NEXT IAS

BIG LEARNINGS MADE EASY

# Microprocessors and Microcomputers

## Electrical Engineering

Comprehensive Theory *with* Solved Examples

## Civil Services Examination

**MADE EASY Publications Pvt. Ltd.**

Corporate Office: 44-A/4, Kalu Sarai (Near Hauz Khas Metro Station), New Delhi-110016
E-mail: infomep@madeeasy.in
Contact: 9021300500

Visit us at: www.madeeasypublications.org

**Microprocessors and Microcomputers**



First Edition : 2018
Second Edition : 2019
Reprint : 2020
Reprint : 2021
Reprint : 2022
Reprint : 2023

**Reprint : 2024**

# Contents
# Microprocessors and Microcomputers



■■■■

CHAPTER 1

# Computer Organisation

## 1.1 Computer Organization

There is a tremendous variety of products, from single-chip microcomputers costing a few dollars to supercomputers costing tens of millions of dollars, that can rightly claim the name computer. Variety is exhibited not only in cost, but also in size, performance and application. Moreover, the rapid pace of change that has always characterised computer technology continues with no letup. These changes cover all aspects of computer technology, from the underlying integrated circuit technology used to construct computer components to the increasing use if parallel organisation concepts in combining those components.

Inspite of the variety and pace of change in the computer field, certain fundamental concepts apply consistently throughout. To be sure, the application of these concepts depend on the current state of technology and the price performance objectives of the designer. To relate these changes to contemporary computer design issue, one has to understand the fundamentals of computer organisation and architecture.

## 1.2 Computer Architecture Vs Computer Organization

Computer architecture refers to those attributes of a system visible to a programmer or put another way, those attributes that have a direct impact on the logical execution of a program. It include the instruction set, the number of bits used to represent various data types (eg. numbers, characters), I/O mechanisms and techniques for addressing memory. Whereas, computer organisation refers to the operational units and their interconnections that realise the architectural specifications. It include those hardware details transparent to the programmer, such as control signals, interfaces between the computer and peripherals, and the memory technology used.

Architecture and organization are independent; you can change the organization of a computer without changing its architecture.

For designing a computer, its architecture is fixed first and then its organization is decided.

***Table*-1.1:** *Difference Between Computer Architecture and Computer Organisation*

| Computer Organization | Computer Architecture |
|---|---|
| • Computer organization deals with structural relationships that are not visible to the programmer (like clock frequency or the size of the physical memory). | • Computer architecture deals with the functional behavior of a computer system as viewed by a programmer (like the size of a data type – 32 bits to an integer). |
| • A computer's organization expresses the realization of the architecture. | • A computer's architecture is its abstract model and is the programmer's view in terms of instructions, addressing modes and registers. |
| • Organization describes how it does it. | • Architecture describes what the computer does. |
| • Organisation reveals its performance. | • Architecture indicate its hardware. |

## Von Neumann Architecture Vs Harvard Architecture

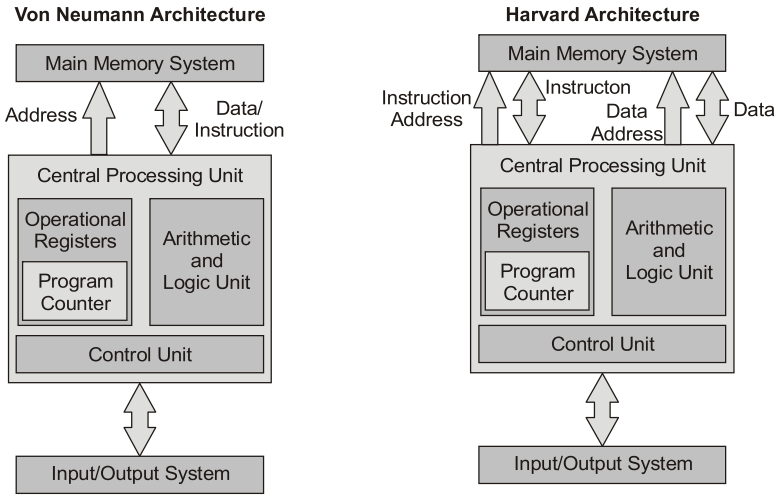



**Figure-1.1:** *Von Neumann Architecture Vs Harvard Architecture*

## 1.3 Evolution of Digital Computers

**First generation:** Vacuum tube computers (1945 ~ 1953)

- Program and data reside in the same memory (stored program concepts: John von Neumann)
- Vacuum tubes were used to implement the functions (ALU & CU design)
- Magnetic core and magnetic tape storage devices are used.
- Using electronic vacuum tubes, as the switching components.
- Assembly level language is used.

**Second generation:** Transistorized computers (1954 ~ 1965)

- Transistor were used to design ALU & CU
- High Level Language is used (FORTRAN)
- To convert HLL to MLL compiler were used.
- Separate I/O processor were developed to operate in parallel with CPU, thus improving the performance.
- Invention of the transistor which was faster, smaller and required considerably less power to operate

**Third generation**: Integrated circuit computers (1965 ~ 1980)

- IC technology improved.
- Improved IC technology helped in designing low cost, high speed processor and memory modules
- Multiprogramming, pipelining concepts were incorporated
- DOS allowed efficient and coordinate operation of computer system with multiple users
- Cache and virtual memory concepts were developed
- More than one circuit on a single silicon chip became available.

**Fourth generation:** Very large scale integrated (VLSI) computers (1980 ~ 2000)

- CPU termed as microprocessor
- INTEL, MOTOROLA, TEXAS, NATIONAL semiconductors started developing microprocessor
- Workstations, microprocessor (PC) & Notebook computers were developed
- Interconnection of different computer for better communication LAN, MAN, WAN
- Computational speed increased by 1000 times
- Specialized processors like Digital Signal Processor were also developed.

**Fifth generation:** System-on-chip (SOC) computers (2000 ~ till date)

- E-Commerce, E-banking, home office
- ARM, AMD, INTEL, MOTOROLA
- High speed processor - GHz speed
- Because of submicron IC technology, more features were added in small size.

## 1.4 Components of Computer Structure

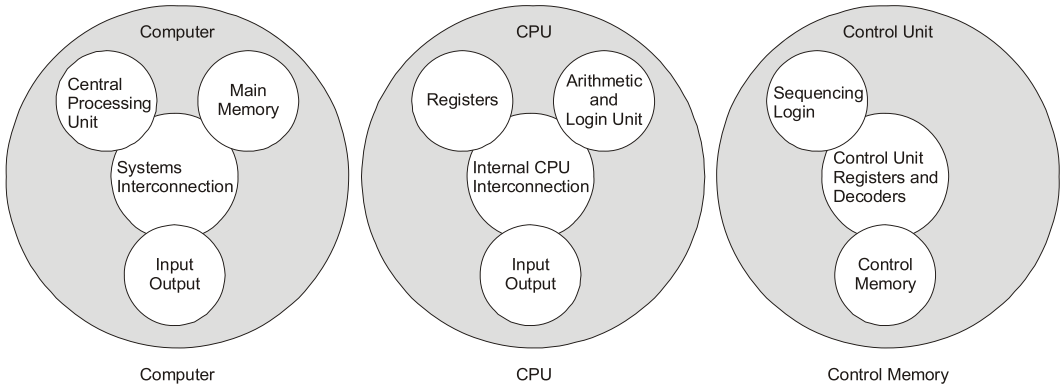



**Figure-1.2:** *Computer Structure vs CPU Structure vs Control Unit*

1. **Input Unit:** Computers can understand only machine language. Therefore for converting data from human language to machine language we use some special peripheral devices which are called input device.

   *Examples*: Keyboard, Mouse, Joystick etc.

2. **Output Unit:** After passing instructions for solving particular problem, the results came out from computer comes in machine language and this is very difficult to convert that results into human language. There are several peripheral devices which help us to convert the machine language data into human acceptable data. These devices are called output devices.

   *Examples*: Monitor, Printer, LCD, LED etc.

3. **Memory Unit:** It is used to store data in computer. Memory unit performs the following functions

   (a) Stores data and instructions required for processing.

   (b) Stores the intermediate results obtained during processing.

   (c) Stores final results before sending it to output unit.

   Two class of storage units: (i) Primary Memory (ii) Secondary Memory

   Further, there are two types of primary memory as RAM (Random Access Memory) and ROM (Read Only Memory). RAM is used to store data temporarily during the program execution. ROM is used to store data and program which is not going to change.

   Secondary Memory is used for bulk storage or mass storage to store data permanently.

4. **CPU:** It is main unit of the computer system. It is responsible for carrying out computational task. The major structural components of a CPU are:

   (a) *Control Unit (CU):* Controls the operation of the CPU and hence the computer.

   (b) *Arithmetic and Logic Unit (ALU):* Perform computer's data processing functions.

   (c) *Register:* Provides internal storage to the CPU.

   (d) *CPU Interconnection:* provides communication among the control unit, ALU, and register.

## 1.5 CISC and RISC Architectures

***Table*-1.2:** *RISC and CISC*

| RISC | | CISC | |
|---|---|---|---|
| 1. | RISC stands for Reduced Instruction Set Computer. | 1. | CISC stands for Complex Instruction Set Computer. |
| 2. | RISC processors have simple instructions taking about one clock cycle. The average clock cycle per instruction (CPI) is 1.5. | 2. | CISC processors have complex instructions that take up multiple clock for execution. The average clock cycle per instruction (CPI) is in the range of 2 and 15. |
| 3. | Performance is optimized with more focus on software. | 3. | Performance is optimized with more focus on hardware. |
| 4. | It has no memory unit and uses a separate hardware to implement instructions. | 4. | It has a memory unit to implement complex instructions. |
| 5. | It has a hard-wired unit of programming. | 5. | It has a microprogramming unit. |
| 6. | The instruction set is reduced i.e. it has only a few instructions in the instruction set. Many of these instructions are very primitive. | 6. | The instruction set has a variety of different instructions that can be used for complex operations. |
| 7. | The instruction set has a variety of different instructions that can be used for complex operations. | 7. | CISC has many different addressing modes and can thus be used to represent higher-level programming language statements more efficiency. |
| 8. | Complex addressing modes are synthesized using the software. | 8. | CISC already supports complex addressing modes. |
| 9. | Multiple register sets are present. | 9. | Only has a single register set. |
| 10. | RISC processors are highly pipelined. | 10. | They are normally not pipelined or less pipelined. |
| 11. | The complexity is RISC lies with the compiler that executes the program. | 11. | The complexity lies in the microprogram. |
| 12. | Execution time is very less. | 12. | Execution time is very high. |
| 13. | Code expansion can be a problem. | 13. | Code expansion is not a problem. |
| 14. | Decoding of instructions is simple. | 14. | Decoding of instructions is complex. |
| 15. | It does not require external memory for calculations. | 15. | It requires external memory is complex. |
| 16. | The most common RISC microprocessors are Alpha, ARC, ARM, AVR, MIPS, PA-RISC, PIC, Power Architecture, and SPARC. | 16. | Examples of CISC processors are the System/360, VAX, PDP-11, Motorola 68000 family, AMD and Intel x86 CPUs. |
| 17. | RISC architecture is used in high-end applications such as video processing, telecommunications and image processing. | 17. | CISC architecture is used in low-end applications such as security systems, home automation, etc. |

## 1.6 Flynn's Classification of Processors

### 1.6.1 Single Instruction Stream, Single Data Stream (SISD)

A computer with a single processor is called a Single Instruction Stream, Single Data Stream (SISD) Computer. It represents the organization of a single computer containing a control unit, a processor unit, and a memory unit. Instructions are executed sequentially and the system may or may not have internal parallel processing. Parallel processing may be achieved by means of a pipeline processing.

In such a computer a single stream of instructions and a single stream of data are accessed by the processing elements from the main memory, processed and the results are stored back in the main memory. SISD computer organization is shown in figure below.

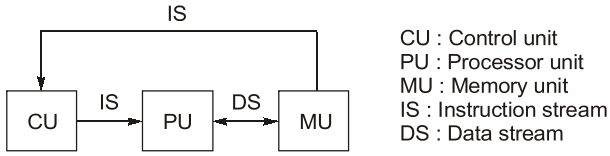



**Figure-1.3:** *SISD Systems*

### 1.6.2 Single Instruction Stream, Multiple Data Stream (SIMD)

It represents an organization of computer which has multiple processors under the supervision of a common control unit. All processors receive the same instruction from the control unit but operate on different items of the data. SIMD computers are used to solve many problems in science which require identical operations to be applied to different data set synchronously. Examples are added a set of matrices simultaneously, such as $\sum_i \sum_k (a_{ik} + a_{ik})$.

Such computers are known as array processors. SIMD computer organization is shown in figure below.

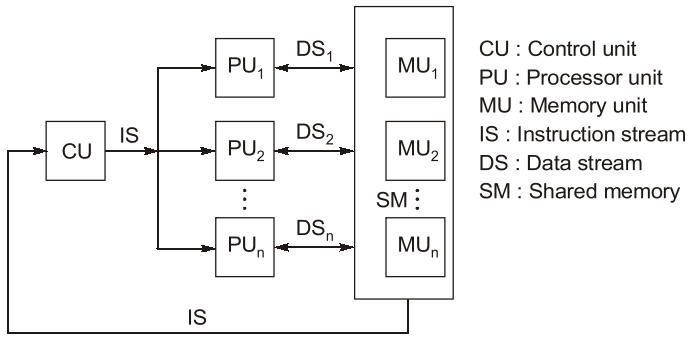



***Figure-1.4:*** *SIMD Systems*

### 1.6.3 Multiple Instruction Stream, Single Data Stream (MISD)

It refers to the computer in which several instructions manipulate the same data stream concurrently. In the structure different processing element run different programs on the same data. This type of processor may be generalized using a 2-dimensional arrangement of processing element. Such a structure is known as systolic processor. MISD computer organization is shown in figure below.

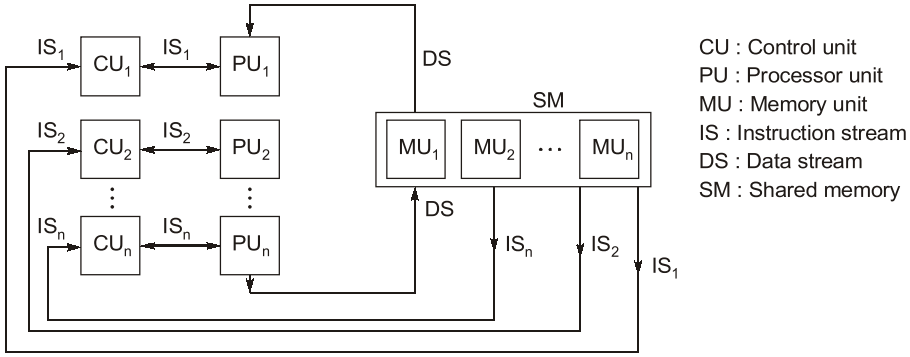



***Figure-1.5:*** *MISD Systems*

### 1.6.4 Multiple Instruction Stream, Multiple Data Stream (MIMD)

MIMD computers are the general purpose parallel computers. Its organization refers to a computer system capable of processing several programs at a same time. MIMD systems include all multiprocessing systems. MIMD computer organization is shown in figure below.

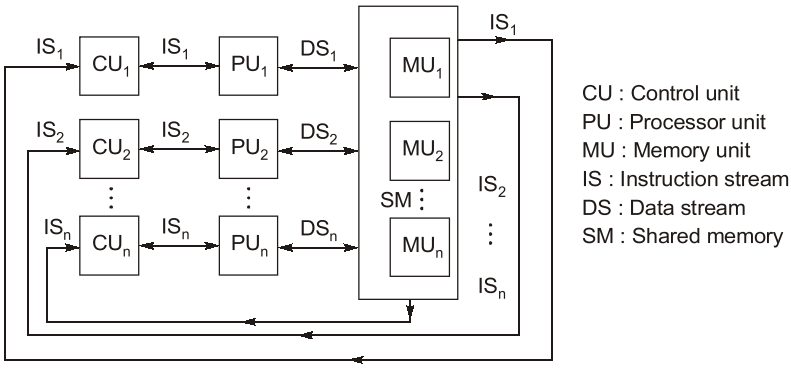



***Figure-1.6:*** *MIMD System*

## 1.7 Control Unit

Control unit generates the signals for sequencing the operations in the datapath. It performs the task by repeatedly cycling through fetch-execute cycle steps:

- Read the instruction that's pointed to by the PC from memory and move it into the IR.
- Increment the PC.
- Decode the instruction in the IR.
- If the instruction has to read an operand from memory, calculate the operand's address (effective address) and read the operand from memory.
- Execute the current instruction from the IR.

To execute an instruction, the control unit of the CPU must generate the required control signal in the proper sequence.

**Functions of Control Unit**

1. **Fetch and instruction sequencing:** Generates control signal to fetch instruction from memory and the sequence of operations involved in processing an instruction.
2. **Instruction interpretation and execution:**
   - Interpreting the operand addressing mode implied in the operation code and fetching the operands.
   - Sequencing the successive micro operations on the data path to execute the operation code specified in the instruction.
3. **Interrupt processing:** Process unmasked interrupts in the interrupt cycle as follows:
   - Suspend execution of current program
   - Save context
   - Set PC to start address of interrupt handler routine
   - Process interrupt
   - Restore context and continue interrupted program

**Control Unit and Datapath**

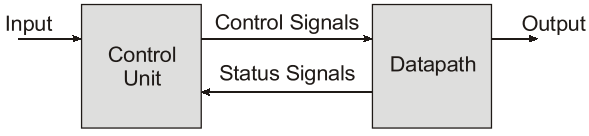



***Figure-1.7***

**Control Unit:** It generates the signals for sequencing the micro-operations in the datapath based on status and input signals.

**Datapath:** It implements the micro-operations under control of the control unit using its functional units (registers, ALU, MUXes, Buses, etc.)

## 1.8 Control Unit Implementation

The main objective of control unit is to generate the control signal in proper sequence. Control unit is implemented in one of two ways either Hardwired control or Micro-programmed control.

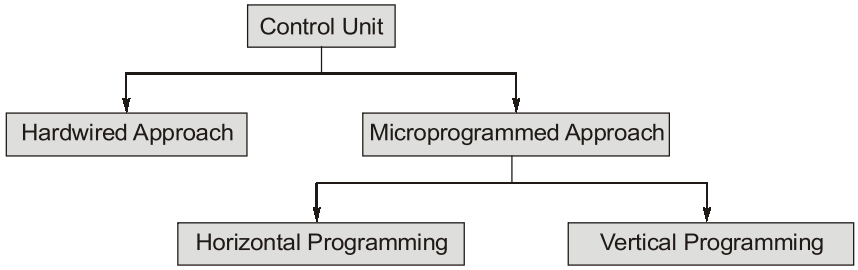



***Figure-1.8***

### 1.8.1 Hardware Control Unit

Hardware Control unit is made up of sequential and combinational circuits (Hardware) to generate the control signals.

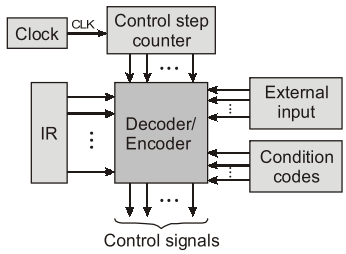



***Figure-1.9:*** *Control Unit Organization*

**Decoder/Encoder Block**

It is a combinational circuit that generates the required control signals depending on the state of all its input.

(i) **Step Decoder:** The decoder part of decoder/encoder provide a separate signal line for each control step, or time slot in the control sequence.

(ii) **Instruction Decoder:** The output of the instructor decoder consists of a separate line for each machine instruction loaded in the IR, one of the output line $INS_1$ to $INS_m$ is set to 1 and all other lines are set to 0.

(iii) **Encoder:** It is required to generate many control signals by the control unit. These are basically coming out from the encoder circuit of the control signal generator.
The control signals are: $PC_{in}$, $PC_{out}$, $Z_{in}$, $Z_{out}$, $MAR_{in}$, ADD, END, etc.

The encoder sends a reset signal after the end of an instruction and a stop signal to the sequencer after the last sequence. The encoder also sends count start signal to let the clock, increment the counter during processing of an instruction.

The advantage of hardwired control is that it is very fast. The disadvantage is that the instruction set and the control logic are directly tied together by special circuits that are complex and difficult to design or modify.

The control signals are expressed as Sum-of-Product (SOP) expression and they are directly realized on the independent hardware.

### 1.8.2 Microprogrammed Control Unit

A control unit whose binary control variables are stored in memory is called a micro-programmed control unit. A control memory (Control storage) on the processor contains micro-programs that activate the necessary control signals whereas hardwired control unit generate control signals by sequential and combinational circuits.

- **Microinstruction:** The microinstruction specifies one or more micro-operations for the system. It Contains a control word and a sequencing word.
- **Microprogram:** A sequence of microinstructions called a microprogram. Program stored in memory that generates all the required control signals to execute the instruction set correctly.

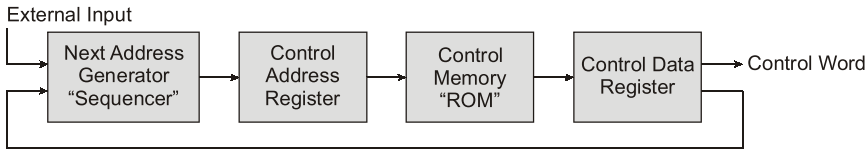



*Figure-1.10*

**Control Memory (Control Storage):** It is Memory unit in the micro-programmed control unit to store the micro-program. Each word in control memory contains within it a microinstruction.

**Control Address Register:** It specifies the address of the microinstruction in control memory.

**Control Data Register:** It holds the microinstruction read from memory.

The location of the next microinstruction may be the one next in sequence, or it may be locate somewhere else in the control memory. For this reason it is necessary to use some bits of the present microinstruction to control the generation of the address of the next microinstruction.

The next address may also be a function of external input conditions. While the microoperations are being executed, the next address is computed in the next address generator circuit and then transferred into the control address register to read the next microinstruction.

**Sequencing Word:** Information needed to decide the next microinstruction address.

**Next Address Generator (Sequencer or Microprogram Sequencer):** It determines the microinstruction Address to be executed in the next clock cycle.

The address of the next microinstruction can be specified several ways, depending on the sequencer inputs as follows:

(i) Incrementing the control address register by one,
(ii) Loading into the control address register an address from control memory,
(iii) Transferring an external address, or
(iv) Loading an initial address to start the control operations.

Determining the address of the next microinstruction depends on one of the following:

- In-line Sequencing
- Branch
- Conditional Branch
- Subroutine
- Loop
- Instruction op-code mapping

**Control word:** The control variables at any given time can be represented by a string of 1's and 0's called a control word. It has all the control information required for one clock cycle. Which can be programmed to perform various operations on the component of the system.

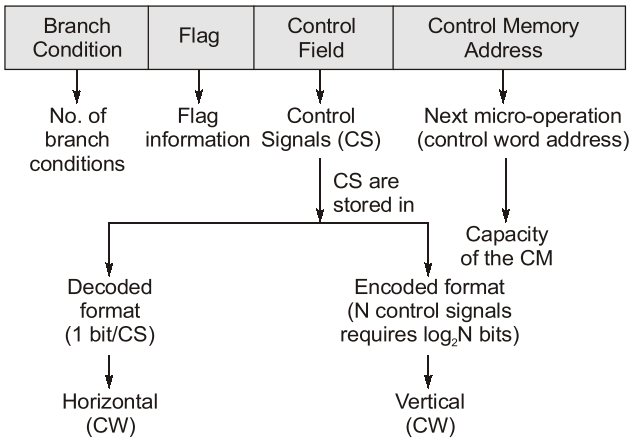



***Figure-1.11:*** *Control Word*

### 1.8.3 Types of Microinstructions

**1. Horizontal Microprogramming:**

- There are no intermediate decoders and the control word bits are directly connected to their destination.
- Each bit in the control word is directly connected to some control signal.
- The total number of bits in the control word is equal to the total number of control signals in the CPU.
- Each Microinstruction specifies many different micro-operations to be performed in parallel.
- All control signals directly in micro-code
- Due to lot of signals, many bits in micro-instruction.

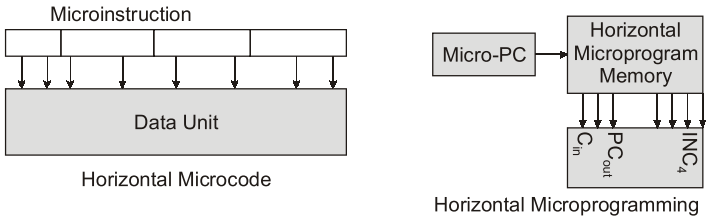



***Figure-1.12:*** *Horizontal microcode and Horizontal Microprogramming*

**2. Vertical Microprogramming**

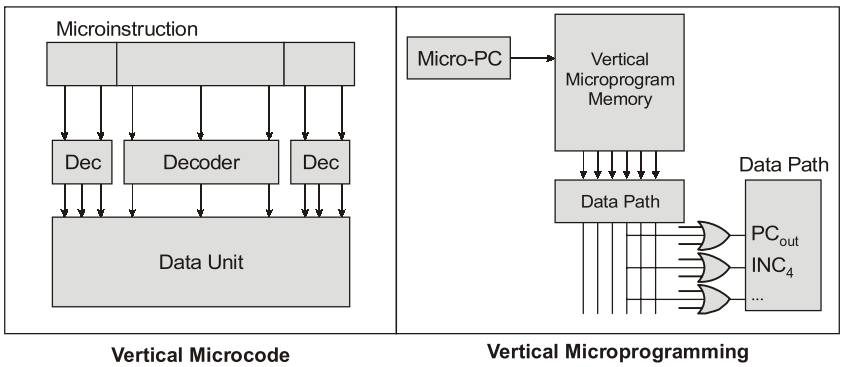



***Figure-1.13:*** *Vertical microcode and Vertical Microprogramming*

- Vertical microcode schemes employ an extra level of decoding to reduce the control word width.
- From an n bit control word we may have $2^n$ bit signal values.
- It takes less space but may be slower
- Actions need to be decoded to signals at execution time
- Each Microinstruction specifies single or few microoperations to be performed.

## 1.9 Main Memory Orgranisation

The memory hierarchy was developed based on a program behavior known as locality of references. Memory references are generated by the CPU for either instruction or data access. These accesses tend to be clustered in certain regions in time, space, and ordering.

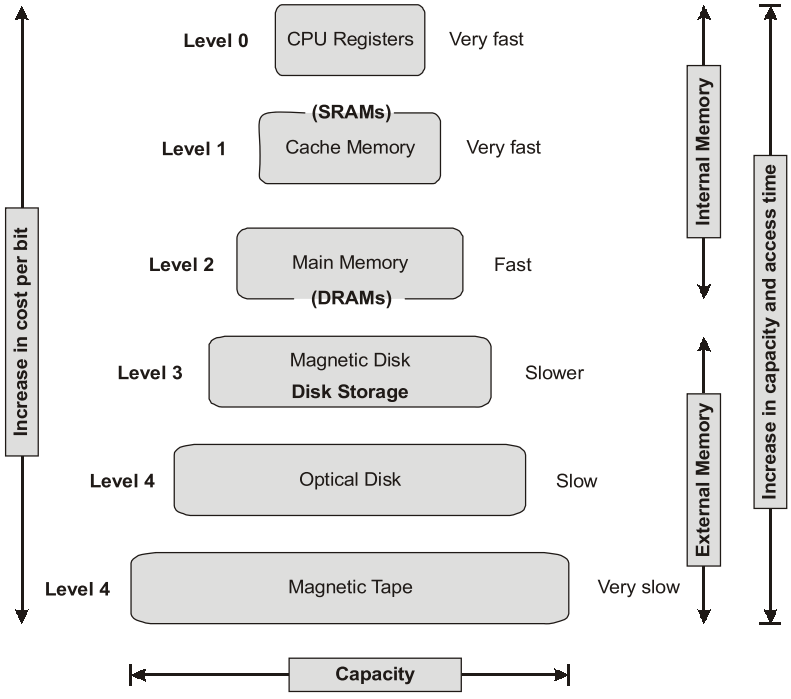



*Figure-1.14*

### 1.9.1 Types of Memory based on Access

1. Serial Access Memory: The system must search the storage device from the beginning of the memory address until it finds the required piece of data. Memory device which supports such access is called a Sequential Access Memory or Serial Access Memory.
   *Example:* Magnetic tape.
2. Direct Access Memory: Direct access memory or Random Access Memory, refers to condition in which a system can go directly to the information that the user wants. Memory device which supports such access is called a Direct Access Memory.
   *Example:* Magnetic disk and optical disks.

### 1.9.2 Memory Access Methods

1. Sequential Access: In this method, the memory is accessed in a specific linear sequential manner. For example, if fourth record (collection of data) stored in a sequential access memory needs to be accessed, the first three records must be skipped. Thus, the access time in this type of memory depends on the location of the data. Magnetic disks, magnetic tapes and optical memories the CD-ROM use this method.

2. **Random Access:** In this mode of access, any location of the memory can be accessed randomly. In other words, the access to any location is not related with its physical location and is independent of other locations. For random access, a separate mechanism is therefore each location. Semiconductor memories (RAM, ROM) use this method.
3. **Direct Access:** This method is basically the combination of previous two methods. Memory devices such as magnetic hard disks contain many rotating storage tracks. If each track has its own read/write head, the tracks can be accessed randomly, but access within each track is sequential. In this case the access is semi-random or direct. The access time depends on both the memory organization and the characteristic of storage technology. Disk use this method.
4. **Associative Access:** This is a special type of random access method that enables one to make a comparison of desired bit locations within a word for a specific match and to do this for all words simultaneously. Thus, based on a portion of a word's content, word is retrieved rather than its address. Cache memory uses this type of access mode.

**Memory or Primary Memory (Core Memory/Store/Storage)**

The memory stores the instructions and data for an executing program. Memory is characterized by the smallest addressable unit as one of the following.

- **Byte addressable:** Smallest unit is an 8-bit byte.
- **Word addressable:** Smallest unit is a word, usually 16 or 32 bits in length.

Most modern computers are byte addressable, facilitating access to character data. Logically, computer memory should be considered as an array. The index into this array is called the **address** or **"memory address"**.

**There are two types of primary memories:** RAM and ROM

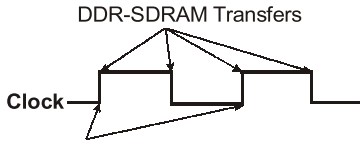
### 1.9.3 Random Access Memory (RAM)

RAM is Read/write memory, Random access, and data is temporarily stored. RAM is further classified into two types:

1. **DRAM (Dynamic Random Access Memory):** It tends to lose its contents, even when powered. Special "refresh circuitry" must be provided.

   SDRAM (Synchronous DRAM): It is a kind of DRAM that is synchronised with the clock speed that the microprocessor is optimised for. The memory bus clock is driven by the CPU system clock, but it is always slower.

   In **SDRAM**, the memory transfers take place on a timing dictated by the memory bus clock rate. This memory bus clock is always based on the system clock. In "plain" SDRAM, all the transfers take place on the rising edge of the memory bus clock. In **DDR SDRAM** (Double Data Rate Synchronous DRAM), the transfers take place on both the rising and falling clock edges.

   
   
   ***Figure-1.15***

   "Plain" SDRAM makes a transfer every cycle of the memory bus. DDR–SDRAM makes two transfers for every cycle of the memory bus, one on the rising edge of the clock cycle and another one on the falling edge of the clock cycle.
2. **SRAM (Static Random Access Memory):** It will keep its contents as long as it is powered. Compared to DRAM, SRAM is faster, more expensive, physically larger (fewer memory bits per square millimeter).

### 1.9.4 Read Only Memory (ROM)

ROM is Read only memory, Random access, and Data is permanently stored. ROM is further classified as following :

1. **MROM (Masked ROM):** The contents of the memory are set at manufacture and cannot be changed without destroying the chip.
2. **PROM (Programmable ROM):** The contents of the chip are set by a special device called a "PROM Programmer". Once programmed the contents are fixed.
3. **EPROM (Erasable and Programmable ROM):** It is same as a PROM, but that the contents can be erased using UV light and reprogrammed by the PROM Programmer.
4. **EEPROM (Electrically EPROM):** The contents can be erased electrically and reprogrammed by the PROM Programmer.

***Table*-1.3**

| Memory Type | Category | Erasure | Write Mechanism | Volatility |
|---|---|---|---|---|
| Random-Access Memory (RAM) | Read-Write Memory | Electrically, byte-level | Electrically | Volatile |
| Read-Only Memory (ROM) | Read-Only Memory | Not possible | Masks | Non-volatile |
| Programmable ROM (PROM) | | | Electrically | |
| Erasable PROM (EPROM) | Read-Mostly Memory | UV light, Chip-level | | |
| Electrically Erasable PROM (EEPROM) | | Electrically, byte-level | | |
| Flash memory | | Electrically, block-level | | |

### 1.9.5 RAM Vs ROM

***Table*-1.4**

| | RAM | ROM |
|---|---|---|
| **Accessibility** | The information stored in the RAM is easily accessed because it communicates directly with the processor. | The processor cannot directly access the information. Hence, the information will be transferred into the RAM and then it gets executed by the processor to access the ROM information. |
| **Volatility** | Volatile in nature, Data is stored as long as the power supply is switched on. Data will erased if the computer crashes or is turned off. | Non-volatile in nature. Data is stored even the power supply is switched off. Data is retain even if the computer crashes or is turned off. |
| **Storage** | Data is temporary.<br>It is only there as long as the computer is on and it can be changed. | Data is permanent.<br>It can never be changed.<br>Contents are remain same. |
| **Speed** | The accessing speed of RAM is faster, it assist the processor to boost up the speed. | Speed is slower compared to RAM, ROM cannot boost up the processor speed. |
| **Data Preserving** | Electricity supply is needed in RAM to flow to preserving information. | Electricity supply is not needed in ROM to flow to preserving information. |
| **Structure** | The RAM is an chip, which is in the rectangle form and is inserted over the mother board of the computer. | ROMs are generally the optical drivers, which are made of magnetic tapes. |
| **Cost** | The price of RAMs are comparatively high. | The price of ROM's are comparatively low. |
| **Chip size** | Physically size of RAM chip is larger than ROM chip. | Physically size of ROM chip is smaller than RAM chip. |
| **Category** | Read-write memory.<br>Data can be written to or read from. | Read-only memory.<br>Data can only be read.<br>User cannot make any changes to the information. |