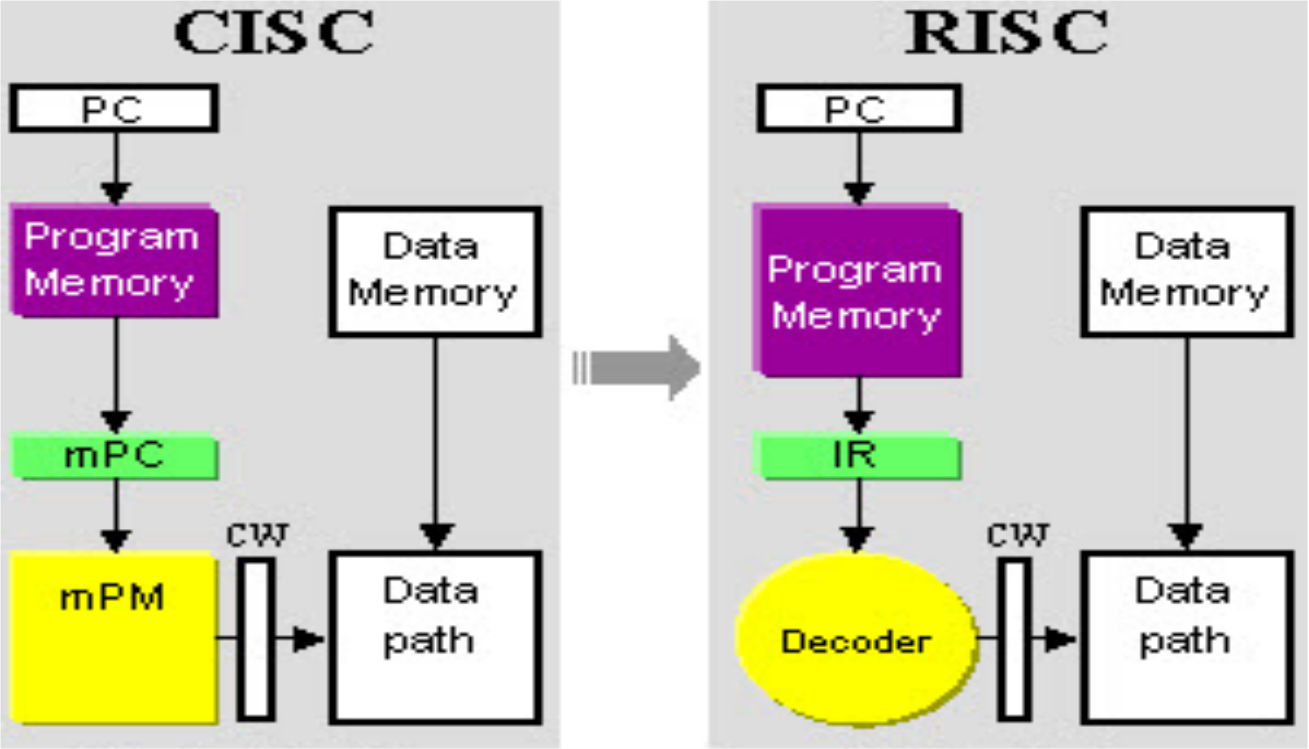
- **Example –** Suppose we have to add two 8-bit number:
- **CISC approach:** There will be a single command or instruction for this like ADD which will perform the task.
- **RISC approach:** Here programmer will write first load command to load data in registers then it will use suitable operator and then it will store result in desired location.

DIFFERENCE BETWEEN RISC AND CISC



RISC	CISC
Focus on software	Focus on hardware
Uses only Hardwired control unit	Uses both hardwired and micro programmed control unit
Transistors are used for more registers	Transistors are used for storing complex Instructions
Fixed sized instructions	Variable sized instructions
Can perform only Register to Register Arthmetic operations	Can perform REG to REG or REG to MEM or MEM to MEM
Requires more number of registers	Requires less number of registers
Code size is large	Code size is small
A instruction execute in single clock cycle	Instruction take more than one clock cycle
A instruction fit in one word	Instruction are larger than size of one word

CONTD..



Complex instructions
One instruction = several CW
PM standard

Simple instructions
One instruction = One CW
PM longer