**NASIR FIRDAUS**

**17/sci01/051**

**CSC 310**

**RISC**, or Reduced Instruction Set **Computer**. is a type of microprocessor **architecture** that utilizes a small, highly-optimized set of instructions, rather than a more specialized set of instructions often found in other types.

However Due to this **characteristic**, we achieve high performance, low code size, low power consumption and low

RISC (Reduced Instruction Set Computer) is used in portable devices due to its power efficiency... 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 CHARACTERISTICS**

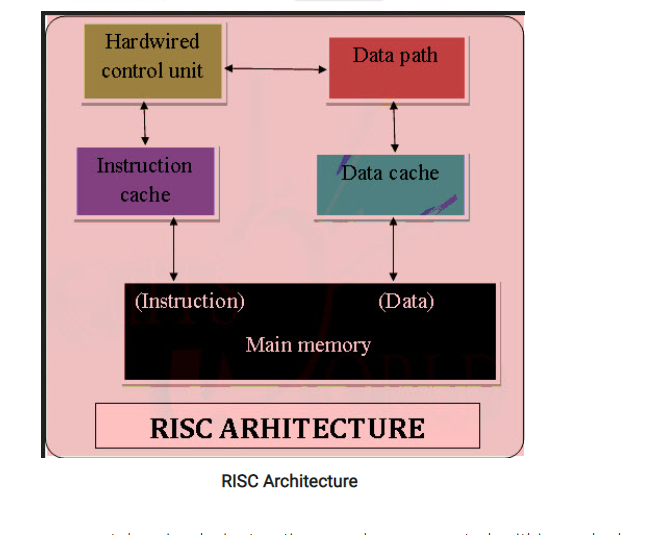
* Simple Instructions are used in RISC architecture.
* RISC helps and supports few simple data types and synthesize complex data types.
* RISC utilizes simple addressing modes and fixed length instructions for pipelining.
* RISC permits any register to use in any context.
* One Cycle Execution Time
* The amount of work that a computer can perform is reduced by separating “LOAD” and “STORE” instructions

**The Advantages of RISC architecture**

* Reduced Instruction Set Computer has a set of instructions, so high-level language compilers can produce more efficient code
* It allows freedom of using the space on microprocessors because of its simplicity.
* Many RISC processors use the registers for passing arguments and holding the local variables.
* RISC functions use only a few parameters, and the RISC processors cannot use the call instructions, and therefore, use a fixed length instruction which is easy to pipeline.

**The Disadvantages of RISC architecture**

* Mostly, the performance of the RISC processors depends on the programmer or compiler as the knowledge of the compiler plays a vital role while changing the CISC code to a RISC code
* While rearranging 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.



### CISC Architecture

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.

#### **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.

#### **CHARACTERISTICS OF CISC ARCHITECTURE**

* Instruction-decoding logic will be Complex.
* One instruction is required to support multiple addressing modes.
* Less chip space is enough for general purpose registers for the instructions that are 0operated directly on memory.
* Various CISC designs are set up two special registers for the stack pointer, handling interrupts,  etc.
* MUL is referred to as a “complex instruction” and requires the programmer for storing functions.

**Advantages of CISC architecture**

* Microprogramming is easy assembly language to implement, and less expensive than hard wiring a control unit.
* The ease of microcoding new instructions allowed designers to make CISC machines upwardly compatible:
* As each instruction became more accomplished, fewer instructions could be used to implement a given task.

**Disadvantages of CISC architecture**

* The performance of the machine slows down due to the amount of clock time taken by different instructions will be dissimilar
* Only 20% of the existing instructions is used in a typical programming event, even though there are various specialized instructions in reality 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.

**VLIW COMPUTER ARCHITECTURE**

Very long instruction word (**VLIW**) describes a **computer** processing **architecture** in which a language compiler or pre-processor breaks program instruction down into basic operations that can be performed by the processor in parallel (that is, at the same time).

VLIW has both advantages and disadvantages. The main advantage is the saving in hardware — the compiler now decides what can be executed in parallel, and the hardware just does it. There is no need to check for dependencies or decide on scheduling — the compiler has already resolved these issues. (Actually, as we shall see, this may not be entirely true either.) This means that much more hardware can be devoted to useful computation, bigger on-chip caches etc., meaning faster processors.

**Advantage**

* the number of FUs can be increased without needing additional sophisticated hardware to detect parallelism, like in superscalars.
* VLIW is less complex than the Superscalar because VLIW.
* Because VLIW is implemented at the software level, all available storage space can be used. But for superscalar, its implemented in the hardware and some space will have to be allocated for the hardware so not all available storage space can be used.

**Disadvantage**

* Superscalar machines are able to dynamically issue multiple instructions each clock cycle from a conventional linear instruction stream while VLIW is static.