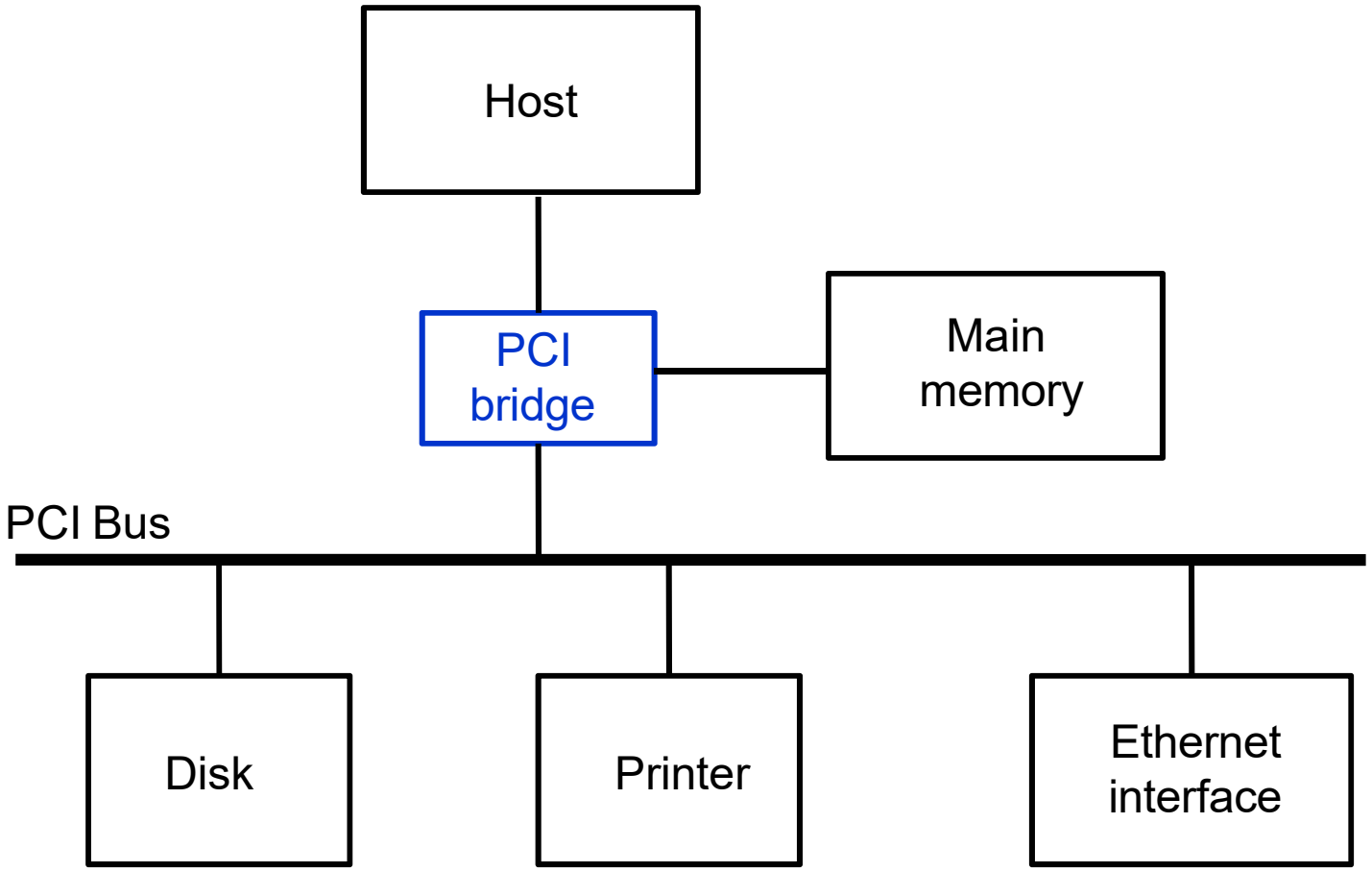
Standard I/O Interfaces

- The processor bus is the bus defined by the signals on the processor chip itself. Devices that require a very high speed connection to the processor, such as the main memory, may be connected directly to this bus
- The motherboard usually provides another bus that can support more devices.
- The two buses are interconnected by a circuit, which we called a bridge, that translates the signals and protocols of one bus into those of the other
- It is impossible to define a uniform standards for the processor bus. The structure of this bus is closely tied to the architecture of the processor
- The expansion bus is not subject to these limitations, and therefore it can use a standardized signaling structure

Peripheral Component Interconnect Bus

- Use of a PCI bus in a computer system



- Peripheral Component Interconnect (PCI) is one of the latest developments in bus architecture and is the current standard for PC expansion cards.

- It was developed by Intel and launched as the expansion bus for the Pentium processor in 1993.

- It is a local bus like VESA i.e. it connects the CPU, memory and peripherals to wider, faster data pathway.

- PCI supports both 32-bit and 64-bit data width; therefore it is compatible with 486s and Pentiums.

- The bus data width is equal to the processor, for example, a 32 bit processor would have a 32 bit PCI bus, and operates at 33MHz.

- PCI was used in developing Plug and Play (PnP) and all PCI cards support PnP i.e. the user can plug a new card into the computer, power it on and it will “self identify” and “self specify” and start working without manual configuration using jumpers.

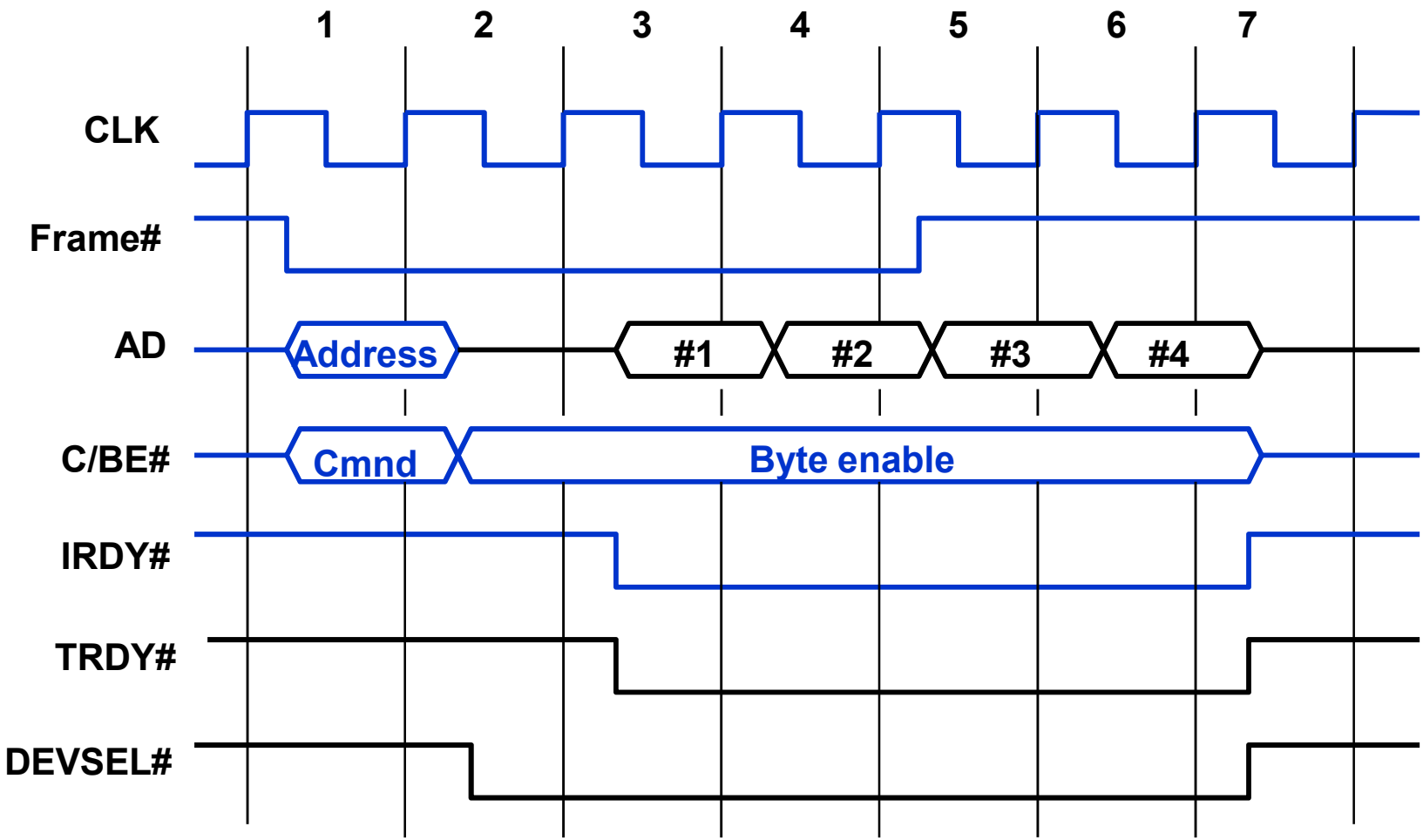
- Unlike VESA, PCI supports bus mastering that is, the bus has some processing capability and therefore the CPU spends less time processing data.

- The bus support three independent address spaces: memory, I / O, and configuration.
- The I / O address space is intended for use with processors, such Pentium, that have a separate I / O address space.
- However, the system designer may choose to use memory-mapped I / O even when a separate I / O address space is available
- The configuration space is intended to give the PCI its plug-and-play capability.
  - A 4-bit command that accompanies the address identifies which of the three spaces is being used in a given data transfer operation

Data Transfer Signals on the PCI Bus

Name	Function
CLK	A 33-MHz or 66MHz clock
FRAME#	Sent by the initiator to indicate the duration of a transaction
AD	32 address/data lines, which may be optionally increased to 64
C/BE#	4 command/byte-enable lines (8 for 64-bit bus)
IRDY#, TRDY#	Initiator-ready and Target-ready signals
DEVSEL#	A response from the device indicating that it has recognized its Address and is ready for a data transfer transaction
IDSEL#	Initialization Device Select

A Read Operation on the PCI Bus



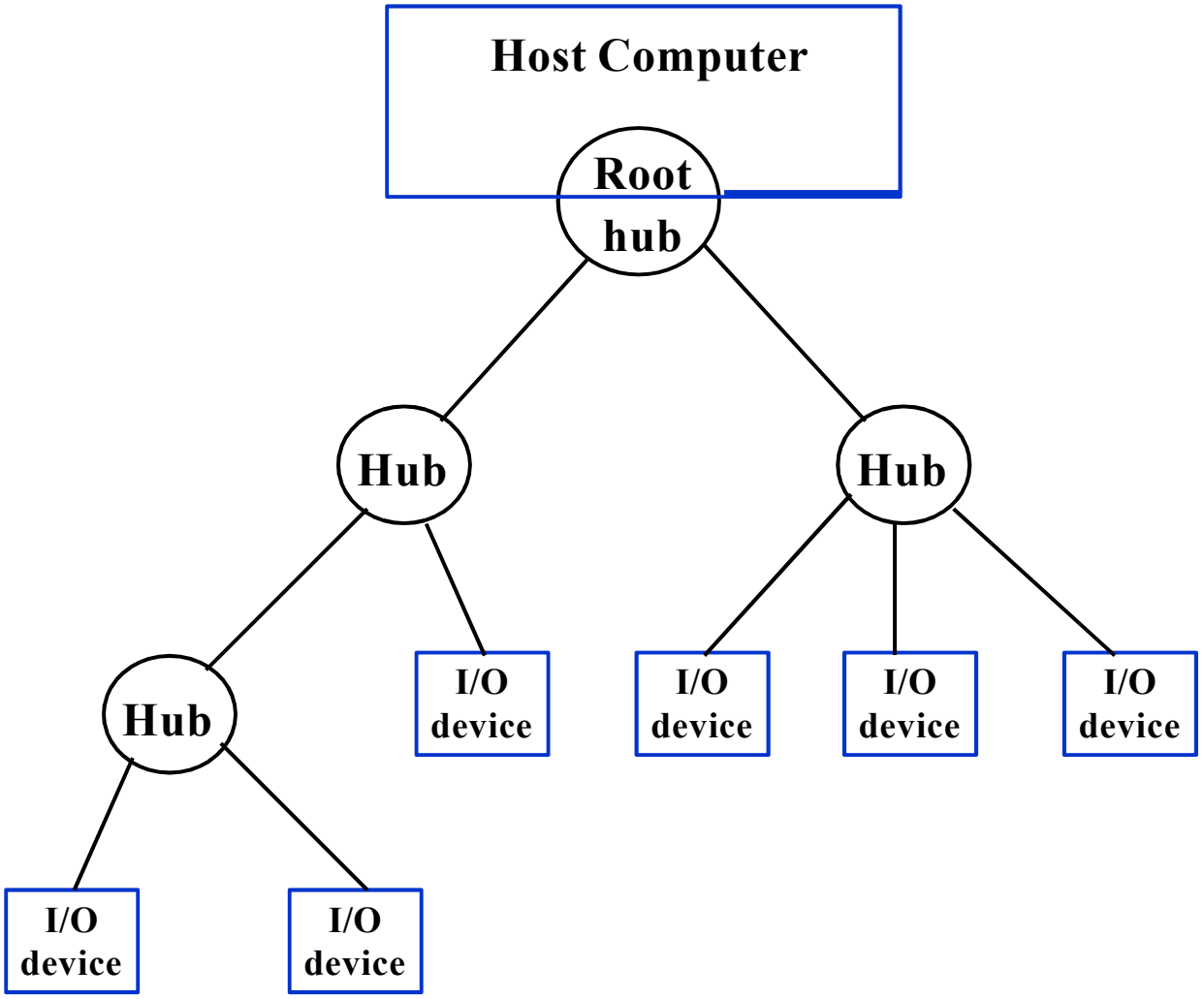
Universal Serial Bus (USB)

- The USB has been designed to meet several key objectives
  - Provide a simple, low-cost, and easy to use interconnection system that overcomes the difficulties due to the limited number of I / O ports available on a computer
  - Accommodate a wide range of data transfer characteristics for I / O devices, including telephone and Internet connections
  - Enhance user convenience through a “plug-and-play” mode of operation

USB Structure

- A serial transmission format has been chosen for the USB because a serial bus satisfies the low-cost and flexibility requirements
- Clock and data information are encoded together and transmitted as a single signal
  - Hence, there are no limitations on clock frequency or distance arising from data skew
- To accommodate a large number of devices that can be added or removed at any time, the USB has the tree structure
  - Each node of the tree has a device called a hub, which acts as an intermediate control point between the host and the I / O device
  - At the root of the tree, a root hub connects the entire tree to the host computer

USB Tree Structure



USB Tree Structure

- The tree structure enables many devices to be connected while using only simple point-to-point serial links
- Each hub has a number of ports where devices may be connected, including other hubs
- In normal operation, a hub copies a message that it receives from its upstream connection to all its downstream ports
  - As a result, a message sent by the host computer is broadcast to all I / O devices, but only the addressed device will respond to that message
- A message sent from an I / O device is sent only upstream towards the root of the tree and is not seen by other devices
  - Hence, USB enables the host to communicate with the I / O devices, but it does not enable these devices to communicate with each other

USB Protocols

- All information transferred over the USB is organized in packets, where a packet consists of one or more bytes of information
- The information transferred on the USB can be divided into two broad categories: control and data
  - Control packets perform such tasks as addressing a device to initiate data transfer, acknowledging that data have been received correctly, or indicating an error
  - Data packets carry information that is delivered to a device. For example, input and output data are transferred inside data packets