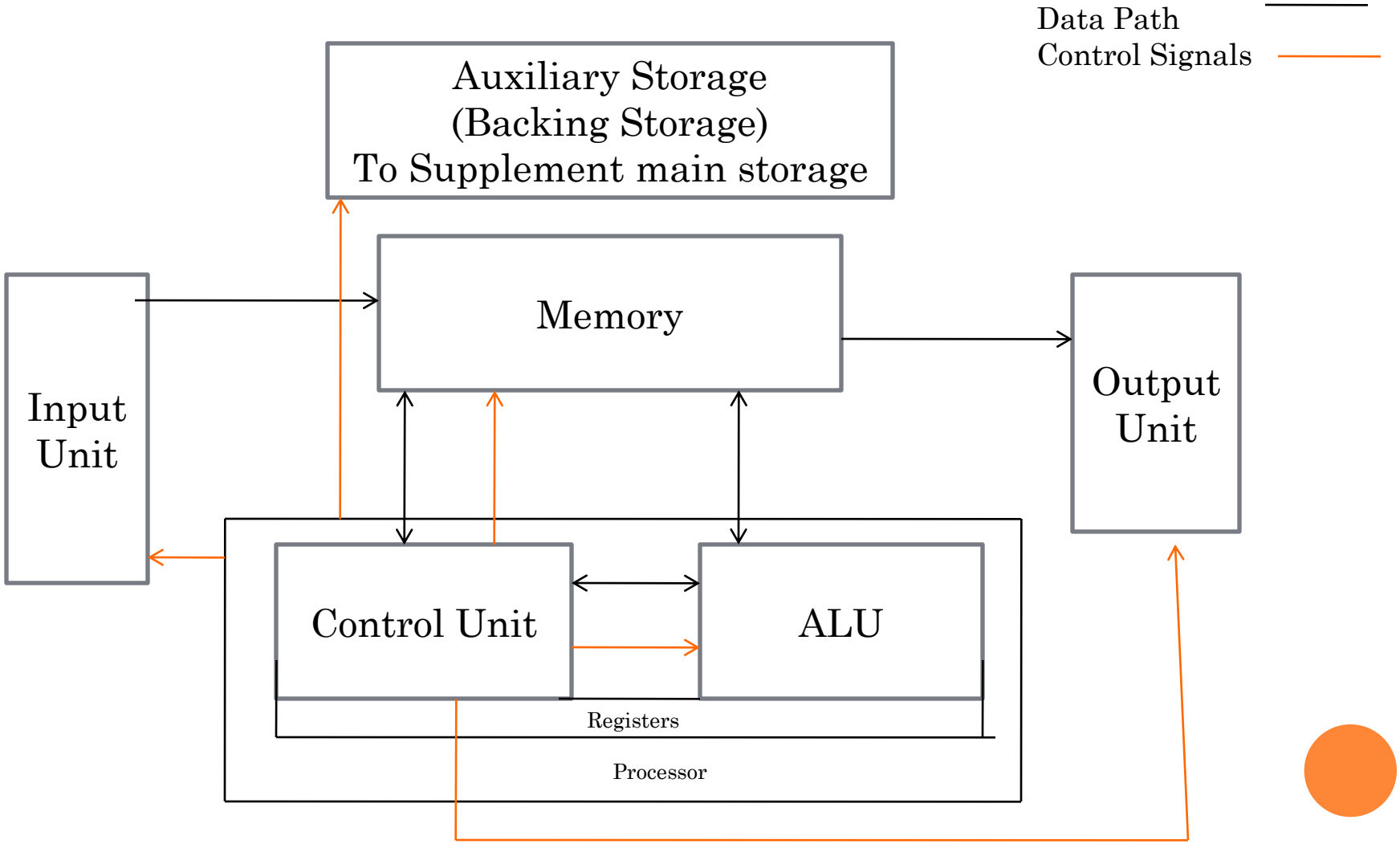
COMPUTER ORGANIZATION AND ARCHITECTURE

Introduction

- Just as buildings, each computer has a visible structure, referred to as its architecture.
- In computer science and engineering computer architecture is the practical art of selecting and interconnecting hardware components to create computers that meet functional, performance and cost goals and the formal modeling of those systems.
- The functional blocks in a computer are of four types:

  1. Central Processing Unit
  2. Memory
  3. Input Unit
  4. Output Unit

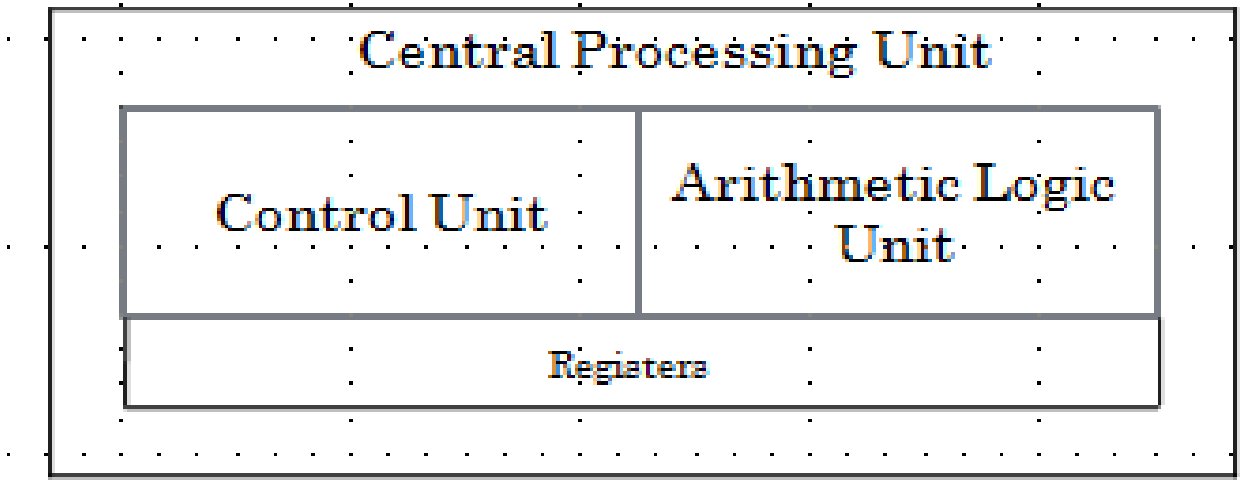
Data flow between CPU, memory and I/O devices



Central Processing Unit (CPU)

- The CPU or the microprocessor (or simply processor) is referred as the brain of a computer system.
- CPU consists of three main subsystems, the Control Unit (CU), the Arithmetic Logic Unit (ALU), and the Registers.
- Speed of the computer system is defined by the architecture of the processor being used.

Central Processing Unit (CPU) (Condt…)



Arithmetic Logic unit

- The ALU contains electronic circuits necessary to perform arithmetic and logical operations.
- The arithmetic operations are ADD, SUBSTRACT, MULTIPLY, DIVIDE, etc.
- The logical operations include COMPARE, SHIFT, ROTATE, AND, OR, etc
- The control unit analyses each instruction in the program and sends the relevant signals to all other units – ALU, Memory, Input unit and Output unit

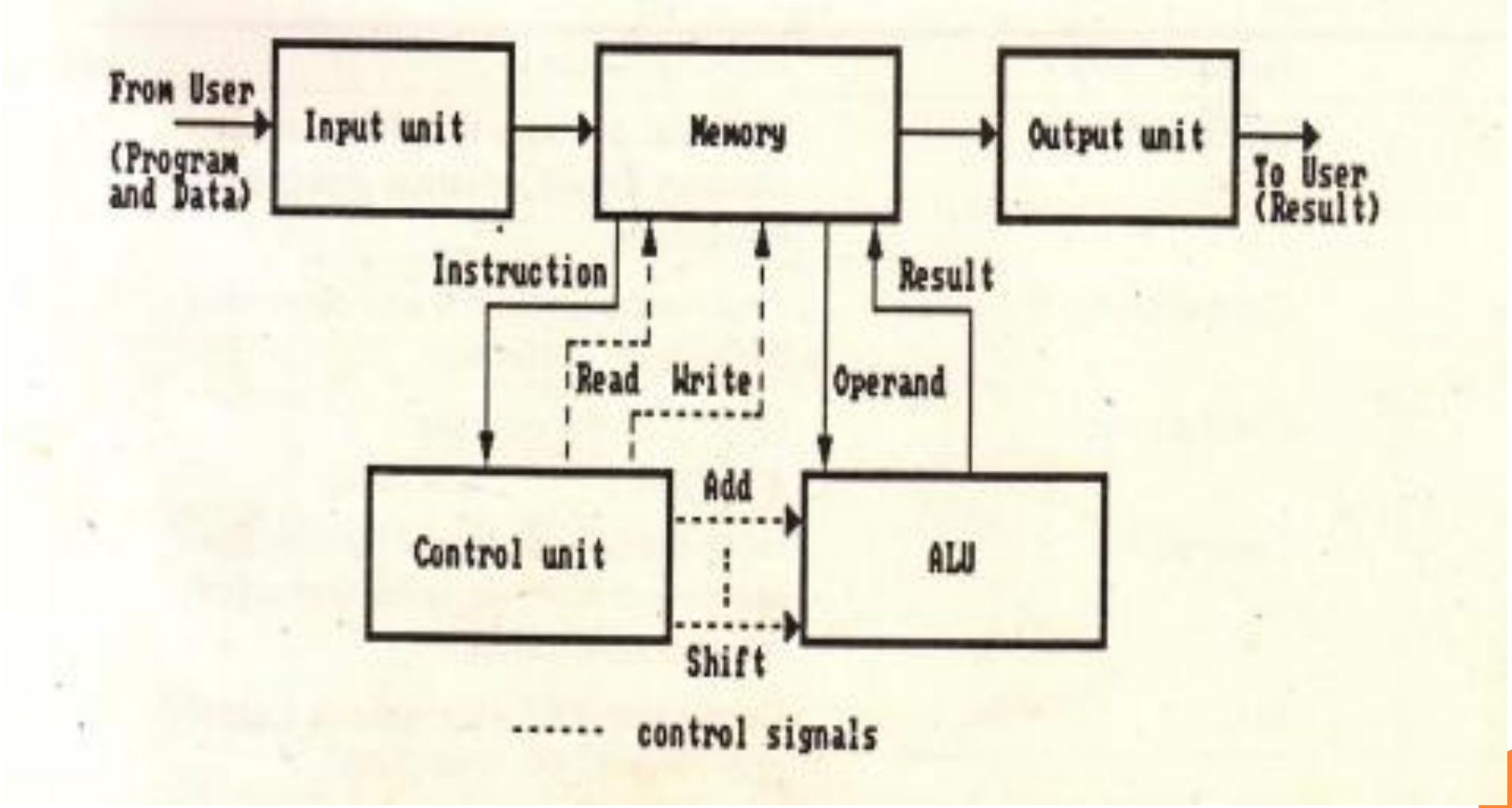
Communication inside a computer..

- A computer program consists of both instructions and data. The program is fed into the computer through the input unit and stored in the memory.
- In order to execute the program, the instructions have to be fetched from memory one by one.
- This fetching of instructions is done by the control unit.
- After an instruction is fetched, the control unit decodes the instruction.
- According to the instruction, the control unit issues control signals to other units.

Communication inside a computer..

- After an instruction is executed, the result of the instruction is stored in memory or stored temporarily in the control unit or ALU, so that this can be used by the next instruction.
- The results of a program are taken out of the computer through the output unit.
- The control unit, ALU and registers are collectively known as Central Processing Unit (CPU)

Communication inside a computer



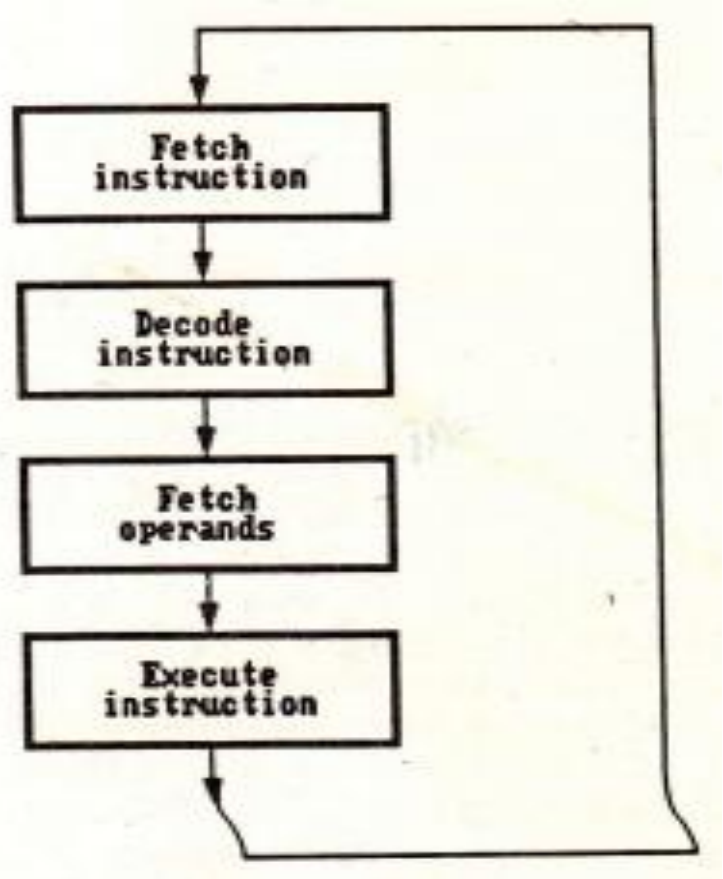
Interconnection Of Units

- A computer program consists of both instructions and data.
- The program is fed into the computer through the input unit and stored in the memory.
- In order to execute the program, the instructions have to be fetched from memory one by one and store it into registers (working memory) for processing.
- This fetching of instructions is done by the control unit.

Instruction cycle

- Instruction are fetched and executed by the control unit one by one. The sequences involved for the fetch of one instruction and its execution are known as instruction cycle.

Instruction Cycle



Instruction Cycle Steps

No.	Activity	Responsibility	Remarks/Actions
1.	Instruction Fetch	Control Unit	Send IAC contents to MAR, perform Memory Read Operation and get the instruction.
2.	Instruction Decode	Control Unit	Analyse the OPCODE and determine the type of instruction
3.	Operand Address Calculation	Control Unit	Determine the operand addresses
4.	Operands Fetch	Control Unit	Fetch the operands, one by one, from Memory or from Registers and supply the operands to ALU
5.	Execute	ALU	Do the required arithmetic or logical operation for the instruction
6.	Result Store	Control Unit	Store result in memory or in registers

Registers

- It is a special temporary storage location within the CPU.
- Registers quickly accept, store and transfer data and instructions that are being used immediately.
- To execute an instruction, the control unit of the CPU retrieves it from main memory and places it onto a register.
- The typical operations that take place in the processing of instruction are part of the instruction cycle or execution cycle.
- The instruction cycle refers to the retrieval of the instruction from main memory and its sub sequence at decoding.
- The time it takes to go through the instruction cycle is referred to as I-time.

Arithmetic Logic Unit (ALU)

- ALU performs all the arithmetic and logical functions.
- It performs arithmetic as well as logical functions.
- The speed of the computer system is defined by the architecture of the processor being used.

Control Unit

- It is responsible for directing and coordinating most of the computer system activities.
- It does not execute instructions by itself. It tells other parts of the computer system what to do.
- It determines the movement of electronic signals between the main memory and arithmetic logic unit as well as the control signals between the CPU and input/output devices.

Control Unit(Condt…)

- To complete an event i.e. processing, control unit repeats a set of four basic operations:
- Fetching is the process of obtaining a program instruction or data item from the memory
- Decoding is the process of translating the instruction into commands the computer can execute.
- Executing is the process of carrying out the commands.
- Storing is the process of writing the result to memory.

CONTROL UNIT(CONDT…)

- The internal communication inside a computer that transforms raw data into useful information is called **processing.**
- To perform this transformation, the computer uses two components- processor and memory
- The program is fed into the computer through the input unit and stored in the memory
- To execute the program, the instructions have to be fetched from memory one by one which is done by control unit
- Then the control unit decodes the instruction.

Control Unit(Condt…)

- According to instruction, control unit issues signals to other units.
- After instruction is executed, the result of the instruction is stored in memory or stored temporarily in the registry, so that this can be used by the next instruction.
- The results of a program are taken out of the computer through the output unit.

Memory

- Memory is the computer's electronic scratchpad or local store in computer terminology.
- Used for temporary storage of calculations, data, and other work in progress.
- Two types: Primary and Secondary
- Primary memory or the main memory is part of the main computer system. The primary memory itself is of two types.
- The first is called random access memory (RAM) and the other is read only memory (ROM).

Random Access Memory (RAM)

- The processor directly stores and retrieves information from it.
- Memory is organized into locations. Each memory location is identified by a unique address. The access time is same for all location.
- It is volatile: when turned off, everything in RAM disappears.
- Two types:

Types of RAM

- Dynamic Random Access Memory (DRAM):

  This type RAM retain the content of any location only for a few milliseconds. Within that period, each location must be written again with the same contents. This is known as refreshing.

- Static Random Access Memory (SRAM):

  This type of RAM preserves the contents of all the locations as long as the power supply is present. SRAM is generally included in a computer system by the name of cache.

Read Only Memory (ROM)

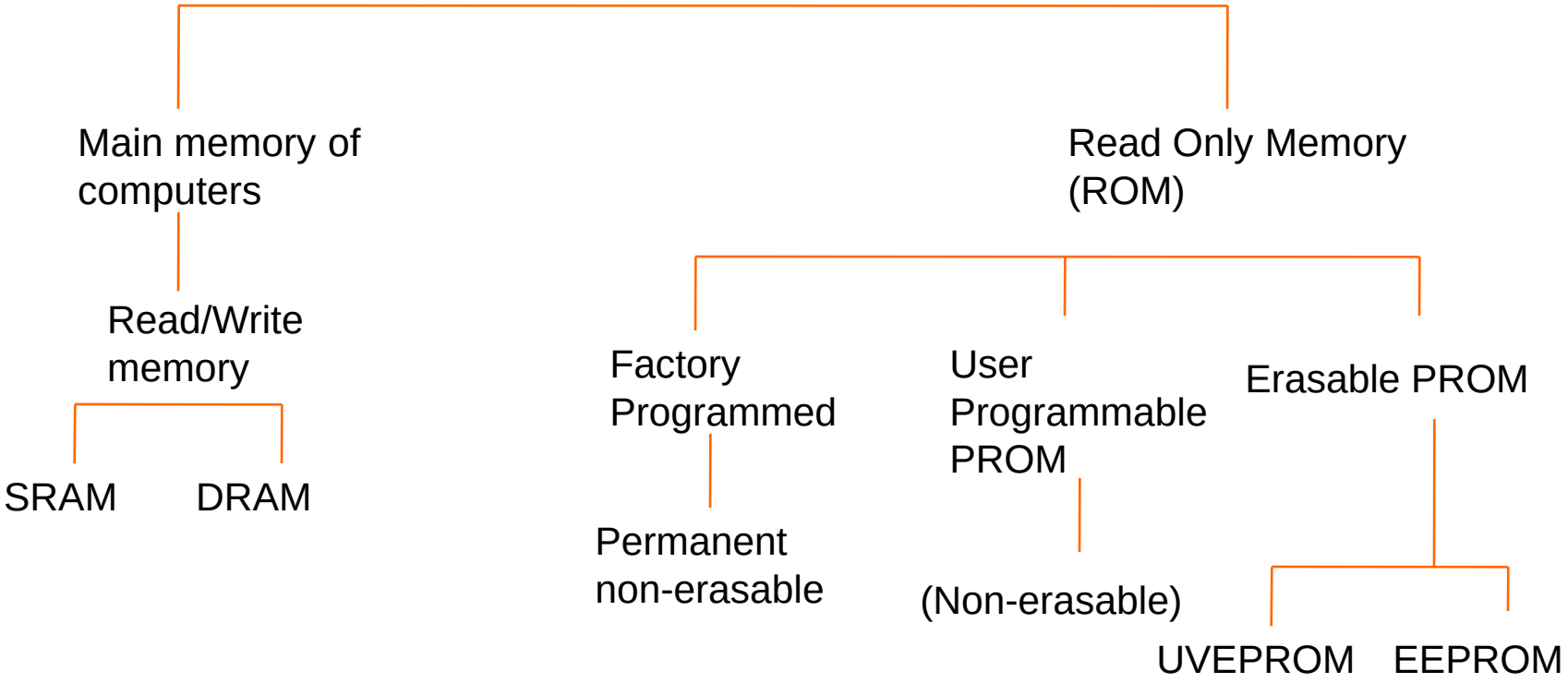
- Data stored in ROM cannot be modified, or can be modified only slowly or with difficulty, so it is mainly used to distribute.
- The instructions in ROM are built into the electronic circuits of the chip which is called firmware.
- Random access in nature and non-volatile.

Types of ROM

- Programmable read-only memory (PROM), or one-time programmable ROM can be written to or programmed via a special device called a PROM programmer.
- Erasable programmable read-only memory (EPROM) can be erased by exposure to strong ultraviolet light then rewritten with a process that again needs higher than usual voltage applied.
- Electrically erasable programmable read-only memory (EEPROM) is based on a similar semiconductor structure to EPROM, but allows its entire contents (or selected banks) to be electrically erased, then rewritten electrically, so that they need not be removed from the computer

- Modern type of EEPROM invented in 1984.
- Random access memories and are non-volatile.
- Use one transistor per memory cell and come in capacities ranging from 1 MB to 32 GB by the year 2007.
- The read time is much smaller (tens of nanoseconds) compared write time (tens of microseconds).

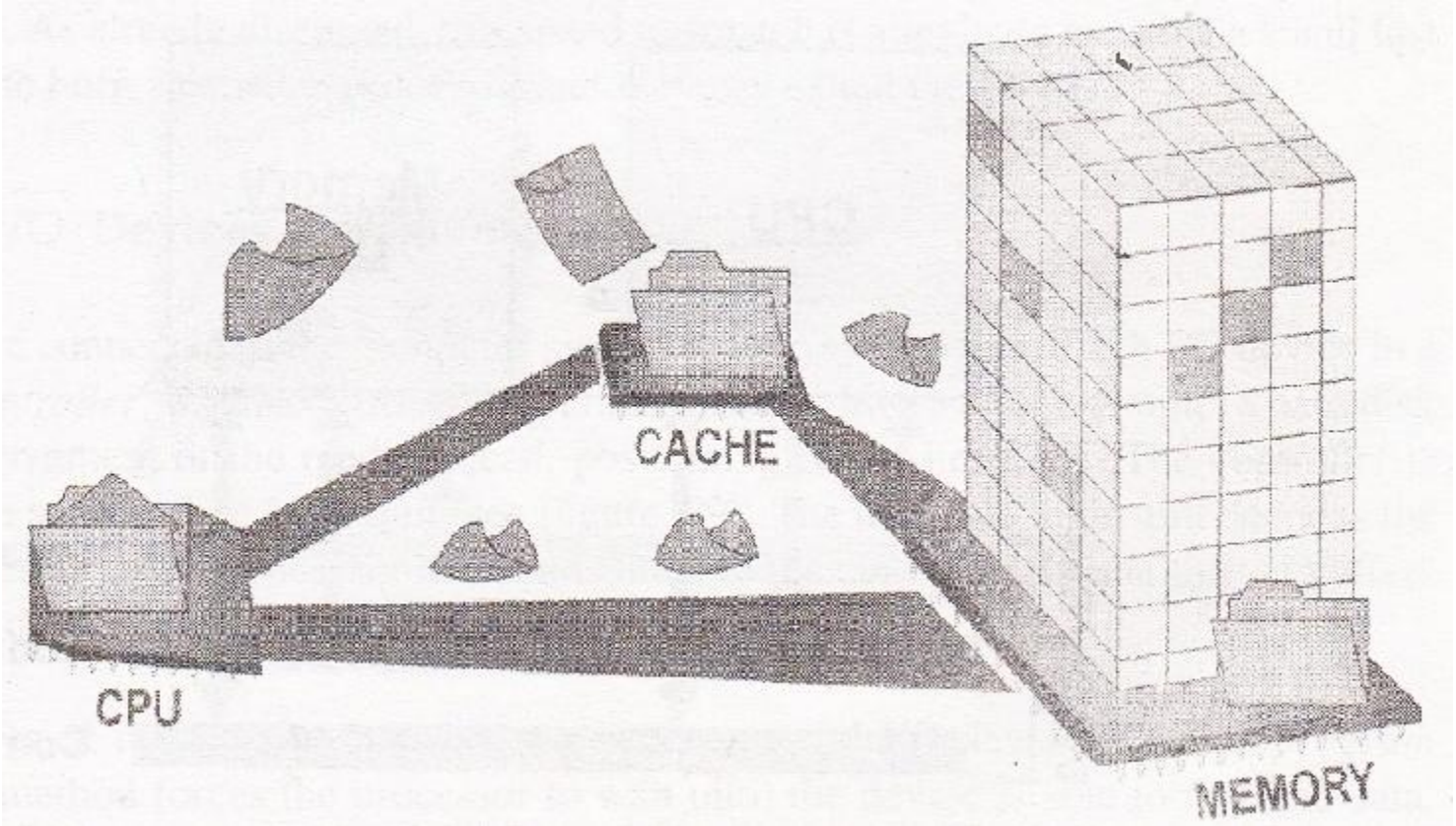
Varieties of Semiconductor Random Access Memories



Cache Memory

- High speed memory kept in between processor and RAM to increase the data execution speed.
- Kept near to the processor.
- Major reason for incorporating cache in the system is that the CPU is much faster than the DRAM and needs a place to store information that can be accessed quickly.
- Cache fetches the frequently used data from the DRAM and buffers (stores) it for further processor usage.

Cache Memory



Different Levels of Cache

- L1-cache is the fastest cache and it usually comes within the processor chip itself. L1 cache typically ranges in size from 8KB to 64KB and uses the high-speed SRAM instead of the slower and cheaper DRAM used for main memory.
- L2 cache comes between L1 and RAM and is bigger than the primary cache.
- L3 cache is not found nowadays as its function is replaced by L2 cache. L3 caches are found on the motherboard rather than the processor. It is kept between RAM and L2 cache.

Processor Speed

- Speed of a computer system is determined by several factors, clock speed of the processor and the speed and size of the data bus.
- Clock speed is the rate at which the processor processes information and this is measured in millions of cycles per second(Megahertz)
- The more the number of hertz, the faster is the processing speed
- The larger the bus width and the faster the bus speed, the greater the amount of data can travel on it in a given amount of time.

Input Devices

- Any peripheral used to provide data and control signals to an information processing system such as a computer or other information appliance.
- Common input devices: Keyboard, Mouse
- Other devices: microphone, digital camera, scanner.

Output Devices

- Any piece of computer hardware equipment used to communicate the results of processed data to the user.
- Examples: Monitors, Printers, Speakers, etc.

Let us Summarise..

- *Computer organization and architecture* is defined as the science of selecting and interconnecting hardware components to create computers that meet functional, performance and cost goals.
- *The central processing unit* is the brain of the computer system where all the computing is done. It consists of three main components, the *control unit (CU), the arithmetic logic unit (ALU) and the registers.*
- The control unit controls the Input/Output devices and transfer of data to and from the primary storage.

Answer in Brief

- Write a note on computer architecture
- What is a system bus? Name the various units of the system bus.
- What is the significance of main memory in proper functioning of a processor.
- What is an Instruction cycle?

Answer in Detail

- What do you understand by Central Processing Unit? Describe in details various units of the CPU.
- Write a detailed note on Instruction Cycle describing the various steps involved.
- Describe in details:
- a. Processor to Memory Communication
- b. Processor to I/O Devices Communication

Let us Summarise..

- The *Arithmetic Unit* is responsible for carrying out the arithmetic calculations such as addition, subtraction, multiplication, and division.
- The *Logic Unit* provides CPU the ability to make logical operations like comparing two data items and taking different actions based on the results of the comparison.
- *Registers* are special purpose, high-speed temporary memory units used by the processor for holding data.

Let us Summarise..

- *The System bus* is a set of wires used for interconnection of different units of a computer system. The three logical units of a system bus are *the address bus, the data bus, and the control bus.*
- *A cache* is a piece of very fast memory, made from high-speed static RAM that reduces the access time of the data. It is very expensive and generally incorporated in the processor, where valuable data and program segments are kept.

Let us Summarise..

- *Instructions* comprise two parts, namely, the *op-code and the operand*. They are transferred one at a time into the processor, where they are decoded and the executed.
- The *Instruction Cycle* details the sequence of events that takes place as an instruction is read from memory and executed.
- In a *Fetch Cycle*, instruction to be executed is fetched from the memory to the processor.
- The *Decode Cycle* is responsible for recognizing which operation the instruction represents activating the correct circuitry to perform that operation.

Let us summarise..

- During the *Execute Cycle*, the operation specified by the op-code is performed on user provided data in the ALU.
- In the *Store Cycle*, the results from the execution cycle are stored back to the memory.
- *Processors* are built with the ability to execute a limited set of basic operations called the *Instruction Set*.
- The speed of the processor is measured in millions of cycles per second or Megahertz (MHz).

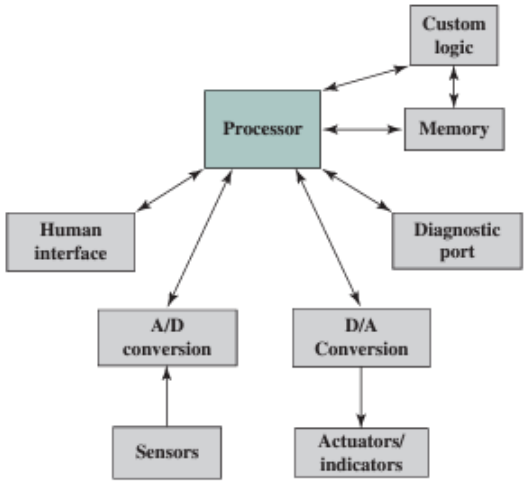
Let Us Summarise..

- Two notables factors on which the speed of a processor depends are the clock speed of the processor and the speed and the size of the data bus

EMBEDDED SYSTEMS:

- The term embedded system refers to the use of electronics and software within a product, as opposed to a general- purpose computer, such as a laptop or desktop system.

Examples include cell phones (not the smart phones), digital cameras, video cameras, calculators, microwave ovens, etc.



Possible Organization of an Embedded System

The Internet of Things:

- The Internet of things (IoT) is a term that refers to the expanding interconnection of smart devices, ranging from appliances to tiny sensors.

Embedded Operating Systems:

- There are two general approaches to developing an embedded operating system:

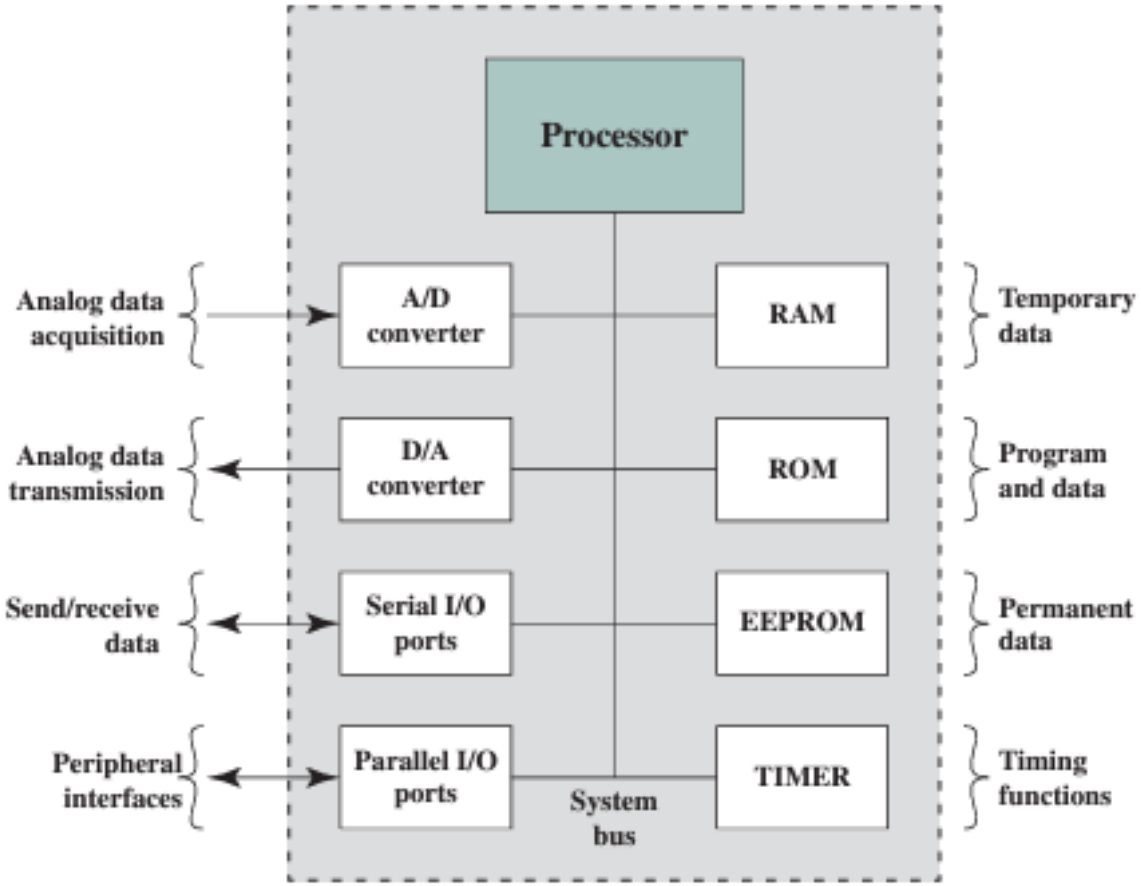
Take an existing OS and adapt it for the embedded application. Ex: embedded versions of Linux, Windows.

The other approach is to design and implement an OS intended solely for embedded use. Ex: TinyOS

Microprocessors and Microcontrollers:

- Microprocessor chips included registers, an ALU, and some sort of control unit or instruction processing logic.

- A microcontroller chip makes a substantially different use of the logic space available. Microcontroller is a single chip that contains the processor, non- volatile memory for the program (ROM) volatile memory for input and output (RAM), a clock, and an I/O control unit.



Typical Microcontroller Chip Elements

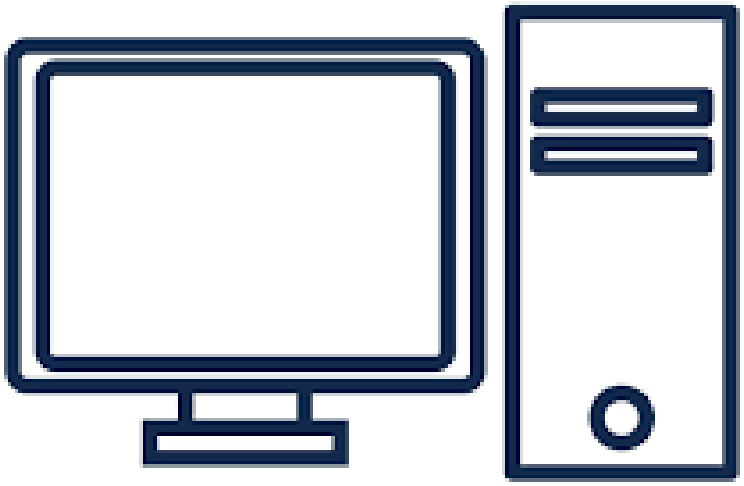
Types of Computers:

- Fixed Program Computers/ Dedicated Device/ Embedded Systems

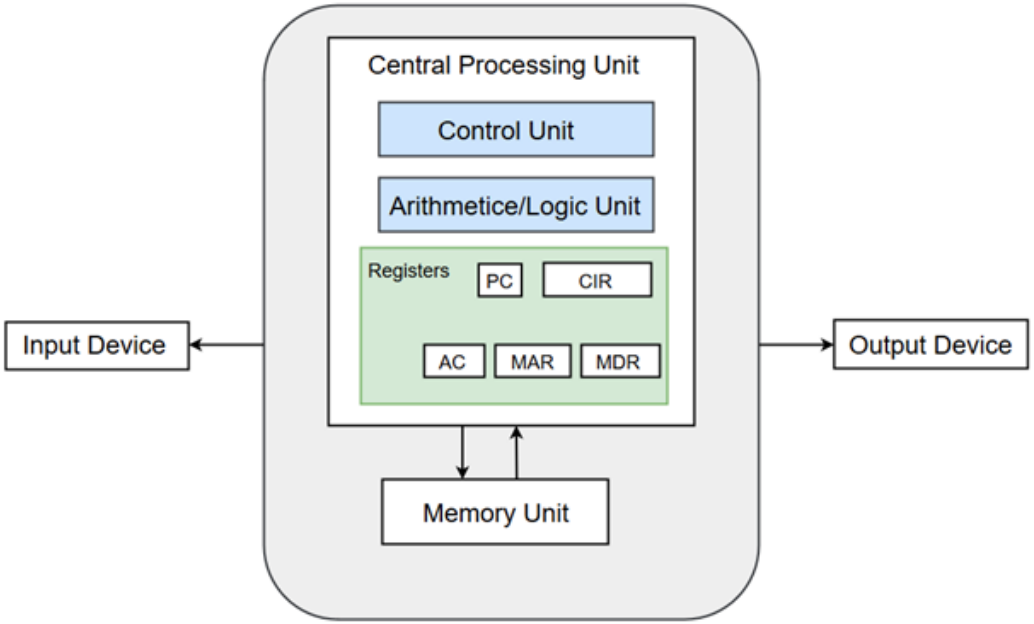
- Stored Program Computers/ General Purpose Computer/ Von Neumann Architecture

Stored Program Computers:

- Stored-program computer, a computer that stores instructions in its memory to enable it to perform a variety of tasks in sequence or intermittently.



Von-Neumann Model:



CPU: The part of the Computer that performs the bulk of data processing operations is called the Central Processing Unit and is referred to as the CPU.

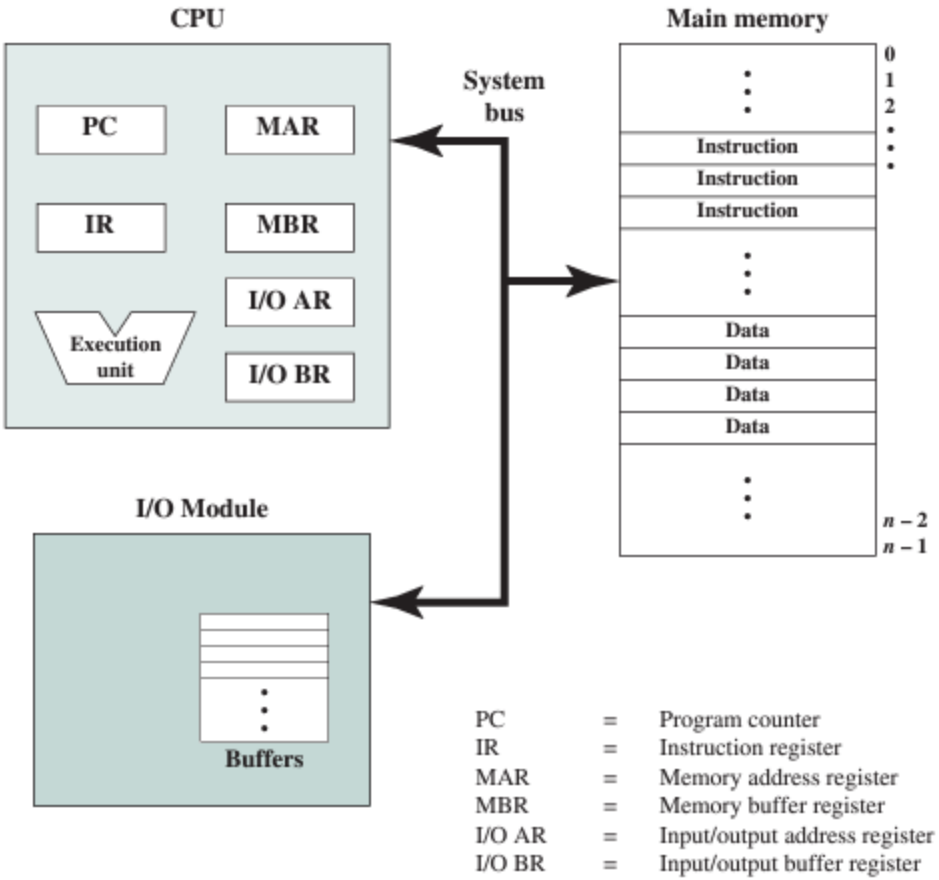
ALU: The Arithmetic and Logic Unit (ALU) performs the required micro-operations for executing the instructions.

Registers: Registers refer to high-speed storage areas in the CPU. The data processed by the CPU are fetched from the registers.

Bus: Buses are the means by which information is shared between the registers in a multiple-register configuration system.

Memory Unit: A memory unit is a collection of storage cells together with associated circuits needed to transfer information in and out of the storage.

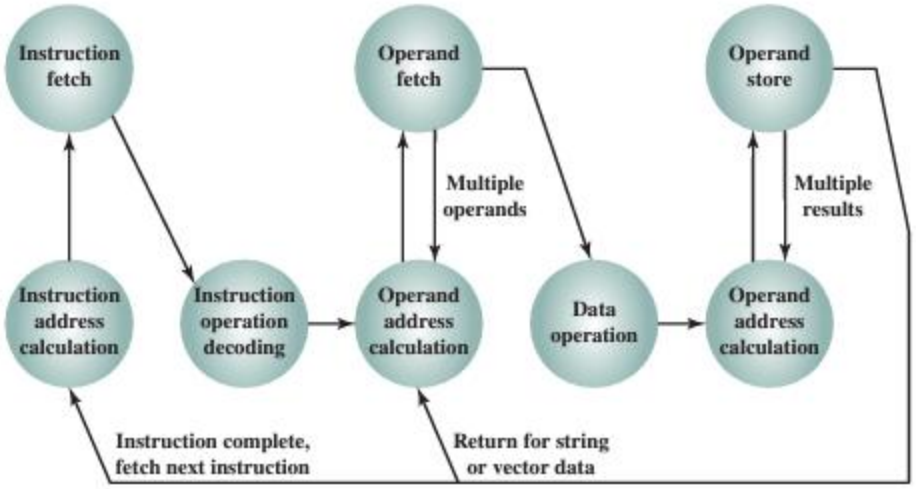
Computer Components: Top-Level View:



- MAR, which specifies the address in memory for the next read or write, and a memory buffer register (MBR), which contains the data to be written into memory or receives the data read from memory.

- I/O address register (I/OAR) specifies a particular I/O device. An I/O buffer register (I/OBR) is used for the exchange of data between an I/O module and the CPU.

Instruction Cycle State Diagram:

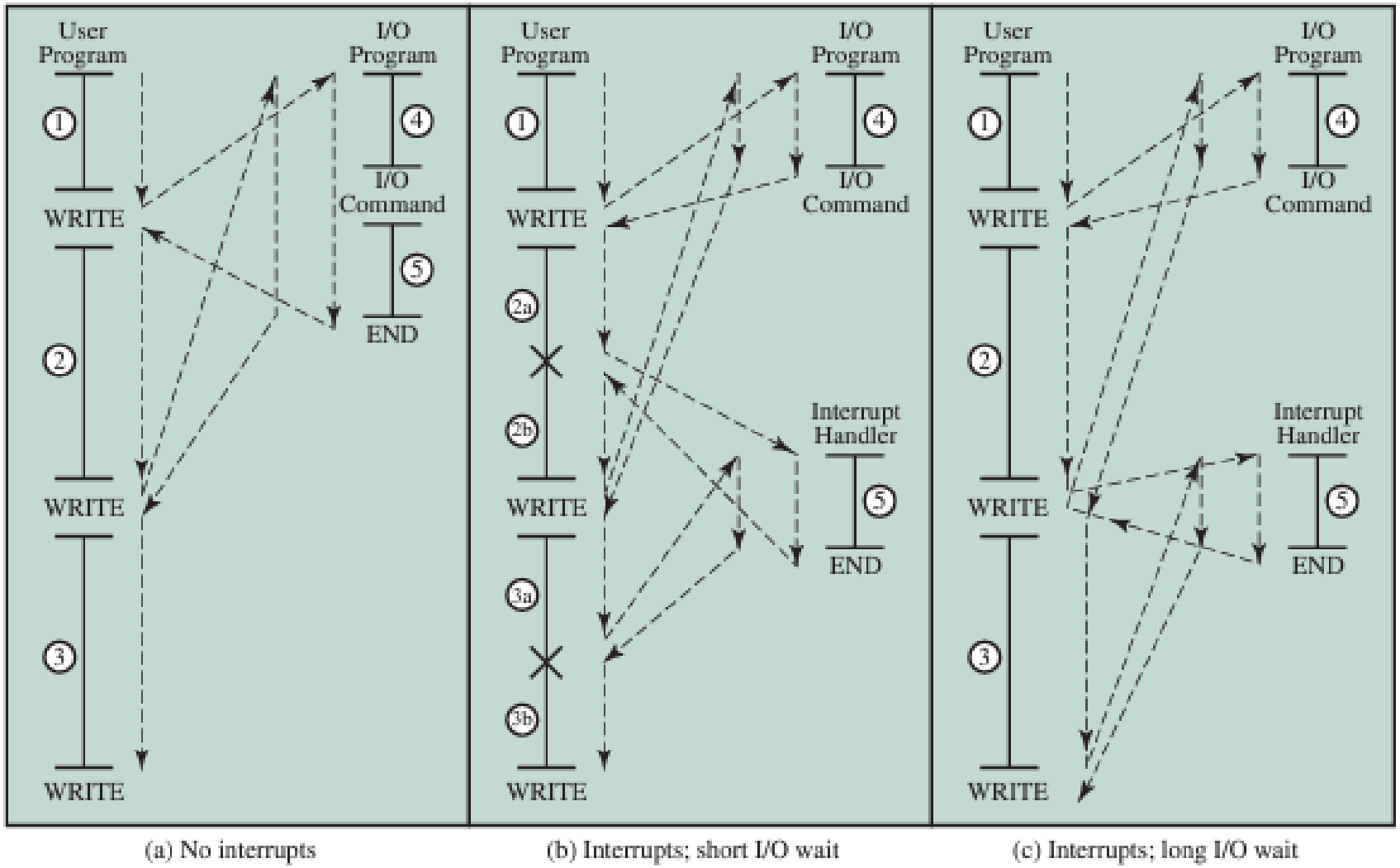


- Instruction fetch (if): Read instruction from its memory location into the processor
- Instruction operation decoding (iod): Analyze instruction to determine type of operation to be performed and operand(s) to be used.
- Operand address calculation (oac): If the operation involves reference to an operand in memory or available via I/O, then determine the address of the operand.
- Operand fetch (of): Fetch the operand from memory or read it in from I/O.
- Data operation (do): Perform the operation indicated in the instruction.
- Operand store (os): Write the result into memory or out to I/O.

INTERRUPTS:

- Computers provide a mechanism by which other modules (I/O, memory) may interrupt the normal processing of the processor; most common classes of interrupts are-

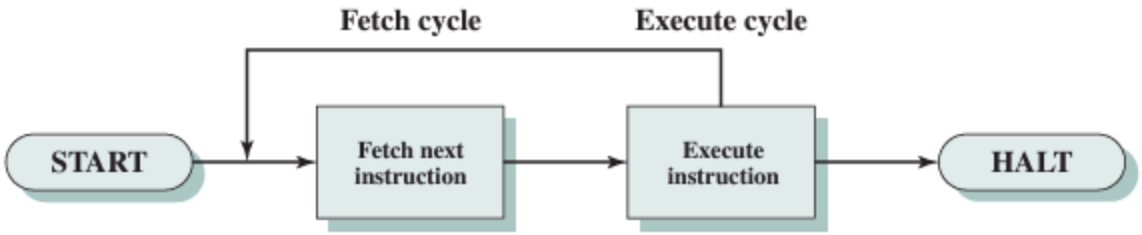
Program	Generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, or reference outside a user's allowed memory space.
Timer	Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.
I/O	Generated by an I/O controller, to signal normal completion of an operation, request service from the processor, or to signal a variety of error conditions.
Hardware Failure	Generated by a failure such as power failure or memory parity error.

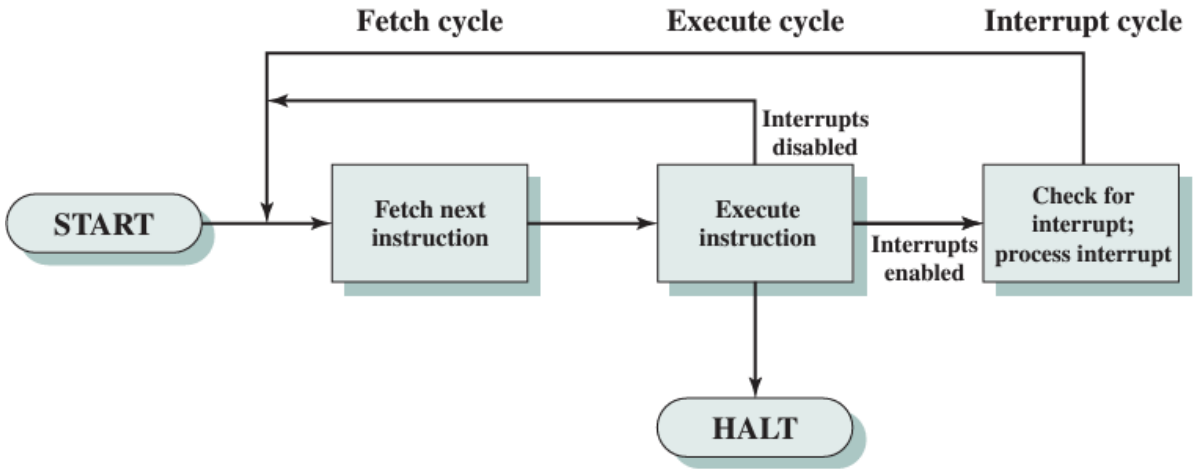


(a) No interrupts (b) Interrupts; short I/O wait (c) Interrupts; long I/O wait

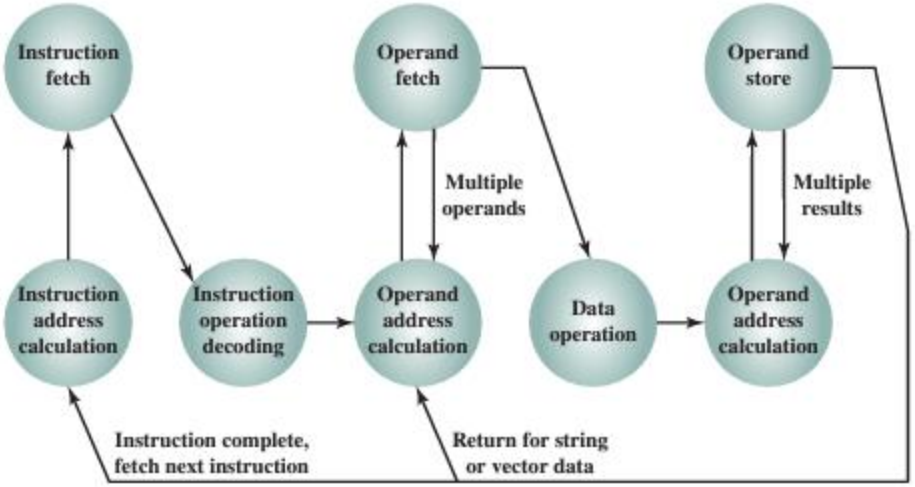
Program Flow Control with and without Interrupt

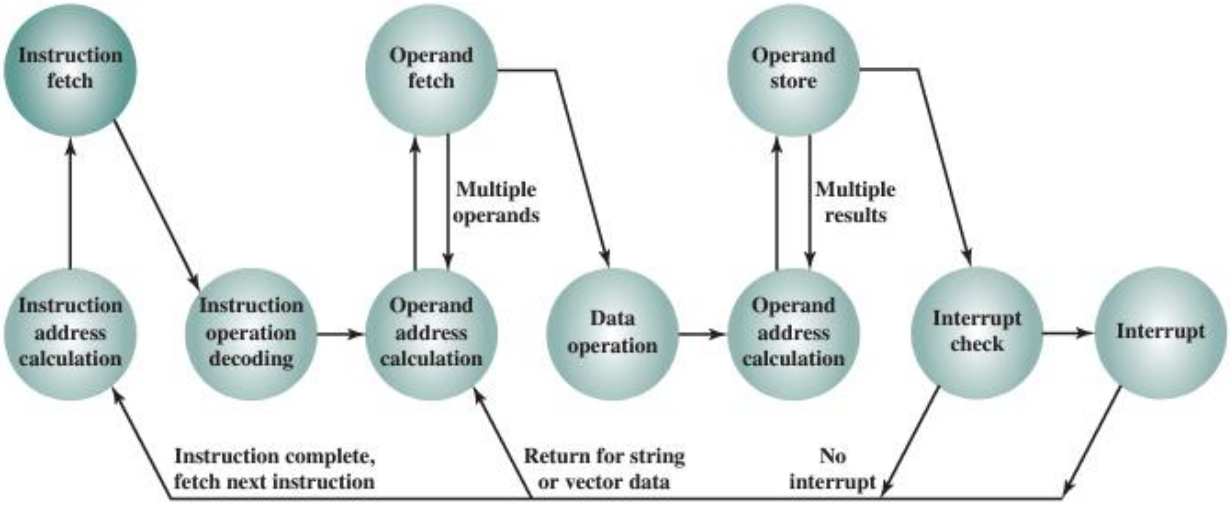
Instruction Cycle with and without Interrupt:





Instr. Cycle State Diag. with and without Interrupt:





INTERCONNECTIONS OF UNITS..

- Set of wires used for interconnection is known as system bus which carry group of bits (information) in a controlled manner.
- It is further divided into three logical units, namely the address bus, the data bus, and the control bus.

Bus Interconnection (Shared Bus):

- A bus is a communication pathway connecting two or more devices. A key characteristic of a bus is that it is a shared transmission medium.
- Multiple devices connect to the bus, and a signal transmitted by any one device is available for reception by all other devices attached to the bus.
- A bus that connects major computer components (processor, memory, I/O) is called a system bus.

System Buses Types

- Data Bus: The data bus is used when any unit is sending data, instruction or command code to some other units.
- Address Bus: The address bus is used when one unit is sending an address information i.e. location of the data residing in the memory to another unit.

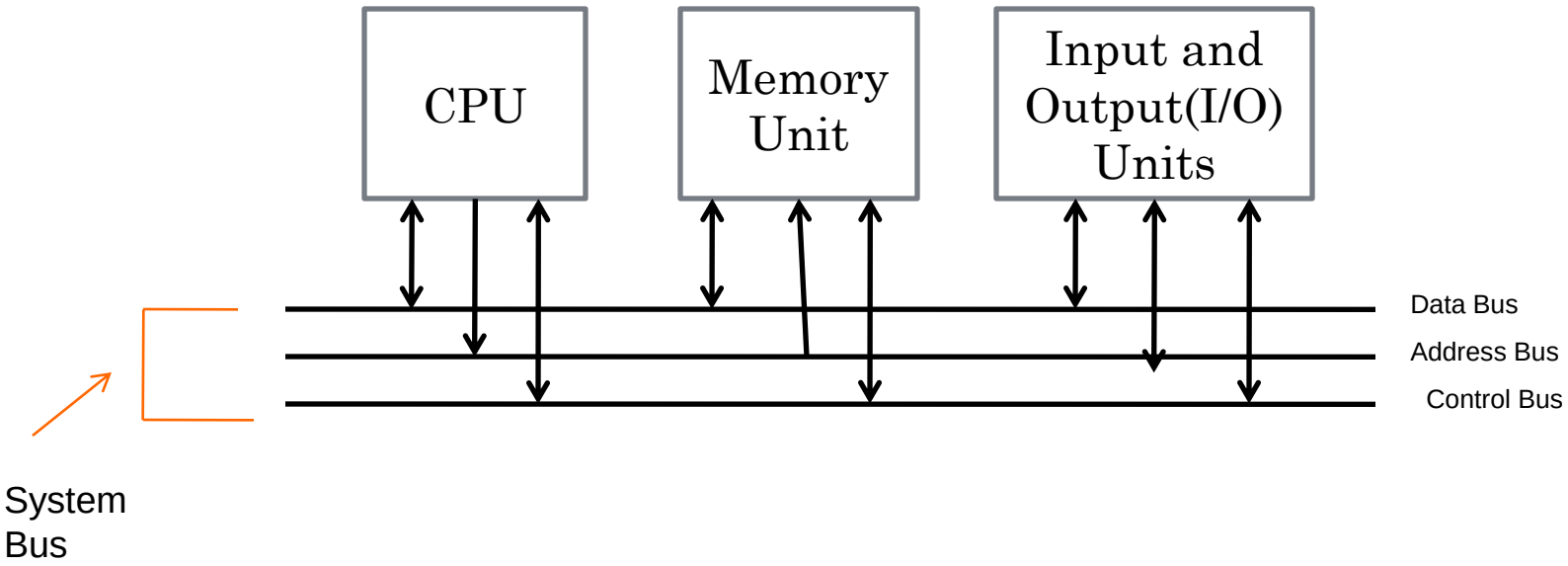
Control Bus

- Control Bus: The control bus is responsible for making CPU, memory and I/O devices work together as a functional system, carrying signals that report the status (ready, not ready) of various units.
- The function of a control bus is to determine and instruct according to the operation type (Read or Write). For example, if the processor or an I/O device wants to read or write a value from memory, the control bus will specify it.

Processing of Information

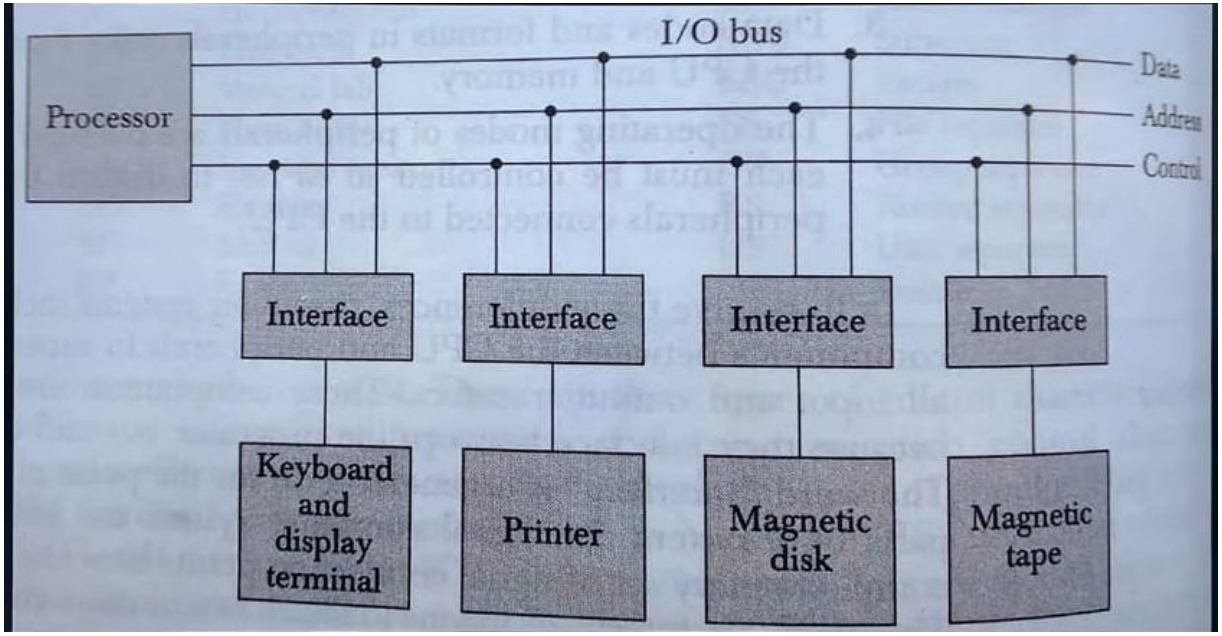
- The bus is common to all the units in the computer. Before sending some information on the bus, a unit should verify whether the bus is free or occupied with some communication started by some other unit.
- CPU is the bus master in a computer which decides who should control the bus when more than one unit wants the bus at the same time.
- A unit who needs the bus makes a request to the CPU and waits sanction. Till the CPU issues sanction, the requesting unit does not attempt to use the bus.

Interconnection of Computers Units via System Bus

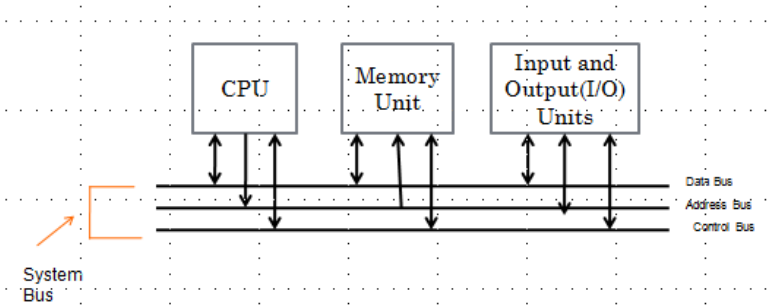


Bus:

- Address Bus: Used to Identify the correct I/O device among the number of I/O Devices
- Control Bus: CPU sends function code through this line
- Data Bus: For data transfer.

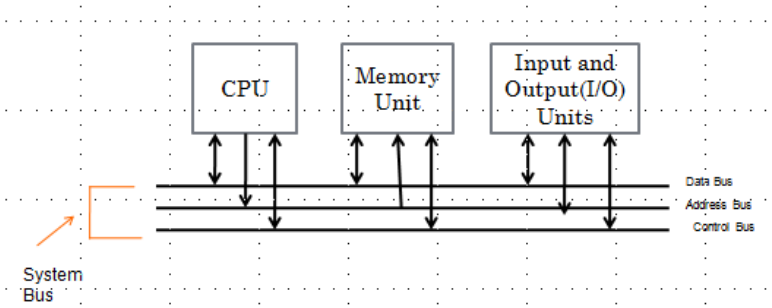


Interconnection of Computers Units via Bus



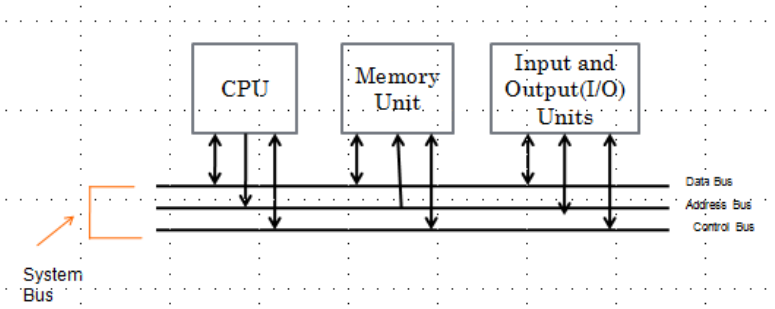
- Shows how the system bus interconnects the processor, memory and I/O devices.
- Both processor and memory units hold a bi-directional relationship with the control and data bus.

Interconnection of Computers Units via Bus



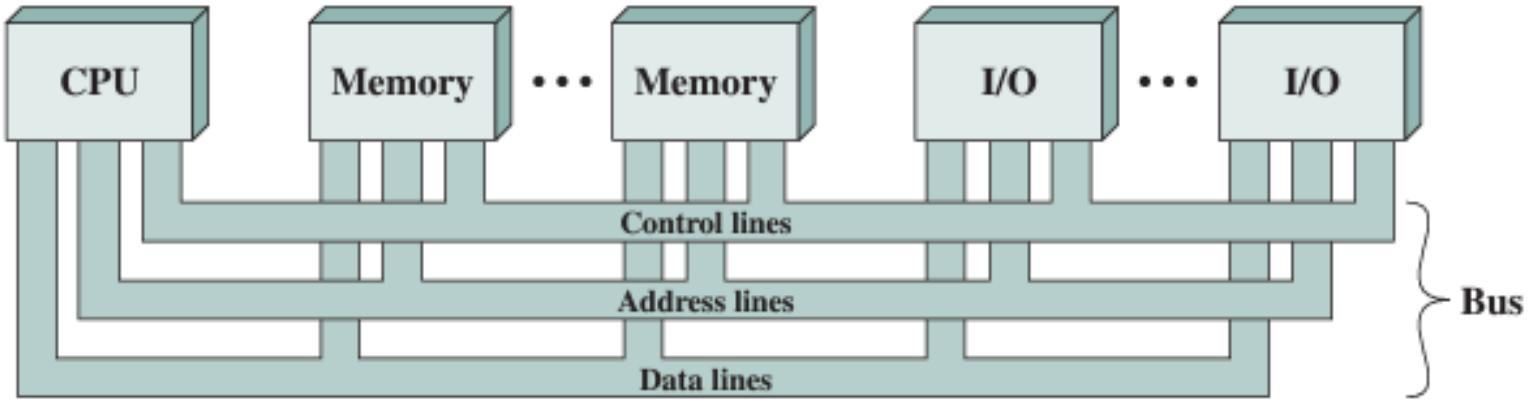
- In case of an address bus, the communication with processor and memory is unidirectional.
- Processor provides location of data (stored in the register) to be fetched from the memory to the address bus and the data carries the required data to the processor.

Interconnection of Computer Units via BUS

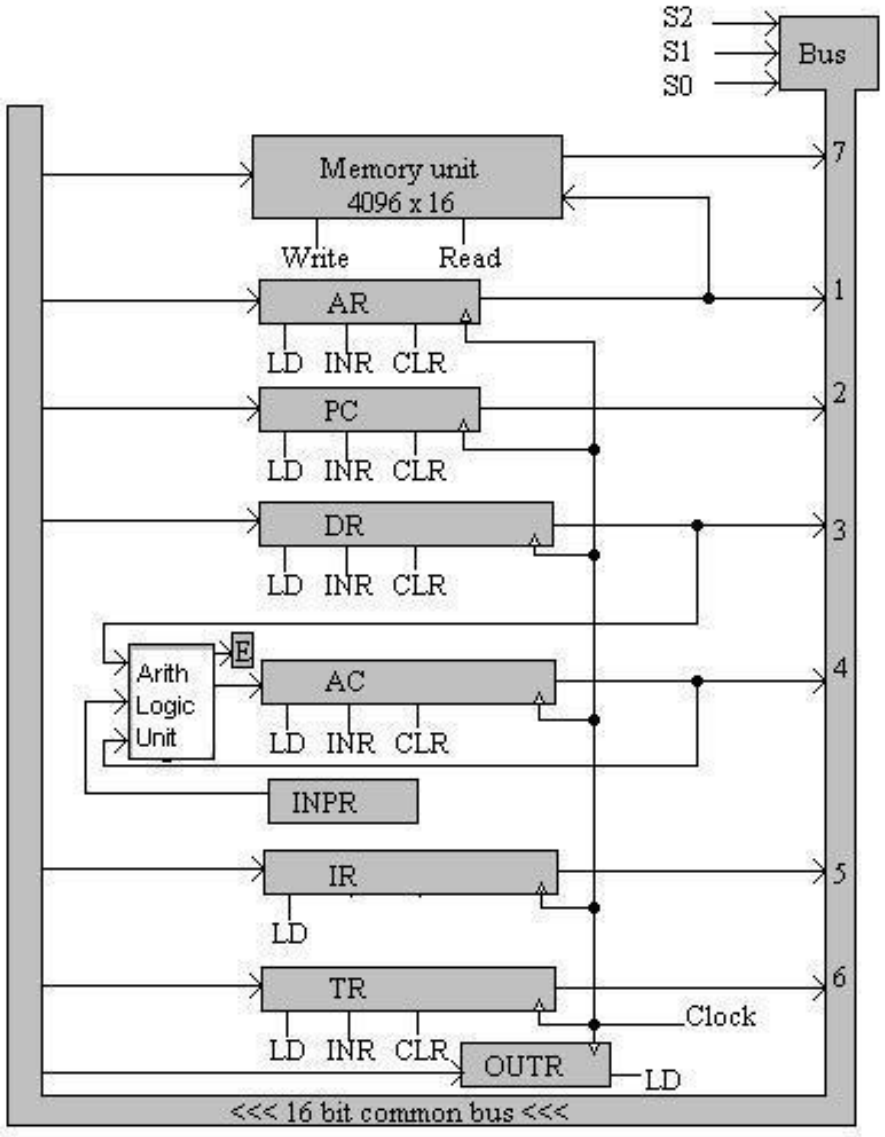


- I/O devices have a bi-directional relationship with the system bus.

- The data lines provide a path for moving data among system modules. These lines, collectively, are called the data bus.
- The address lines are used to designate the source or destination of the data on the data bus.
- The control lines are used to control the access to and the use of the data and address lines.



Common Bus System (Processor Organization):



List of registers for the basic computer

Register symbol	Number of bits	Register name	Function
DR	16	Data register	Holds memory operand
AR	12	Address register	Holds address for memory
AC	16	Accumulator	Processor register
IR	16	Instruction register	Holds instruction code
PC	12	Program counter	Holds address of instruction
TR	16	Temporary register	Holds temporary data
INPR	8	Input register	Holds input character
OUTR	8	Output register	Holds output character

Point To Point Interconnect:

- The principal reason driving the change from bus to point-to-point interconnect was the electrical constraints encountered with increasing the frequency of wide synchronous buses.

- Compared to the shared bus, the point-to-point interconnect has lower latency, higher data rate, and better scalability.

- Example of the point-to-point interconnect approach: Intel's **QuickPath Interconnect (QPI), which was introduced in 2008**.

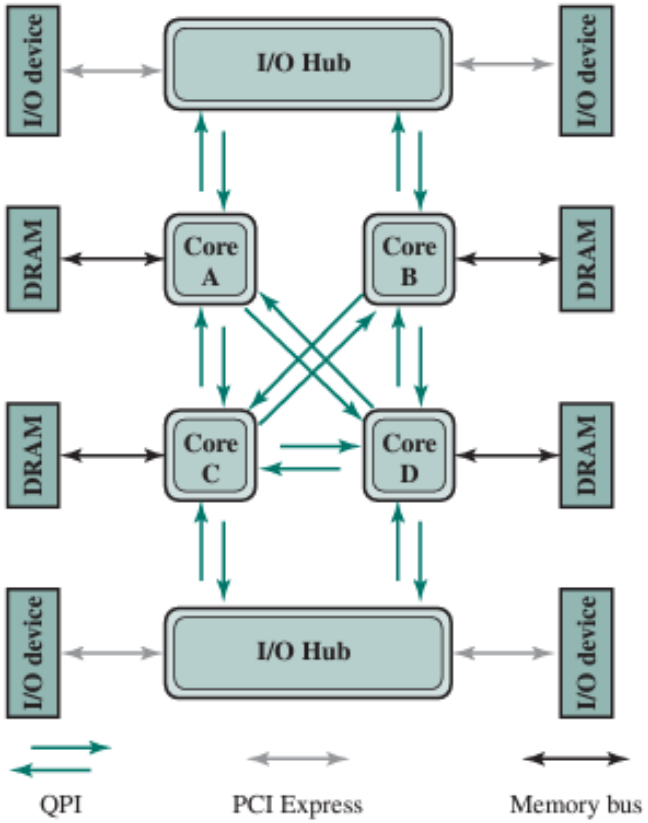
- Characteristics of QPI and other point-to-point interconnect schemes:

Multiple direct connections: Multiple components within the system enjoy direct pairwise connections to other components. This eliminates the need for arbitration found in shared transmission systems.

Layered protocol architecture: As found in network environments, such as TCP/IP-based data networks, these processor-level interconnects use a layered protocol architecture, rather than the simple use of control signals found in shared bus arrangements.

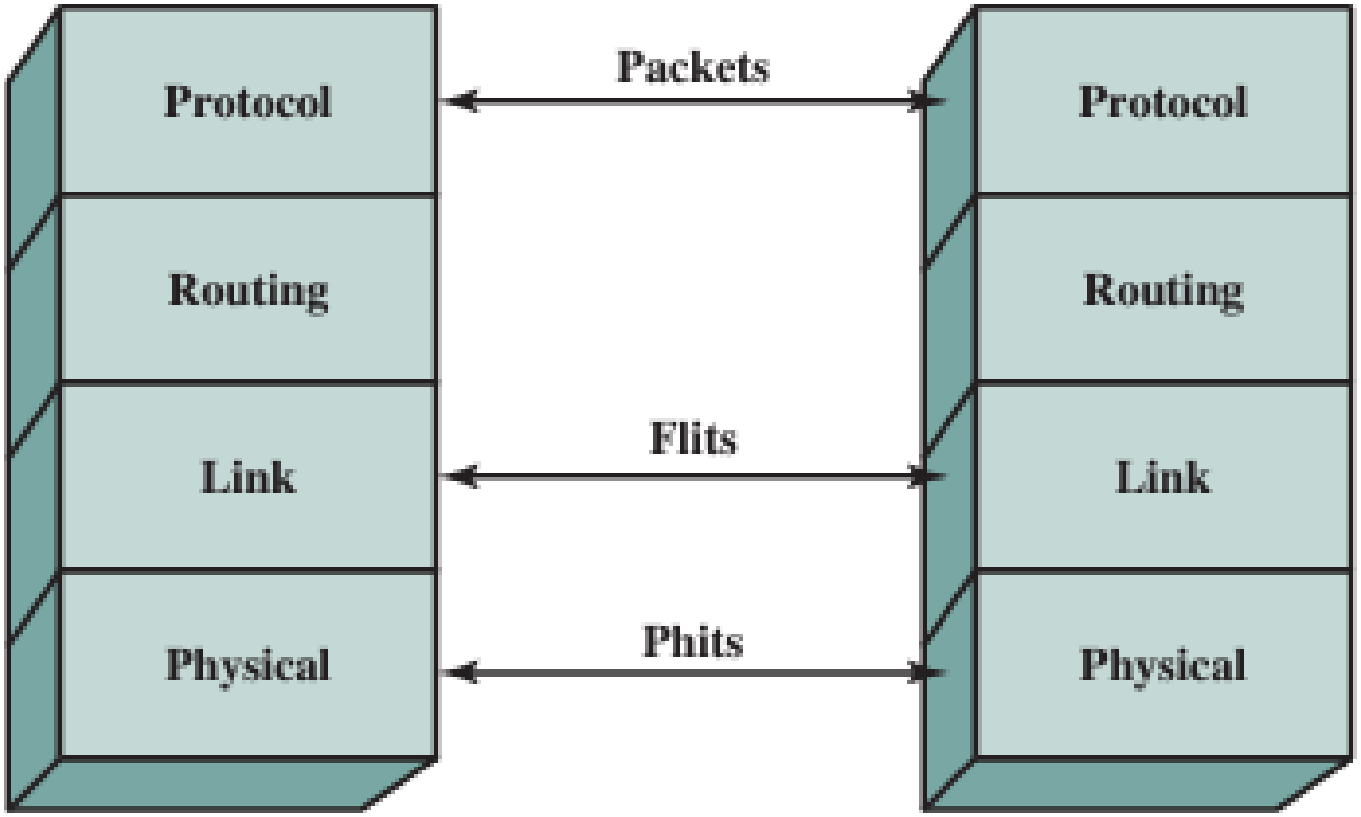
Packetized data transfer: Data are not sent as a raw bit stream. Rather, data are sent as a sequence of packets, each of which includes control headers and error control codes.

TYPICAL USE OF QPI ON A MULTICORE COMPUTER:



QPI A FOUR-LAYER PROTOCOL:

- Physical: Consists of the actual wires carrying the signals, as well as circuitry and logic to support ancillary features required in the transmission and receipt of the 1s and 0s. The unit of transfer at the Physical layer is 20 bits, which is called a Phit (physical unit).
- Link: Responsible for reliable transmission and flow control. The Link layer's unit of transfer is an 80-bit Flit (flow control unit).
- Routing: Provides the framework for directing packets through the fabric.
- Protocol: The high-level set of rules for exchanging packets of data between devices. A packet is comprised of an integral number of Flits.


Protocol
Routing
Link
Physical
Packets
Flits
Phits
Protocol
Routing
Link
Physical

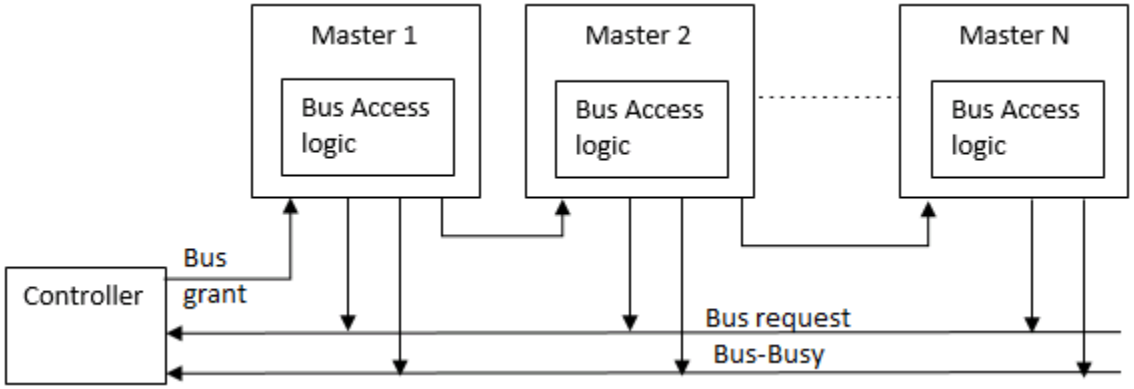
Bus Arbitration:

Method used to decide which device gets the access of common BUS. (Ensuring Data Integrity and system stability). Methods:

- Daisy Chaining (Centralised Arbitration)
- Polling (Centralised Arbitration)
- Fixed Priority or Independent Request Method (Distributed Arbitration)

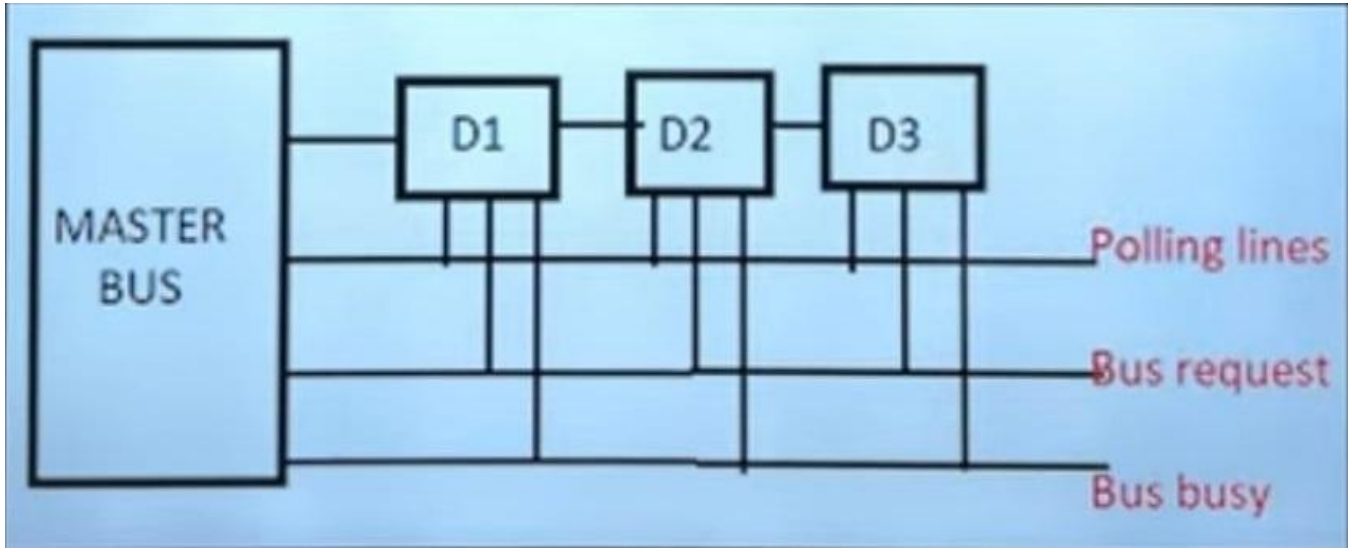
Daisy Chaining:

- If more than one device makes a request at the same time, then the device that is closer to the arbiter will get the bus. (Hardware Solution)
- Priority can not be changed in random.
- Serial Propagation
- Relatively Fast



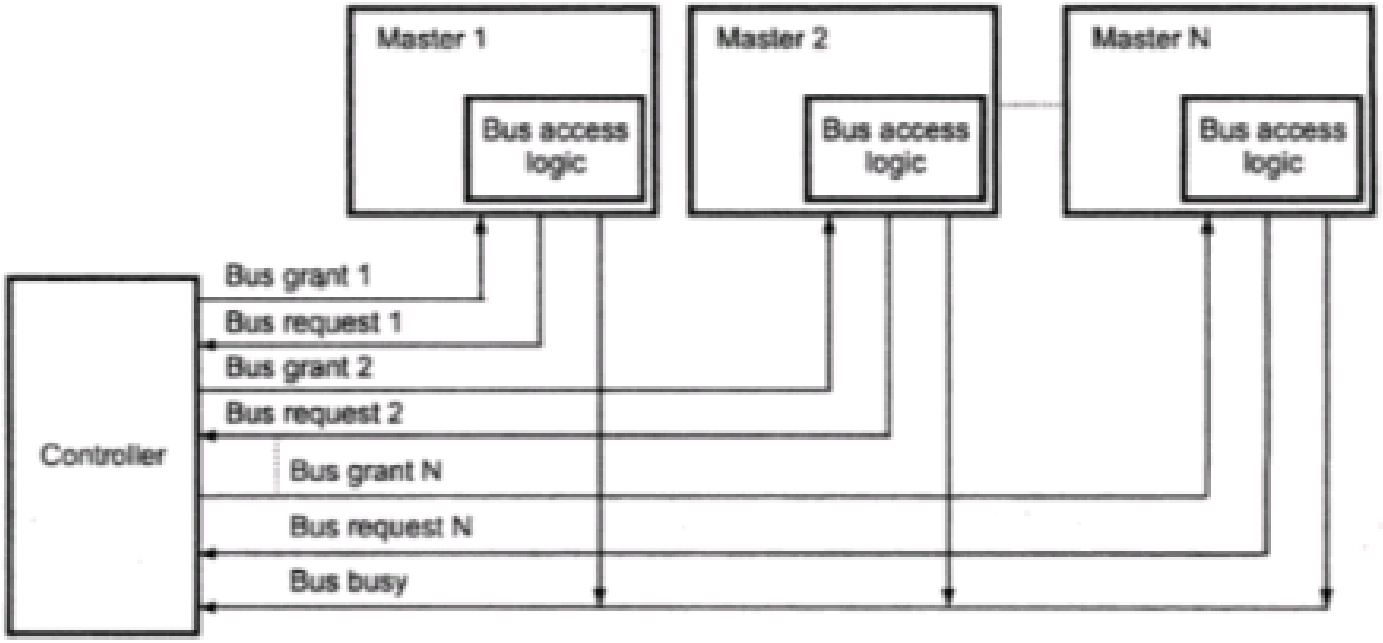
POLLING:

- The controller is used to generate the address for the master(unique priority), the number of address lines required depends on the number of masters connected in the system. (Software Solution)
- Priority can be changed in random.

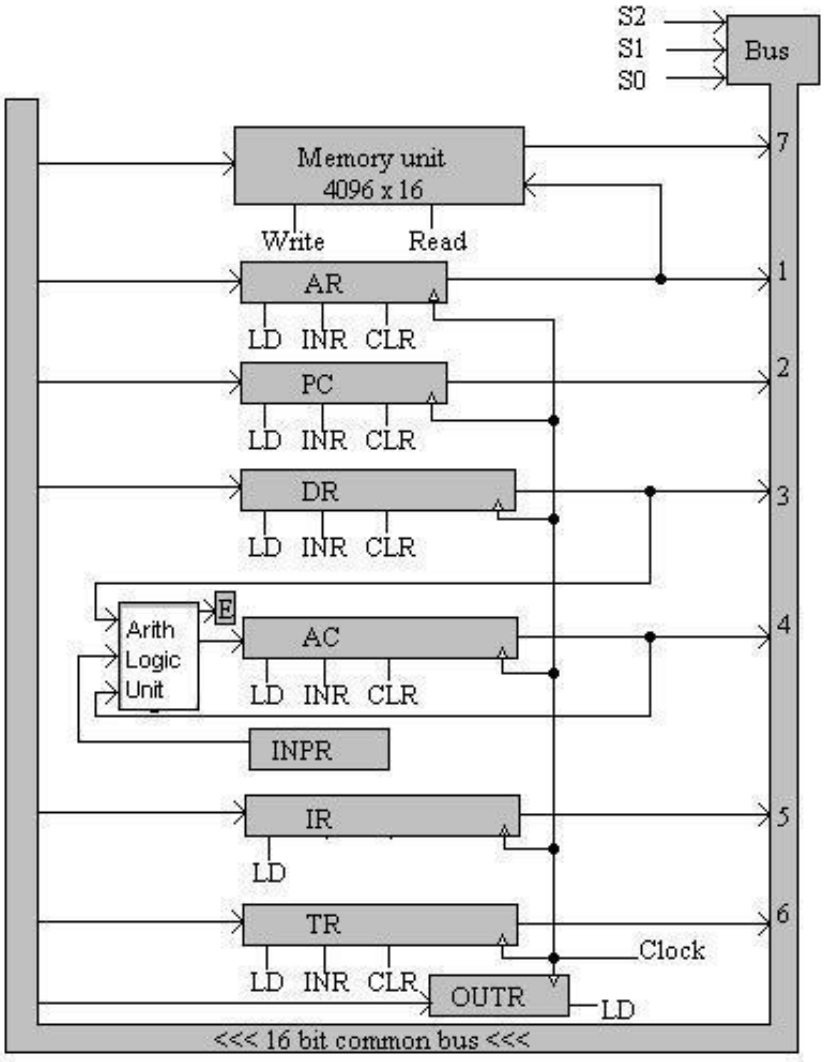


Fixed Priority (Independent Req. Method):

- Each master has a separate pair of Bus request and grant lines, and each pair has a priority assigned to it.



Common Bus System (Processor Organization):



List of registers for the basic computer

Register symbol	Number of bits	Register name	Function
DR	16	Data register	Holds memory operand
AR	12	Address register	Holds address for memory
AC	16	Accumulator	Processor register
IR	16	Instruction register	Holds instruction code
PC	12	Program counter	Holds address of instruction
TR	16	Temporary register	Holds temporary data
INPR	8	Input register	Holds input character
OUTR	8	Output register	Holds output character

General Register organization:

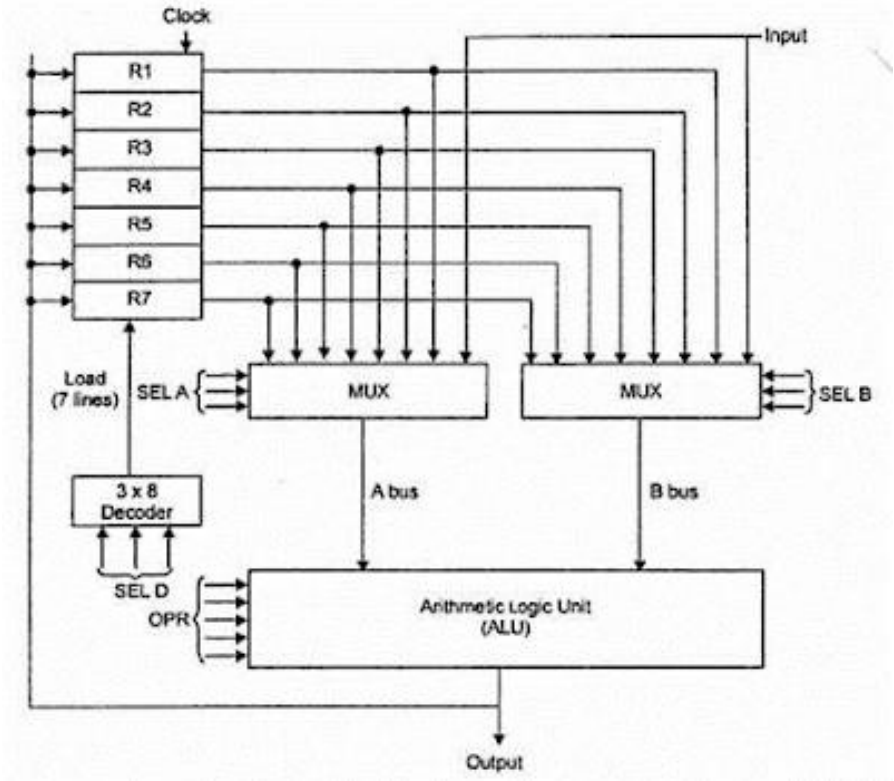


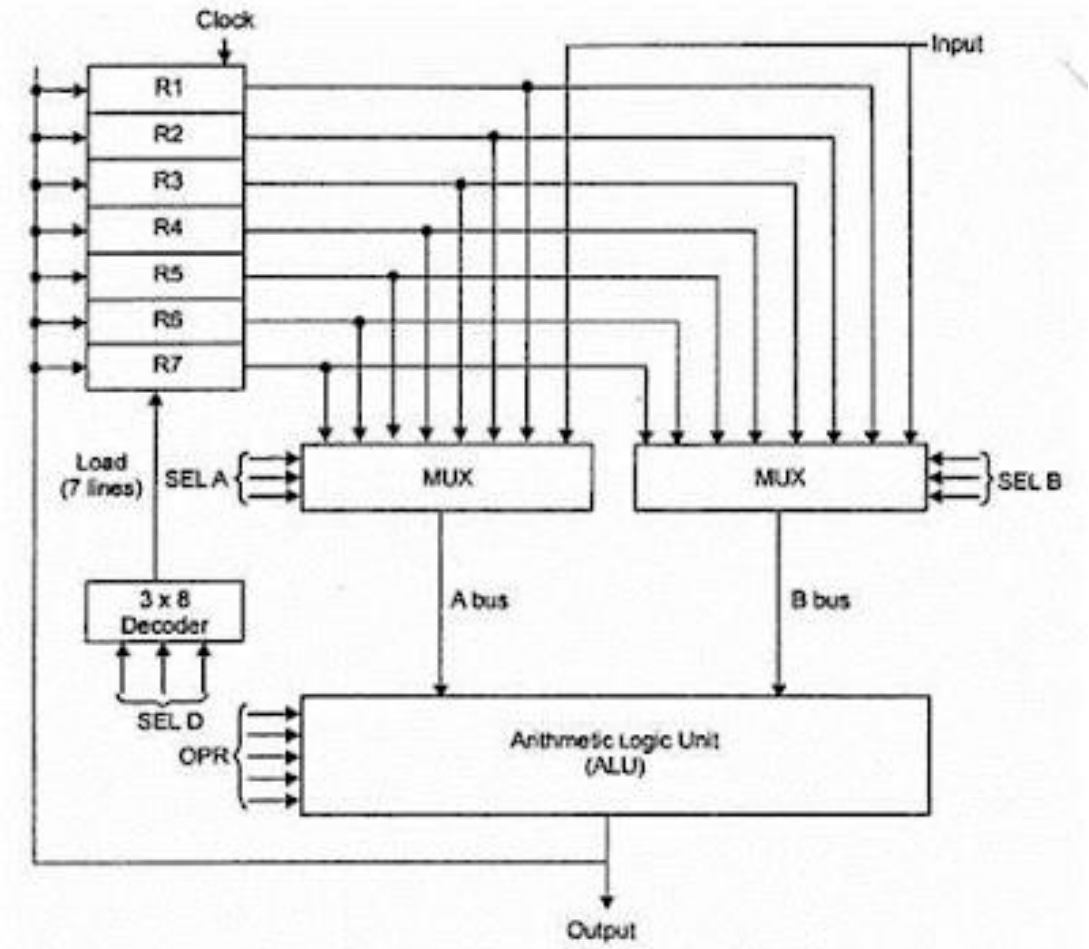
TABLE 8-1 Encoding of Register Selection Fields

Binary Code	SELA	SELB	SELD
000	Input	Input	None
001	R1	R1	R1
010	R2	R2	R2
011	R3	R3	R3
100	R4	R4	R4
101	R5	R5	R5
110	R6	R6	R6
111	R7	R7	R7

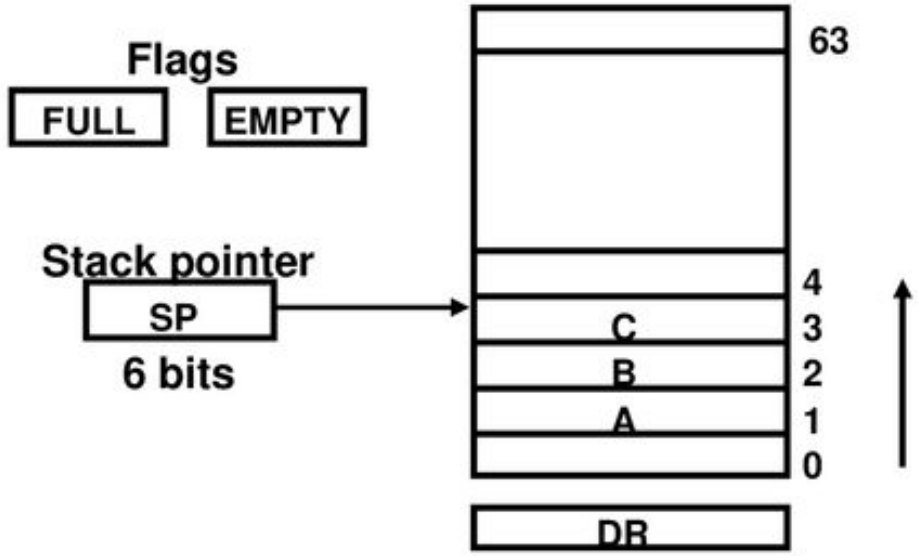
TABLE 8-2 Encoding of ALU Operations

OPR Select	Operation	Symbol
00000	Transfer *A*	TSFA
00001	Increment *A*	INCA
00010	Add *A* + *B*	ADD
00101	Subtract *A* − *B*	SUB
00110	Decrement *A*	DECA
01000	AND *A* and *B*	AND
01010	OR *A* and *B*	OR
01100	XOR *A* and *B*	XOR
01110	Complement *A*	COMA
10000	Shift right *A*	SHRA
11000	Shift left *A*	SHLA

Microoperation	Symbolic Designation SELA	SELB	SELD	OPR	Control Word
R1 ← R2 − R3	R2	R3	R1	SUB	010 011 001 00101
R4 ← R4 ∨ R5	R4	R5	R4	OR	100 101 100 01010
R6 ← R6 + 1	R6	—	R6	INCA	110 000 110 00001
R7 ← R1	R1	—	R7	TSFA	001 000 111 00000
Output ← R2	R2	—	None	TSFA	010 000 000 00000
Output ← Input	Input	—	None	TSFA	000 000 000 00000
R4 ← sh1 R4	R4	—	R4	SHLA	100 000 100 11000
R5 ← 0	R5	R5	R5	XOR	101 101 101 01100



Stack Organization:



SP: Points the top of the stack
Full: Set to 1 when stack is full
Empty: Set to 1 when stack is empty.
DR: Data register

Memory Stack:

- Works on LIFO principle

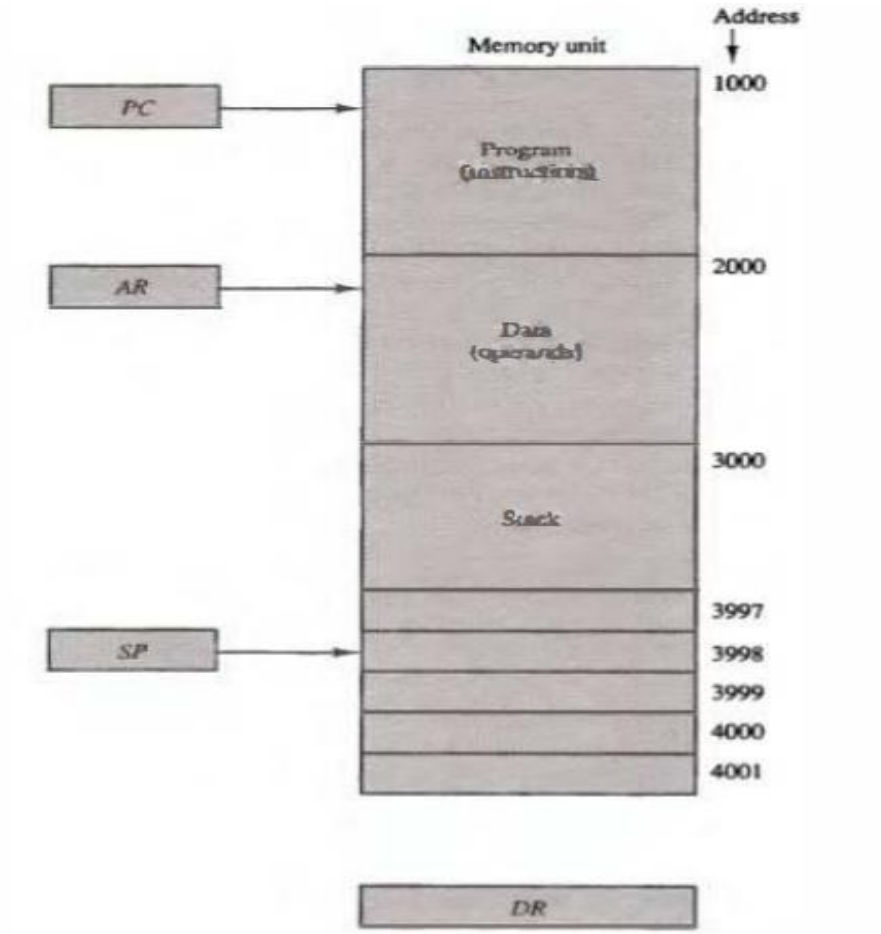


Figure 4 Computer memory with program, data, and stack segments.

Addressing Mode:

- Specifies different ways possible in which reference to the operand can be made.
- Effective address: Final address of the location where the operand is stored. Calculation of effective address can be done in two ways:

1. Non computable addressing

2. Computable addressing (involves arithmetic)

Criteria for different addressing mode:

- It should be fast
- The length of the instruction must be small
- They should support pointers
- Should support looping constructs, indexing of data structure.
- Program relocation.

Addressing Modes types:

- Immediate addressing mode
- Direct Addressing Mode (Absolute address mode)
- Indirect Addressing Mode
- Implied mode Addressing
- Register mode Addressing
- Register indirect mode Addressing
- Base register (off set) addressing mode
- Index addressing mode
- Relative addressing mode

Immediate Addressing mode:

- Operand is itself is part of the instruction.

E.g. ADD 3; means add 3 to the accumulator

Advantages:

- Can be used for constants
- Extremely fast, no memory reference is required

Disadvantages:

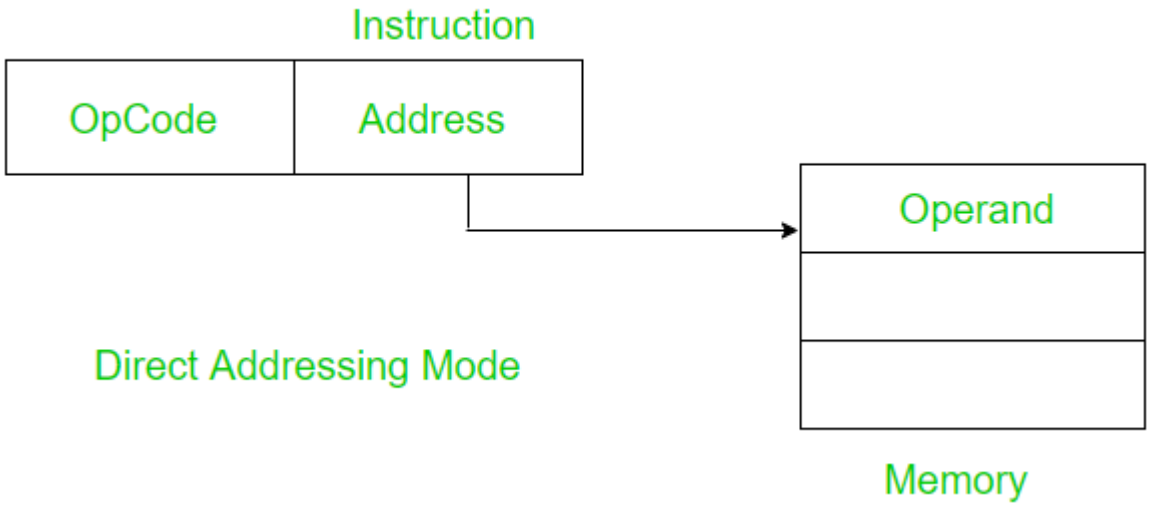
- Cannot be used with variables
- Cannot be used for large constant values

Application:

- Mostly used when required data is directly moved to required register or memory

Direct addressing mode (Absolute address mode):

- Instruction contains address of the memory location where data (operand) is present (Effective address).
- Only one memory reference operation is required to access data.



Advantages:

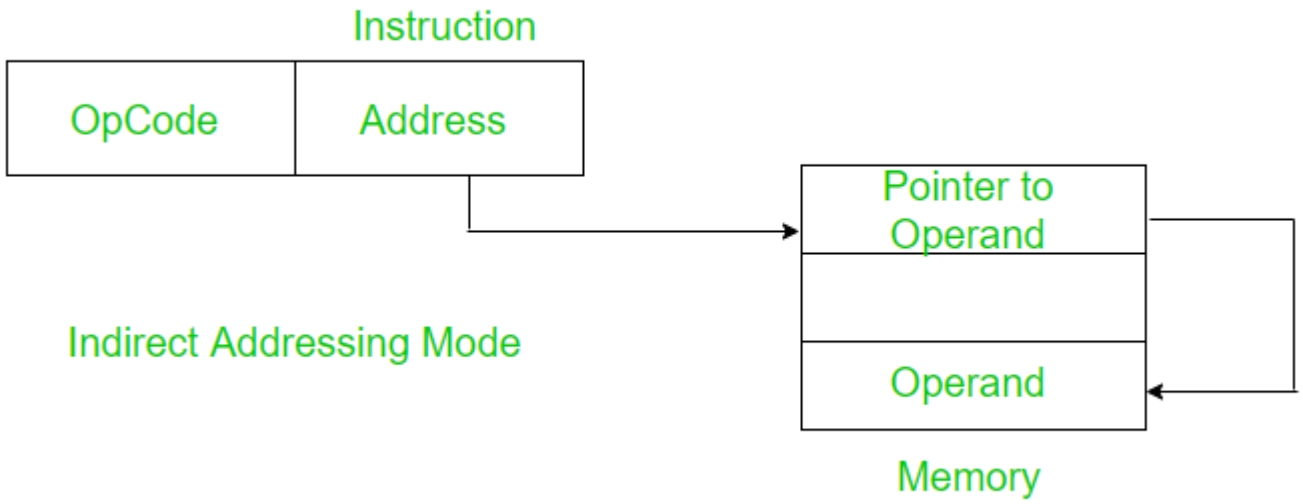
- With variables, simplest addressing mode
- No restriction on range of data.
- Can be used to access global variables whose address is known at compile time.

Disadvantages:

- Relatively Slow
- Number of variables used are limited.
- In large calculation it will fail.

Indirect Addressing Mode:

- Here in instruction, we store the address where the effective address is stored, using which we can access the actual data.
- Two references are required.
- 1st is to get EA, and 2nd reference is to access the data.



Advantages:

- No limitation on number of variable or size of variable
- Implementation of pointer are feasible and relatively more secure.

Disadvantage:

- Relatively slow as memory must be referred more than one time.

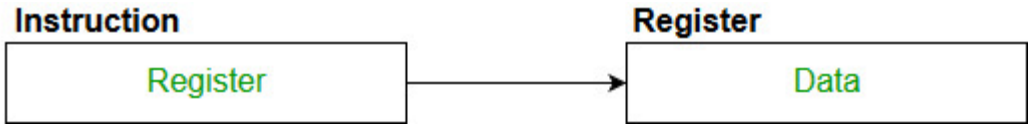
Implied mode addressing:

- Operands are specified implicitly in the definition of the instruction.
- Register reference instructions use an accumulator are implied mode
- Zero address Instruction, also known as stack addressing mode.

E.g. Increment accumulator, complement accumulator.

Register Addressing mode:

- Variables are stored in registers of the CPU instead of memory, in the instruction we will give the register number.



Advantages:

- Extremely fast.
- Bits required to specify a register is less

Disadvantage:

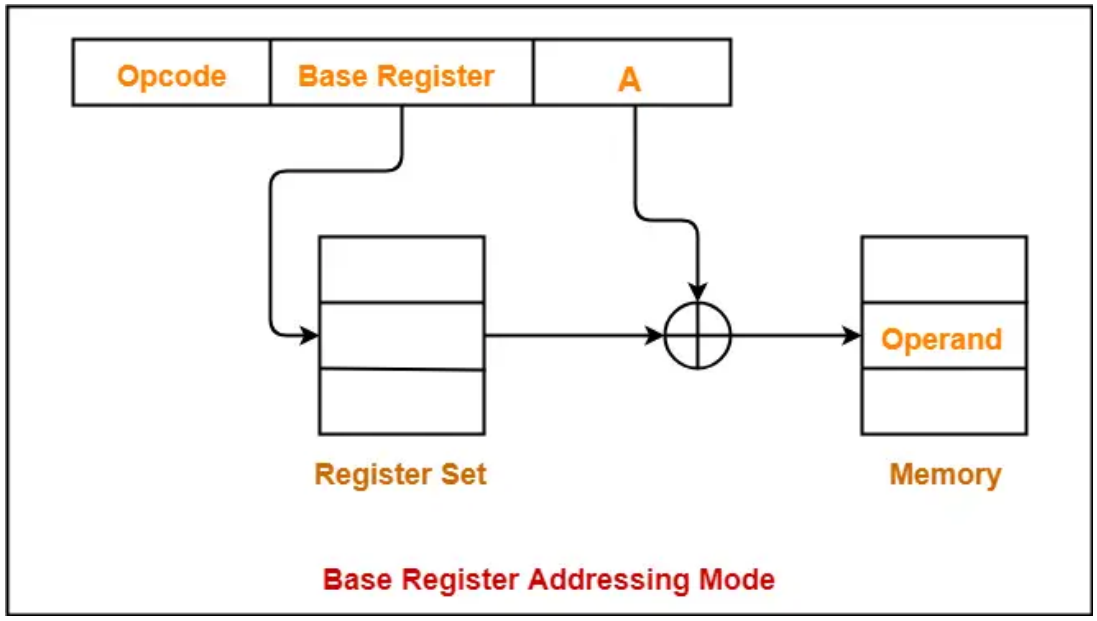
- Number of registers is less

Register Indirect mode addressing:

- We specify the address of the register where inside the register we will get the memory address of the variable (EA).
- When same address is required, this becomes very useful.
- It can be further improved by auto increment or auto decrement for operations on data structures like matrix or array.

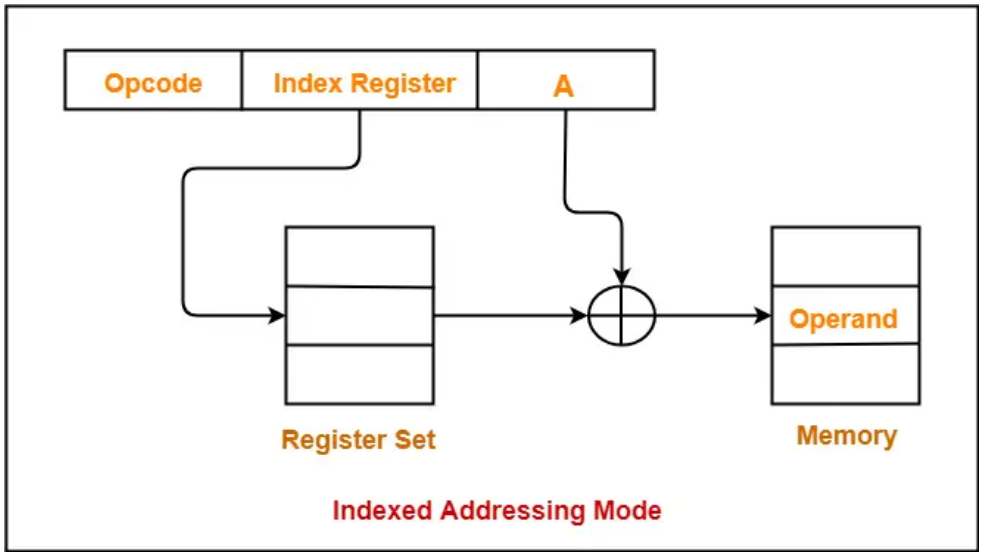
Base Register (off set) addressing mode:

- In multiprogramming environment, the location of the process keeps on changing, which is why direct address will create problem here.
- To solve this, we save the starting of the program address in a register (BR).



INDEX ADDRESSING MODE:

- Generally used when CPU has number of registers, out of which one can be used as a index register.
- Base is present inside the instruction and index is present inside the register.
- Especially useful for large sized array.



RELATIVE ADDRESSING MODE:

- Effective address of the operand is obtained by adding the content of the program counter with the address part of the instruction.