Listen Learn Lead

Engineering Excellence for Empowerment

Nadar Saraswathi College of Engineering & Technology

NSCET E-LEARNING PRESENTATION

LISTEN … LEARN… LEAD…

Computer Science Engineering

II^{nd} YEAR / Iv^{th} SEMESTER

CS8491-Computer Architecture

B.Sri Chithra Devi,M.E
Assistant Professor
Nadar Saraswathi College of Engineering & Technology,
Vadapudupatti, Annanji (po), Theni – 625531.

UNIT-I
BASIC STRUCTURE OF A COMPUTER SYSTEM

Computer Architecture

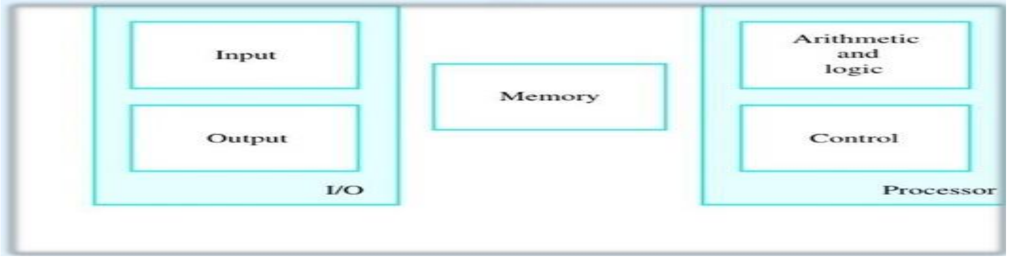
- Computer architecture is concerned with the structure and behavior of the various functions modules of the computer and how they interact to provide the processing needs of the user
- It refers to the operational units and their interconnections that realize the architectural specification.

Functional Units

- Basic functional units of a computer are I/O Processor Output Memory Input and Arithmetic logic Control.

Input devices :

- Receive both data and program statements to function properly and be able to solve problems feeding data and programs Eg : keyboard, a mouse, or a scanner Input devices



Nadar Saraswathi College of Engineering & Technology

Functional Units

- **Memory Unit :** Memory Unit Store programs and data Two classes of storage Primary storage Fast Programs must be stored in memory while they are being executed Large number of semiconductor storage cells Processed in words Address RAM and memory access time Memory hierarchy cache, main memory Secondary storage larger and cheaper.
- **Central Processing Unit(CPU) :** The brain of a computer system is the central processing unit (CPU). computing center of the system processes data transferred to it from one of the various input devices consists of a control section, an arithmetic-logic section and an internal storage section Central Processing Unit(CPU)

Functional Units

Arithmetic and Logic Unit (ALU) :

- Arithmetic and Logic Unit (ALU) Most computer operations are executed in ALU. Performs all arithmetic calculations and take logical decisions. Tests various conditions encountered during processing and takes action based on the result

Control Unit :

- Control Unit All computer operations are controlled by the control unit. The control section directs the flow of traffic (operations) and data. It also maintains order within the computer Operations of a computer: Accept information in the form of programs and data through an input unit and store it in the memory Fetch the information stored in the memory, under program control, into an ALU, where the information is processed Output the processed information through an output unit Control all activities inside the machine through a control unit

Output Devices :

Output Devices convert information coming from a computer system into some form perceptible by humans Eg:Monitor

Eight Ideas

- Design for Moore's Law
- Use Abstraction to Simplify Design
- Make the common case fast
- Performance via parallelism
- Performance via pipelining
- Performance via prediction
- Hierarchy of memories
- Dependability via redundancy

Design for Moore's Law

- States that integrated circuit resources doubles every 2 years or months
- Important key note is that the start and end point of completion may varies because of many changes in-between of period.
- So designers must anticipate the end design at the beginning stage.

Use Abstraction to Simplify Design

- Both computer architects and programmers had to invent techniques to make themselves more productive, for otherwise design time would lengthen as dramatically as resources grew by Moore's Law..
- A major productivity technique for hardware and soft ware is to use abstractions to represent the design at different levels of representation.
- lower-level details are hidden to off er a simpler model at higher levels.
- We'll use the abstract painting icon to represent this second great idea.

Make the common case fast

- Making the common case fast will tend to enhance performance better than optimizing the rare case.
- the common case is oft en simpler than the rare case and hence is oft en easier to enhance
- This common sense advice implies that you know what the common case is, which is only possible with careful experimentation and measurement.

Performance via parallelism

- Performance of computer can be increased by parallelism.
- Because parallel processing will increase the speed of performance execution.
- Computer architecture design has more performance by performing operations in parallel.

Performance via pipelining

- Pipeline is used to fetch the instruction for execution in parallel.
- Operation of pipeline: first instruction output may be taken as the input of next instruction for processing.
- Example: output of first instruction is 10.
- Input of second instruction is 10,20

Performance via prediction

- In some cases it is quit to use the predicted processing instead of accurate.
- Because more are less both will be giving the similar output.
- When the end design comes to know at the initial stage.

Hierarchy of memories

- Programmers want memory to be fast, large, and cheap, as memory speed often shapes performance, capacity limits the size of problems that can be solved, and the cost of memory today is often the majority of computer cost.
- Architects have found that they can address these conflicting demands with a hierarchy of memories, with the fastest, smallest, and most expensive memory per bit at the top of the hierarchy and the slowest, largest, and cheapest per bit at the bottom.
- Caches give the programmer the illusion that main memory is nearly as fast as the top of the hierarchy and nearly as big and cheap as the bottom of the hierarchy.

Hierarchy of memories

- We use a layered triangle icon to represent the memory hierarchy.
- The shape indicates speed, cost, and size: the closer to the top, the faster and more expensive per bit the memory.
- The wider the base of the layer, the bigger the memory.

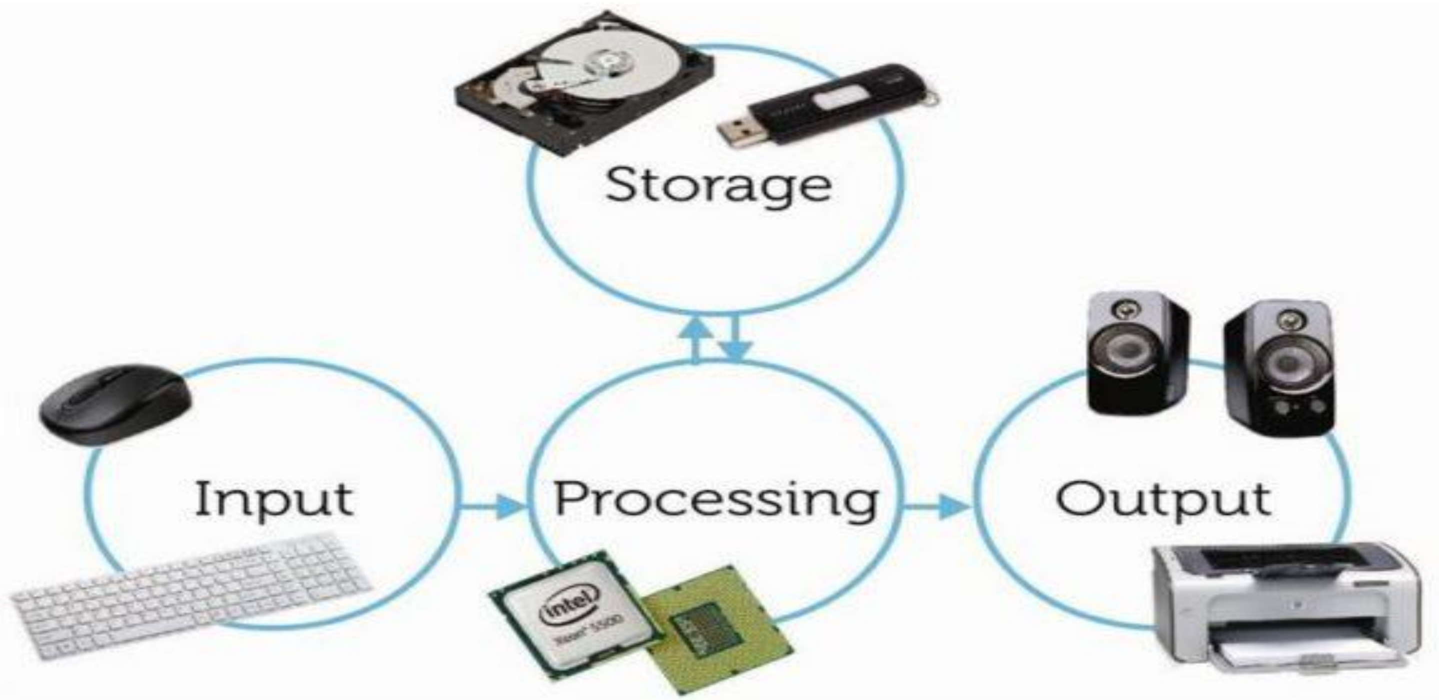
Dependability via redundancy

- Computers only need to be fast. They need to be dependable.
- Since any device can fail, we make systems dependable by including redundant components that take over when a failure occurs and to help detect failures.

Components of computer systems

- Every computer system has the following three basic components:
- Input unit
- Central processing unit
- Output unit
- While there are other components as well, these three are primarily responsible for making a computer function.
- They must work in complete synergy because that will ensure smooth overall functioning. Hence, we can even call them building blocks of a computer system.

Elements of a computer system



Input Unit

- These components help users enter data and commands into a computer system.
- Data can be in the form of numbers, words, actions, commands, etc.
- The main function of input devices is to direct commands and data into computers.
- Computers then use their CPU to process this data and produce output.

Central Processing Unit

- CPU is a complex set of electronic circuit which acts as a control central for the system.
- After receiving data and commands from users, a computer system now has to process it according to the instructions provided. Here, it has to rely on a component called the central processing unit.

- The CPU further uses these three elements:

Memory Unit

- The function of the memory unit is to store programs and data.There are two classes of memory

  Main memory or primary memory

  Secondary memory

Main memory

- Main memory is a fast memory that operates at electronic speeds.
- Programs must be stored in the memory while they are being executed.
- The memory contains a large number of semiconductor storage cells,each capable of storing one bit of information.
- Main memory can be classified into two types

  RAM-Random Access Memory

  ROM-Read Only Memory

Arithmetic and Logic Unit

- An Arithmetic and Logic unit is a digital circuit used to perform arithmetic and logic operations.
- An ALU represents the fundamental building block of the central processing unit of a computer.

Registers

- A register may hold a computer instruction storage address or any kind of data

Data Register

- It is the register of a computers control unit that contain the data to be stored in the computer storage.

Arithmetic and Logic Unit

Address Register

- It is a CPU register that either stores the memory address from which data will be fetched to the CPU or the address to which data will be sent and stored.

Status Register

- The status register is a hardware register which contains information about the state of the processor.

Program Counter

- A program counter is a register in a computer processor that contains the address of the instruction being executed at the current time.

Control Unit

- This unit is the backbone of computers. It is responsible for coordinating tasks between all components of a computer system.

- The control unit collects data from input units and sends it to processing units depending on its nature.

- Finally, it also further transmits processed data to output units for users.

Output Unit

- The third and final component of a computer system is the output unit.
- After processing of data, it is converted into a format which humans can understand.
- After conversion, the output units displays this data to users.
- Examples of output devices include monitors, screens, printers and speakers.
- Thus, output units basically reproduce the data formatted by the computer for users’ benefit.

Technology

- To improve the performance of computer two things are more important.
- One is the processor and another is memory.Because computer designers must know the latest technology to design a better computer.
- Technology shapes what computers will be able to do and how quickly they will evolve.

Technology

S.NO	TECHNOLOGY	RELATIVE PERFORMANCE
1951	VACUUM TUBES	1
1965	TRANSISTORS	35
1975	INTEGRATED CIRCIUTS	900
1995	VERY LARGE SCALE IC	2,400,000
2013	ULTRA LARGE SCALE IC	6,200,000,000

Tuesday, December 11, 2018 27

- Relative performance per unit cost of technologies used in computer.
- From the table it is known that all computer professional should be familiar with the basics of Integrated circuits.

Technology

- To improve the performance of computer two things are more important.
- One is the processor and another is memory. Because computer designers must know the latest technology to design a better computer.
- Technology shapes what computers will be able to do and how quickly they will evolve.

Vaccum Tube

- The computers of first generation used vacuum tubes as the basic components for memory and circuitry for CPU (Central Processing Unit).
- These tubes, like electric bulbs, produced a lot of heat and the installations used to fuse frequently.
- Therefore, they were very expensive and only large organizations were able to afford it.
- In this generation, mainly batch processing operating system was used. Punch cards, paper tape, and magnetic tape was used as input and output devices.
- The computers in this generation used machine code as the programming language.

Transistor

- It is a smaller than other device and consume less power,high speed ,more memory storage and size is small.
- It helps to handle both floating point and fixed point operations.
- Separate input processor.
- Support higher level programming languages.

Integrated Circuits

- The fourth generation of computers took advantage of the invention of the microprocessor more commonly known as a CPU. Microprocessors, along with integrated circuits, helped make it possible for computers to fit easily on a desk and for the introduction of the laptop.
- Some of the earliest computers to use a microprocessor include the Altair 8800, IBM 5100, and Micral.
-

Very Large Scale Integrated Circuit(VLSI)

- It is possible to manufacture the entire CPU,main memory into single CPU.
- This can be implemented in PC and high performance parallel processor.
- Concept supported are concurrency,pipeline,cache and virtual memories.
- This evolved to perform high performance computing.

Chip

- Chip is the basic element for manufacturing integrated circuits.
- The manufacture of a chip begins with silicon.Because silicon does not conduct electricity well,it is called a semiconductor.
- With a special chemical process it is possible to add materials to silicon that allow tiny areas to transform into one of three devices such as
- Excellent conductors of electricity
- Excellent insulators from electricity.
- Areas that can conduct or insulate under special conditions
- The manufacturing process for IC is critical to the cost of the chips and hence important to computer designers.

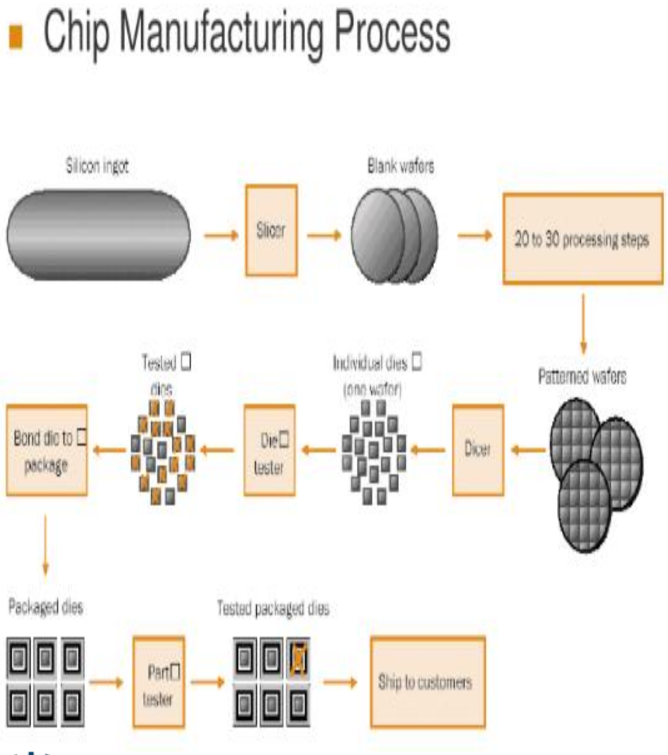
Chip

- The process starts with a silicon crystalingot.
- Silicon crystalin got is a rod composed of silicon crystal that is between 8-12 inches in diameter and 12-24 inches long.
- An ingot is finely sliced into wafers no more than 0.1 inches thick.
- Wafer is a slice from a silicon ingot no more than 0.1 inches thick,used to create chips.
- These wafer then go through a series of processing steps to create the transistors,conductors and insulators.

Chip

- Die is the individual rectangular sections that are cut from a wafer and more informally known as chips.
- Dicing enables us to discard only those dies that were unlucky enough to contain the flows,rather than the whole wafer.
- This concept is known as yield of a process.
- Yield process is defined as the percentage of good dies from the total number of dies on the wafer.
- Smaller sizes transistors and wires improve the die count count per wafer.

Chip



- Die is the individual rectangular sections that are cut from a wafer and more informally known as chips.
- Dicing enables us to discard only those dies that were unlucky enough to contain the flows,rather than the whole wafer.
- This concept is known as yield of a process.
- Yield process is defined as the percentage of good dies from the total number of dies on the wafer.
- Smaller sizes transistors and wires improve the die count count per wafer.

Chip

Bonding:

- once we have found good dies,they are connected to the input/output pins of a package using a process called bonding.
- These Packaged parts are tested a final time,because mistake can occur in package.
- If there is no mistake it can be shipped to customers.

Elaboration:

- The cost of an IC can be expressed in three simple equations

cost per die=cost per wafer/dies per wafer*yield

Chip

Elaboartion(contd…)

Dies per wafer =wafer area/die area

Yield=1/[1+defects per area *die area/2)^2

- First equation is straight forward to derive.
- Second equation is an approximation.it does not subtract the area near the border of the round wafer that cannot accommodate the rectangular dies.
- Final equation is based on emprical observations of yields at IC factories, with the exponent related to the number of critical processing steps.

Performance

- Accessing the performance of computers is not easy task.
- Performance of computer depends on many things such as modern software systems and wide range of performance improvemnet techniques in hardware side.
- To choose better computer among different computers,performance is an important attribute.
- For selecting a computer it is nescessary to know,how to measure performance and limitations of performance measurements.

Performance

Response time

- It is the metric used to know which system has the best performance.
- Response time also called as executiom time it can be defined as the total time required for the computer to complete a task,including disk accesses,memory accesses,I/o activities,operating system overhead,CPU execution time and so on.

Bandwidth:

- Bandwidth is also known as throughput,it then another measure of performance.
- Bandwidth can be defined as the"number of tasks completed per unit time"

Performance

Throughput and response time

- To understand the relationship between throughput and response time let us consider the following equation

performance=1/Execution time

Relative performance

- Performance and execution time are reciprocals,increasing performance decreasing execution time.

Measuring performance

Types of CPU time

CPU time can be classified into two types

- User CPU time
- System CPU time.

User CPU time

- CPU time spent in the program called user CPU time.

System CPU time

- CPU time spent in the operating system performing taks on behalf of the program.
- Differentiating between system and user CPU time is difficult to do accurately.

Measuring performance

Clock Cycles

- Clock cycles also called tick,clock tick,clock period,clock or cycle.
- It is the time for one clock period usually of the processor clock,which runs at constant rate.

Clock period

- The length of each clock cycle is known as clock period.

CPU execution time for a program=CPU clock cycles for a program*clock cycle time

Power Wall

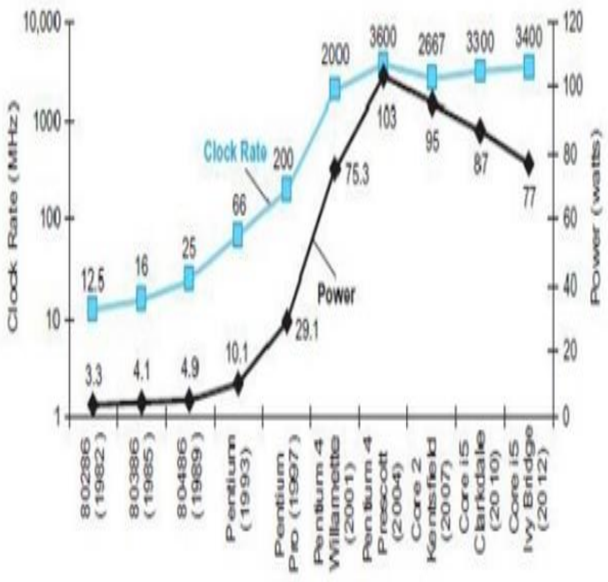


Fig.1.3 Clock rate and power for Intel x86 microprocessors

- The dominant technology for integrated circuits is called CMOS (complementary metal oxide semiconductor).
- For CMOS, the primary source of energy consumption is so-called dynamic energy— that is, energy that is consumed when transistors switch states from 0 to 1 and vice versa.
- The dynamic energy depends on the capacitive loading of each transistor and the voltage applied.

Instructions

Instruction format

- The instruction format of an instruction is usually depicted in a rectangular box symbolizing the bits of the instruction as they appear in memory words or in a control register.
- An instruction format defines the layout of the bits of an instruction, in terms of its constituent parts.
- The bits of an instruction are divided into groups called fields. The most common fields found in instruction formats are
- An operation code field that specifies the operation to be performed.
- An address field that designates a memory address or a processor register.
- A mode field that specifies the way the operand or the effective address is determined

Instruction Format

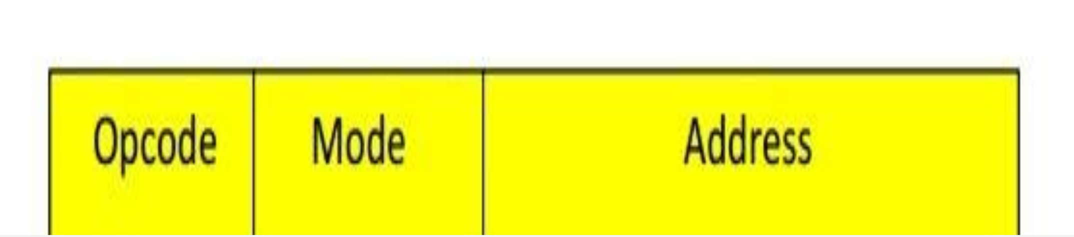
•Other special fields are sometimes employed under certain circumstances.

• The operation code field of an instruction is a group of bits that define various processor operations, such as add, subtract, complement and shift.

•Address fields contain either a memory address field or a register address.

•Mode fields offer a variety of ways in which an operand is chosen.

•



Instructions

Instruction format

There are mainly four types of instruction formats:

- · Three address instructions
- · Two address instructions
- · One address instructions
- · Zero address instructions

Three address instructions

- Computers with three address instructions formats can use each address field to specify either a processor register or a memory operand. The program in assembly language that evaluates X=(A+B)*(C+D) is shown below, together with comments that explain the register transfer operation of each instruction.

Add R1, A, B **R1ßM [A] + M [B]**
Add R2, C, D **R2ßM [C] + M [D]**

Nadar Saraswathi College of Engineering & Technology

Instructions

Two address instructions

- Two address instructions are the most common in commercial computers. Here again each address field can specify either a processor register or a memory word. The program to evaluate X= (A+B)*(C+D) is as follows:

MOV	R1, A	R1←M [A]
ADD	R2, B	R1←R1 + M [B]
MOV	R2, C	R2←M [C]
ADD	R2, D	R2←R2 + M [D]
MUL	R1, R2	R1←R1 * R2
MOV	X, R1	M [X]←R1

Instructions

One address instructions

- One address instructions use an implied accumulator (AC) register for all data manipulation. For multiplication and division there is a need for a second register. However, here we will neglect the second register and assume that the AC contains the result of all operations. The program to evaluate X= (A+B)*(C+D) is

LOAD	A	AC←M [A]
ADD	B	AC←AC + M [B]
STORE	T	M [T]←AC
LOAD	C	AC←M [C]
ADD	D	AC←AC +M [D]
MUL	T	AC←AC * M [T]
STORE	X	M [X]←AC

Instructions

Zero address instructions

- A stack organized computer does not use an address field for the instructions ADD and MUL. The PUSH and POP instructions, however, need an address field to specify the operand that communicates with the stack. The following program shows how X=(A+B)*(C+D) will be written for a stack organized computer.(TOS stands for top of stack.)

PUSH	A	TOS←A
PUSH	B	TOS←B
ADD		TOS←(A + B)
PUSH	C	TOS←C
PUSH	D	TOS←D
ADD		TOS←(C+D)
MUL		TOS←(C+D) * (A+B)
POP	X	M[X]←TOS

ADDRESSING MODES

- To perform any operation, the corresponding instruction is to be given to the microprocessor. In each instruction, programmer has to specify 3 things:

- Operation to be performed.
- Address of source of data.
- Address of destination of result
- The method by which the address of source of data or the address of destination of result is given in the instruction is called Addressing Modes

IMPLEMENTATION OF VARIABLES AND CONSTANTS

- Variables and constants are the simplest data types and are found in almost every computer program. A variable is represented by allocating a register or a memory location to hold its value. Thus, the value can be changed as needed using appropriate instructions.

ADDRESSING MODES

Register addressing mode -

- The operand is the contents of a processor register; the name (address) of the register is given in the instruction.

Example: **MOVE R1,R2**

This instruction copies the contents of register R2 to R1.

Absolute addressing mode

- The operand is in a memory location; the address of this location is given explicitly in the instruction. (In some assembly languages, this mode is called Direct.)

Example: **MOVE LOC,R2**

This instruction copies the contents of memory location of LOC to register R2.

ADDRESSING MODES

Immediate addressing mode -

- The operand is given explicitly in the instruction.

Example: **MOVE #200 , R0**

- The above statement places the value 200 in the register R0. A common convention is to use the sharp sign (#) in front of the value to indicate that this value is to be used as an immediate operand.

Indirect addressing mode

- The effective address of the operand is the contents of a register or memory location whose address appears in the instruction.

Example **Add (R2),R0**

- Register R2 is used as a pointer to the numbers in the list, and the operands are accessed indirectly through R2.
- The initialization section of the program loads the counter value n from memory location N into Rl and uses the Immediate addressing mode to place the address value NUM 1, which is the address of the first number in the list, into R2.

MIPS addressing

Instruction fetch cycle (IF):

IR = Mem[PC];

NPC = PC + 4; Operation:

- Send out the PC and fetch the instruction from memory into the instruction register (IR). Increment the PC by 4 to address the next sequential instruction.

IR - holds instruction that will be needed on subsequent clock cycles

Register NPC - holds next sequential PC.

MIPS addressing

Execution/effective address cycle (EX):

- ALU operates on the operands prepared in the prior cycle, performing one of four functions depending on the MIPS instruction type.

Memory reference:

ALUOutput = A + Imm;

Register-Register ALU instruction:

ALUOutput = A func B;

Memory access/branch completion cycle (MEM):

LMD = Mem[ALUOutput] or

Mem[ALUOutput] = B;

Operation:

- Access memory if needed.
- Instruction is load-data returns from memory and is placed in LMD (load memory data)
- Instruction is store-data from the B register is written into memory