# Corso di Architettura dei Sistemi a Microprocessore

## *BUS*



**Luigi Coppolino**

# Contact info

Prof. Luigi Coppolino
luigi.coppolino@uniparthenope.it

Università degli Studi di Napoli "Parthenope"
Dipartimento di Ingegneria

Centro Direzionale di Napoli, Isola C4
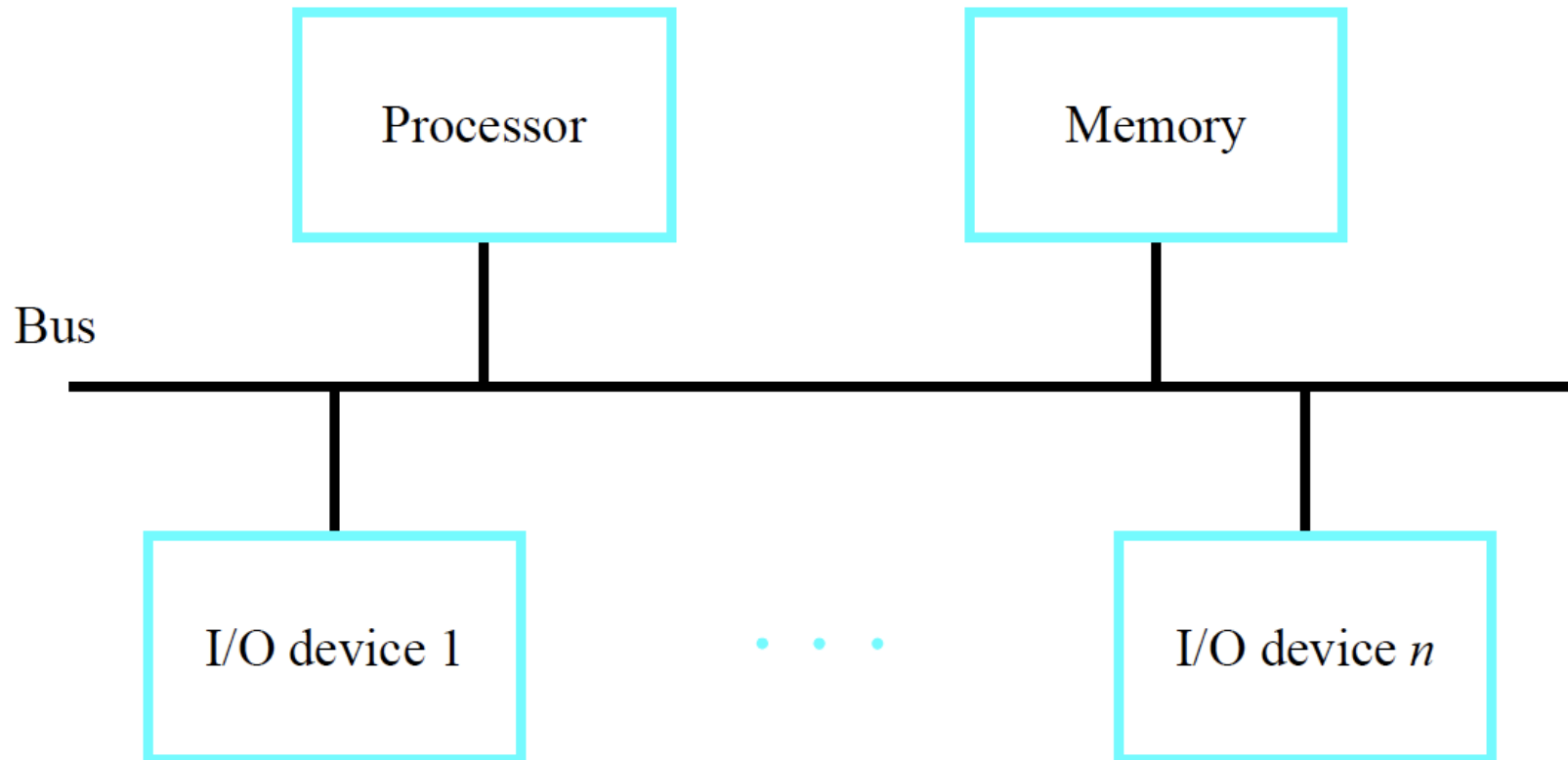V Piano lato SUD - Stanza n. 512
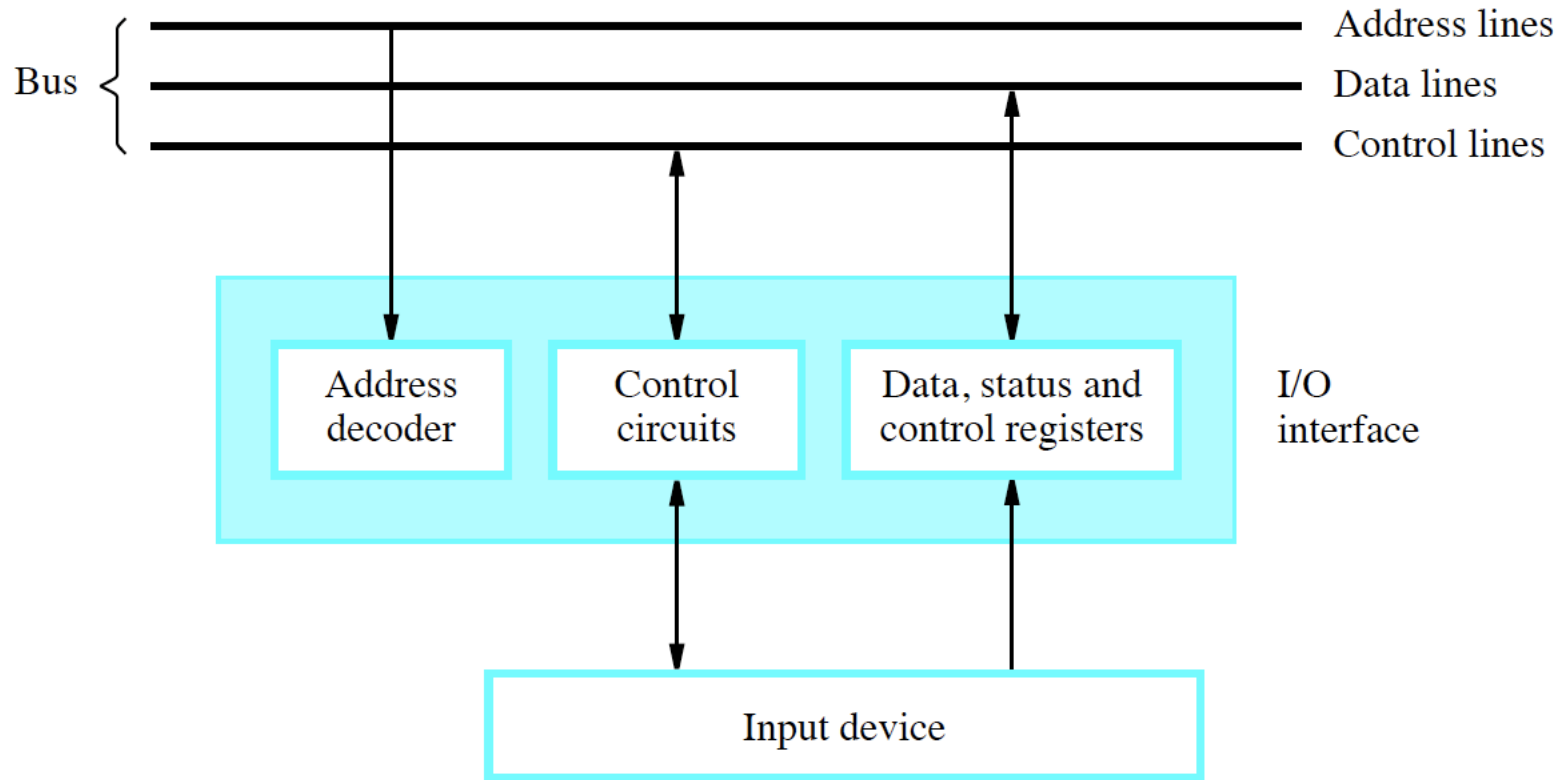
Tel: +39-081-5476702
Fax: +39-081-5476777

# *Chapter Outline*

➢ Hardware needed to access I/O devices

➢ Synchronous and asynchronous bus operation

➢ Interface circuits

➢ Commercial standards, such as USB, SAS and PCI Express

# *A Single-Bus System*

# I/O interface for an input device

# *I/O interface for an input device*

➢ Interface circuit responds when it recognizes its address.

➢ Needs to know exactly when to check address information and when to send input data or receive output data.

➢ Control circuit coordinates the interface's operation.

➢ Two approaches to timing of bus transfers:

- o Synchronous
- o Asynchronous

# Input transfer timing on a synchronous bus
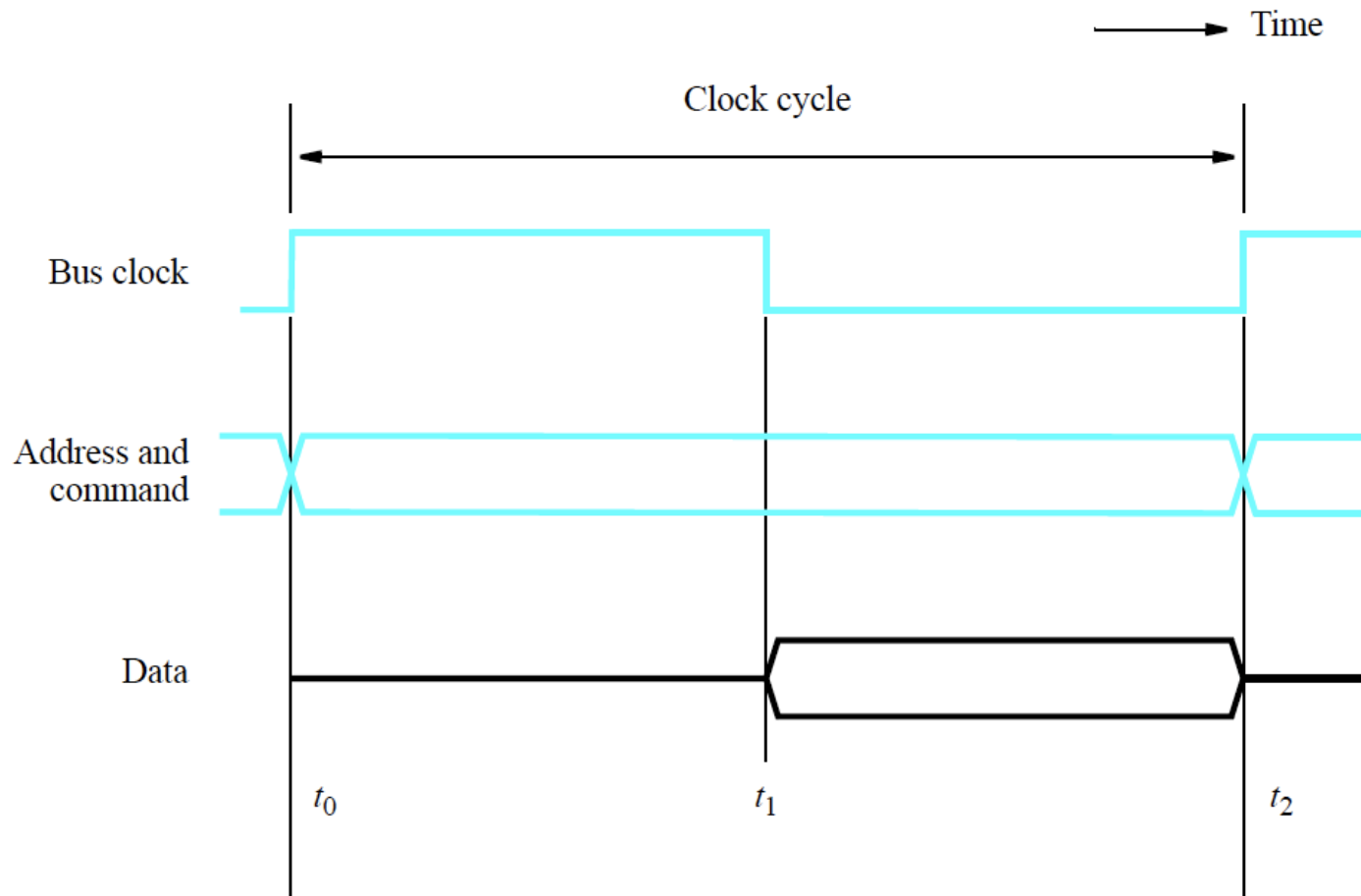
# *Input transfer timing on a synchronous bus*

➢ A delay is introduced by the bus drivers that place new information on the address or data lines.

➢ Additional delay is encountered as information propagates along the bus.

➢ Receiver requires a minimum amount of setup time to receive data correctly

# A detailed timing diagram for the input transfer
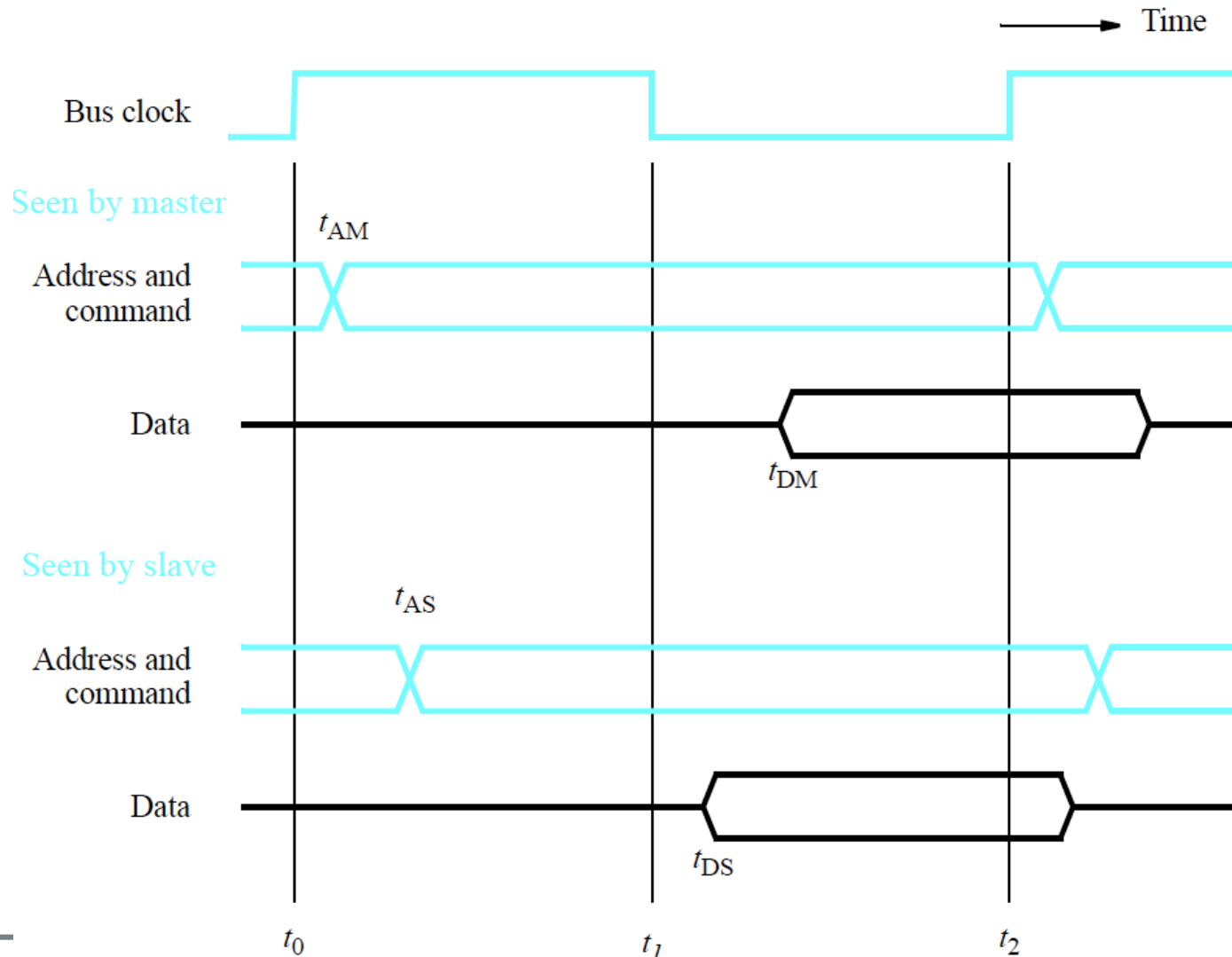
# A detailed timing diagram for the input transfer

➢ Not all devices operate at the same speed.

➢ After sending a transfer request, some bus protocols allow device interface to respond in a later clock cycle.

➢ In this case, the clock period need not be designed to accommodate the slowest device.

# An input transfer using multiple clock cycles
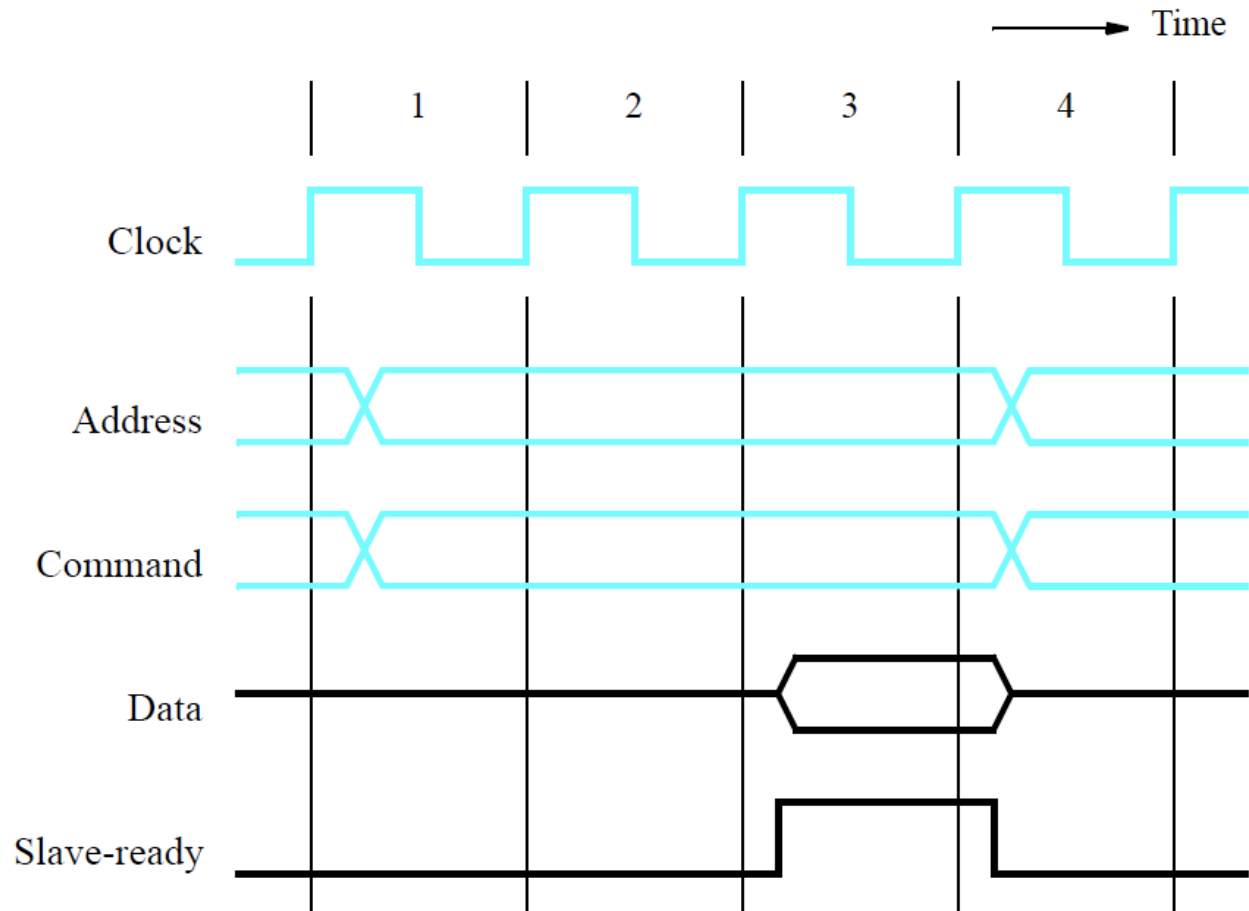
# *Asynchronous protocol*

➢ Timing automatically adjusts to delays.

➢ Data transfer is controlled by two interlocked signals: Master-ready and Slave-ready.

➢ Whenever the processor takes an action, it waits for the device interface to respond before taking the next action, and vice-versa.
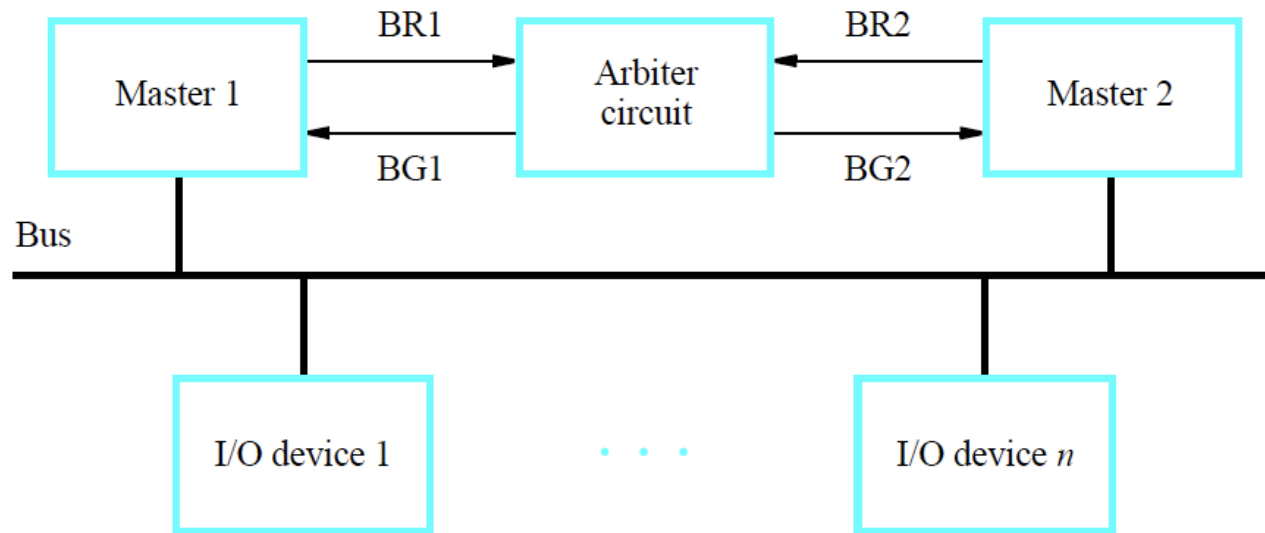
# Handshake control: input operation
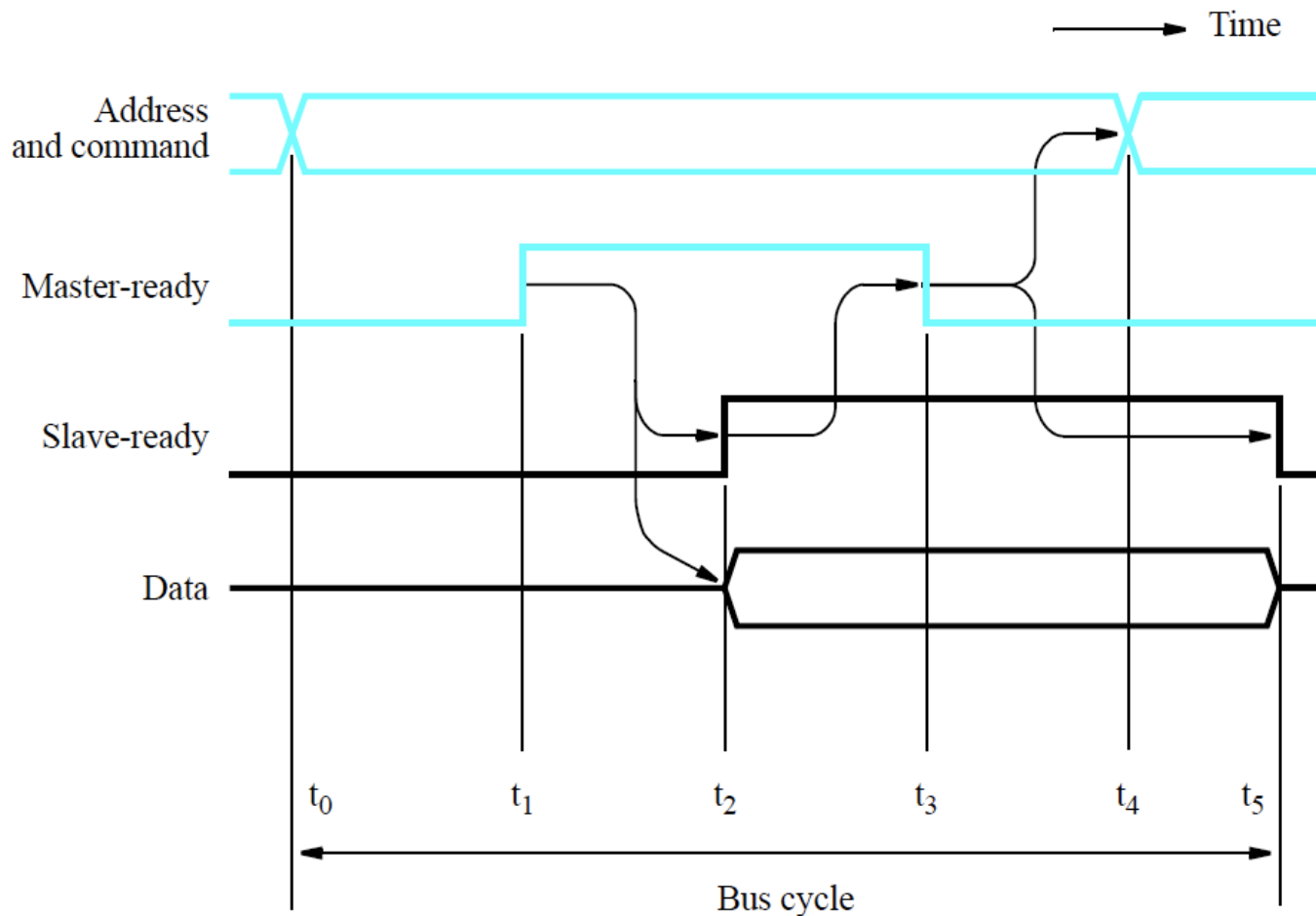
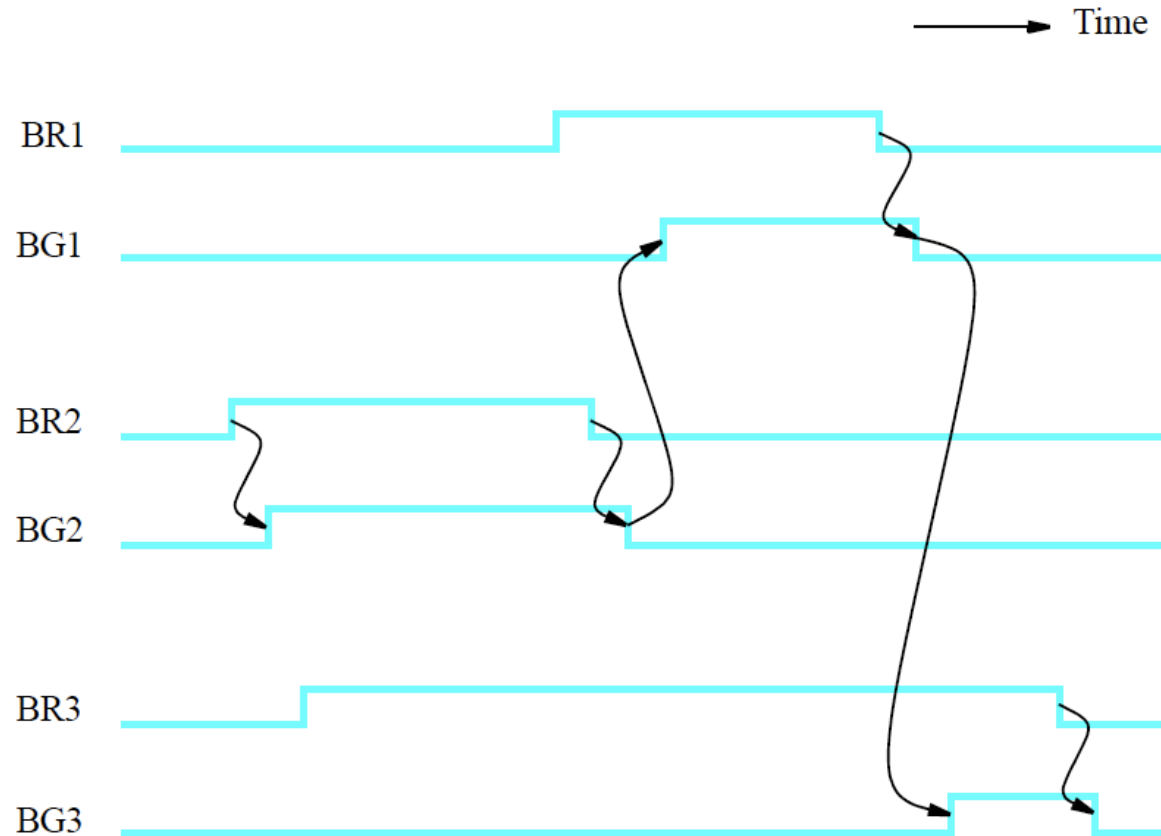# *Handshake control: output operation*
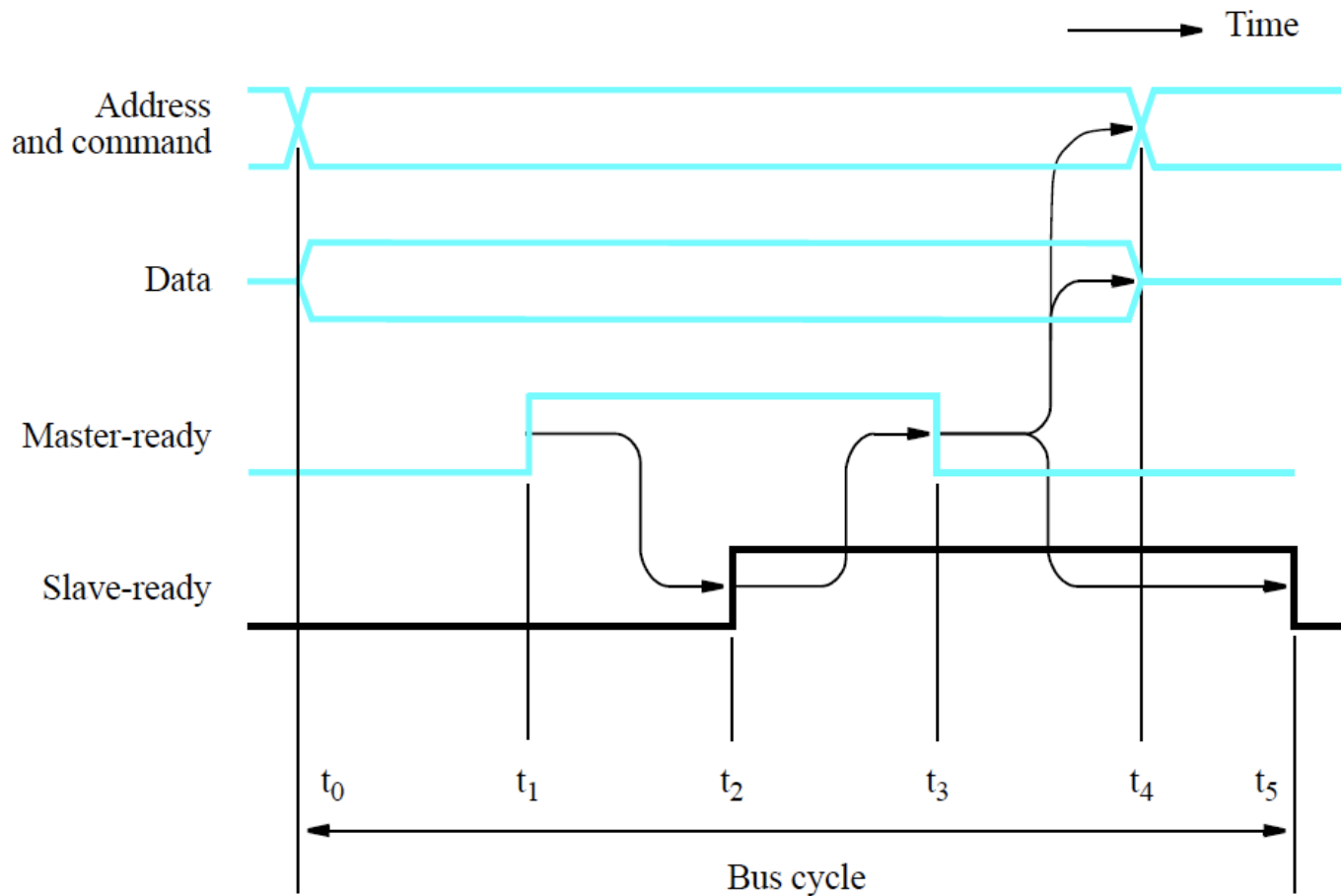
# Bus Master

➢ The device initiating read and write requests is called a bus master.

➢ Devices request bus mastership.

➢ An arbiter grants the bus to the highest priority device.

➢ Control lines on the bus are used to request and grant the bus.

# Bus arbitration

# Granting the bus, priorities: BR1 > BR2 > BR3

# *Examples of interface circuits*

➤ Consider an interface circuit that connects a keyboard to the bus of a processor.

➤ When a key is pressed, the character code is loaded into an interface register, and a status bit is set.

➤ These data are sent to the processor in response to a read operation.

# Keyboard-to-processor connection
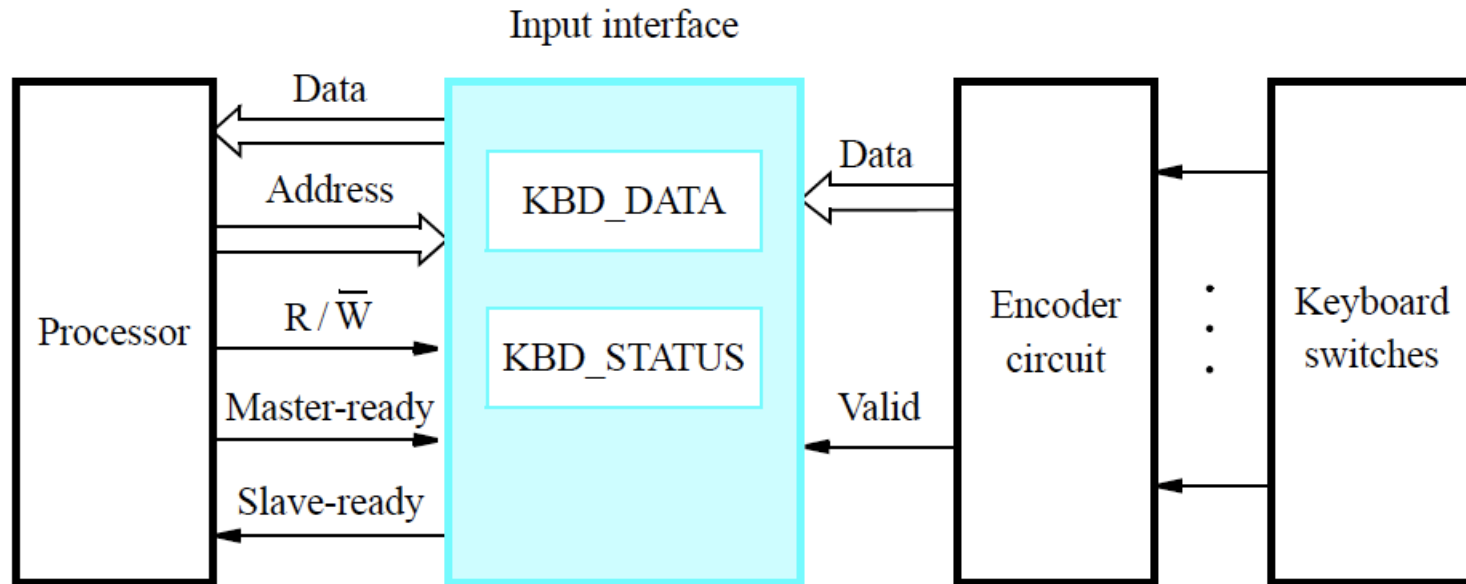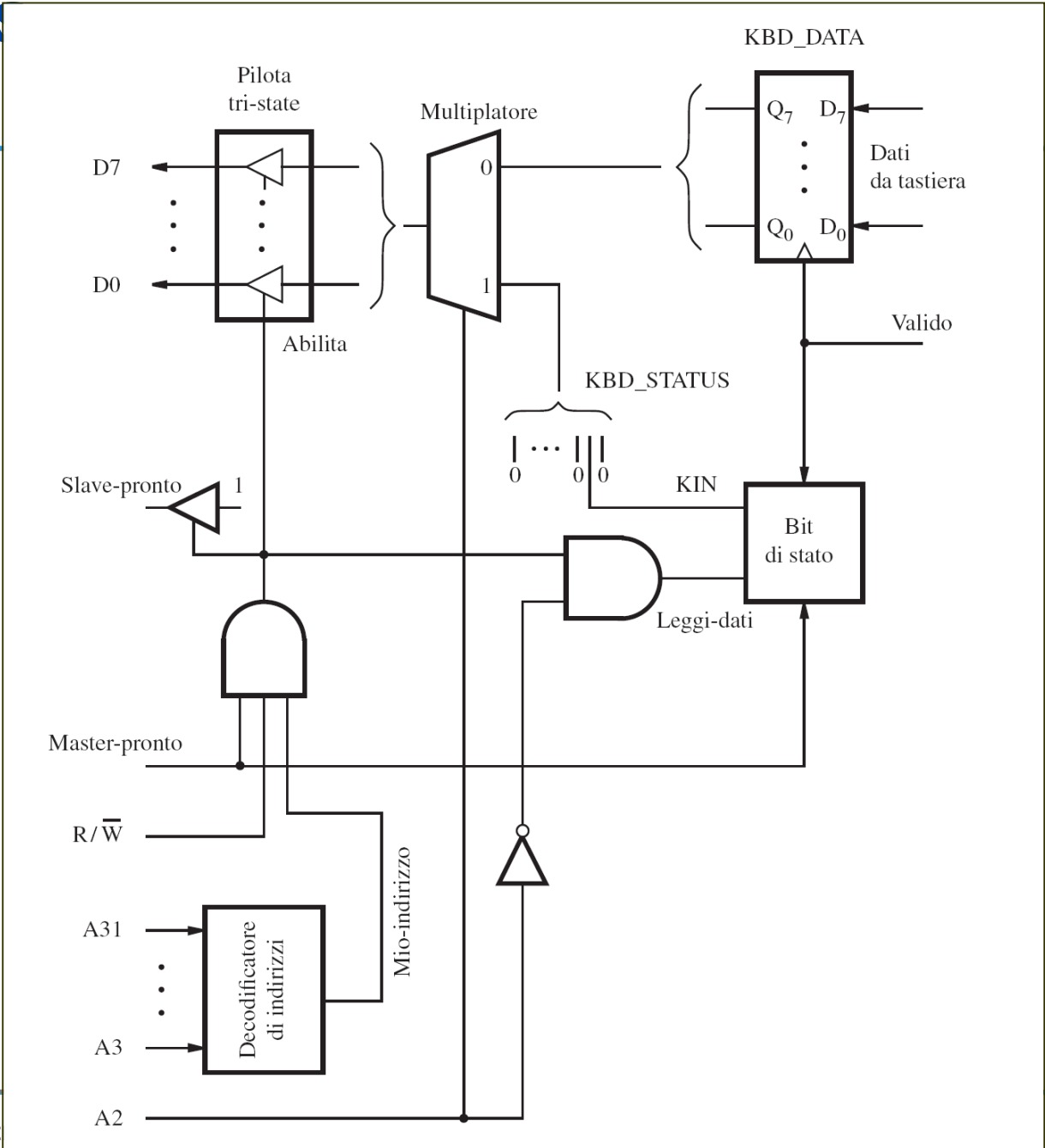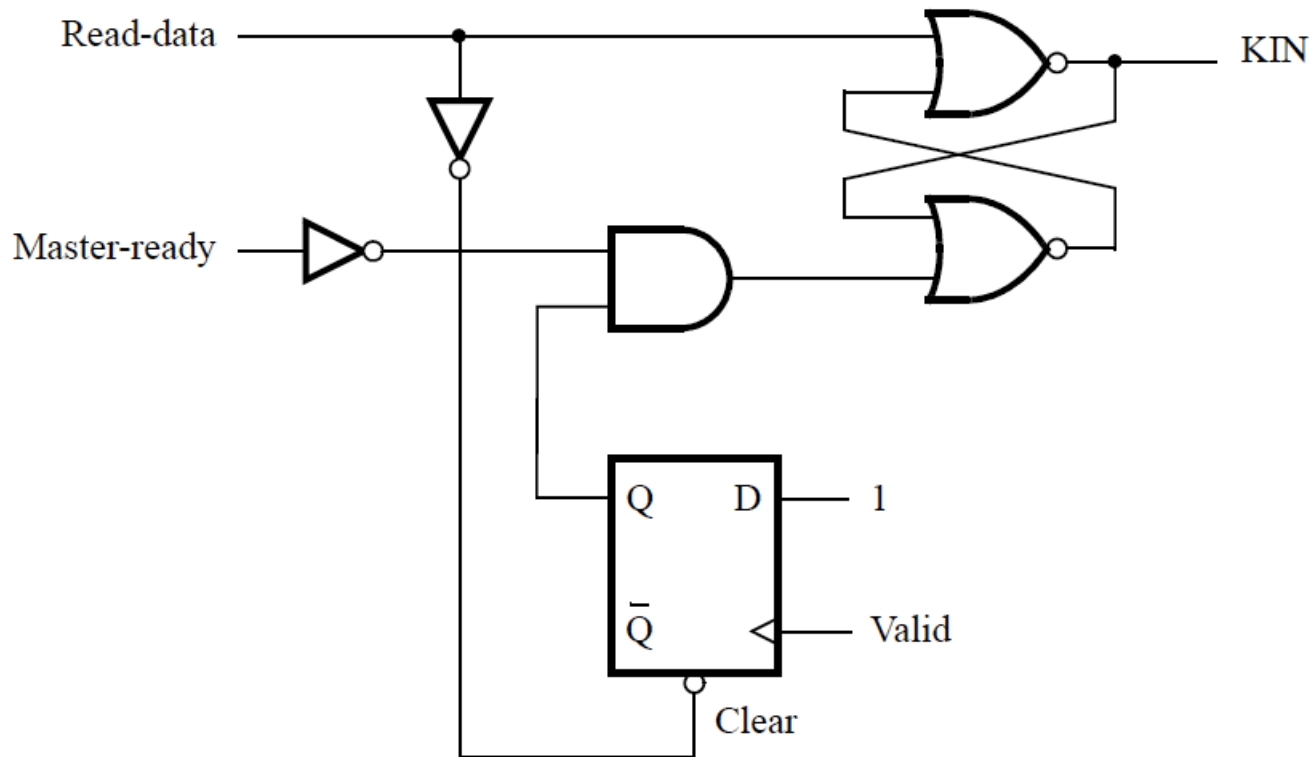
# Data and status information



**Figura 7.11**  Un circuito di interfaccia d'ingresso.

# *Detail of status flag control*

➢ KIN is set by Valid and cleared by a read operation, but only when Master-ready is not asserted.

# An output interface

➢ Data are written into DISP-DATA register.

➢ Transferred to the display under control of New-data.

➢ Ready sets status bit.

Output interface

DISP_DATA



> An implementation of an output interface

FITNESS
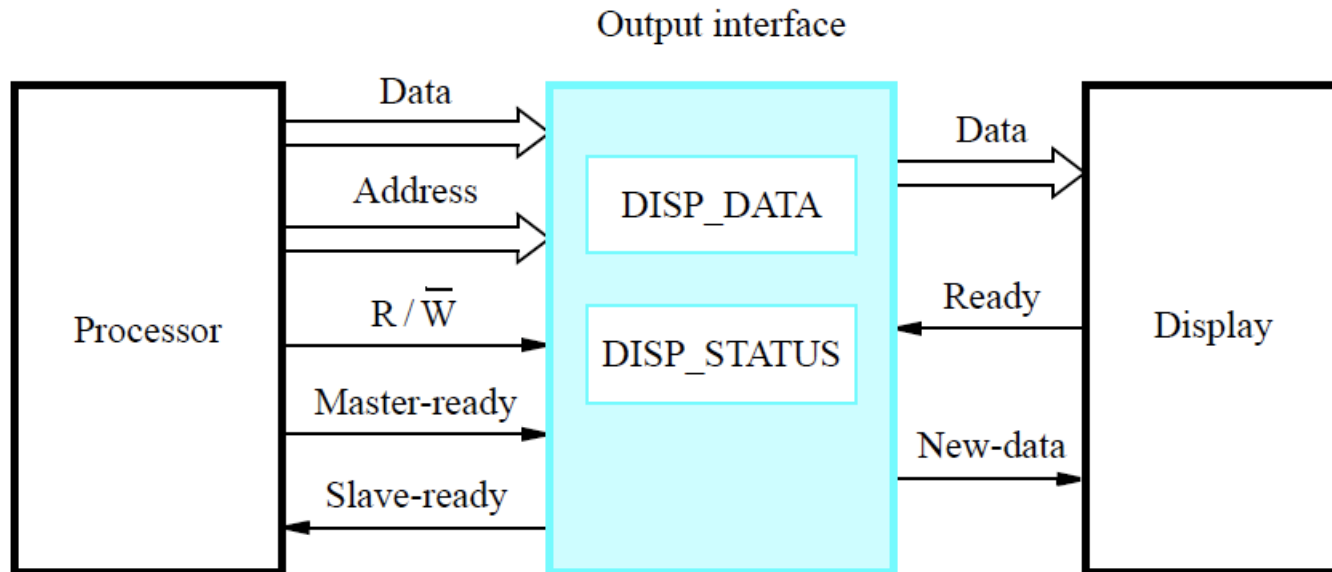
# *Serial links*

➤ Many I/O interconnections use serial data transmission.

  o More suitable for longer distances

  o Less expensive

➤ Data are transmitted one bit at a time.

➤ Requires a means for the receiver to recover timing information.

➤ A simple scheme for low-speed transmission is known as "start-stop," using a Universal Asynchronous Receiver Transmitter (UART).

# UART
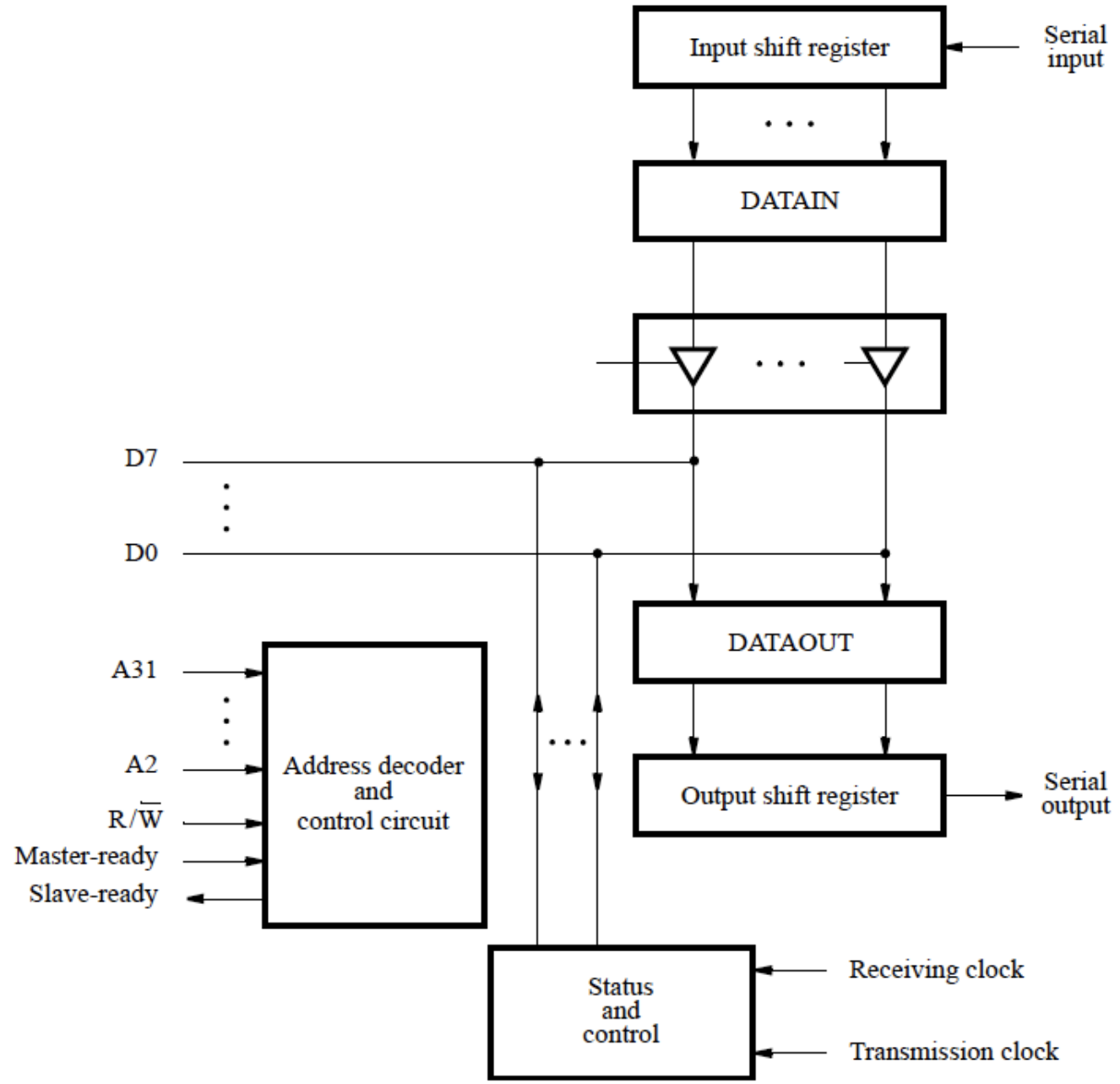
# *Start-stop transmission*

➢ Start bit signals the beginning of a character.
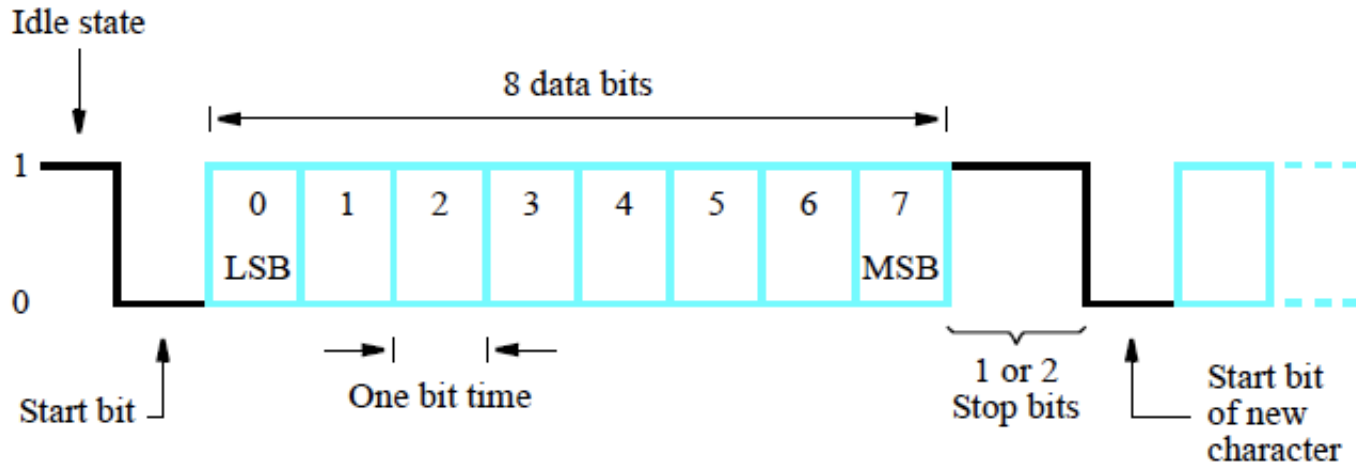
# I/O interconnection standards

- ➢ Standards facilitate system integration from a variety of sources.
- ➢ They encourage the development of many plug-compatible devices.
- ➢ They help industry grow.
- ➢ Perhaps the most commonly encountered I/O standard is USB
- ➢ USB is found on many devices:
  Memory keys, printers, external disk drives, cameras, etc.

# USB (Universal SERIAL BUS )

- ➤ Point-to-point connections
- ➤ Tree structure
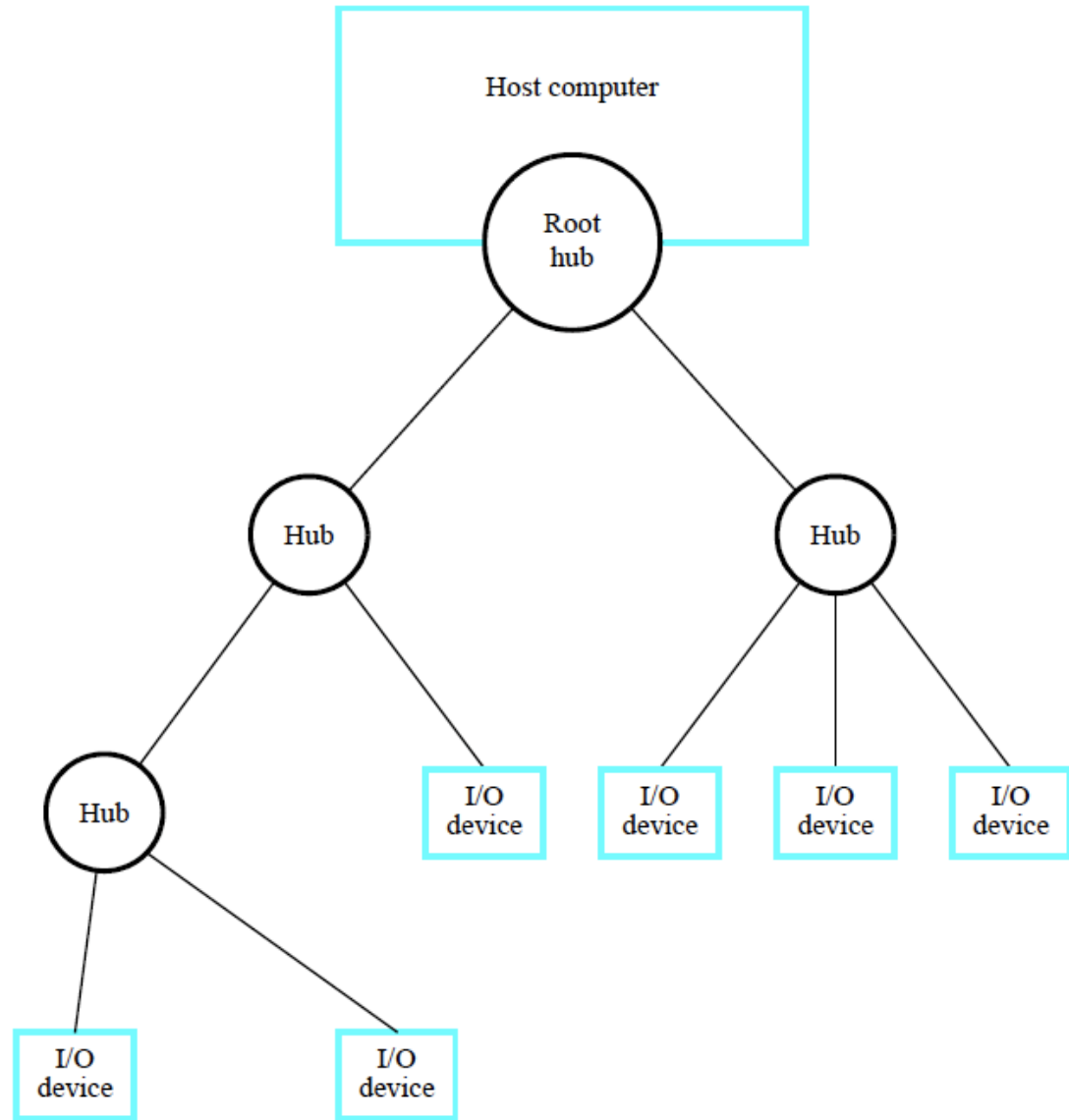
# USB features

➢ Point-to-point connections using two twisted pairs

➢ Wide range of speeds

➢ Plug-and-play

➢ Supports a wide variety of devices:

  o Asynchronous, e.g. mouse, keyboard, memory key

  o Isochronous: e.g. sampled audio

➢ Several versions, with speeds up to 5 Gb/s

# PCI (Peripheral Component Interconnect) bus

➢ Used most often on the motherboard of a computer.
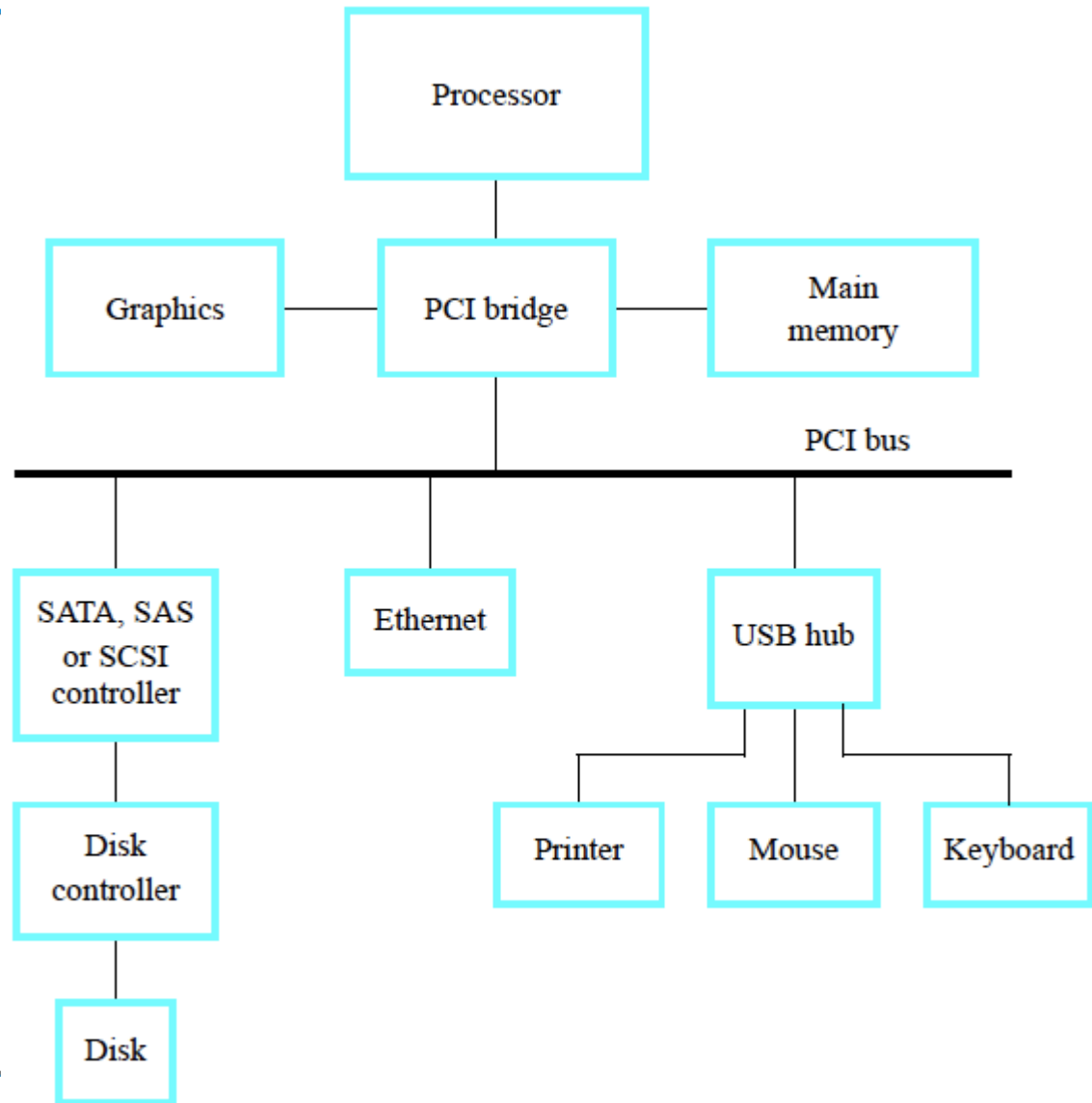
# An input operation on the PCI bus
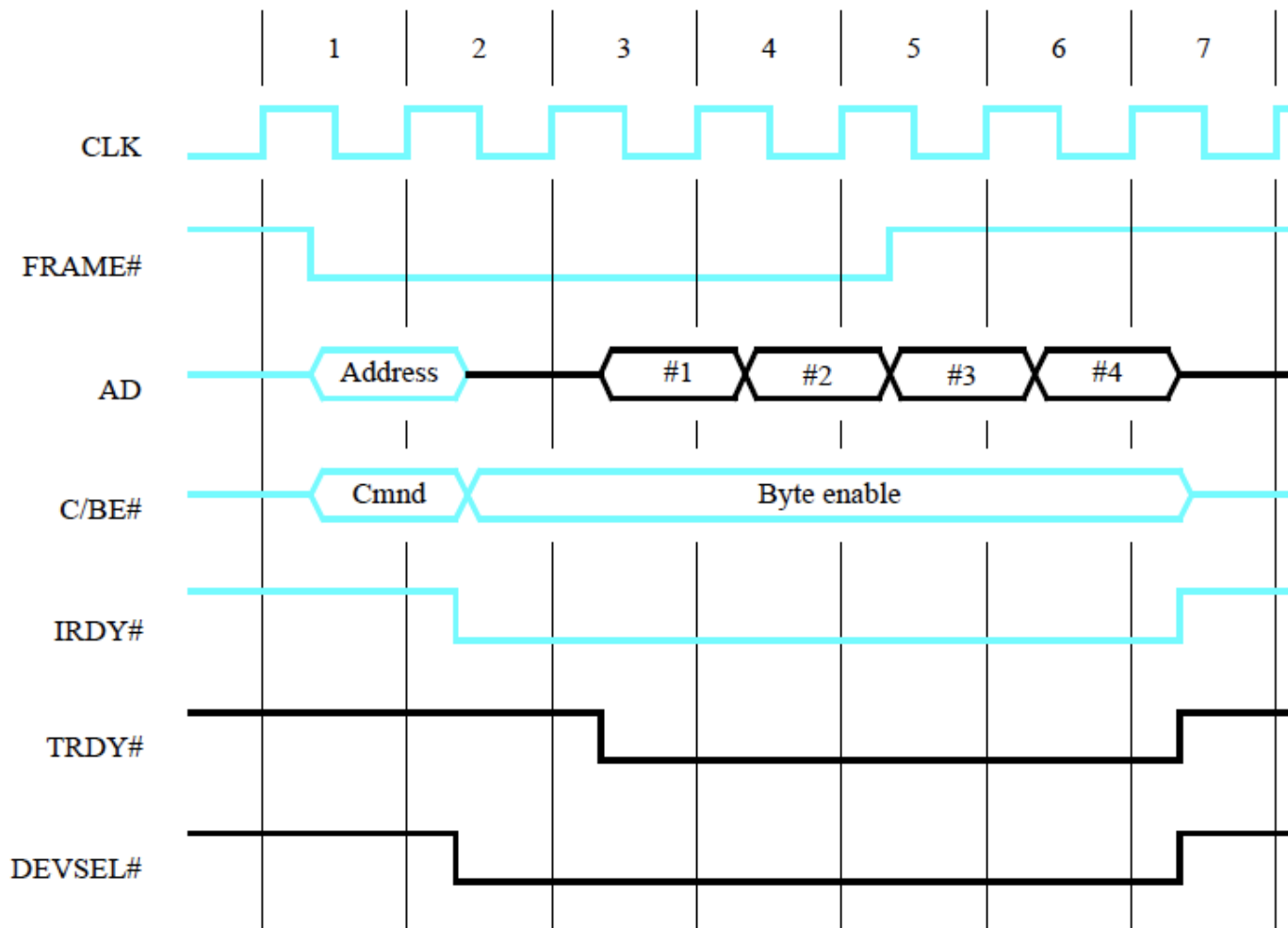
# *Plug-and-play*

- ➢ PCI pioneered the plug-and-play feature, made possible by the bus's initial connection protocol.

- ➢ There are up to 21 device connectors on the PCI bus.

- ➢ Each PCI-compatible device has a small ROM with information on the device characteristics.

- ➢ Processor scans all connectors to determine whether a device is plugged in.

- ➢ It assigns an address to each device and reads the contents of its ROM.

- ➢ With this information, it selects the appropriate device driver software, performs any initialization that may be needed, etc.
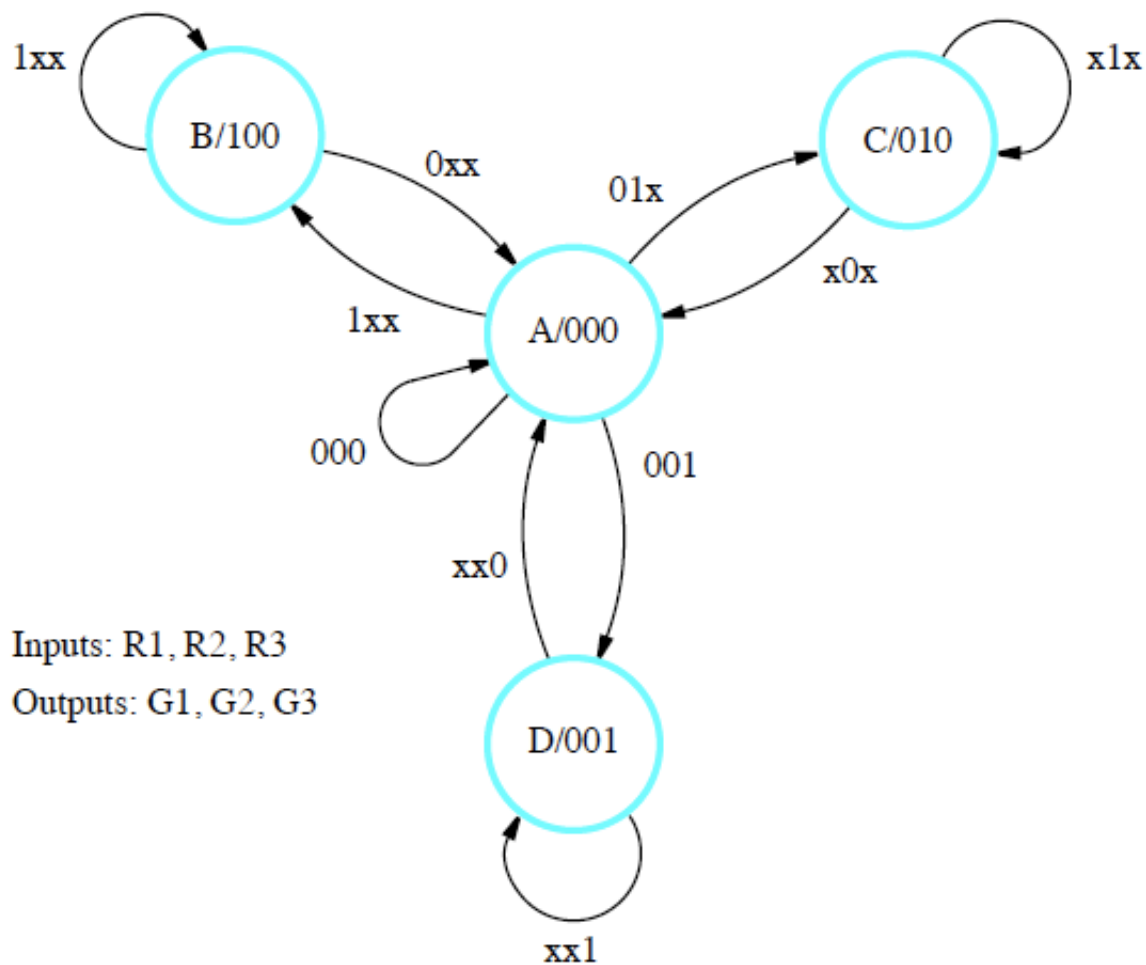
# PClexpress

- ➤ Point-to-point connections with one or more switches forming a tree.

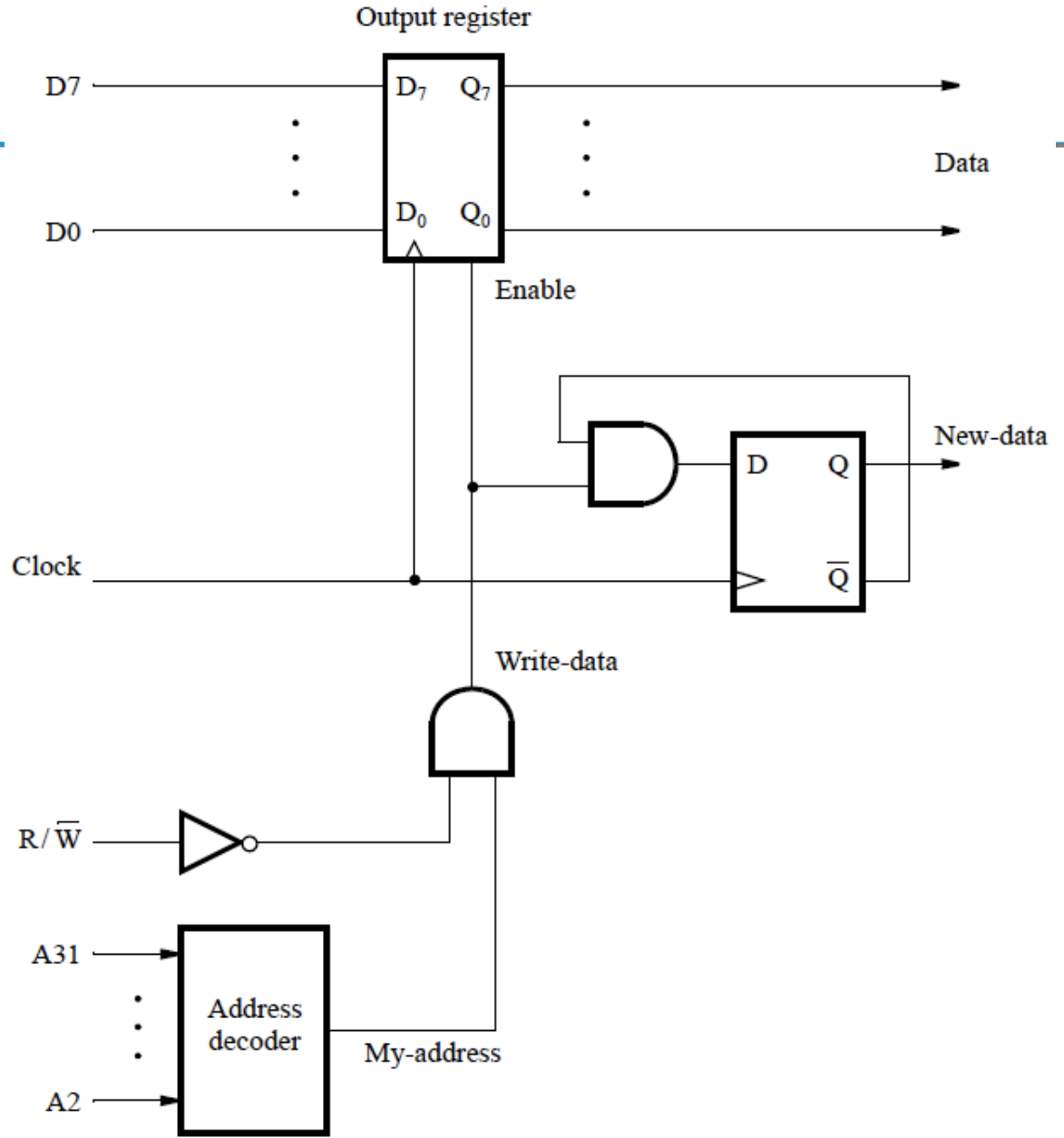- ➤ Root complex provides high-speed ports for memory and other devices.
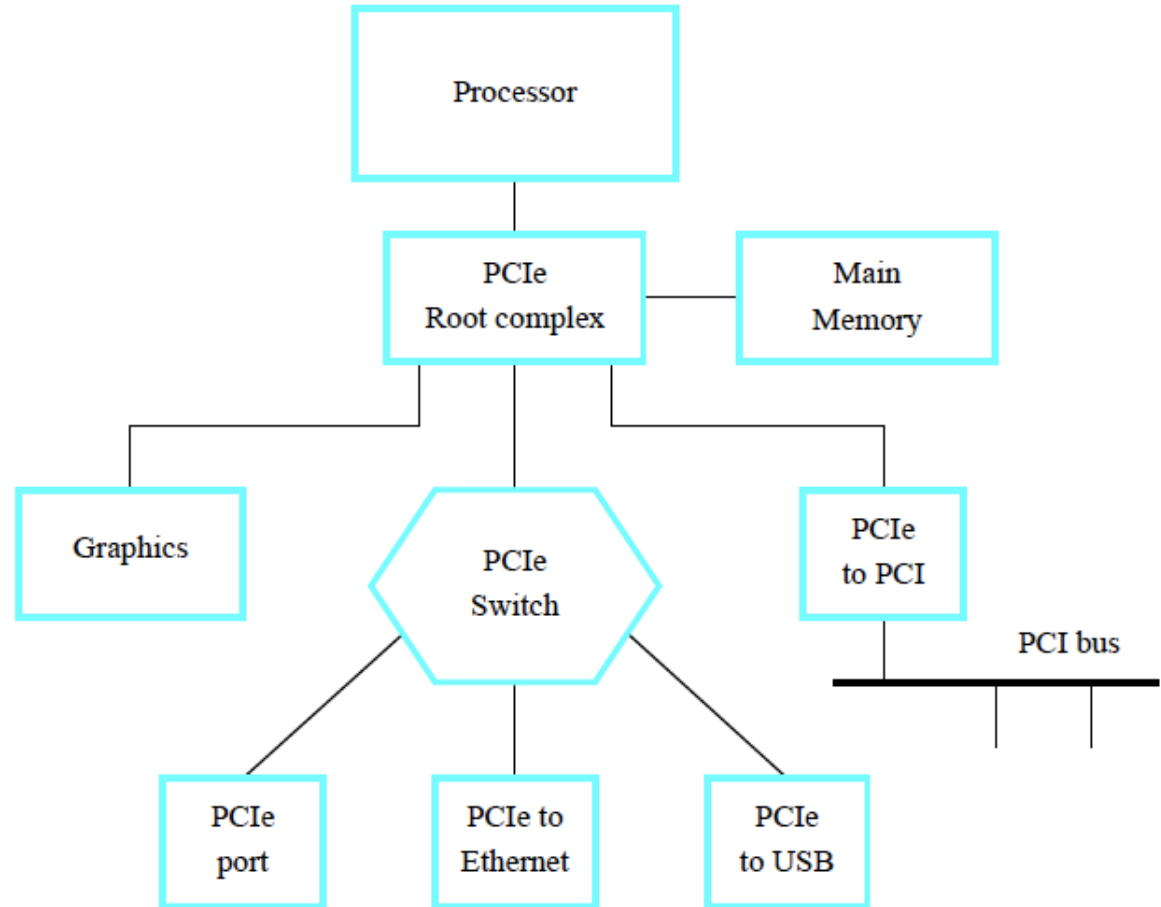
# PCIe links

➢ Basic connection called a lane

➢ A lane consists of two twisted-pairs or optical lines for each direction of transmission.

➢ Data rate is 2.5 Gb/s in each direction.

➢ Connection to a device (link) may use up to 16 lanes.

➢ Protocols fully compatible with PCI, e.g. same initial connection protocol.
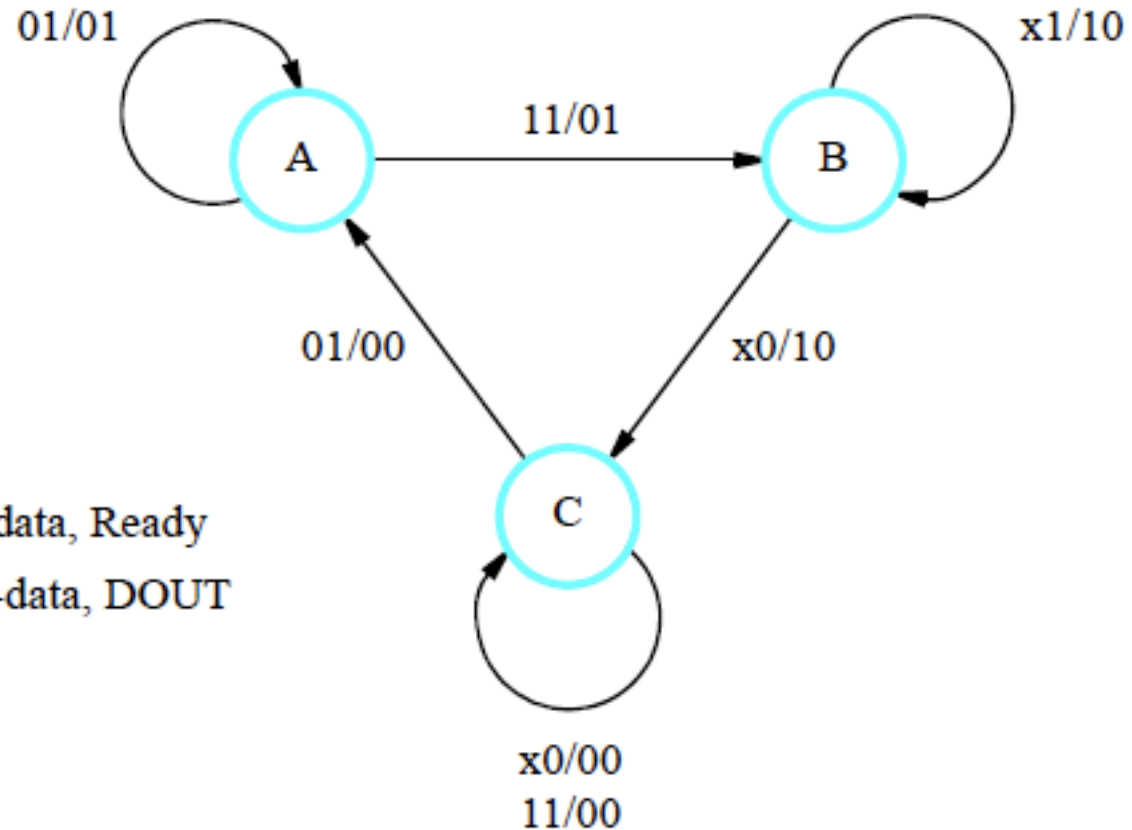
# *Example 7.2*



Inputs: R1, R2, R3
Outputs: G1, G2, G3

# Example 7.3

# Example 7.4



01/01

x1/10

11/01

A → B

01/00          x0/10

C

Inputs: Write-data, Ready

Outputs: New-data, DOUT

x0/00
11/00