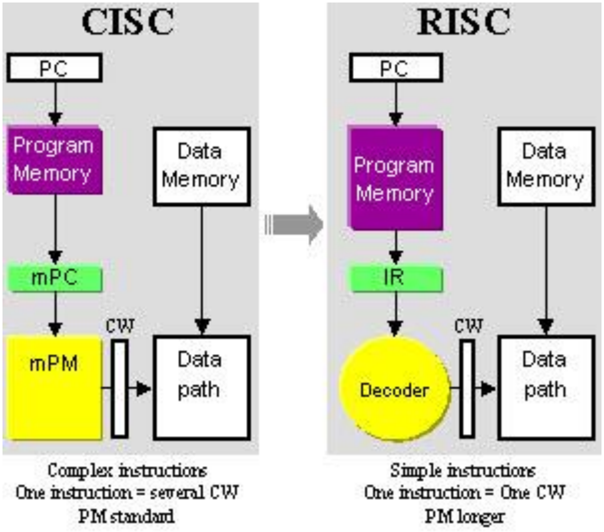
Instruction set architecture is a part of processor architecture, which is necessary for creating machine level programs to perform any mathematical or logical operations. Instruction set architecture acts as an interface between hardware and software. It prepares the processor to respond to the commands like execution, deleting etc given by the user.

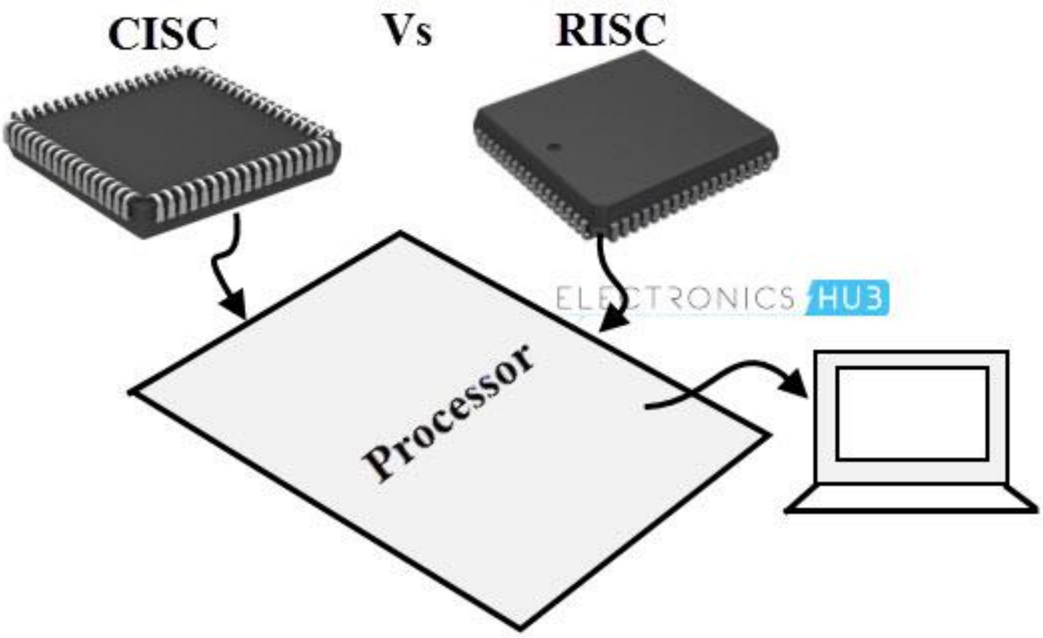
The performance of the processor is defined by the instruction set architecture designed in it. As both software and hardware are required for functioning of a processor, there is dilemma in deciding which should play a major role. Major firms like Intel argues that hardware should play a major role than software. While, Apple's argument is that software should play a major role in processors architecture.

The two major instruction sets architectures are

1) CISC (Complex Instruction Set Computing)

2) RISC (Reduced Instruction Set Computing)





What is RISC and CISC Architectures

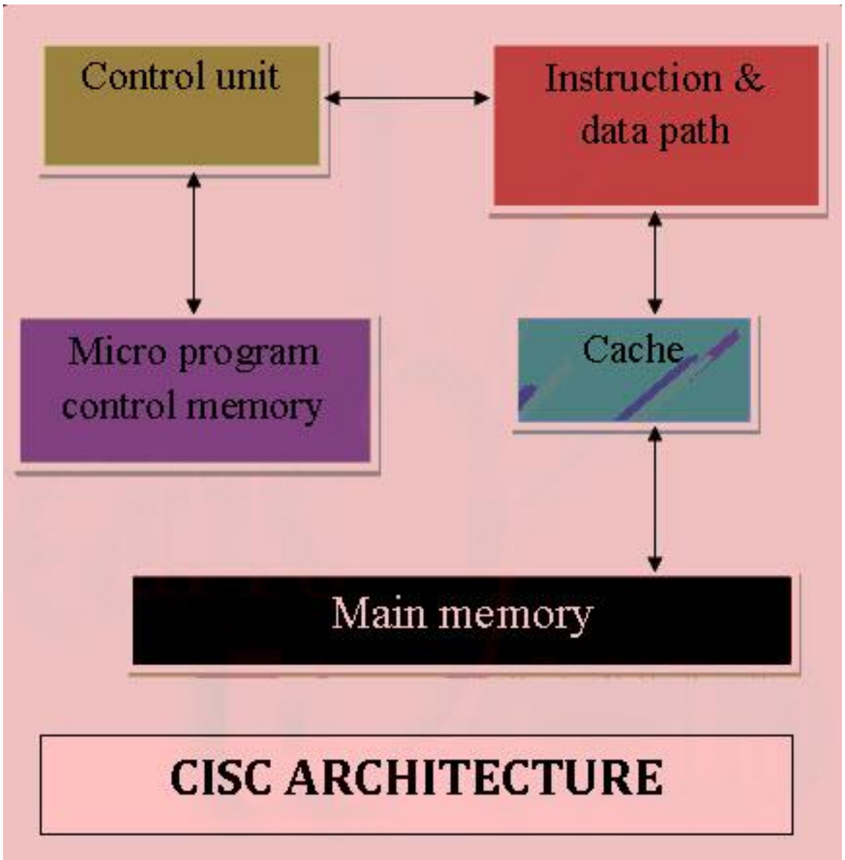
Hardware designers invent numerous technologies & tools to implement the desired architecture in order to fulfill these needs. Hardware architecture may be implemented to be either hardware specific or software specific, but according to the application both are used in the required quantity. As far as the processor hardware is concerned, there are 2 types of concepts to implement the processor hardware architecture. First one is RISC and other is CISC.

CISC ARCHITECHTURE

In the early days machines were programmed in assembly language and the memory access is also slow. To calculate complex arithmetic operations, compilers have to create long sequence of machine code.

This made the designers to build an architecture , which access memory less frequently and reduce burden to compiler. Thus this lead to very power full but complex instruction set.

The CISC approach attempts to minimize the number of instructions per program, sacrificing the number of cycles per instruction. Computers based on the CISC architecture are designed to decrease the memory cost. Because, the large programs need more storage, thus increasing the memory cost and large memory becomes more expensive. To solve these problems, the number of instructions per program can be reduced by embedding the number of operations in a single instruction, thereby making the instructions more complex.



CISC Architecture

Instruction Set: Group of instructions given to execute the program and they direct the computer by manipulating the data. Instructions are in the form – Opcode (operational code) and Operand. Where, opcode is the instruction applied to load and store data, etc. The operand is a memory register where instruction applied.

Addressing Modes: Addressing modes are the manner in the data is accessed. Depending upon the type of instruction applied, addressing modes are of various types such as direct mode where straight data is accessed or indirect mode where the location of the data is accessed. Processors having identical ISA may be very different in organization. Processors with identical ISA and nearly identical organization is still not nearly identical.

CPU performance is given by the fundamental law

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

Examples of CISC PROCESSORS

IBM 370/168 – It was introduced in the year 1970. CISC design is a 32 bit processor and four 64-bit floating point registers.

VAX 11/780 – CISC design is a 32-bit processor and it supports many numbers of addressing modes and machine instructions which is from Digital Equipment Corporation.

Intel 80486 – It was launched in the year 1989 and it is a CISC processor, which has instructions varying lengths from 1 to 11 and it will have 235 instructions.

Features of CISC Architecture

- To simplify the computer architecture, CISC supports microprogramming.
- CISC have more number of predefined instructions which makes high level languages easy to design and implement.
- CISC consists of less number of registers and more number of addressing modes, generally 5 to 20.
- CISC processor takes varying cycle time for execution of instructions – multi-clock cycles.
- Because of the complex instruction set of the CISC, the pipelining technique is very difficult.
- CISC consists of more number of instructions, generally from 100 to 250.
- Special instructions are used very rarely.
- Operands in memory are manipulated by instructions.

Advantages of CISC architecture

- Each machine language instruction is grouped into a microcode instruction and executed accordingly, and then are stored inbuilt in the memory of the main processor, termed as microcode implementation.
- As the microcode memory is faster than the main memory, the microcode instruction set can be implemented without considerable speed reduction over hard wired implementation.
- Entire new instruction set can be handled by modifying the micro program design.
- CISC, the number of instructions required to implement a program can be reduced by building rich instruction sets and can also be made to use slow main memory more efficiently.
- Because of the superset of instructions that consists of all earlier instructions, this makes micro coding easy.

Drawbacks of CISC

- The amount of clock time taken by different instructions will be different – due to this – the performance of the machine slows down.

- The instruction set complexity and the chip hardware increases as every new version of the processor consists of a subset of earlier generations.
- Only 20% of the existing instructions are used in a typical programming event, even though there are many specialized instructions in existence which are not even used frequently.
- The conditional codes are set by the CISC instructions as a side effect of each instruction which takes time for this setting – and, as the subsequent instruction changes the condition code bits – so, the compiler has to examine the condition code bits before this happens.

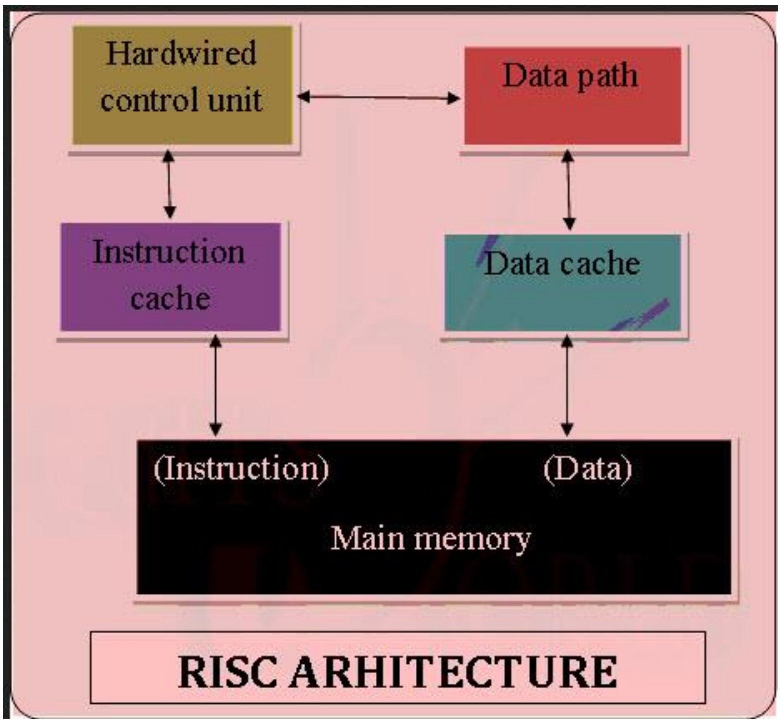
RISC ARCHITECHTURE

Although CISC reduces usage of memory and compiler, it requires more complex hardware to implement the complex instructions.

In RISC architecture, the instruction set of processor is simplified to reduce the execution time. It uses small and highly optimized set of instructions which are generally register to register operations.

The speed of the execution is increased by using smaller number of instructions .This uses pipeline technique for execution of any instruction.

RISC (Reduced Instruction Set Computer) is used in portable devices due to its power efficiency. For Example, Apple iPod and Nintendo DS. RISC is a type of microprocessor architecture that uses highly-optimized set of instructions. RISC does the opposite, reducing the cycles per instruction at the cost of the number of instructions per program Pipelining is one of the unique feature of RISC. It is performed by overlapping the execution of several instructions in a pipeline fashion. It has a high performance advantage over CISC.



RISC Architecture

RISC processors take simple instructions and are executed within a clock cycle

RISC ARCHITECTURE CHARACTERISTICS

- Simple Instructions are used in RISC architecture.
- RISC helps and supports few simple data types and synthesize complex data types.
- RISC utilizes simple addressing modes and fixed length instructions for pipelining.
- RISC permits any register to use in any context.
- One Cycle Execution Time
- The amount of work that a computer can perform is reduced by separating “LOAD” and “STORE” instructions.
- RISC contains Large Number of Registers in order to prevent various number of interactions with memory.
- In RISC, Pipelining is easy as the execution of all instructions will be done in a uniform interval of time i.e. one click.
- In RISC, more RAM is required to store assembly level instructions.
- Reduced instructions need a less number of transistors in RISC.
- RISC uses Harvard memory model means it is Harvard Architecture.
- A compiler is used to perform the conversion operation means to convert a high-level language statement into the code of its form.

Advantages of RISC processor architecture

- Because of the small set of instructions of RISC, high-level language compilers can produce more efficient code.
- RISC allows freedom of using the space on microprocessors because of its simplicity.
- Instead of using Stack, many RISC processors use the registers for passing arguments and holding the local variables.

- RISC functions uses only a few parameters, and the RISC processors cannot use the call instructions, and therefore, use a fixed length instructions which are easy to pipeline.
- The speed of the operation can be maximized and the execution time can be minimized.
- Very less number of instruction formats (less than four), a few number of instructions (around 150) and a few addressing modes (less than four) are needed.

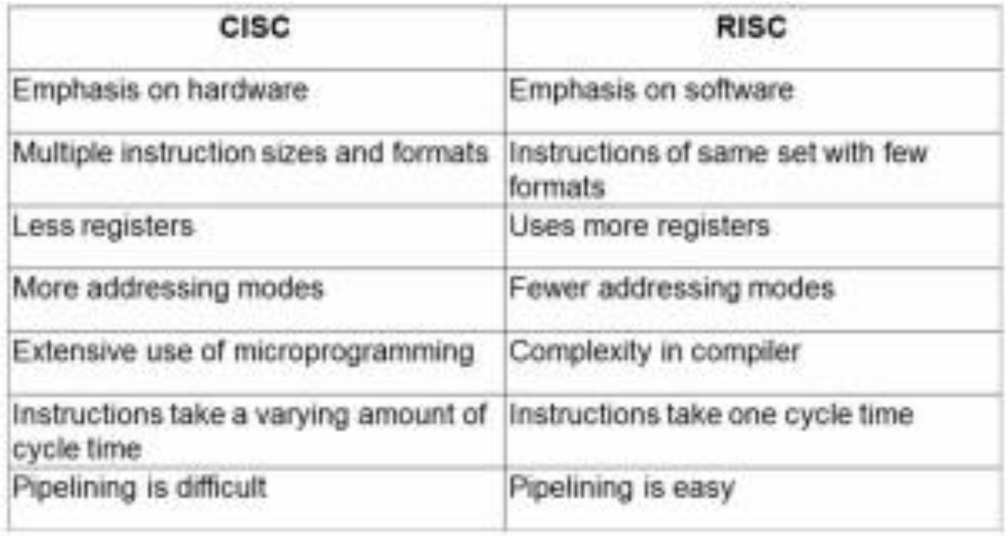
Drawbacks of RISC processor architecture

- With the increase in length of the instructions, the complexity increases for the RISC processors to execute due to its character cycle per instruction.
- The performance of the RISC processors depends mostly on the compiler or programmer as the knowledge of the compiler plays a major role while converting the CISC code to a RISC code; hence, the quality of the generated code depends on the compiler.
- While rescheduling the CISC code to a RISC code, termed as a code expansion, will increase the size. And, the quality of this code expansion will again depend on the compiler, and also on the machine's instruction set.
- The first level cache of the RISC processors is also a disadvantage of the RISC, in which these processors have large memory caches on the chip itself. For feeding the instructions, they require very fast memory systems.

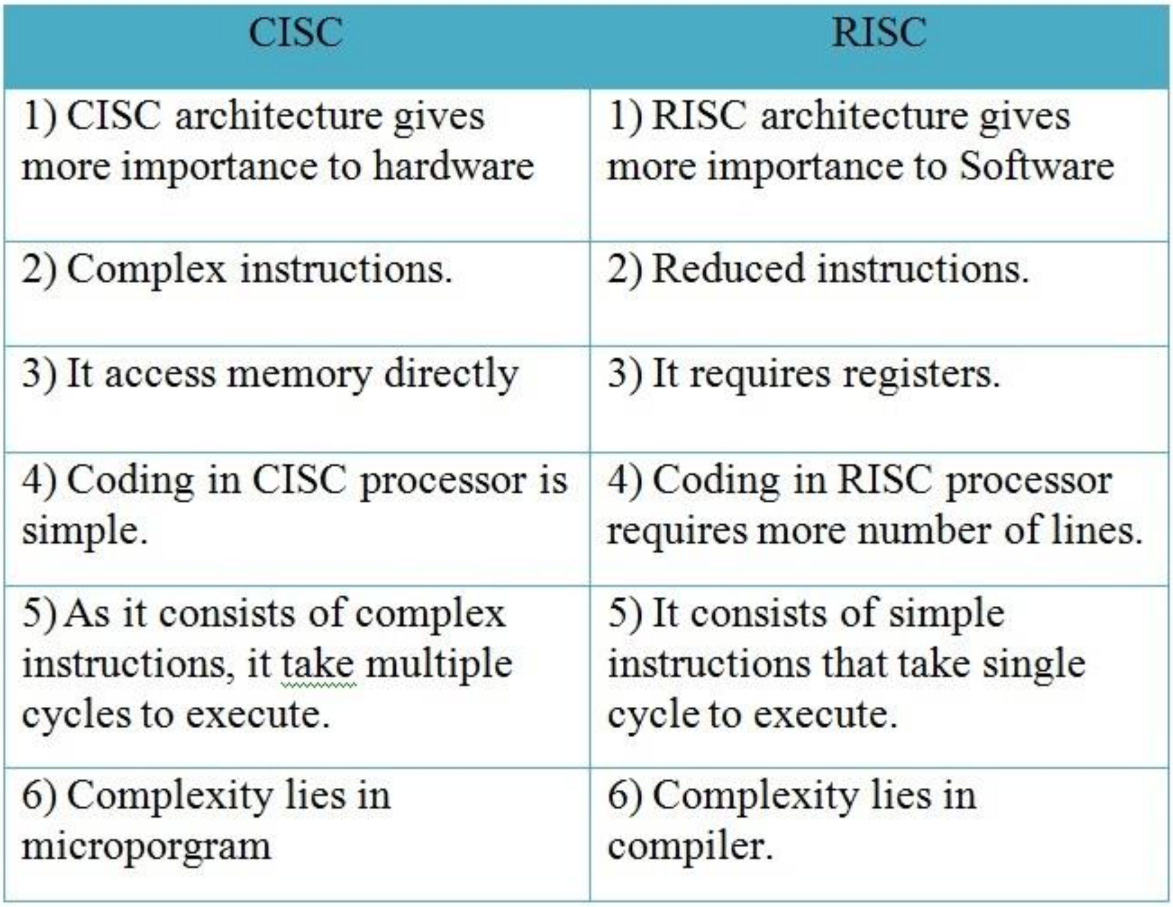
RISC vs. CISC

- The wasting cycles can be prevented by the programmer by removing the unnecessary code in the RISC, but, while using the CISC code leads to wasting cycles because of the inefficiency of the CISC.
- In RISC, each instruction is intended to perform a small task such that, to perform a complex task, multiple small instruction are used together, whereas only few instructions are required to do the same task using CISC – as it is capable of performing complex task as the instructions are similar to a high-language code.
- CISC is typically used for computers while RISC is used for smart phones, tablets and other electronic devices.

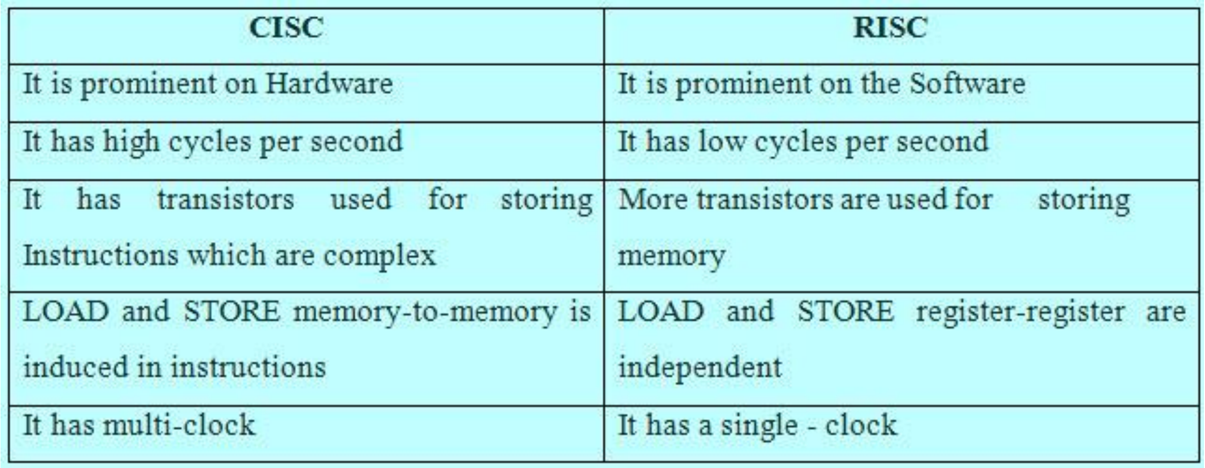
The following figure shows more differences between RISC and CISC



CISC	RISC
Emphasis on hardware	Emphasis on software
Multiple instruction sizes and formats	Instructions of same set with few formats
Less registers	Uses more registers
More addressing modes	Fewer addressing modes
Extensive use of microprogramming	Complexity in compiler
Instructions take a varying amount of cycle time	Instructions take one cycle time
Pipelining is difficult	Pipelining is easy

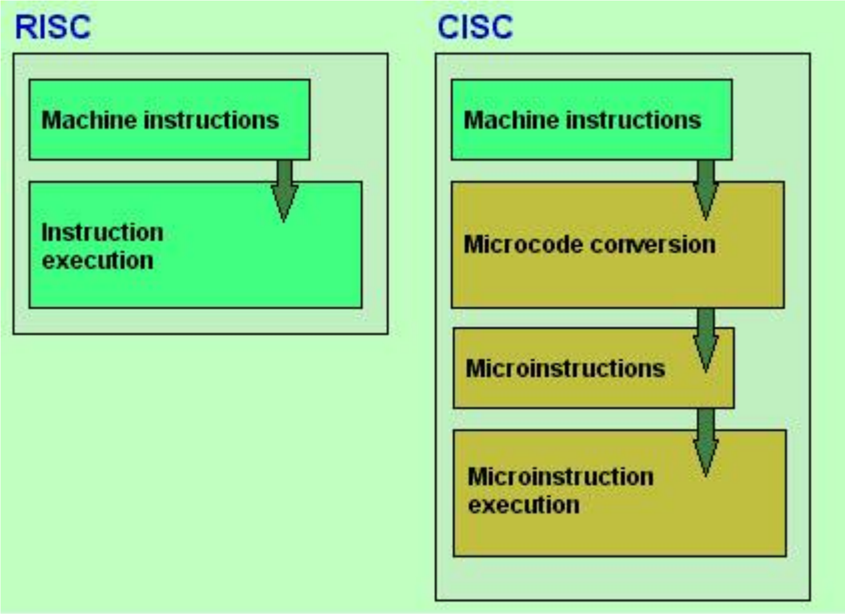


CISC	RISC
1) CISC architecture gives more importance to hardware	1) RISC architecture gives more importance to Software
2) Complex instructions.	2) Reduced instructions.
3) It access memory directly	3) It requires registers.
4) Coding in CISC processor is simple.	4) Coding in RISC processor requires more number of lines.
5) As it consists of complex instructions, it take multiple cycles to execute.	5) It consists of simple instructions that take single cycle to execute.
6) Complexity lies in microporgram	6) Complexity lies in compiler.



CISC	RISC
It is prominent on Hardware	It is prominent on the Software
It has high cycles per second	It has low cycles per second
It has transistors used for storing Instructions which are complex	More transistors are used for storing memory
LOAD and STORE memory-to-memory is induced in instructions	LOAD and STORE register-register are independent
It has multi-clock	It has a single - clock

LAST DIAGRAM


RISC
Machine instructions
Instruction execution
CISC
Machine instructions
Microcode conversion
Microinstructions
Microinstruction execution