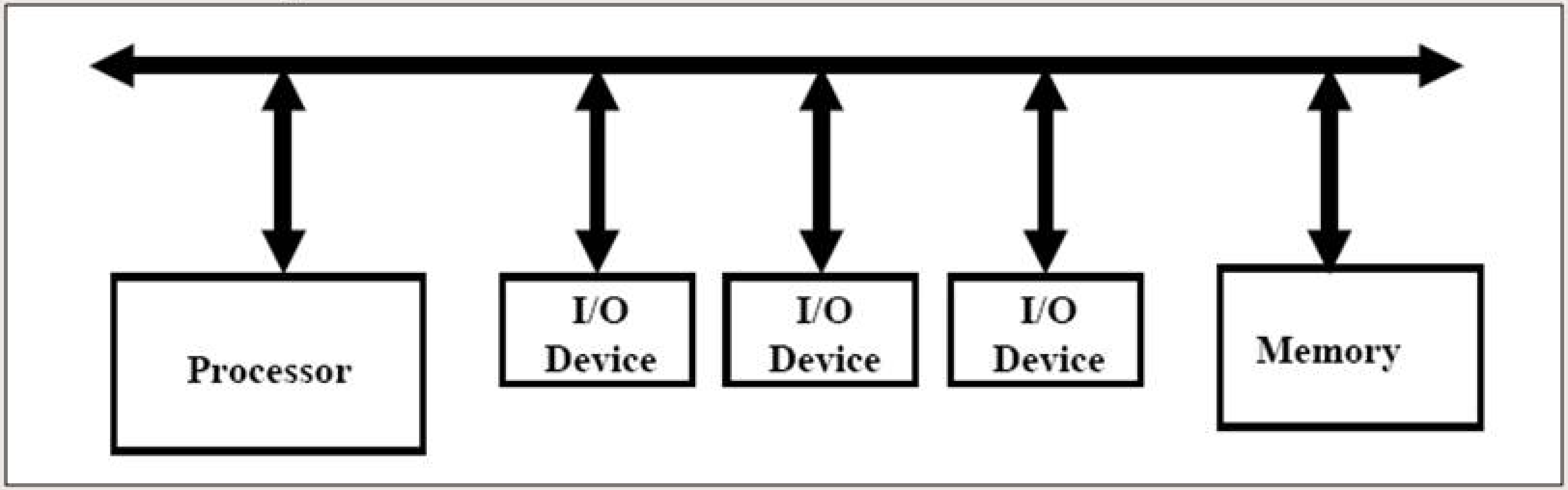
BUSES

INTRODUCTION

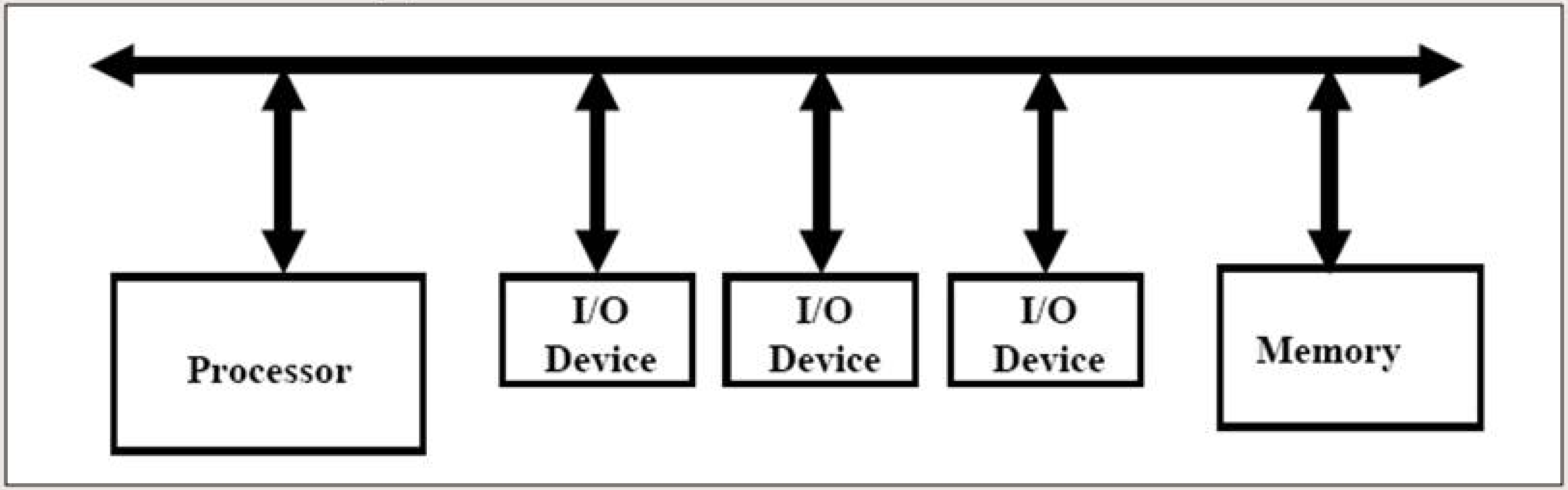
Advantages of Buses



- ❑ New devices can be added easily
- ❑ Peripherals can be moved between computer systems that use the same bus standard
- ❑ Low Cost: A single set of wires is shared in multiple ways

INTRODUCTION

Disadvantages of Buses



- ❑ It creates a communication bottleneck:
  - The bandwidth of the bus can limit the maximum I/O throughput
- ❑ The maximum bus speed is largely limited by:
  - The length of the bus
  - The number of devices on the bus

BUS

Characteristics

❑ We measure data transfer by two metrics:

- ❖ Total number of bits we can transfer in parallel. This is called the width of the data.
- ❖ The clock rate or frequency (in Hertz) of the bus

❑ Total Transfer Speed: Bandwidth

A bus with a width of 16 bits and a frequency of 133 MHz, has a transfer speed equal to:

$16 * 133*10^{6} = 2128*10^{6}$ bit/s,

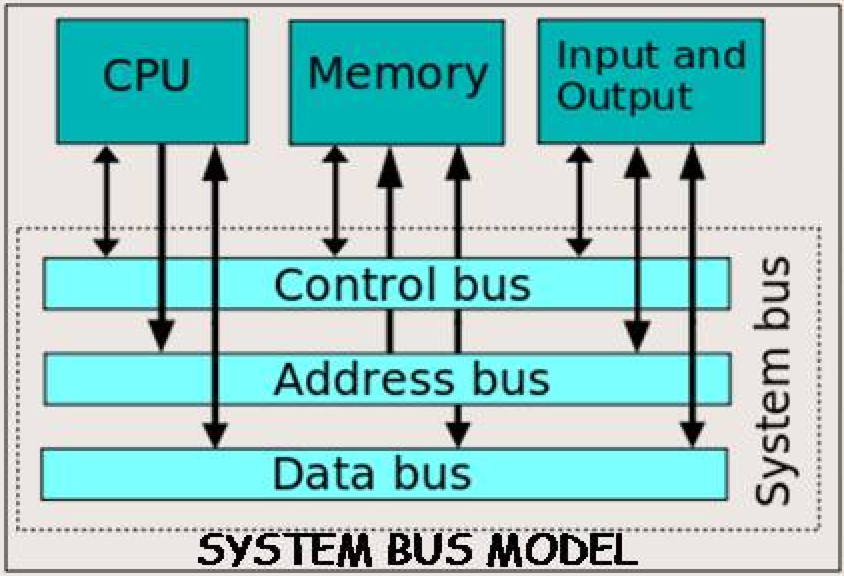
or $2128*10^{6}/8 = 266*10^{6}$ bytes/s

or $266*10^{6} /1000 = 266*10^{3}$ KB/s

or $266*10^{3} /1000 = 266$ MB/s

SYSTEM BUS

General Organization



SYSTEM BUS MODEL

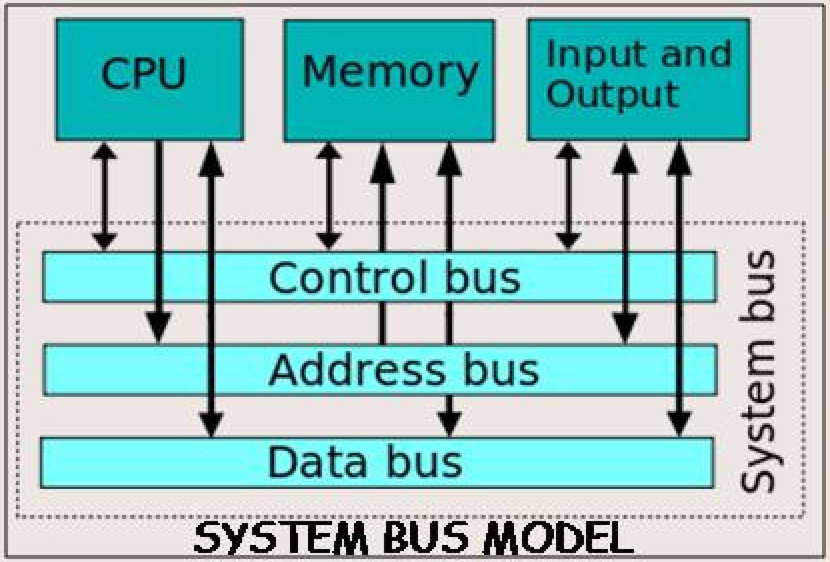
- ❑ A bus that connects major computer components (processor, memory, I/O) is called a system bus.

- ❑ System bus usually is separated into three functional groups .
  1. Data Bus
  2. Address Bus
  3. Control Bus

- ❑ In addition, there may be power distribution lines that supply power to the attached modules.

SYSTEM BUS

General Organization (DATA BUS)

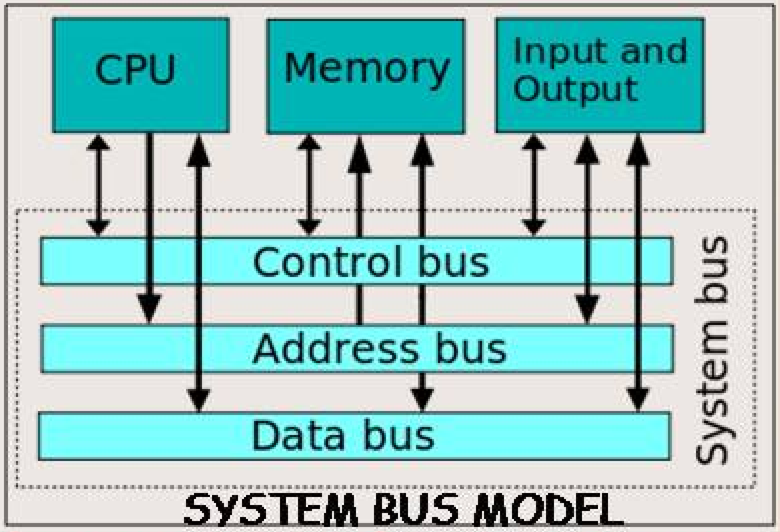


SYSTEM BUS MODEL

- □ It is a bidirectional bus.
- □ The size (width) of bus determines how much data can be transmitted at one time. e.g. :
  - 16-bit bus can transmit 2 bytes of data at a time.
  -32-bit bus can transmit 4 bytes) at a time.
- □ The size (width) of bus is a critical parameter in determining system performance.
- □ The wider the data bus, the better, but they are expensive.

SYSTEM BUS

General Organization (ADDRESS BUS)



SYSTEM BUS MODEL

- ❑ It is an unidirectional bus.
- ❑ A collection of wires used to identify particular location in main memory is called Address Bus.
- ❑ Width of the address bus determines the maximum possible memory capacity of the system.
- ❑ N address lines directly address 2^N memory locations.
  - ❖ 20 address lines could address 1 MB of memory
  - ❖ 32 address lines could address 4 GB of memory
  - ❖ 64 address lines could address 2^{64} bytes of memory

SYSTEM BUS

General Organization (CONTROL BUS)

❑ Because the data and address lines are shared by all components, there must be a means of controlling their use.

❑ The control lines regulates the activity on the bus.

❑ Control signals transmit both command and timing information among system modules.

❑ The control bus carries signals that report the status of various devices.

❑ Typical control bus signals are :

Memory Read : causes data from the addressed location to be placed on the data bus.

Memory Write : causes data on the bus to be written into the addressed location

I/O write: causes data on the bus to be output to the addressed I/O port

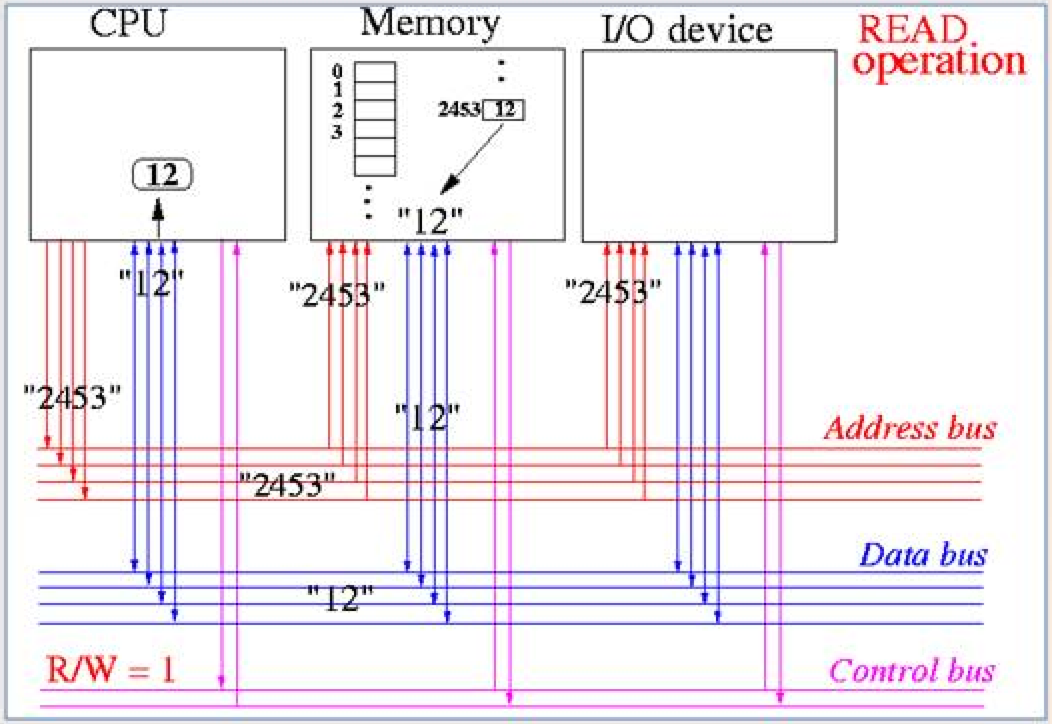
I/O read: causes data from the addressed I/O port to be placed on the bus

SYSTEM BUS

General Organization (CONTROL BUS)

Example : *Memory Read* (The following figure shows how the *CPU* reads the value 12 from the memory location 2453):

- ❑ *CPU* sends out the address value 2453 on the address bus
- ❑ Simultaneously, *CPU* sends out the signal R/W = 1 on the control bus, which indicates a READ operation
- ❑ *CPU* then waits for the data from memory on the data bus



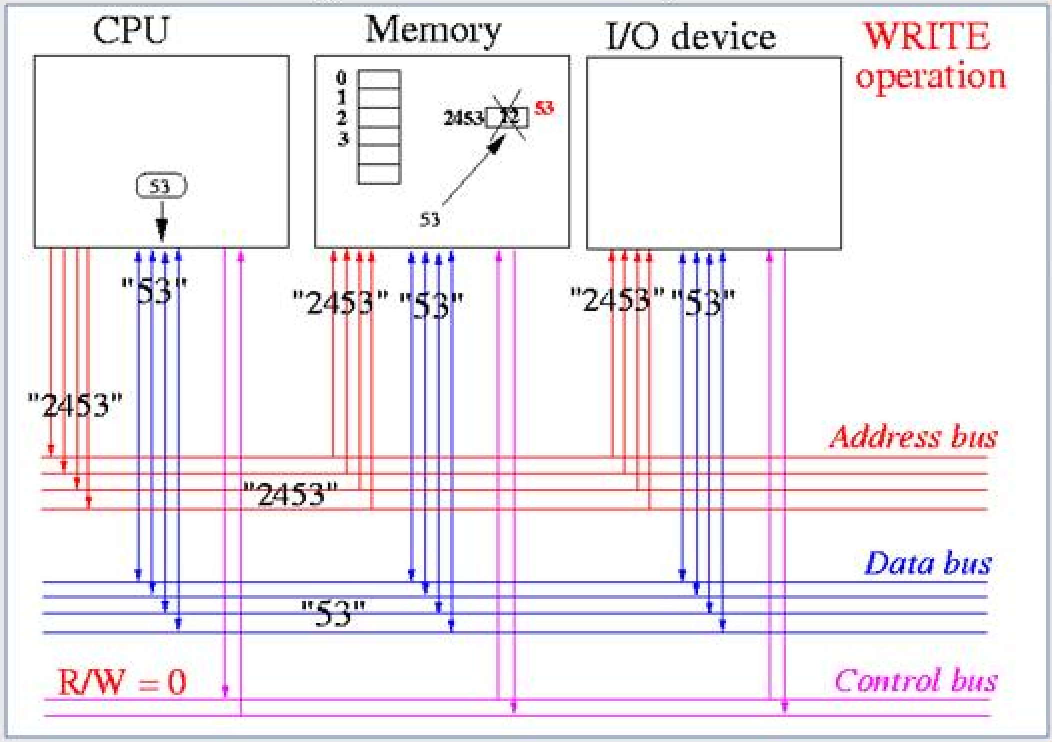
- ❑ The R/W = 1 signal and the address bus value 2453 will cause the memory to retrieve the value at memory location 2453 to be sent out on the data bus

SYSTEM BUS

General Organization (CONTROL BUS)

Example : Memory Write (The following figure shows how the CPU writes the value 53 into the memory location 2453):

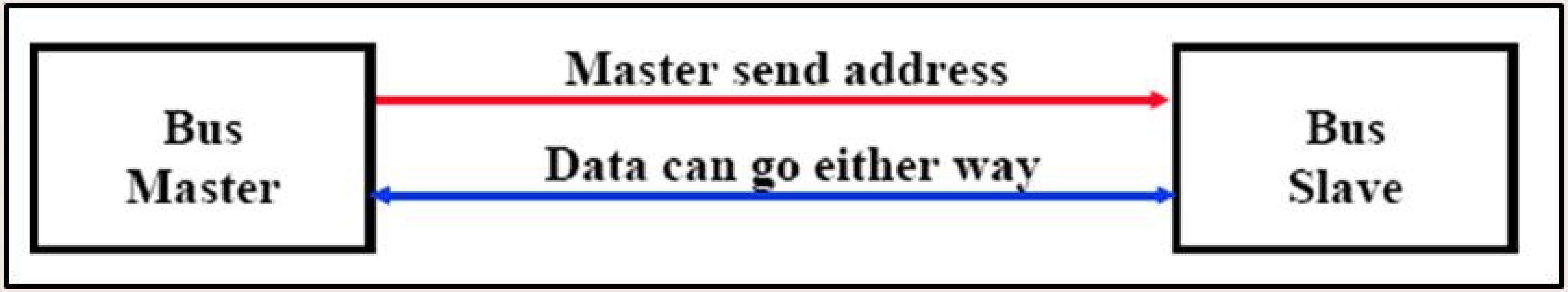
- ❑ CPU sends out the address value 2453 on the address bus
- ❑ Simultaneously, CPU also sends out the value 53 on the data bus, and the signal R/W = 0 on the control bus which indicating a WRITE operation



- ❑ The R/W = 0 signal along with the address bus value 2453 and data bus value 53 will cause the memory to store the value 53 at the location 2453…

SYSTEM BUS

Master versus Slave



- ❑ Master is the one who starts the bus transaction by:
  - ❖ Sending the address
- ❑ Slave is the one who responds to the address by:
  - ❖ Sending data to the master if the master ask for data
  - ❖ Receiving data from the master if the master wants to send data

SYSTEM BUS

Bus Design Issues

Need to consider several design issues :

- Bus width
  - Data and address buses.
- Bus type
  - Dedicated or multiplexed.
- Bus operations
  - Read, write, block transfer, interrupt, …
- Bus timing
  - Synchronous or asynchronous
- Bus arbitration
  - Centralized or distributed.

SYSTEM BUS

Bus Type

- □ Dedicated buses
  - ❖ Separate buses dedicated to carry data and address information.
  - ❖ Good for performance.

    But increases cost.
- □ Multiplexed buses
  - ❖ Data and address information is time multiplexed on a shared bus.
  - ❖ Poor Performance

    But Reduces cost.

SYSTEM BUS

Bus Type

- □ Time Multiplexed buses
  - ❖ Address and data information may be transmitted over the same set of lines using an Address Valid control line.
  - ❖ This method of using the same lines for multiple purposes is known as time multiplexing.
  - ❖ The advantage of time multiplexing is the use of fewer lines.

SYSTEM BUS

Bus Operations

- ❖ Basic operations

  Read and write.
- ❖ Block transfer operations.

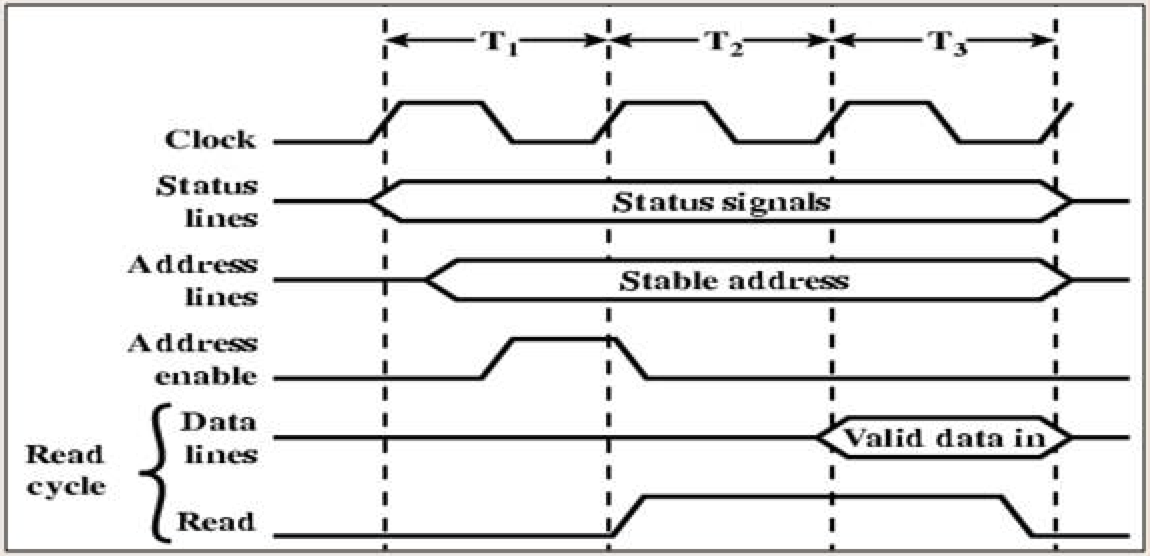
  Read or write several contiguous memory locations.

  Example: cache line fill.
- ❖ Interrupt operation.

SYSTEM BUS

Bus Timing Synchronous Timing - Read Diagram

- ❑ All other devices on the bus can read the clock line, and all events start at the beginning of a clock cycle.
- ❑ The processor places a memory address on the address lines and set status lines during the first clock cycle
- ❑ Once the address lines have stabilized, the processor issues an address enable signal.

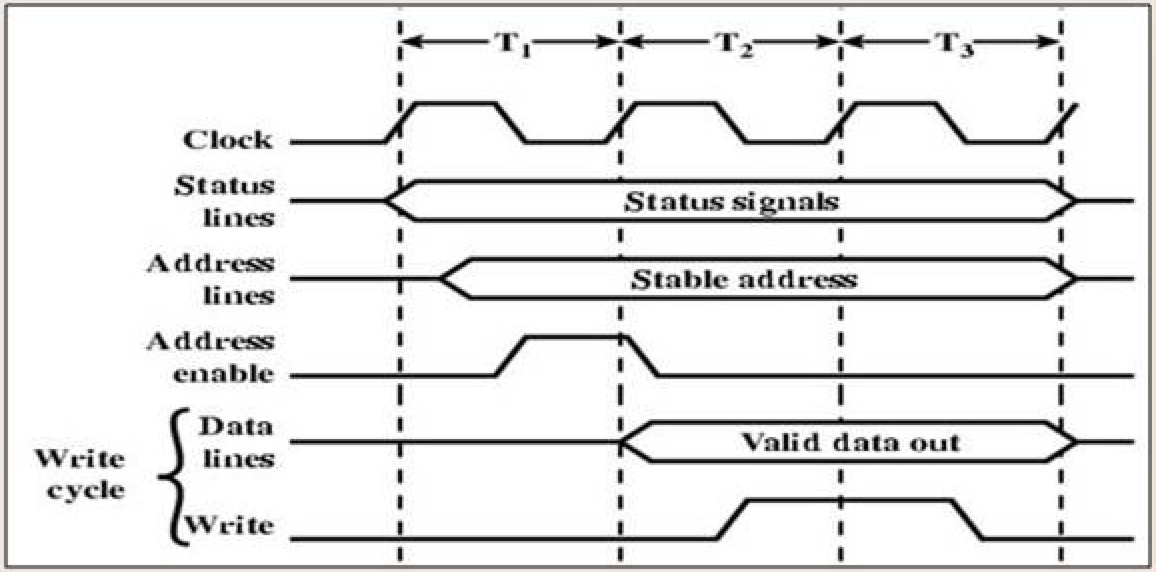


- ❑ The processor issues a read command at the start of the second cycle.
- ❑ A memory module recognizes the address and, after a delay of one cycle, places the data on the data lines.
- ❑ The processor reads the data from the data lines and drops the read signal.

SYSTEM BUS

Bus Timing Synchronous Timing - Write Diagram

- ❑ All other devices on the bus can read the clock line, and all events start at the beginning of a clock cycle.
- ❑ The processor places a memory address on the address lines and set status lines during the first clock cycle
- ❑ Once the address lines have stabilized, the processor issues an address enable signal.



- ❑ The processor puts the data on the data lines at the start of the second cycle and issues a write command after the data lines have stabilized.
- ❑ The memory module copies the information from the data lines during the third clock cycle.

SYSTEM BUS

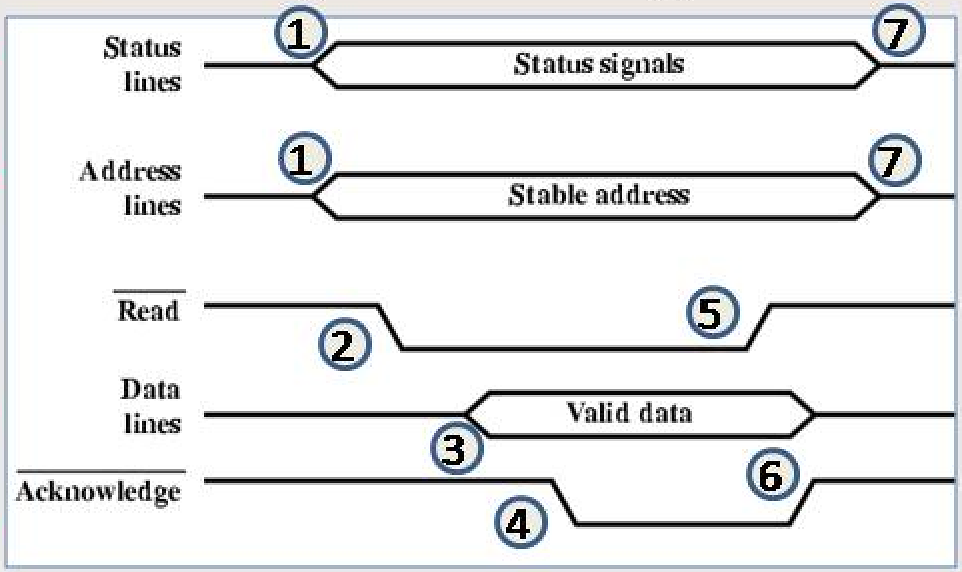
Bus Timing

Synchronous Bus:

- □ Includes a clock in the control lines
- □ A fixed protocol for communication that is relative to the clock
- □ Advantage: involves very little logic and can run very fast
- □ Disadvantages:
  - ❖ Every device on the bus must run at the same clock rate
  - ❖ To avoid clock skew, they cannot be long if they are fast

SYSTEM BUS

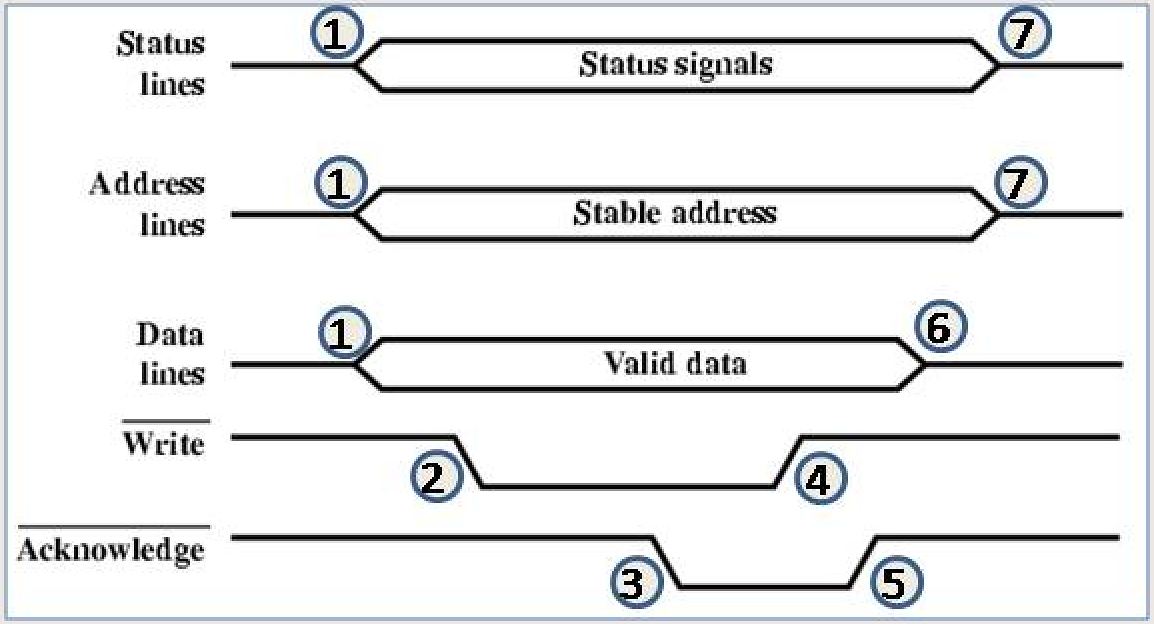
Bus Timing Asynchronous Timing - Read Diagram



- ❑ The processor places address and status signals on the bus.
- ❑ After that, it issues a read command, indicating the presence of valid address and control signals.
- ❑ The appropriate memory decodes the address and responds by placing the data on the data line.
- ❑ Once the data lines have stabilized, the memory module set the acknowledged line to signal the processor that the data is available.
- ❑ Once the master has read the data from the data lines, it reset the read signal.
- ❑ This causes the memory module to drop the data and acknowledge lines.
- ❑ Finally, once the acknowledge line is dropped, the master removes the address information.

SYSTEM BUS

Bus Timing Asynchronous Timing - Write Diagram



- ❑ The master places the data on the data line at the same time that it puts signals on the status and address lines.
- ❑ The memory module responds to the write command by copying the data from the data lines and then set the acknowledge line.
- ❑ The master then drops the write signal and the memory module drops the acknowledge signal.

SYSTEM BUS

Bus Timing

Asynchronous Bus:

- ❑ It is not clocked
- ❑ It can accommodate a wide range of devices
- ❑ It can be lengthened without worrying about clock skew
- ❑ It requires a handshaking protocol

SYSTEM BUS

Bus Arbitration

- In a computer system there may be more than one bus master such as processor, DMA controller etc. They share the system bus.
- When current master relinquishes control of the bus, another bus master can acquire the control of the bus.
- Bus arbitration is the process by which the next device to become the bus master is selected and bus mastership is transferred to it.
- There are two approaches to bus arbitration:
    - Centralized.
    - Distributed.

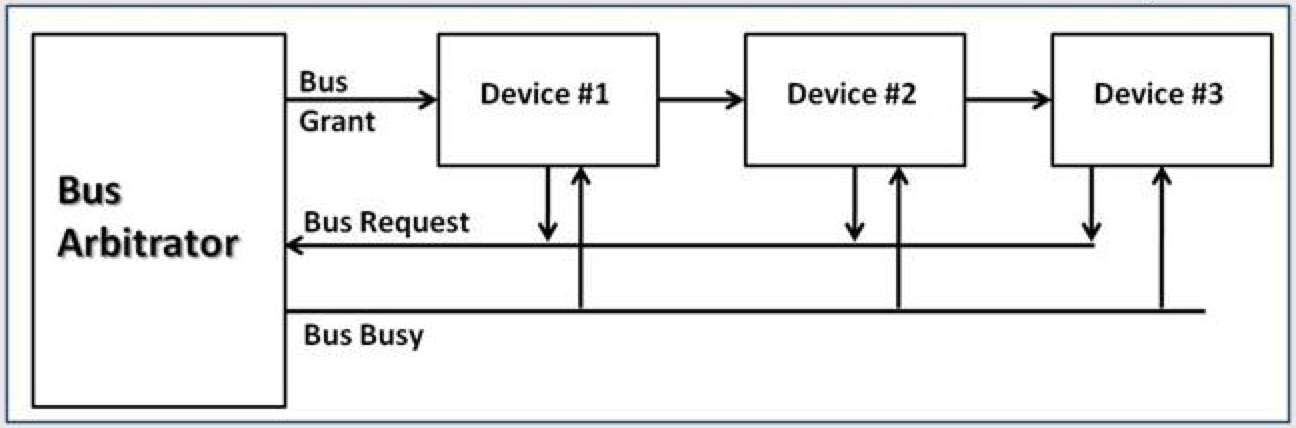
SYSTEM BUS

Bus Arbitration - Centralized

- ❑ In centralized bus arbitration, a single bus arbiter performs the required arbitration.
- ❑ The bus arbiter may be the processor or a separate controller connected to the bus.
- ❑ There are three centralized arbitration schemes:
  - ❖ Daisy chaining
  - ❖ Polling
  - ❖ Independent requesting

SYSTEM BUS

Bus Arbitration - Centralized - Daisy Chain

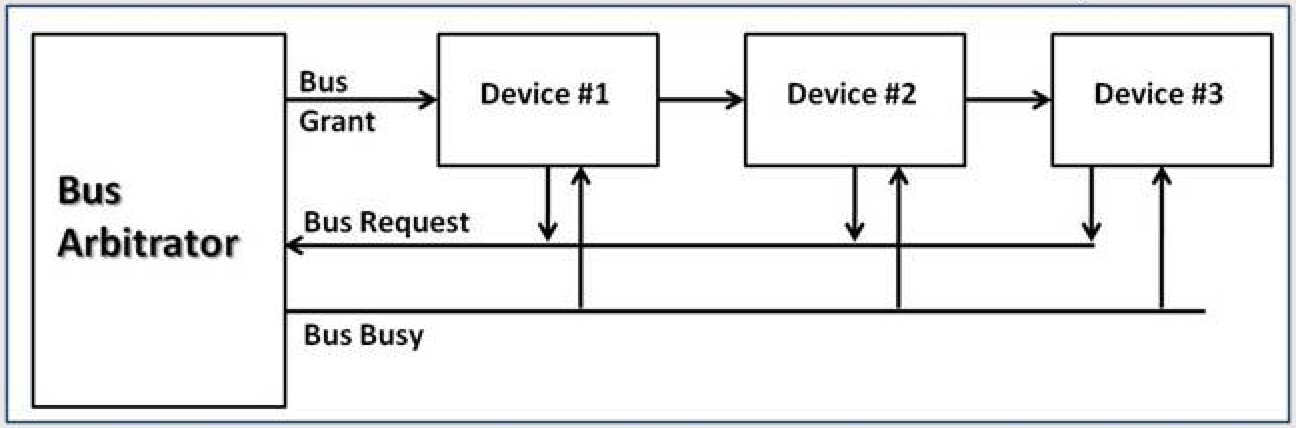


- ❑ All requesting components are attached serially to the bus.
- ❑ This method involves three control signals:

  1- BUS REQUEST 2- BUS GRANT 3- BUS BUSY
- ❑ All the bus units are connected to BUS REQUEST line. When activated, it indicates that one or more devices are requesting to use the bus.
- ❑ Bus Controller responds to a BUS REQUEST only if BUS BUSY is inactive. When bus control is given to requesting device, it enables its physical bus connection and activates BUS BUSY.

SYSTEM BUS

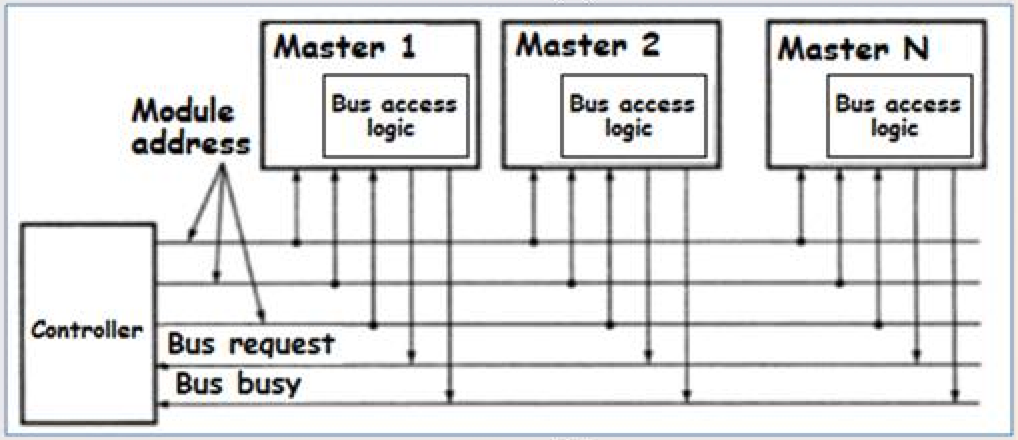
Bus Arbitration - Centralized - Daisy Chain



- ❑ When the 1st requesting device gets control of the bus and receives BUS GRANT signal, it blocks further propagation of signals, activates BUS BUSY and begins to use bus.
- ❑ When a non requesting device receives BUS GRANT signal, it forwards the signal to next device.
- ❑ Thus if two devices simultaneously request bus access, the device that is closer to the bus controller receives BUS GRANT and receives the bus control.
- ❑ Means the devices that are closed to the bus controller are of higher priority than those of the other devices.

SYSTEM BUS

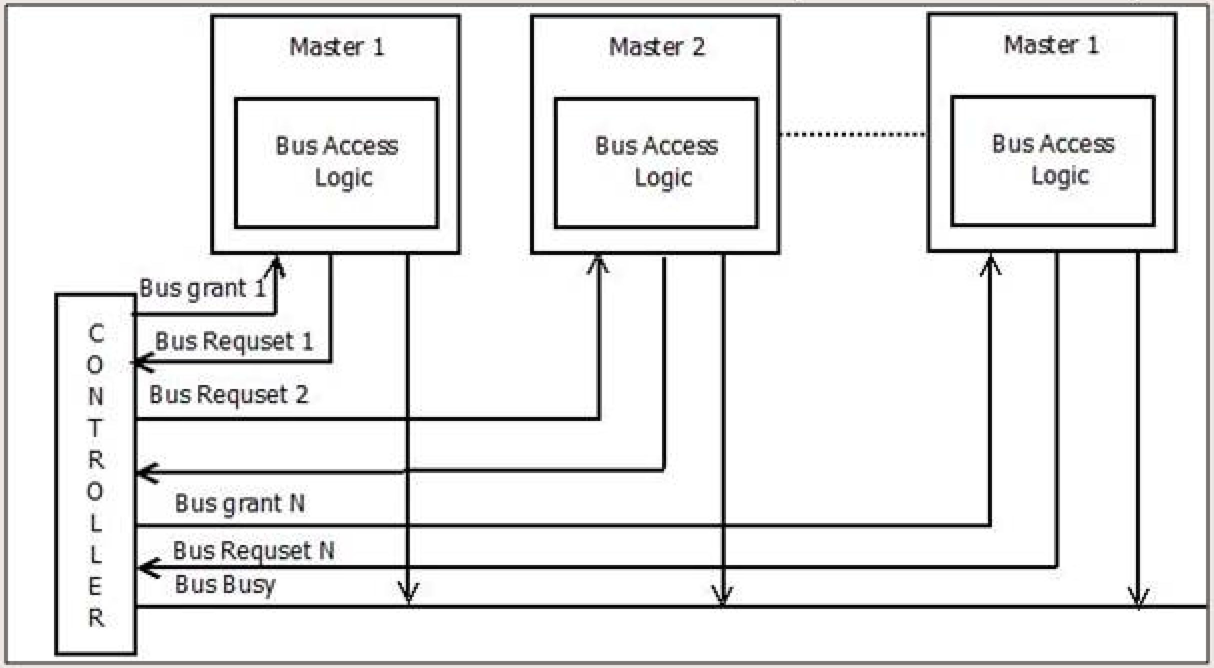
Bus Arbitration - Centralized - Polling



- ❑ Devices request access to the bus via a common BUS REQUEST line.
- ❑ In respond to BUS REQUEST, Bus controller generates a sequence of numbers on the poll count lines.
- ❑ Each device compares these numbers as their device address already assigned to them.
- ❑ When a requesting device finds that its address matches the numbers on the poll-count lines, the device activates BUS BUSY.
- ❑ The bus controller responds by terminating the polling process and the device connects to the bus.

SYSTEM BUS

Bus Arbitration - Centralized - Independent requesting



- In this scheme each master has a separate pair of BUS REQUEST and BUS GRANT lines and each pair has a priority assigned to it.
- The built in priority decoder within the controller selects the highest priority request and asserts the corresponding BUS GRANT signal.

SYSTEM BUS

Bus Arbitration - Distributed

- In distributed arbitration, All devices participate in the selection of the next bus master.
- In this scheme each device on the bus is assigned a 4-bit identification number. Device having highest ID has highest priority.
- When one or more devices request for the control of bus, each device place their 4-bit ID numbers on arbitration lines, ARB0 through ARB3
- To identify highest ID number from the status of bus line, each device compares the code formed on the arbitration line to its own ID, starting from the most significant bit. If it finds the different at any bit position, it disables its drive at that bit position and for all lower-order bits.
- The decentralized arbitration offers high reliability because operation of the bus is not dependent on single device.

SYSTEM BUS

Bus Arbitration - Distributed

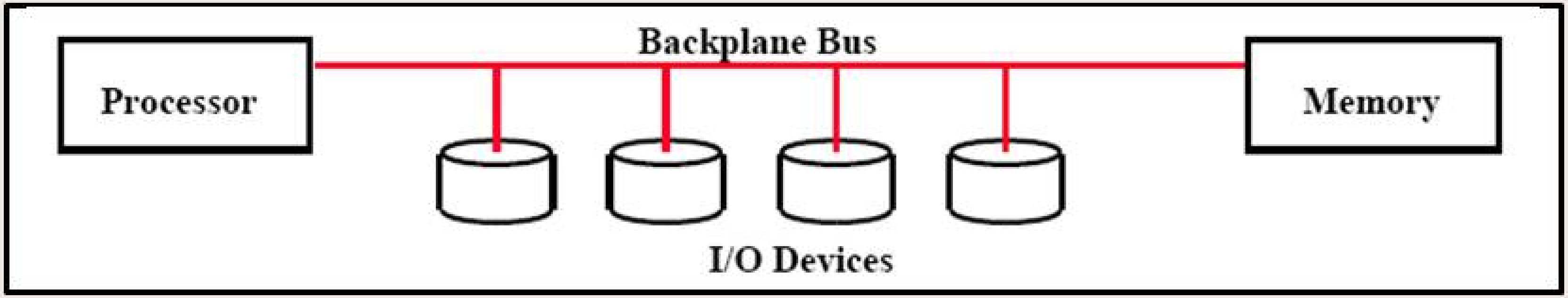
Example:

- □ Consider that two devices A and B, having ID number 1 and 6, respectively are requesting the use of the bus

Device A puts the bit pattern (ID) => 0001
Device B puts the bit pattern (ID) => 0110
Status of bus-line seen by both devices is 0111 (A or B)

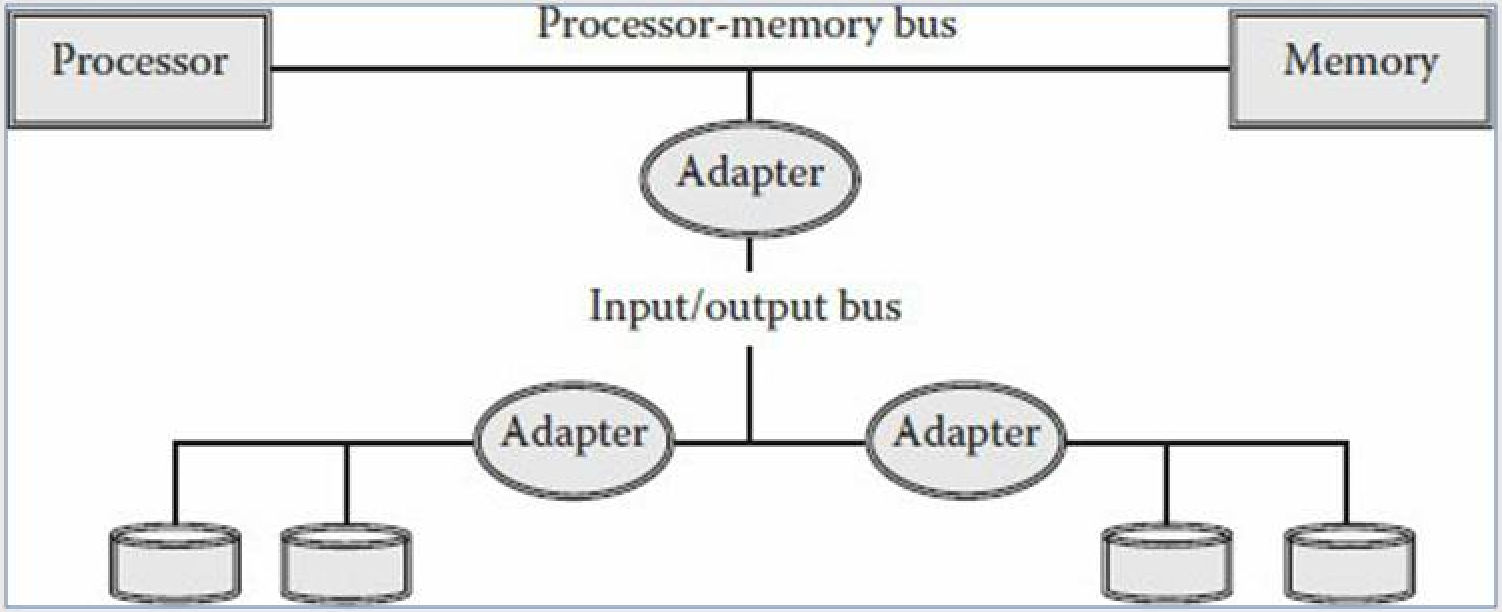
- □ Device 1 compares the ID formed on the arbitration line to its own ID, starting from the most significant bit. Detects a different on line ARB2 and hence set ARB2,ARB1,ARB0 to 0.Then ID seen by device 1 is 0000 which is not his code.
- □ Device 6 compares the ID formed on the arbitration line to its own ID. ID seen by device 6 is 0110 which is device its device ID. This means that device B has won the race.

Backplane Bus



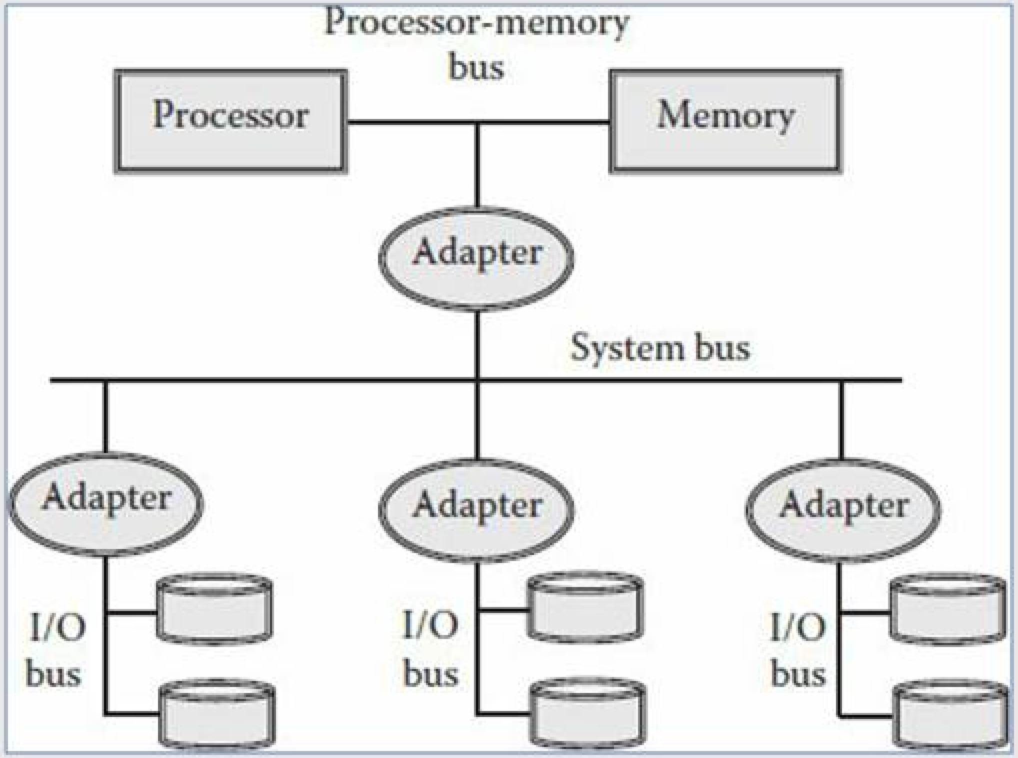
- ❑ Backplane: an interconnection structure within the chassis
- ❑ Cost advantage: one single bus for all components
  - ❖ A Computer System with One Bus
  - ❖ Single bus (the backplane bus) is used for:
    - ➢ Processor to memory communication
    - ➢ Communication between I/O devices and memory
  - ❖ Example: IBM PC

Two-Bus System



- ❑ I/O buses tap into the system bus via bus adaptors:
  - System bus: mainly for processor-memory traffic
  - I/O buses: provide expansion slots for I/O devices
- ❑ Apple Macintosh-II
  - NuBus: Processor, memory, and a few selected I/O devices
  - SCCI Bus: the rest of the I/O devices

Three-Bus System



- ❑ A small number of backplane buses tap into the processor-memory bus
  - ❖ Processor-memory bus is used for processor memory traffic
  - ❖ I/O buses are connected to the backplane bus
- ❑ Advantage: loading on the processor bus is greatly reduced