+

Bahria University
Discovering Knowledge

Computer Architecture and Logic Design (CALD)

Lecture 05

Dr. Sorath Hansrajani

Assistant Professor

Department of Software Engineering

Bahria University Karachi Campus

Email: sorathhansrajani.bukc@bahria.edu.pk

Input / Output

+

I/O Module

I/O module has two major functions:

- Interface to the processor and memory via the system bus
- Interface to one or more peripheral devices by tailored data links

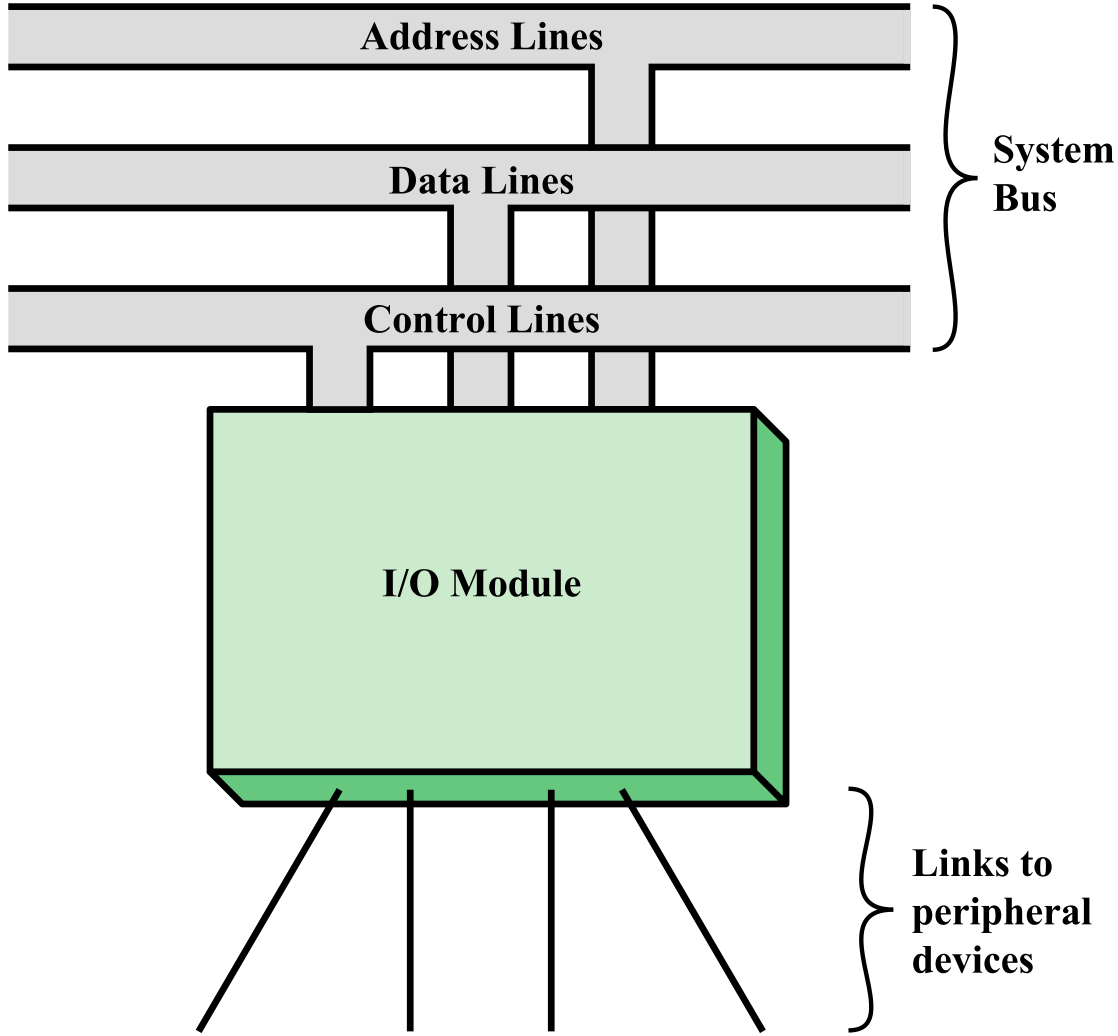
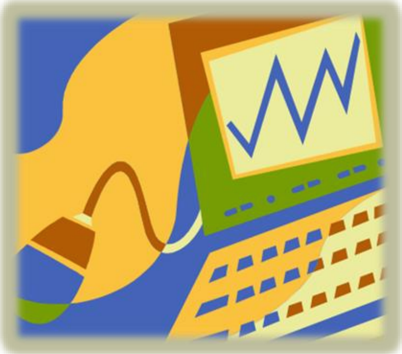


Figure 7.1 Generic Model of an I/O Module

+

External Devices



- Provide a means of exchanging data between the external environment and the computer
- Attach to the computer by a link to an I/O module
  - The link is used to exchange control, status, and data between the I/O module and the external device
- *Peripheral device*
  - An external device connected to an I/O module

Three categories:

- Human readable
  - Suitable for communicating with the computer user
  - Video display terminals (VDTs), printers
- Machine readable
  - Suitable for communicating with equipment
  - Magnetic disk and tape systems, sensors and actuators
- Communication
  - Suitable for communicating with remote devices such as a terminal, a machine readable device, or another computer

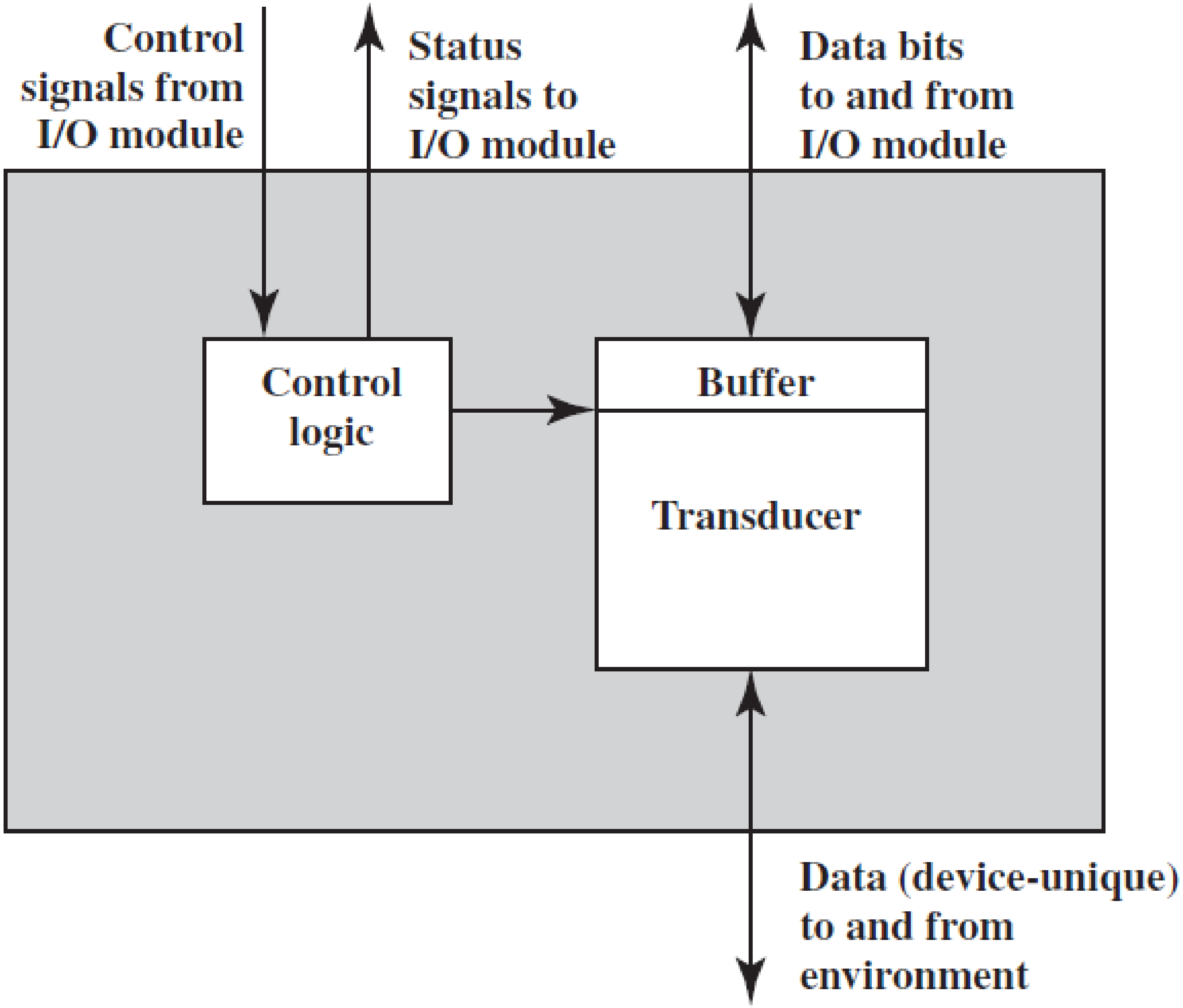


Figure 7.2 Block Diagram of an External Device

+
I/O Module Interface

- Control Signals
  - Control signals determine the function that the device will perform, such as
  - send data to the I/O module (INPUT or READ),
  - accept data from the I/O module (OUTPUT or WRITE)
- Status Signals
  - Status signals indicate the state of the device.
  - Examples are READY/NOT-READY to show whether the device is ready for data transfer.
- Data are in the form of a set of bits to be sent to or received from the I/O module.

The major functions for an I/O module fall into the following categories:

Control and timing
- Coordinates the flow of traffic between internal resources and external devices

Processor communication
- Involves command decoding, data, status reporting, address recognition

Device communication
- Involves commands, status information, and data

Data buffering
- Performs the needed buffering operation to balance device and memory speeds

Error detection
- Detects and reports transmission errors

Transfer of Data from External Device to Processor

1. The processor interrogates the I/O module to check the status of the attached device.
2. The I/O module returns the device status.
3. If the device is operational and ready to transmit, the processor requests the transfer of data, by means of a command to the I/O module.
4. The I/O module obtains a unit of data (e.g., 8 or 16 bits) from the external device.
5. The data are transferred from the I/O module to the processor.

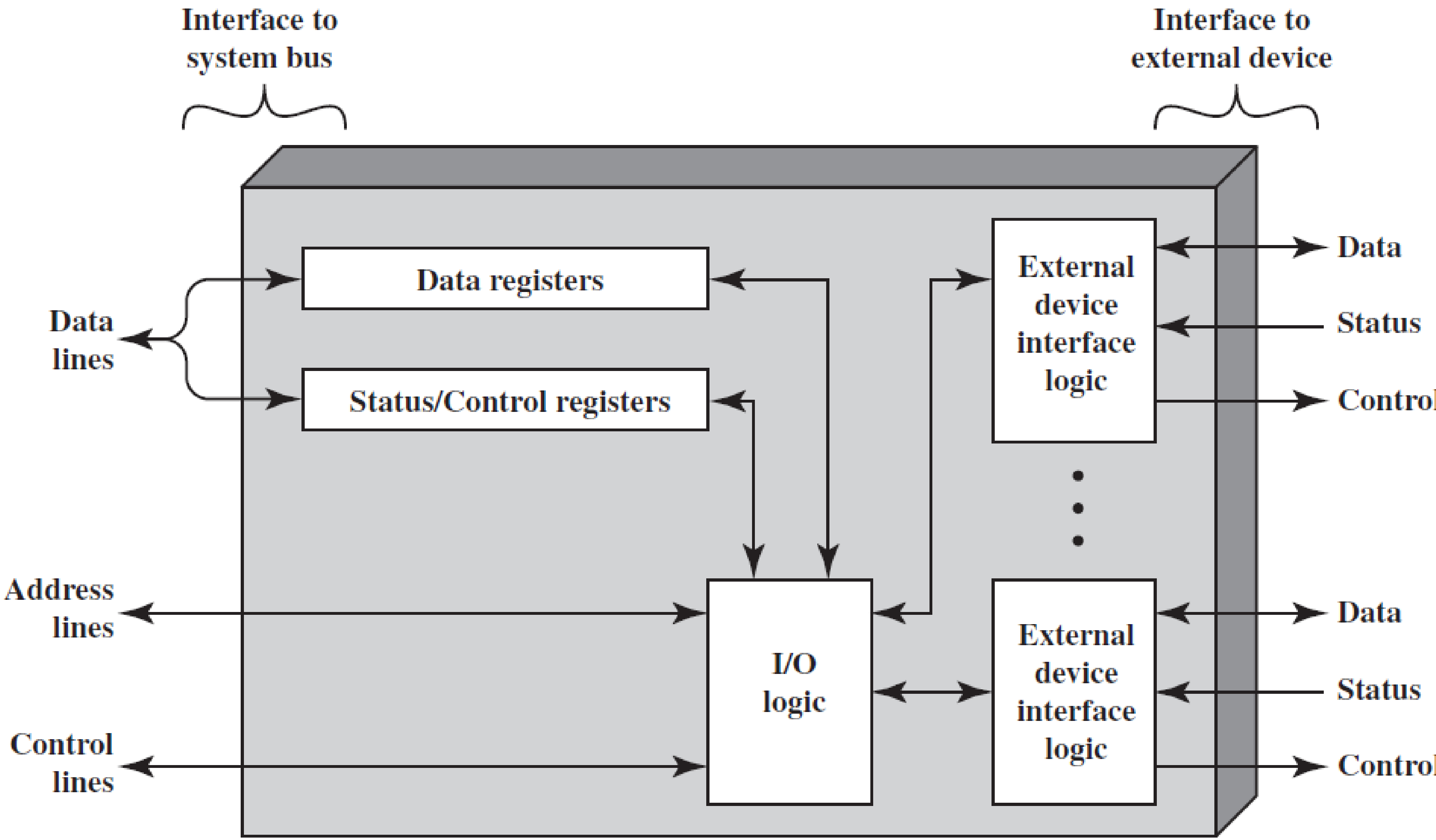


Figure 7.3 Block Diagram of an I/O Module

+

I/O Module Functions

- The module connects to the rest of the computer through a set of signal lines (e.g., system bus lines).
- Data transferred to and from the module are buffered in one or more data registers.
- There may also be one or more status registers that provide current status information.
- The logic within the module interacts with the processor via a set of control lines.
- The processor uses the control lines to issue commands to the I/O module.
- the I/O module contains logic specific to the interface with each device that it controls.

+ I/O Techniques

Three techniques are possible for I/O operations:

- Programmed I/O
  - Data are exchanged between the processor and the I/O module
  - Processor executes a program that gives it direct control of the I/O operation
  - When the processor issues a command it must wait until the I/O operation is complete
  - If the processor is faster than the I/O module this is wasteful of processor time
- Interrupt-driven I/O
  - Processor issues an I/O command, continues to execute other instructions, and is interrupted by the I/O module when the latter has completed its work
- Direct memory access (DMA)
  - The I/O module and main memory exchange data directly without processor involvement

+

I/O Commands

- There are four types of I/O commands that an I/O module may receive when it is addressed by a processor:

 1) Control

 - used to activate a peripheral and tell it what to do

 2) Test

 - used to test various status conditions associated with an I/O module and its peripherals

 3) Read

 - causes the I/O module to obtain an item of data from the peripheral and place it in an internal buffer

 4) Write

 - causes the I/O module to take an item of data from the data bus and subsequently transmit that data item to the peripheral

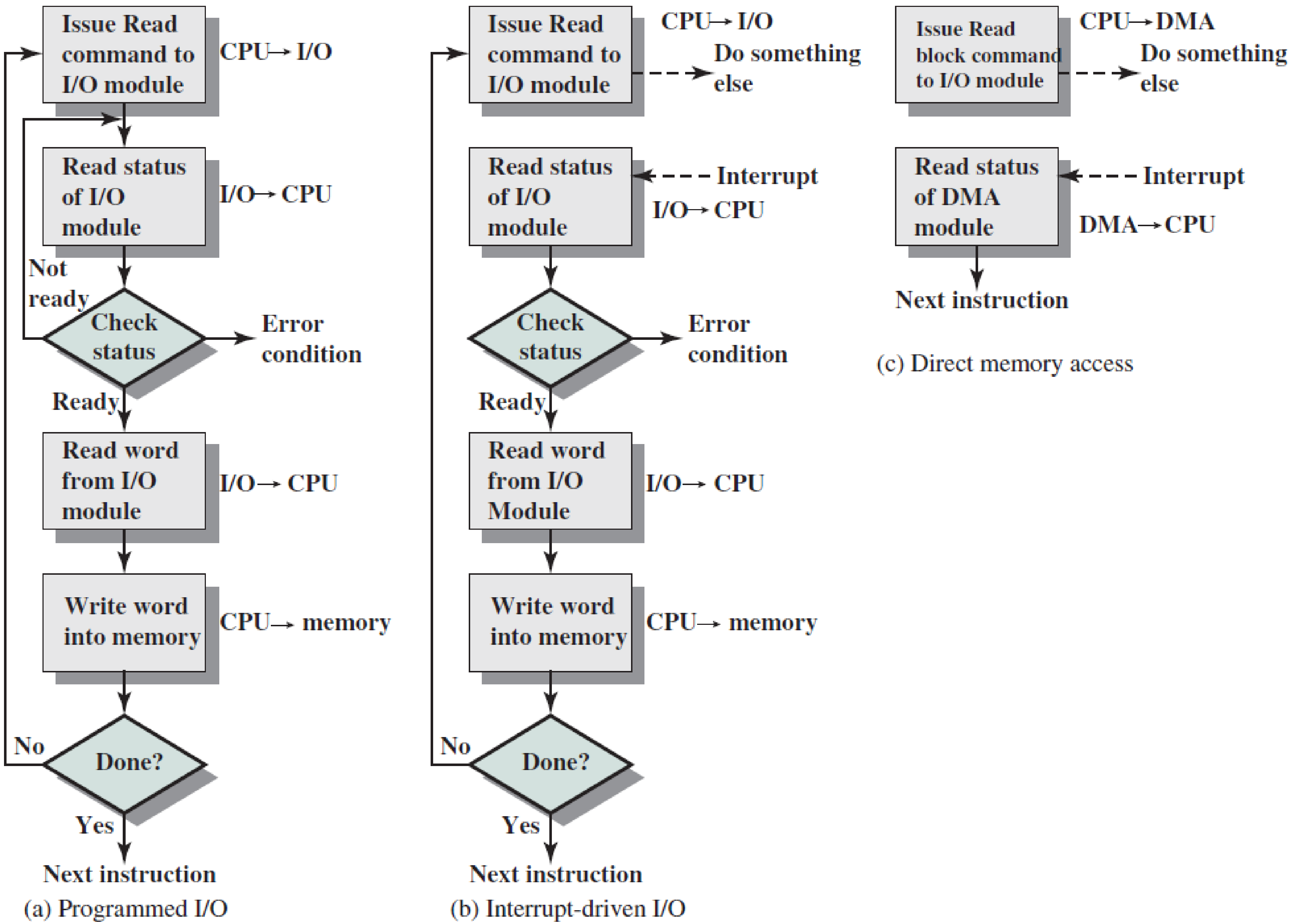


Figure 7.4 Three Techniques for Input of a Block of Data

I/O Instructions

With programmed I/O there is a close correspondence between the I/O-related instructions that the processor fetches from memory and the I/O commands that the processor issues to an I/O module to execute the instructions

The form of the instruction depends on the way in which external devices are addressed

Each I/O device connected through I/O modules is given a unique identifier or address

When the processor issues an I/O command, the command contains the address of the desired device

Thus each I/O module must interpret the address lines to determine if the command is for itself

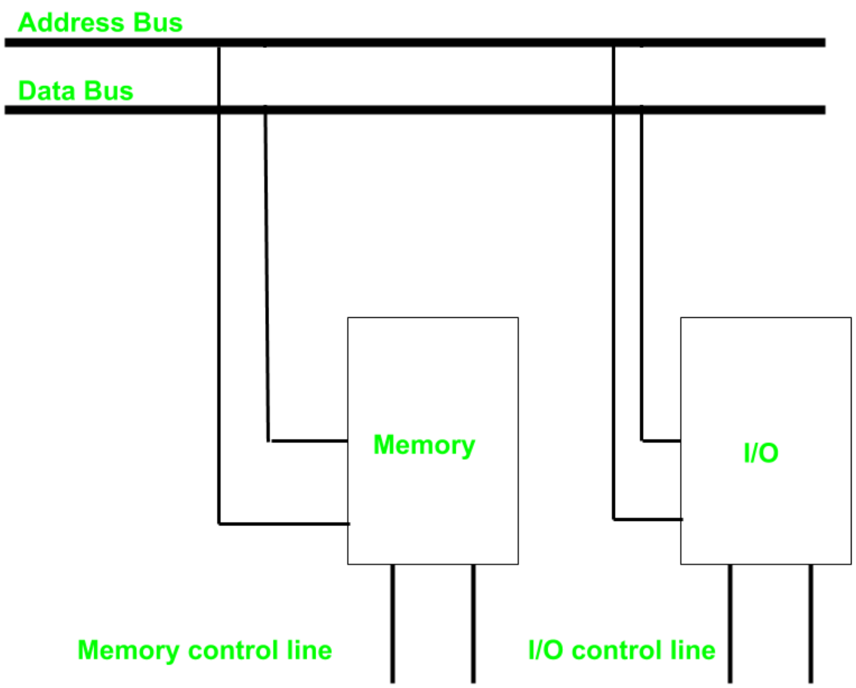
Memory-mapped I/O

There is a single address space for memory locations and I/O devices

A single read line and a single write line are needed on the bus

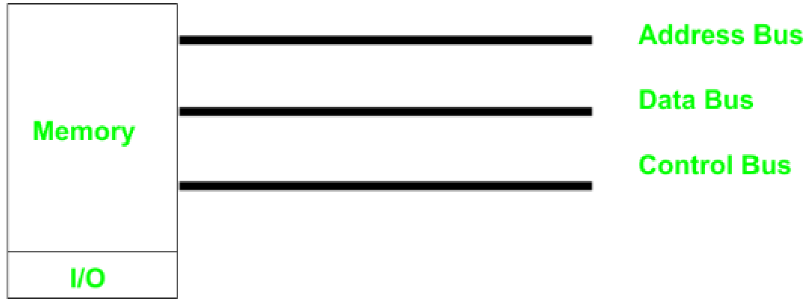
Isolated I/O

- common bus(data and address) for I/O and memory
- but separate read and write control lines for I/O.
- So when CPU decode instruction then if data is for I/O then it places the address on the address line and set I/O read or write control line on due to which data transfer occurs between CPU and I/O.
- different read-write instruction for both I/O and memory.



Memory-mapped I/O

- Every bus in common
- same set of instructions work for memory and I/O.
- Hence we manipulate I/O same as memory and both have same address space,
- due to which addressing capability of memory become less because some part is occupied by the I/O.

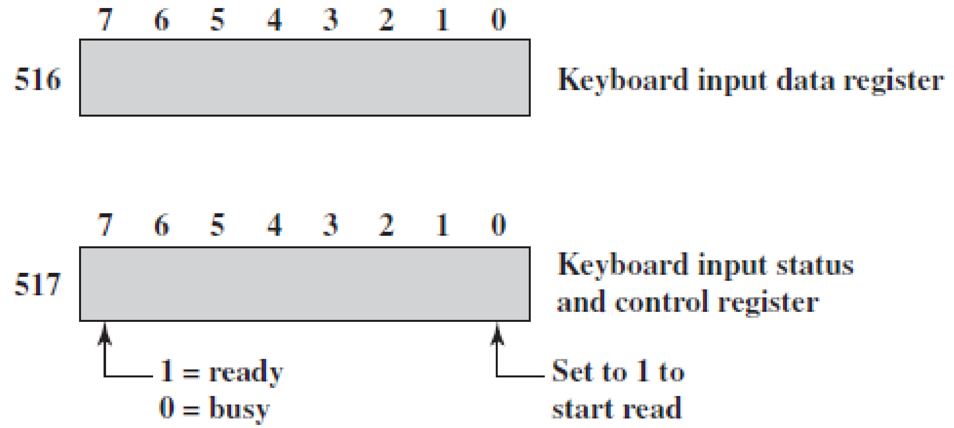


I/O Mapping Summary

- Memory mapped I/O
  - Devices and memory share an address space
  - I/O looks just like memory read/write
  - No special commands for I/O
    - Large selection of memory access commands available
- Isolated I/O
  - Separate address spaces
  - Need I/O or memory select lines
  - Special commands for I/O
    - Limited set

Isolated vs. Memory-mapped I/O

Isolated I/O	Memory Mapped I/O
Memory and I/O have separate address space	Both have same address space
All address can be used by the memory	Due to addition of I/O addressable memory become less for memory
Separate instruction control read and write operation in I/O and Memory	Same instructions can control both I/O and Memory
In this I/O address are called ports.	Normal memory address are for both
More efficient due to separate buses	Lesser efficient
Larger in size due to more buses	Smaller in size
It is complex due to separate logic is used to control both.	Simpler logic is used as I/O is also treated as memory only.



ADDRESS	INSTRUCTION	OPERAND	COMMENT
200	Load AC	"1"	Load accumulator
	Store AC	517	Initiate keyboard read
202	Load AC	517	Get status byte
	Branch if Sign = 0	202	Loop until ready
	Load AC	516	Load data byte

(a) Memory-mapped I/O

ADDRESS	INSTRUCTION	OPERAND	COMMENT
200	Load I/O	5	Initiate keyboard read
201	Test I/O	5	Check for completion
	Branch Not Ready	201	Loop until complete
	In	5	Load data byte

(b) Isolated I/O

Figure 7.5 Memory-Mapped and Isolated I/O

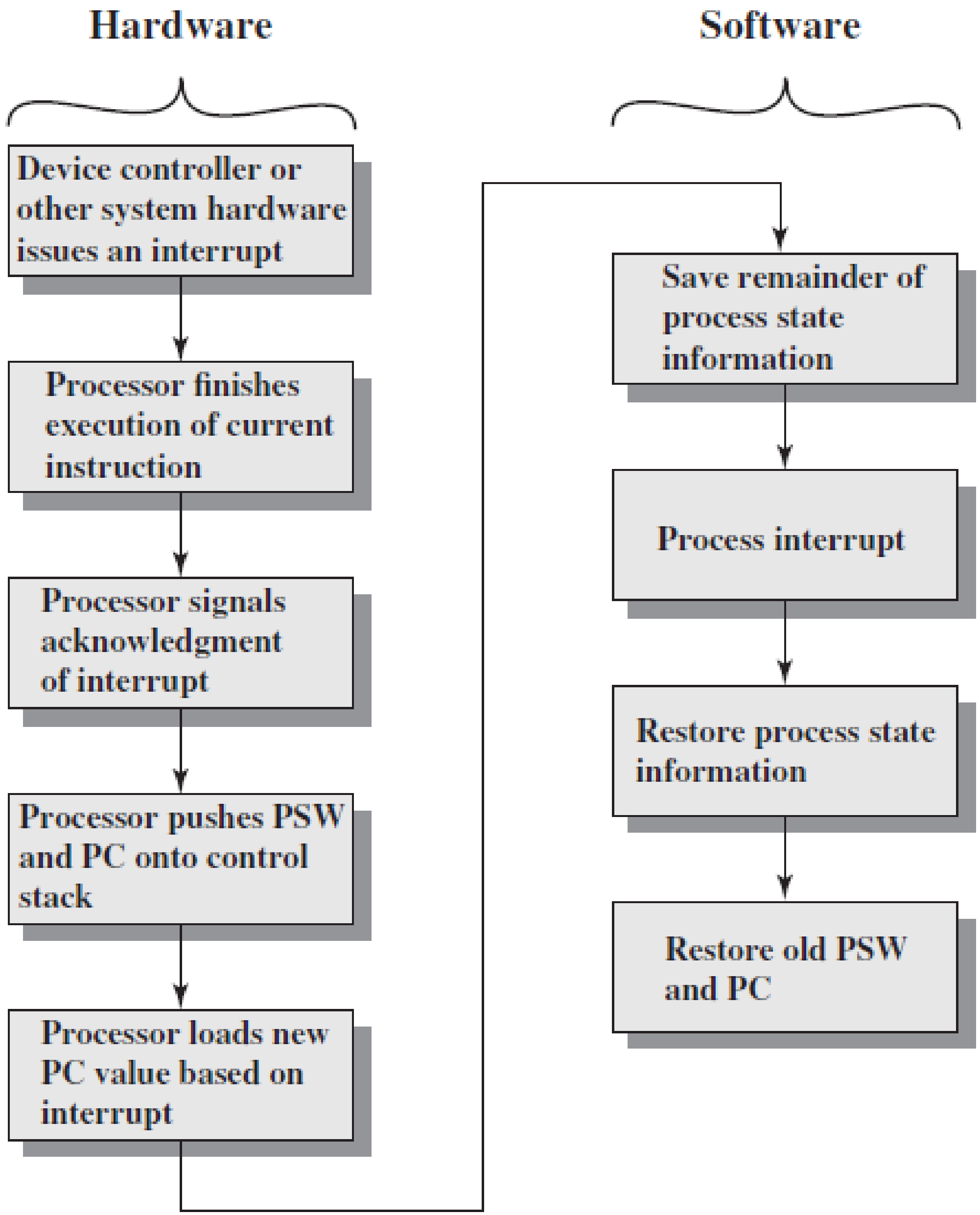
Interrupt-Driven I/O

The problem with programmed I/O is that the processor has to wait a long time for the I/O module to be ready for either reception or transmission of data

An alternative is for the processor to issue an I/O command to a module and then go on to do some other useful work

The I/O module will then interrupt the processor to request service when it is ready to exchange data with the processor

The processor executes the data transfer and resumes its former processing



PSW =
Program Status Word

Figure 7.6 Simple Interrupt Processing

Drawbacks of Programmed and Interrupt-Driven I/O

- Both forms of I/O suffer from two inherent drawbacks:

    1) The I/O transfer rate is limited by the speed with which the processor can test and service a device

    2) The processor is tied up in managing an I/O transfer; a number of instructions must be executed for each I/O transfer

- When large volumes of data are to be moved a more efficient technique is *direct memory access* (DMA)

+

Trade-offs

- There is somewhat of a trade-off between these two drawbacks.
- Consider the transfer of a block of data.
- Using simple programmed I/O, the processor is dedicated to the task of I/O and can move data at a rather high rate, at the cost of doing nothing else.
- Interrupt I/O frees up the processor to some extent at the expense of the I/O transfer rate.
- Nevertheless, both methods have an adverse impact on both processor activity and I/O transfer rate.

+

Direct Memory Access (DMA)

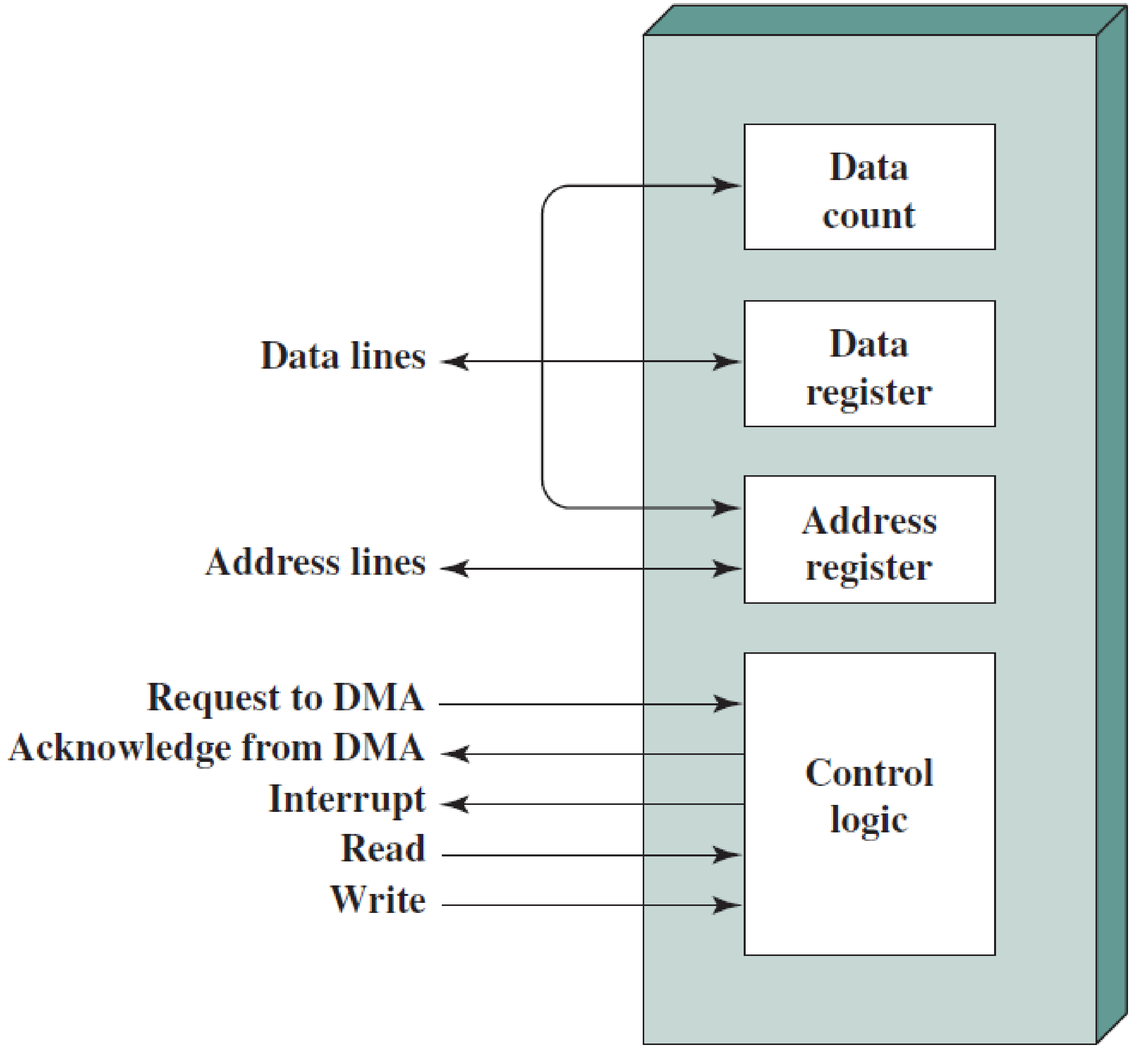


Figure 7.12 Typical DMA Block Diagram

+
Direct Memory Access (DMA)

- The DMA module is capable of taking over control of the system from the processor.
- It needs to do this to transfer data to and from memory over the system bus.
- Cycle Stealing:
  - the DMA module must use the bus only when the processor does not need it, or it must force the processor to suspend operation temporarily.
  - the DMA module in effect steals a bus cycle.

Direct Memory Access (DMA)

- The processor issues a command to the DMA module
  - Read or Write Signal
  - The address of the I/O device involved
  - The starting location in memory to read from or write to
  - The number of words to be read or written
  - The processor then continues with other work.
  - It has delegated this I/O operation to the DMA module.
  - The DMA module transfers the entire block of data, one word at a time, directly to or from memory, without going through the processor.
  - When the transfer is complete, the DMA module sends an interrupt signal to the processor.
  - Thus, the processor is involved only at the beginning and end of the transfer

+ Summary

Chapter 7

Input/Output

- External devices
  - Keyboard/monitor
  - Disk drive
- I/O modules
  - Module function
  - I/O module structure
- Programmed I/O
  - Overview of programmed I/O
  - I/O commands/instructions
- Direct memory access
  - Drawbacks of programmed and interrupt-driven I/O
  - DMA function
- Interrupt-driven I/O