© January 2023| IJIRT | Volume 9 Issue 8 | ISSN: 2349-6002

Central Processing Unit

Ricky Parmar[1], Omkar Raval[2]
[1,2]*Student, K.J. Somaiya Polytechnic, Vidhyavihar*

I.INTRODUCTION

The Central Processing Unit (CPU) is the heart of a computer system, responsible for executing instructions and performing calculations. It is a complex piece of technology that has undergone significant evolution over the years, with advancements in processing power, energy efficiency, and architecture.

One of the key aspects of the CPU is its ability to perform a wide variety of tasks. It can execute instructions that control basic operations such as arithmetic, logic, and data movement, as well as more complex tasks such as video rendering and machine learning. The CPU is also capable of multitasking, allowing multiple programs to run simultaneously.

Another important aspect of the CPU is its architecture. The most common architecture is the Von Neumann architecture, which consists of a single processing unit that can access both memory and storage. However, there are also other architectures such as the Harvard architecture, which separates memory and storage, and the Systolic architecture, which is optimized for parallel processing.

One of the major advancements in CPU technology in recent years has been the development of multi-core processors. These processors contain multiple processing units, allowing for increased processing power and energy efficiency. Additionally, the use of specialized processing units, such as graphics processing units (GPUs) and digital signal processors (DSPs), has become increasingly popular for specific tasks.

Another area of research in CPU technology is the development of new materials and manufacturing processes. The use of silicon-on-insulator (SOI) and high-k metal gate (HKMG) materials has improved energy efficiency and allowed for smaller transistors. Additionally, advances in lithography techniques, such as extreme ultraviolet (EUV) lithography, have allowed for the production of smaller and more densely packed transistors.

In conclusion, the CPU is a critical component of computer systems, responsible for executing instructions and performing calculations. Advancements in architecture, materials, and manufacturing have led to increased processing power and energy efficiency, and the development of specialized processing units has allowed for more efficient performance of specific tasks. Future research will focus on continuing to improve these areas, as well as exploring new architectures and materials for even greater performance and energy efficiency.

The CPU architecture refers to the internal design of the CPU and how it handles the execution of instructions. The most common architecture is the Von Neumann architecture, which consists of a single processing unit that can access both memory and storage. However, there are also other architectures such as the Harvard architecture, which separates memory and storage, and the Systolic architecture, which is optimized for parallel processing.

Register Organization:
The register organization of a CPU refers to the number and types of registers that are built into the processor. Registers are small, fast memory locations within the CPU that are used to store data and instructions. The number and types of registers can vary depending on the CPU architecture. For example, a CPU may have general-purpose registers, special-purpose registers, and control registers.

Addressing Modes:
Addressing modes refer to the ways in which the CPU accesses memory to retrieve data. The most common addressing modes are:

Immediate addressing: the operand is part of the instruction itself.

Direct addressing: the operand is stored at the memory location specified in the instruction.

Indirect addressing: the instruction contains the memory address of the operand.

Register addressing: the operand is stored in a register.

Instruction Cycle:
The instruction cycle is the process by which the CPU fetches, decodes, and executes instructions. It consists of several steps:

Fetch: the CPU retrieves the instruction from memory.

Decode: the CPU interprets the instruction and determines what operation to perform.

Execute: the CPU performs the operation specified by the instruction.

Writeback: the CPU writes the results of the operation back to memory.
The instruction cycle is repeated continuously, allowing the CPU to perform multiple tasks simultaneously. The speed at which the instruction cycle is executed, known as the clock speed, is measured in hertz (Hz) and is one of the key factors that determine the performance of a CPU.
In conclusion, the CPU architecture, register organization, addressing modes and instruction cycle are the important aspect of the CPU that affect its performance and functionality. Understanding these concepts and their impact on the CPU can help in the design and optimization of computer systems.

The Control Unit (CU) is a component of the CPU that is responsible for fetching and executing instructions. There are several design approaches for the control unit, including hardwired control, micro-programmed control, and Wilkes control.

Hardwired Control:
In a hardwired control unit, the logic for fetching and executing instructions is implemented using digital logic circuits such as gates and flip-flops. The control signals for the CPU are generated by these circuits, which are wired together to form a fixed sequence of operations. Hardwired control units are fast, but they are inflexible and difficult to modify.

Micro-programmed Control:
In a micro-programmed control unit, the logic for fetching and executing instructions is implemented using a microcode stored in memory. The microcode consists of a series of microinstructions that the control unit fetches and executes to perform the necessary operations. Micro-programmed control units are more flexible than hardwired control units, as the microcode can be easily modified to change the behavior of the CPU.

Wilkes Control:
Wilkes control is a hybrid approach that combines the best features of hardwired and micro-programmed control. It uses a small, hardwired control unit to fetch and decode instructions, and a microcode memory to execute the instructions. This approach allows for the fast operation of the hardwired control and the flexibility of the micro-programmed control.

Comparison between Hardwired and Micro-programmed control:
Hardwired control units are faster than micro-programmed control units as the control signals are generated in parallel using digital circuits. On the other hand, micro-programmed control units are more flexible, as the microcode can be easily modified to change the behaviour of the CPU.

Microinstruction format:
Microinstructions are the individual instructions that make up the microcode in a micro-programmed control unit. The format of a microinstruction can vary depending on the specific design of the control unit, but generally it includes control signals, memory addresses, and data. It can also include fields such as opcode, addressing mode, and register specification.
In conclusion, the design of the control unit is a critical aspect of the CPU, and there are several approaches that can be used including hardwired, micro-programmed and Wilkes control. Each approach has its own advantages and disadvantages, and the choice of approach will depend on the specific requirements of the system. Understanding the microinstruction format is also important when working with micro-programmed control units.

RISC, which stands for Reduced Instruction Set Computer, is a type of computer architecture that utilizes a small, highly optimized set of instructions to perform basic operations, as opposed to a larger, more complex instruction set found in CISC (Complex Instruction Set Computer) architectures.

RISC architectures are designed to simplify the instruction set and allow for faster instruction execution. This is achieved by reducing the number of instructions, utilizing simple instructions that can be executed quickly, and by hardwiring frequently used instructions into the processor.

CISC architectures, on the other hand, have a larger and more complex instruction set, which allows for more flexibility but at the cost of slower instruction execution. CISC processors also tend to have more memory, which is used to store the larger instruction set.

In general, RISC processors are more efficient and faster than CISC processors, but CISC processors have more functionality. RISC processors are most commonly used in embedded systems and mobile devices, while CISC processors are more common in desktops and servers.

REFERNECE

[1] https://www.wikipedia.org/
[2] https://www.geeksforgeeks.org/