RISC and CISC:

**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.

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.

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.

**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.

**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.

**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 –**

| **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 Arithmetic 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 |