KCA-105

Sub:- Computer Architecture or organisation :-

* Introduction to computer organisation :-

(I) Computer organisation refers to the operational units and their interconnection.

(II) It Implements the provided computer architecture.

(III) Organisation is how features are implemented.

- Control singles, interfaces, memory technology.
- for Ex- Is their are hardware multiply unit or is it done by repeated edition.

(IV) Computer architecture is a Blue print for design or implementation of a computer system.

(V) It provides the functional details or behaviour of a computer system or comes before computer organisation

Note:-
- Computer organisation deals with how to do ?
- Computer architecture deals with what do to ?

* Difference between computer architecture and computer organisation –

Computer architecture	Computer organisation
• It is concern with the way hardware components are connected together to form a computer system.	• It is concern with the structur and behaviour of a computer system as seen by the user
• Acts as the interface between Hardware and Software.	• Deals with the component of a connection in a system.
• It helps us to understand the functionality of a System.	• It tells us how excatly all the the units in the system are arranged and interconnected.
• A programmer can view architecture interm of instruction addressing mode and instruction.	• Organisation involves physical components & circuits, design, address, signal, peripherals.

• A programmer can view architecture in term of instruction addressing modes and most register.	• Organisation expresses the realisation of architecture.
• While designing a computer System architecture is considered first	• An organisation is done on the basis of architecture.
• It deals with high level design issues.	It deals with low design issue.
• Architecture involves logic (instruction sets, addressing modes, data types, cache memory)	Organisation involves physical components (circuits, design, address, signal, peripherals).

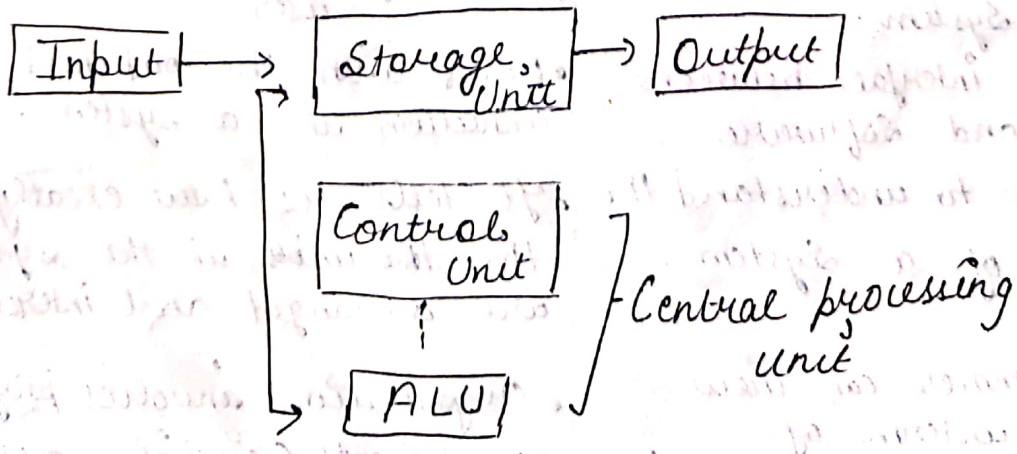
* Functional Units of Digital System :-

(I) A general purpose computer system is the best known example of a digital system.

(II) Other Ex- include telephone switching exchanges, Digital voll meters, Digital counters, Electronic calculator and digital display etc.

(III) A compuder consists of 5 main components -

1. Input
2. Output
3. Memory
4. ALU
5. CU



* Input Unit :-

(i) It is used by the computer to read the Data.

(ii) The most commonly used input devices are Keyboards, mouse, joystick, trackballs, microphones etc.

(iii) Whenever a key is pressed the crossponding letter or digit is autometically translated into its corrosponding Binary code and transmited over a cable to either the memory or the processor.

* Control Unit :-

(i) It is a component of a computer CPU the coordinates the operations of the processor.

(ii) It tells the computer memory, ALU and I/O devices how to responds to a programs instruction.

(iii) The control unit is also known as the nerve centre of a computer System.

* Central processing Unit :-

(i) It is commonaly known as CPU can be reffered as electronic cercuitry within a computer.

(ii) It carries out the instructions given by a computer program by performing the basic Airthmetic, logical, control and I/O operations specify by the Instruction.

* Memory Unit :-

(i) It is reffered as the storage area in which programs are kept which are running and that contains data needed by the running programs.

(ii) The memory unit can be categrized as primary memory and Secondary memory.

- **Primary Memory :-** Also known as volatile form of memory means when the computer is shut down anything contained in RAM is lost.

  Ex - RAM - Quickly excess
  ROM -

- **Secondary Memory :-** It is used when a large amount of data and program have to be stored for a long term basis. Also known as non-volatile memory. i.e. Data is stored permanently irrespective of shut down for Ex- Magnetic disk, Magnetic tap or optical disk.

* **Arithmetic and Logical unit :-**

(I) Most of all the AL operation of a computer are exicuted in the Arithmetic and logical unit.

(II) It performs arithmetic operations like - addition, substraction multiply, Devision or also the logical operation like AND or NOT operation -

* **Output Unit :-**

(I) Output devices displays information in a way that the user can understand.

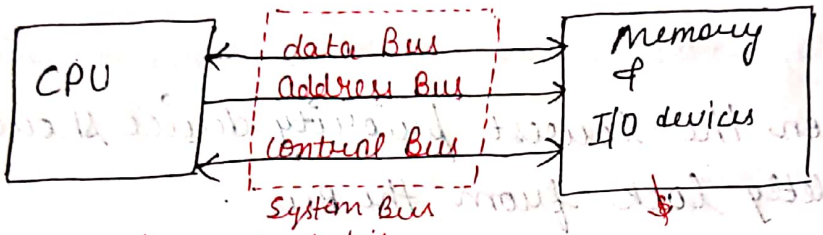
(II) These devices display information that has been held ~~held~~ on generated with in a computer.

(III) The most common example of an output devices is a monitor.

* Interconnection between functional components :-

- The majors part of micro computer are CPU, memory, I/O devices/unit -
- To connect these parts together through three sets of parallel lines called Bus.
- There are Three types of Buses — (connects all internal computer components to the CPU main memory)

i) Address Bus
ii) Data Bus
iii) Control Bus



* One line transfer only 1 bit.

* ~~Bus~~ address Bus is undirectional because CPU does not store anything it have no memory.

Set of Parallel lines

Address generated by CPU called logical address.

* Bus arbitration :-

- A device that initiate data transfer on the Bus at any given time is called bus master.
- In a computer system their may be more than one bus master such as DMA. Direct memory access as a DMA controller or processor etc.

(III) These devices share the system bus and when a current master relinquishes another Bus can aquired the control of the process.

(IV) Bus arbitration is a process by which next device becomes the controller by transffering Bus mastership to another Bus.

(V) Bus arbitration schemes usally tried to Balance two factors –

- **Bus priority** :- The highest priority device should by serviced first.
- **fairness** :- Even the lowest priority device should never be completly lock from the Bus.

Types of Bus arbitration

Central arbitration

- In centralized arbitration a single Bus are arbiter performs the required arbitration.
- The bus arbiter may be the processor or a separate controller connected to the Bus.

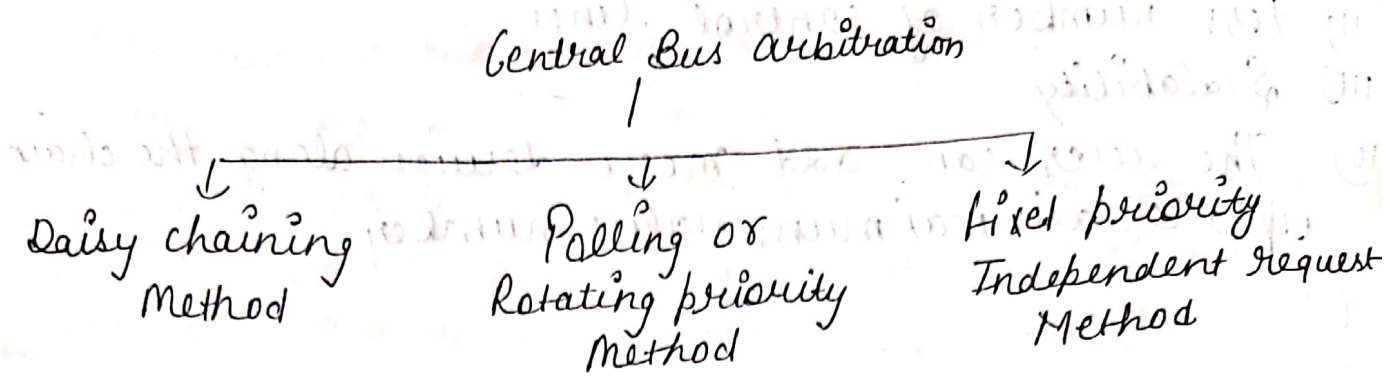
Distributed arbitration

- In distributed arbitration all devices participate in the next Bus master.

* **Centralized Bus arbitration** :-

(I) In Centralized arbitration a single Bus arbiter performs the required arbitration.

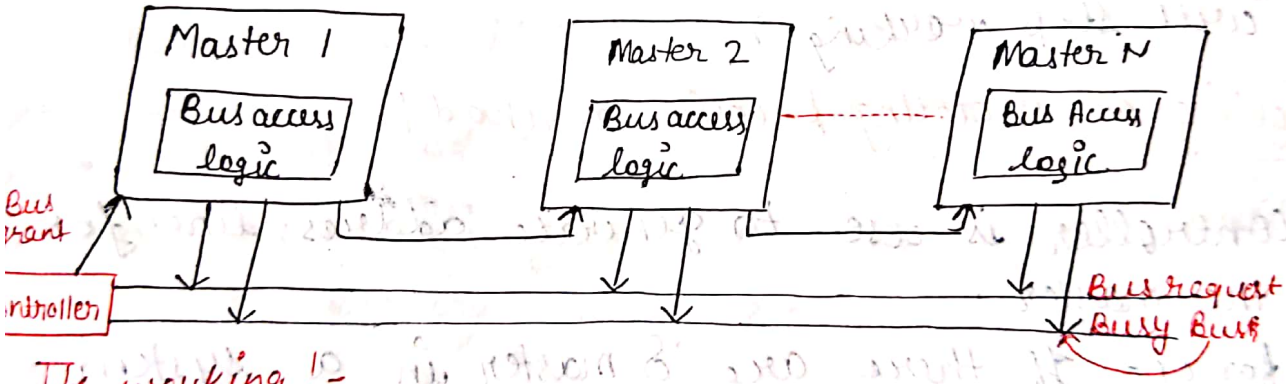
(11) There are three different arbitration schems that used the centralized Bus arbitration approach.



(1) Daisy chaining Method :-

(I) It is a simple and cheaper method.

(II) All the masters used the same line for making Bus request.



Its working :-

- All Bus masters use the same line for bus request.
- If the Bus busy line is inactive the bus controller gives the bus grant.
- Bus grant single is propegated serialy through all master starting from nearest one.
- The bus master which requires system bus stops this single, activates the bus busy line and takes control of System bus.

Advantages!-

(i) Simplicity
(ii) Less number of control lines
(iii) Scalability
(iv) The user can add more devices along the chain upto certain maximum value number.

Disadvantages!-

(i) Priority assign to a device depends on the position of the bus master.

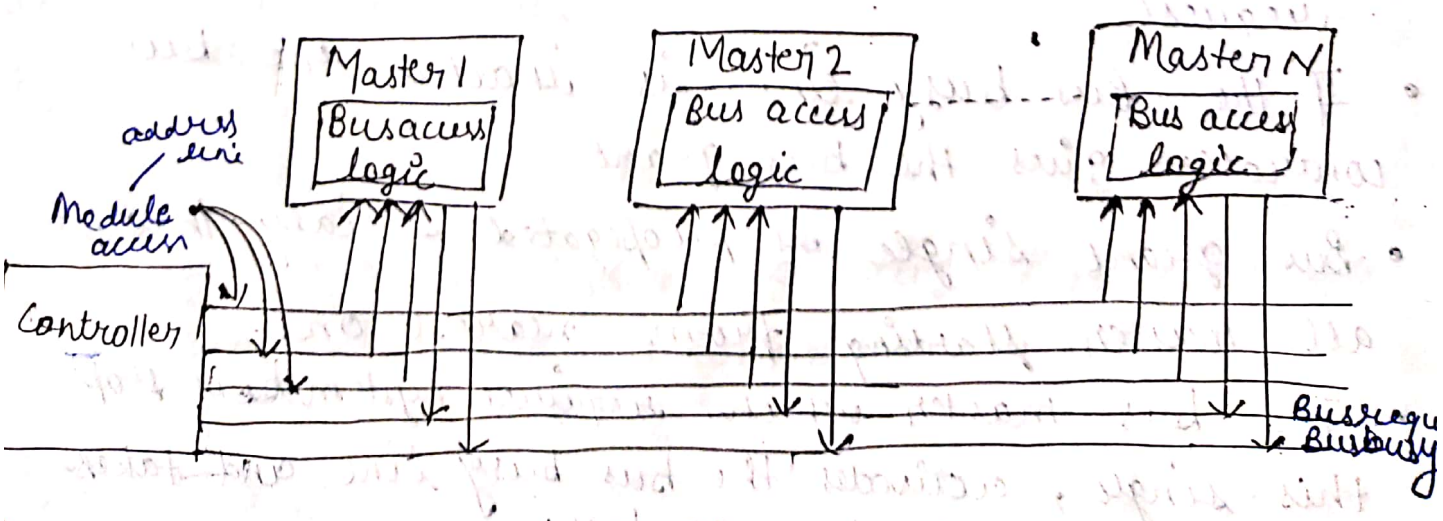
(ii) Propagation delay arises due to serially granding of bus.

(iii) If one device fails then the entire system will stop working.

* Polling or rotating priority method!-

(i) Controller is use to generate address lines for the master.

(ii) for Ex- If there are 8 master in a system atleast three address lines are required.



Working!-

(i) All bus master use the same lines for bus request.

(ii) Controller generates Binary address for the master (To connect 8 bus master we need three address line $2^3 = 8$)

(iii) In response to a bus request the controller poles the bus master by sending a sequence of bus masters address on address lines.

(iv) When requesting master recongises its address it activates the bus busy line and takes control of the bus.

Advantages!-

(i) This method does not favor any particular device and processor.

(ii) Simple method.

(iii) If one device fails than the System will not Staping working.

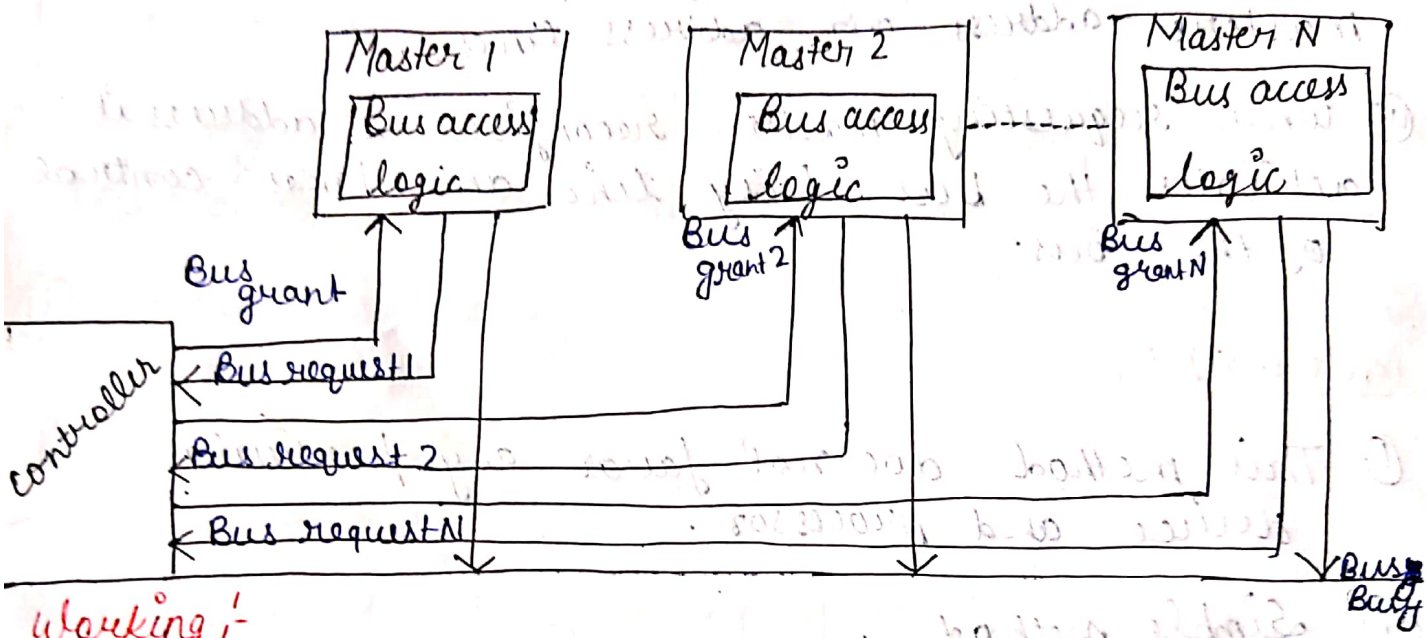
Disadvantages!-

(i) Adding Bus masters is difficult as it increase the number of address lines of the circuit.

(iii) Fixed Priority or independent request method :-

(i) Each bus master has its own Bus request and a grant.

(ii) The built in priority decoder selects the highest priority request and asserts the system.



Working :-

(i) All bus masters have there individual bus request and bus grant lines.

(ii) The controller thus knows which master has request so bus is granted to that master.

(iii) Priority of the masters are predefined so bus request the bus is granted based on the priority provided line is not active.

(iv) The controller consists of encoder or decoder logic for priorities.

* Advantages :-

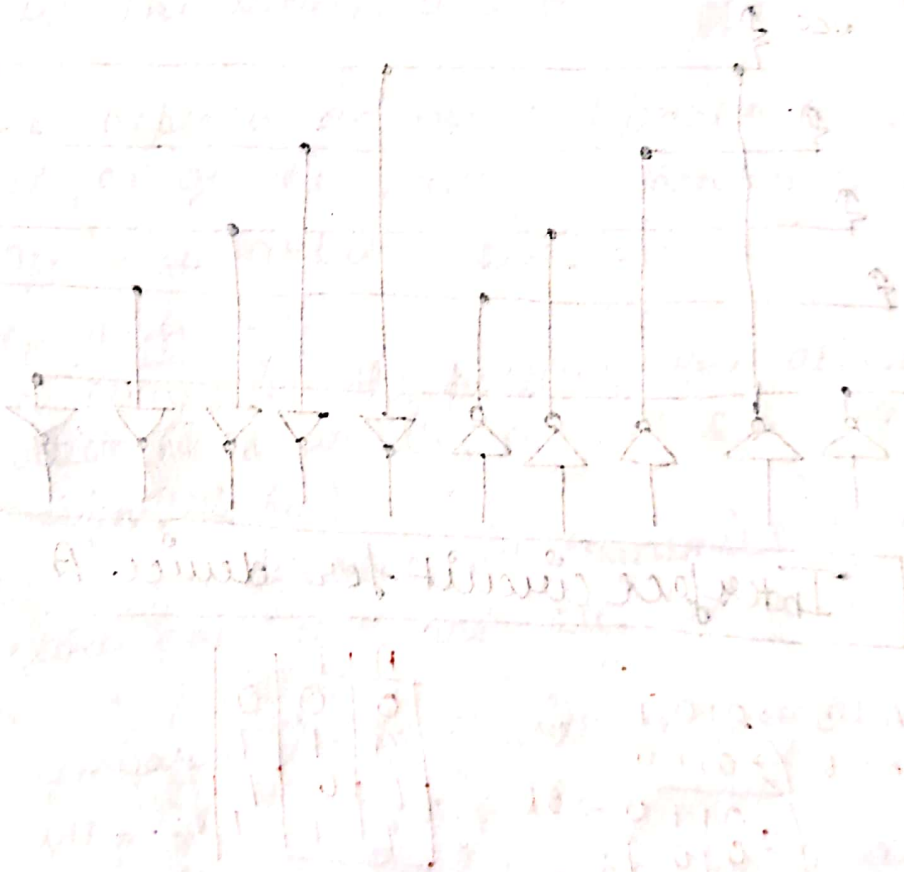
(I) Fast response

(II) Speed independ of number of devices connected.

* Disadvantages :-

(I) Hardware cost to high as a large number of control line is required.

(II) Number of control lines required is more therefore connecting large no. of bus master is difficult.



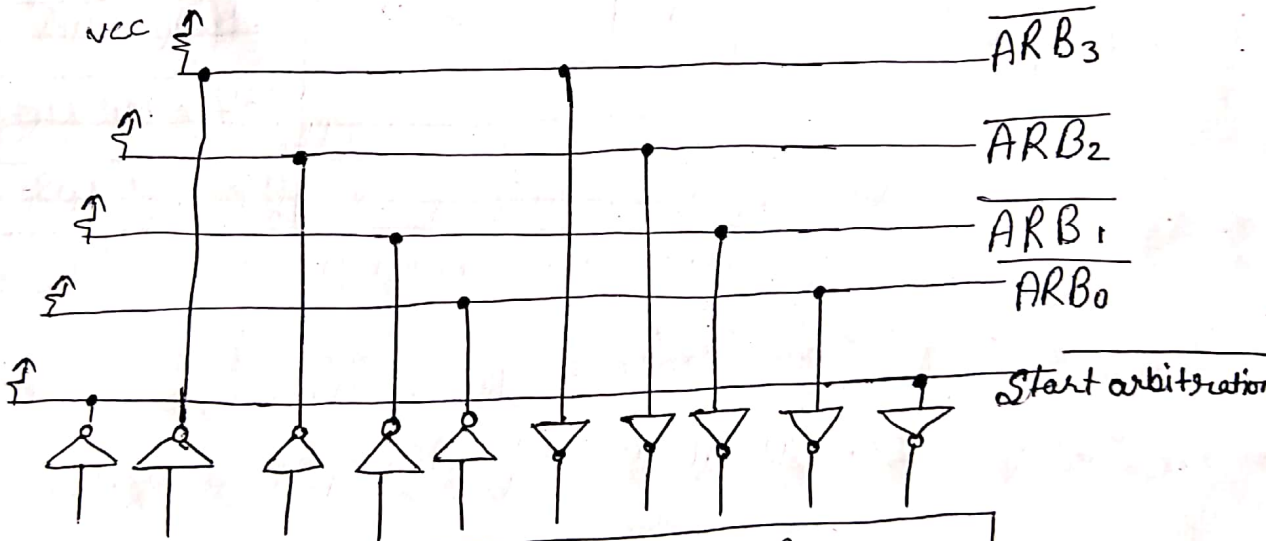
* Distributed arbitration :-

(i) All devices waiting to use the bus share the responsibility of carrying out the arbitration process.

(ii) Arbitration process does not depend on a central arbiter and hence distributed arbitration has hier rilability.

(iii) Each device is assigned a 4 bit Id number.

(iv) All the devices are connected using 5 lines, 4 arbitration lines to transmit the id, and one line for the start arbitration.



Interface circuit for device A

A	B	
0	0	0
0	1	1
1	0	1
1	1	1

Device A = 5 → 0101
B = 6 → 0110

0110 → OR
0101

0111 0
0111 0

0110

Working -

* To request the Bus a Device :-

(I) Asserts the Start arbitr signal signal.

(II) Places its 4 bit Id number on the arbitration lines.

(III) The Pattern that appear on the arbitration lines its the logical or of all the 4 bit devices id place on the arbitration lines.

(IV) deviceA has the Id 5 and wants to request the bus transmits the pattern 0101 on the arbitration lines.

(V) device B has the id 6 and want to request the Bus transmits the pattern 0110 on the arbitration lines.

(VI) Pattern the appears on the arbitration lines is the logical or of the patterns. Patterns 0111 appears on the arbitration lines.

Arbitration process :-

(I) Each device compares the pattern that appears on the arbitration lines to its own Id starting with MSB (Most Significic Bit)

(II) If it detects a difference its transmits 0's on the arbitration lines for that and all lower bit position.

(III) Device A compares its Id 5 with a pattern 0101. to pattern 0111.

(IV) It detects a difference at bit position 0 as a result it transmit a pattern 0100 on the arbitration line.

(V) The pattern that appears on the arbitration lines is the logical or of 0100 and 0110 which is 0110.

(VI) This pattern is the same as device id of B enhence B has and Hence we has won the arbitration.

Introduction to Bus :-

(I) Bus is a sub system that is used to transfer Data and other information between devices.

(II) Various devices in computer such as memory, CPU, I/O etc communication with each other through buses.

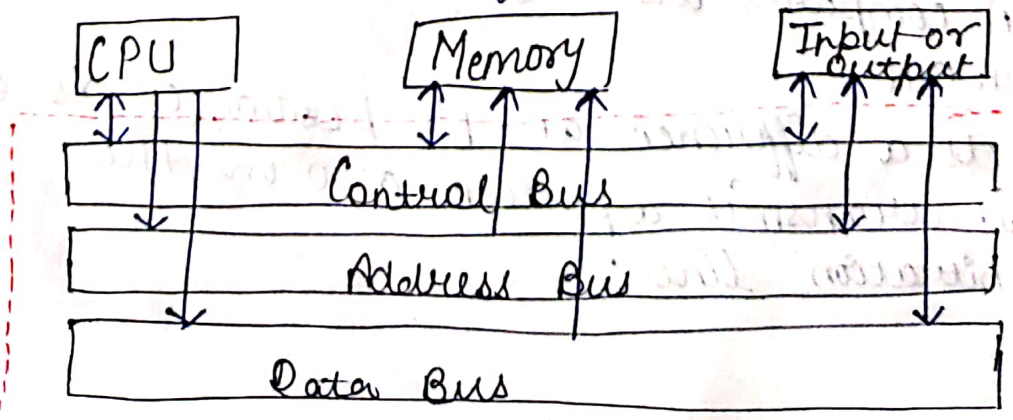
(III) In general a bus is said to as a communication way connecting two or more devices.

(IV) A key characteristics of a bus is that it is a shared transmission mediam because multiple devices are attached to a bus.

(V) A bus consists of multiple communication path ways or lines which are either in a form of wire ore metallines etched in a card or board (Printed circuit board)

(VI) Each line is capable of transmitting Binary 1 and Binary 0.

(VII) Computer System contains a number of different buses that provides path ways between components at various levels of computer System hierarchy



Three types of buses are -

(I) Address Bus
(II) Data Bus
(III) Control Bus

Data Bus :-

(I) Data lines provides a path for moving data b/w System modules.

(II) It is bidirectional which means data lines are used to transfer data in both directions.

(III) for ex - CPU can read data on these lines from memory as well as send data out of these lines to a memory location or to a port.

(IV) In any Bus the number of lines in data lines are either 8, 16, 32 or more depending on size of Bus, these lines collectively are called as data bus.

(II) Address Buses :-

(I) Address lines are collectively called as address Bus.

(II) In any bus the number of lines in addresses are usally 16, 20, 24 or more depending on types of architecture of Bus.

(III) On these lines CPU Sends out the address of memory location on the I/O Port.

(IV) The communication is one way ~~that~~ i.e. the address is send from the CPU to memory and I/O Port but not memory or I/O Port send address to CPU on that lines ehence these lines are Unidirectional.

3- Control Bus:-

(i) Control lines are collectively called as control Bus.

(ii) They are used by CPU's for communicating with other devices within the computer.

for ex- CPU sends signals on the control Bus to enable the outputs of addressed memory devices and port devices.

(iii) Typical control line signals are -
   (i) Memory read, memory write
   (ii) I/O read, I/O write
   (iii) Bus request
   (iv) Bus grant

* Operation of Bus :-

The operation of Bus is as follow :-

1) If one module wishes to send data to another it must do two things
   - obtained the use of module bus
   - transfer data to the bus

If one module wishes to request data from another module it must
   - obtained the use of Bus
   - transfer a request to other module over the appropriate control or address line and then it must wait for that second module to send the data.

Types of Buses :-

(i) There are variety of Buses But some of widly use bus are —

(1) System Bus :-

* A Bus that connects major computer components (Processor, memory, I/O) is called a System Bus.

(ii) It is signal computer bus among all buses that connects all these components of a computer system.

(iii) It is the only bus in which data lines address and control lines all are present, it is also known as front side bus.

(iv) It is faster then peripheral bus (PCI, ISA) etc. but slower than back side bus.

* (2) Peripheral Bus (I/O Bus or External bus) :-

(i) peripheral Bus also known as I/O Bus.

(ii) It is data path way that connects peripheral ~~bus~~ devices to the CPU.

(iii) In computing a peripheral bus is a computer bus design to support computer peripheral like- printers, harddrives etc.

(iv) The PCI and USB buses are commonly used peripheral buses and are today used in commonly PC's.

* PCI (Peripheral component Interconnect) :-

(i) PCI Bus connects the CPU and expansion board such as modem cards, network card and sound card.

(ii) These expansion boards are normally plugged into expansion slot on the motherboard.

(iii) That is ~~obg~~ by PCI bus is also known as expension or external bus.

* USB (Universal Serial bus) :-

(i) Universal serial bus is use to attach USB devices like Pendrive etc to CPU.

* (3) Local Bus :-

(i) Local Bus are the traditional I/O (peripheral bus) such as ISA, MCA or EISA Buses.

- ISA - Industry Standard Architecture Bus -

(I) The ISA bus permit Bus mastering i.e it unable to peripheral connected directly to the bus to communicate directly with other peripheral without going to processor.

(II) One of the consequences of Bus mastering is direct memory access

(III) Upto end the 1990's almost all pc's were equipped with ISA Bus it was progressive replaced by the PCI bus which offered a better performancing.

- MCA - Micro channel architecture -

(I) It is an improve proprietary bus designed by IBM in 1987 to be used in there PS/2 lines by computer.

(II) This 16 to 32 bit bus was incompatable with the IAS bus ~~but~~ and put reach throughput~~cost~~ of 20 Mbps

- EISA - Extended Industry Standard architecture -
  - It was developed in 1988 by a consortium team.
  - The EISA bus use connectors ~~th~~ that were same size as the ISA connectors with 4 rows of contacts instead of 2(two) for 32 bit addressing.

Bus Structure :-

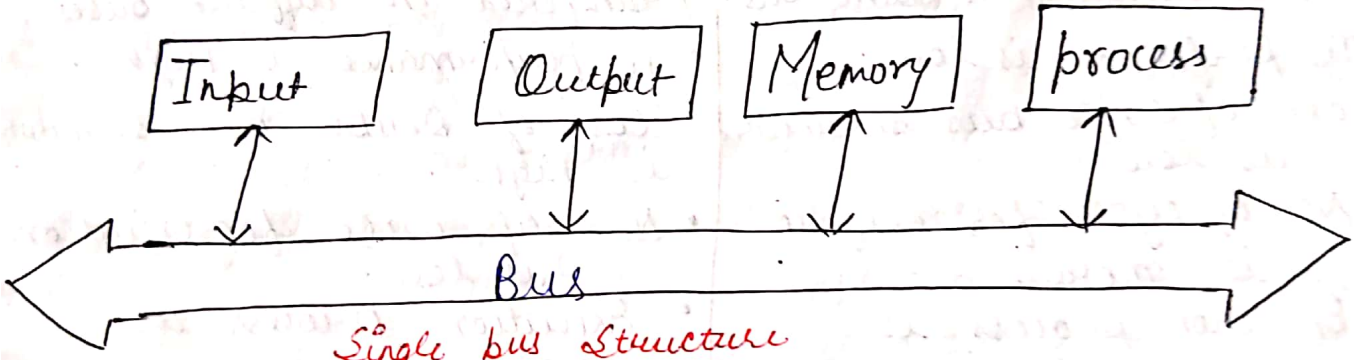
① Single Bus Structure :-

(I) All units are connected to a single bus so it provides the sole means of interconnection.

(II) It has advantages of simplicity, no cost, and many devices can be connected.

(III) It has disadvantages of limited speed, since usually only two units can participate in a data transfer at any one time therefore and arbitration system is required and those units will be forced to weight

(IV) Bus control lines are use to arbitrate multiple requests for the use of the bus.



Single bus Structure

② Double Bus Structure :-

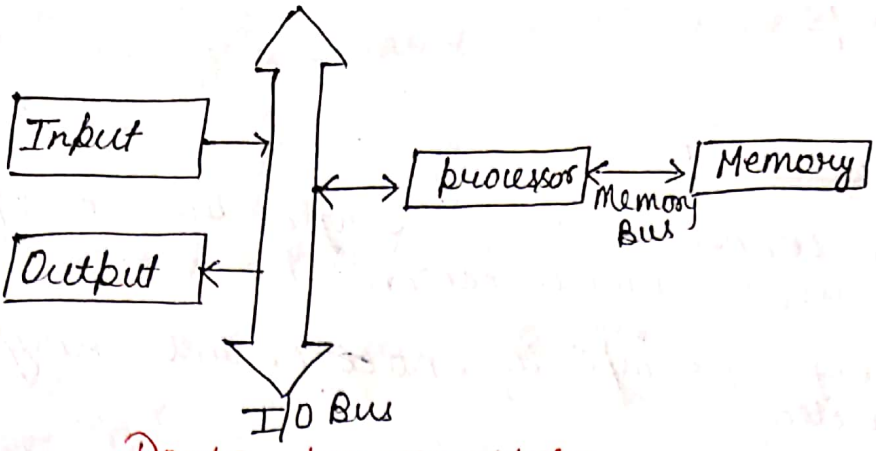
(I) Double Bus structure is used to overcome the bottleneck of single bus structure.

(II) It uses two buses
- One bus is used to fetch instructions
- Other is used to fetch data required for exiution.

(III) In the first configuration the processor is placed between the Input output unit and the memory unit.

(IV) The processor is responsible for any data transfer between the I/O unit and the memory unit.

(V) The processor acts as a messanger.



Double bus structure

* Single bus V/s Double bus :—

Single bus	Double Bus
* One common bus is used for communication between peripheral or processor.	• Two Buses are used one for communication from peripheral and other for processor.
* Instructions and data both are transfered in same bus.	• Instruction and data both are transfered in diffrent buses.
* Its performance is low.	• Its performance is High.
* Cost of Single bus structure is low.	• Cost of Double bus structure is high.
* No. of cycle for exicution is more.	• No. of cycle for exicution is less.
* Exicution process is slow.	• Exicution process is fast.
* No. of registers associated are less.	* No. of registers associated are more.
* At a time single operand can be read from bus.	• At a time two operand can be read.

* Introduction to Register :-

(I) Register is a very fast computer memory used to store data / instruction in execution.

(II) A register is a group of flip-flops with each flip-flop capable of storing 1 bit of information.

(III) A n-bit register has a group of n flip-flop and capable of storing binary information of n bits.

(IV) A register consists of a group of flip-flops and gates.

(V) The flip-flops hold the binary information and gates control when and how new information is transferred into a register.

(VI) Various types are register are available commericial

(VII) The simplest is one that consists of only flip-flops and with no external gates.

Some of the commonly used register are :-

(I) **Accumulator** :- This is the most common register used to store data taken out from the memory

(II) **General Purpose register** :- This is used to store data intermediate result during program execution it can be access by assembly programming.

(III) **Special purpose register** :- User do not access these register these register are for computer ~~regular~~ System.

① MAR (Memory address register), are these register that holds the address for memory unit.

① MDR/MBR (Memory Data / Buffer register) :- Store instruction and data recieve from the memory and send ~~from~~ to the memory.

- PC (Program counter) – Program counter point to the next instruction to be executed.
- IR (Instruction register) :- Holds the instruction to be executed.

Imp :-

Addressing modes :-

(I) Implied mode -

INCA
increament accumulator

(II) Immediate mode -

(III) Direct -

(IV) Indirect -

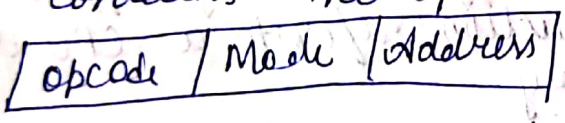
Addressing modes!- This different ways of specifying the location of an operand in an instruction are called as addressing modes.

Types of Addressing modes!-

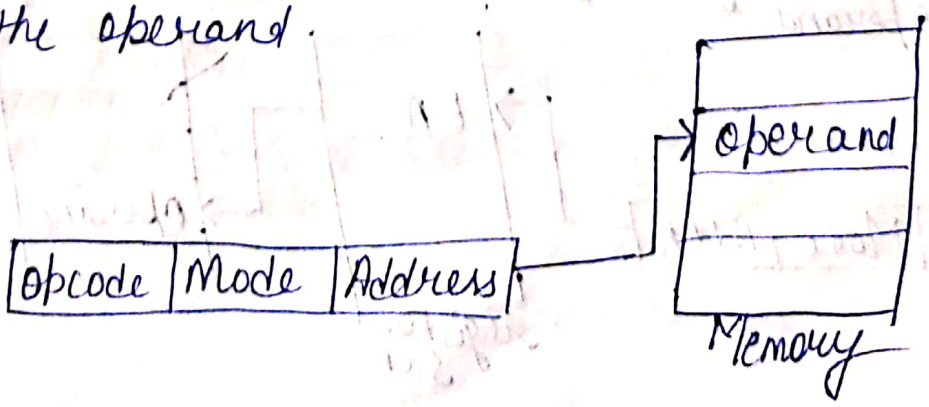
① Implied!-(Implict) The definition of the instruction itself specify the operands implicity.
- Zero address instruction are implied mode instruction
  ex- Increment, Decrement.

② Stack!- The operand is contained at the top of the stack.
- This instruction simply pops out two symbols contained at top of the stack.
- The addition of these two operand is performed.
- The result so obtained after addition is pushed again at the top of the stack.

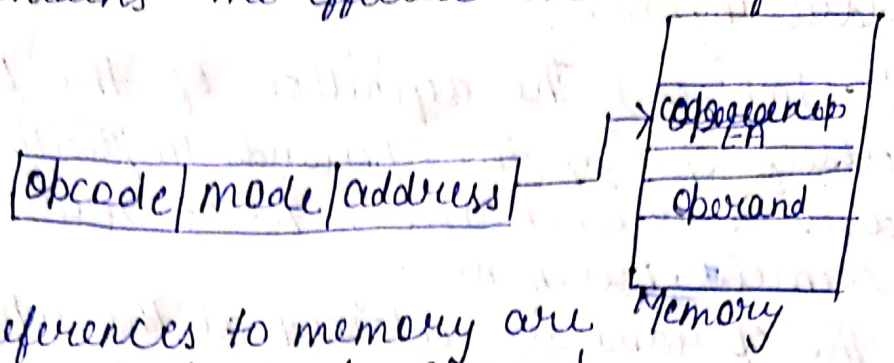
③ Immediate!-
- The operand is specified in the instruction explicity.
- Insted of address field, an operand field is present that contains the operand.



④ Direct addressing mode! Absolute-
- The address field of the instructions contain the effective address of the operand.
- Only one reference to memory is required to fetch the operand.



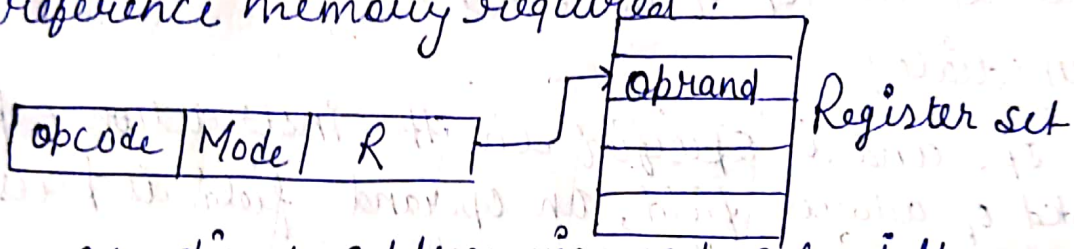
⑤ Indirect addressing mode - The address field of the instruction specifies the address of memory location that contains the effective address of the operand.



- Two references to memory are required to fetch the operand.

⑥ Register direct addressing mode :-

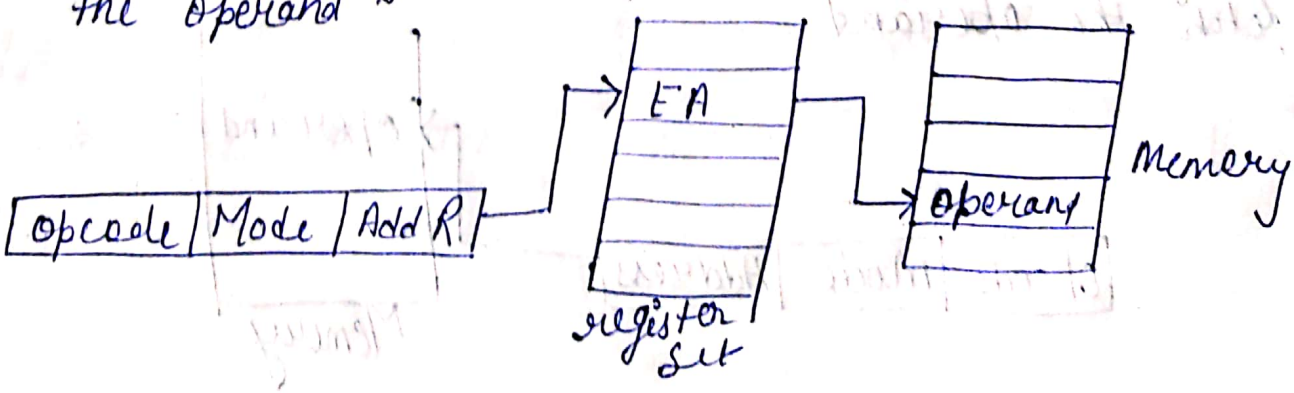
- The operand is contained in a register set.
- The address field of the instruction refers to a CPU register that contain the operand.
- No reference memory required.



- Same as direct addressing mode only difference is address field of the instruction refers to a CPU register instead of main memory.

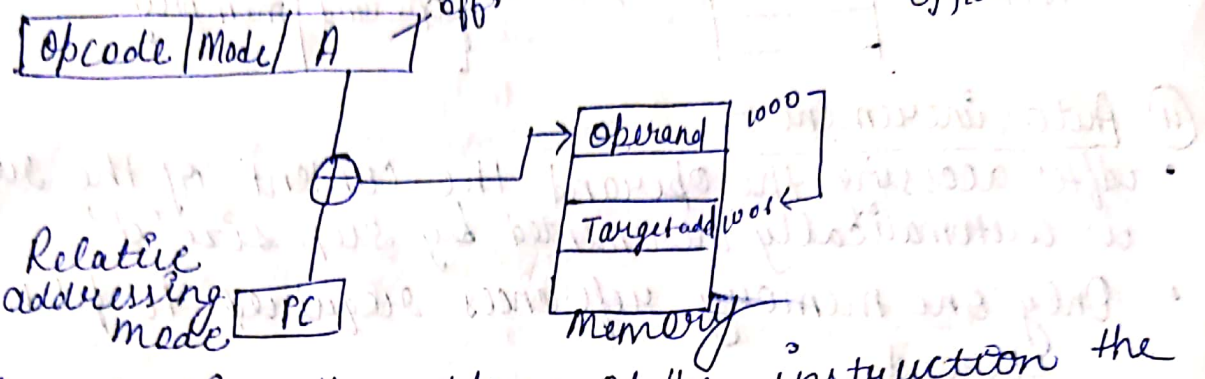
⑦ Register - Indirect addressing mode :- The address file of the instruction refers to a CPU register that contain the effective address of operand.

- Only one memory references is require to fetch the operand



(8) Relative addressing mode :- Effective address of operand is obtained by adding the content of program counter with the address part of the instruction.

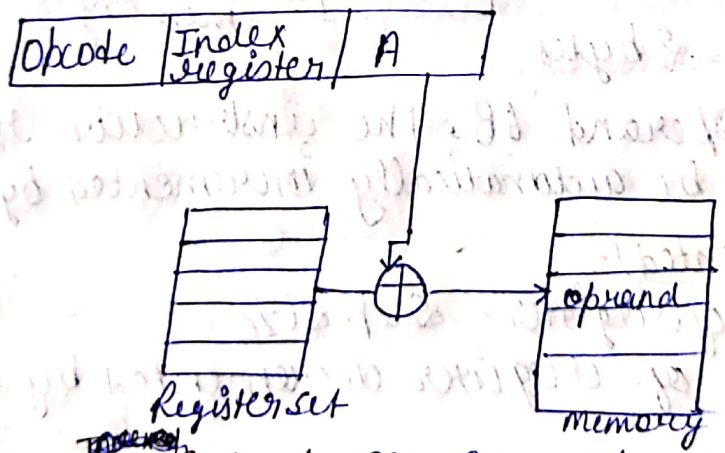
Effective address = Content of program counter + address part of the instruction



- After fetching the address of the instruction the value of program counter immediately increase.

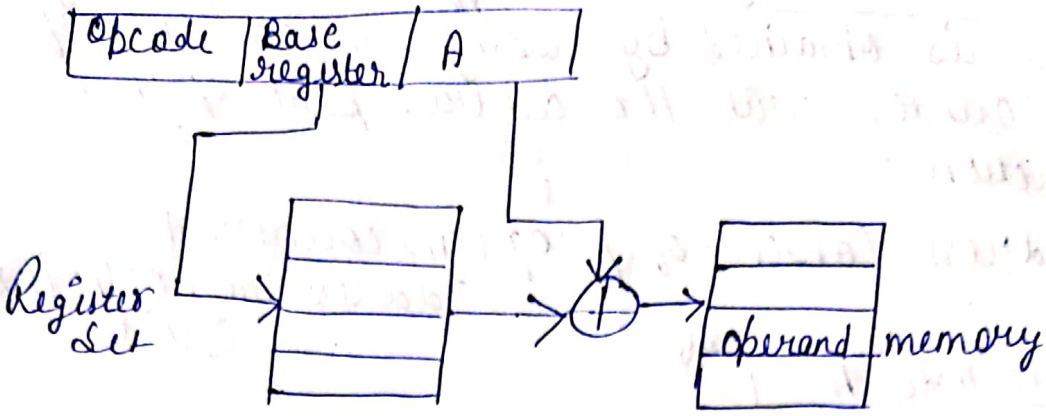
(9) Indexed addressing mode :- Effective address of the operand is obtained by adding the content of index register with the address part of instruction.

Effective = Content of Index register + address part of the instruction.



(10) Base register :- Effective address of the operand is obtained by adding the content of base register with the address part of instruction.

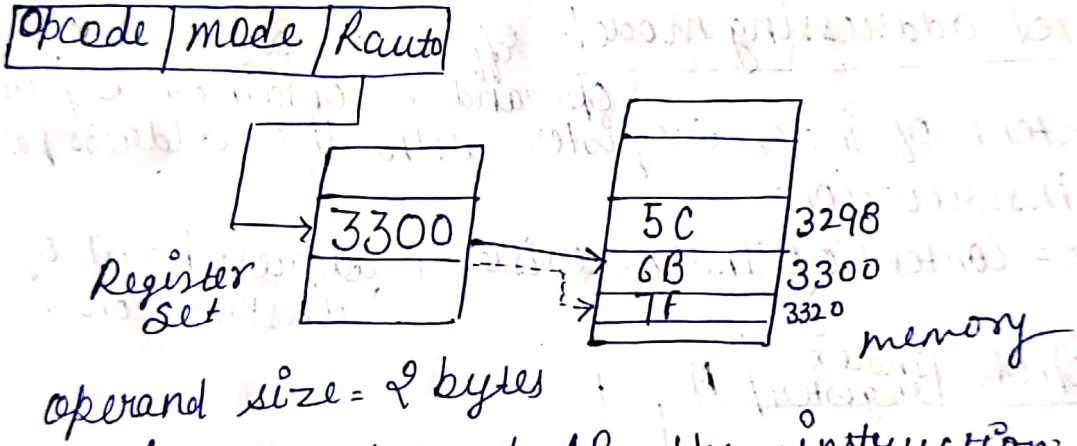
EA = Content of base R + address part of instruction



(11) Auto-increment –

- After accessing the operand the content of the register is automatically incremented by step size 'd'
- Only one memory references required to fetch the Operand

EA = Content of Register



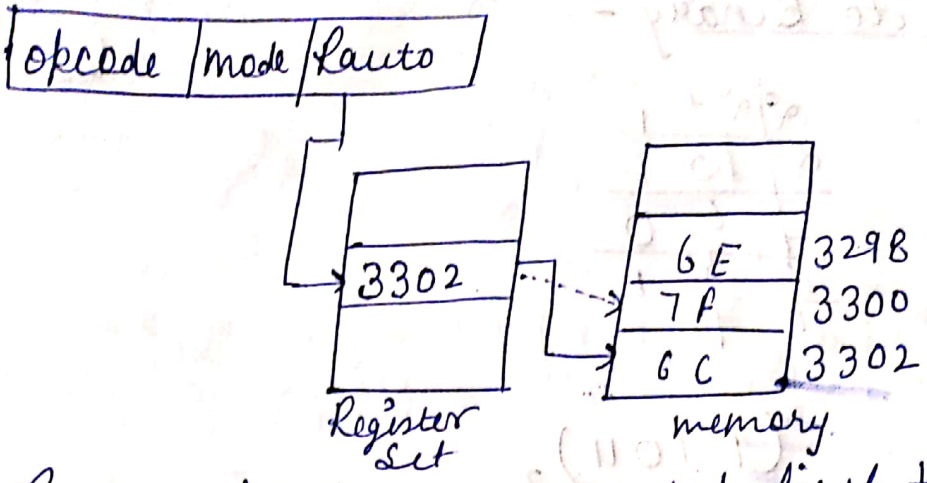
operand size = 2 bytes

- After fetching Operand 6B, the instruction register to Rauto will be automatically incremented by 2.

(12) Auto Decremented :-

EA = Content of register – Step size

- First the content of register decremented by step size 'd'
- Only one reference to memory is required to fetch the Operand.



- Rauto will be decremented first then update value.

* Applications of addressing modes :-

⊛ Immediate :- To initialize registers to a constant value.

* Register / Direct addressing mode :- To access static data To implement variables

* Register / Indirect :- To implement pointers because pointers are memory location that store address of another variable.

* Relative - for program relocation at run time i.e. for position independent code for branch type instruction.

* Index - For array implementation ara or array addressing.

* Base register - for relocation of program in memory even at run time.

* Auto increment / decrement :- for implementing loops.
- for stepping through arrays in a loop
- for push, pop.