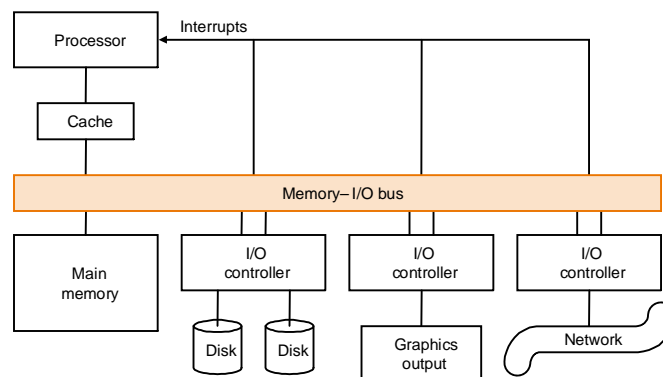


Chapter 8

1

A Typical collection of I/O devices



2

Interfacing Processors and Peripherals

- I/O Design affected by many factors (expandability, resilience)
- Performance:
 - access latency
 - throughput
 - connection between devices and the system
 - the memory hierarchy
 - the operating system
- A variety of different users (e.g., banks, supercomputers, engineers)

3

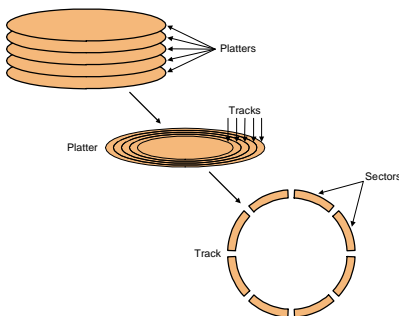
I/O Devices

- Very diverse devices
 - behavior (i.e., input vs. output)
 - partner (who is at the other end?)
 - data rate

Device	Behavior	Partner	Data rate (KB/sec)
Keyboard	input	human	0.01
Mouse	input	human	0.02
Voice input	input	human	0.02
Scanner	input	human	400.00
Voice output	output	human	0.60
Line printer	output	human	1.00
Laser printer	output	human	200.00
Graphics display	output	human	60,000.00
Modem	input or output	machine	2.00-8.00
Network/LAN	input or output	machine	500.00-6000.00
Floppy disk	storage	machine	100.00
Optical disk	storage	machine	1000.00
Magnetic tape	storage	machine	2000.00
Magnetic disk	storage	machine	2000.00-10,000.00

4

I/O Example: Disk Drives



To access data:

- seek: position head over the proper track (8 to 20 ms. avg.)
- rotational latency: wait for desired sector (.5 / RPM)
- transfer: grab the data (one or more sectors) 2 to 15 MB/sec

5

Buses

- Buses connect I/O devices to processors and memory
- A bus is a shared communication link which uses one set of wires to connect multiple systems
- Advantages
 - Shared link, therefore cost effective
- Disadvantage
 - Shared link can become a communication bottleneck
- Each bus has a set of data lines and a set of control lines

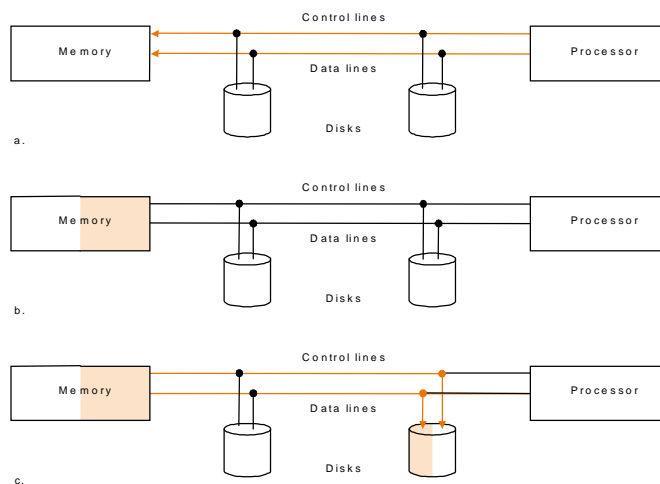
6

Using a Bus for Input/Output

- Each bus transaction has two parts
 - Sending the address
 - Receiving/Sending the data
- Input transaction
 - Inputs data from a device to memory
- Output transaction
 - Outputting data from memory to a device

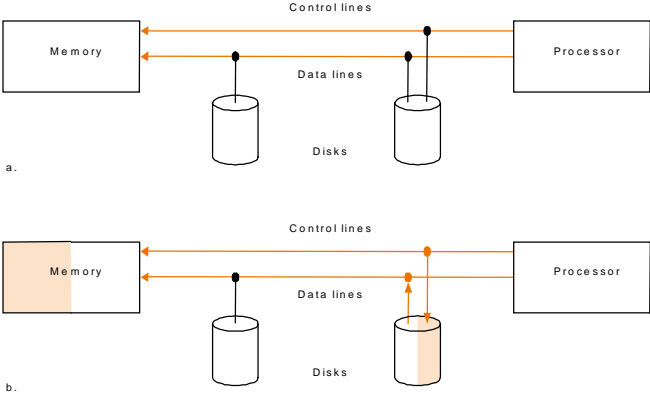
7

The three steps of an output transaction

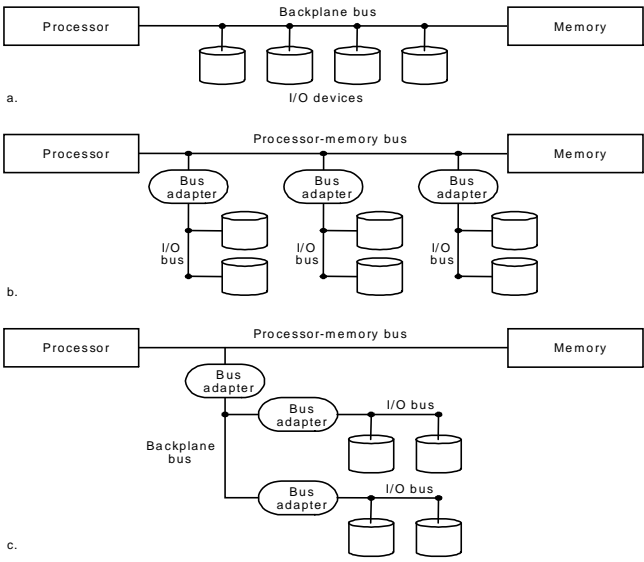


8

An input transaction



Types of buses



Types of buses

- Processor-Memory bus
 - Short
 - High-speed
 - Matched to the memory system
- I/O buses
 - Lengthy
 - Can have many different devices attached to them
 - Wide range of data transfer speeds
- Backplane buses
 - Can have processor, memory, I/O devices coexisting on a single bus
 - Balance demands of processor-memory communication with demands of device memory communication

11

Synchronous and Asynchronous Buses

- Synchronous
 - Driven by a clock
 - Clock input to control lines
 - Protocol for communication is relative to this clock
 - Example: processor-memory communication
 - Processor sends address and read command in first clock cycle
 - Memory sends data on fifth clock cycle
 - **Protocol is predetermined and driven by the clock**
 - Disadvantage
 - Every device on bus must be able to run at same clock rate
 - Synchronous buses have to be short to avoid clock skew

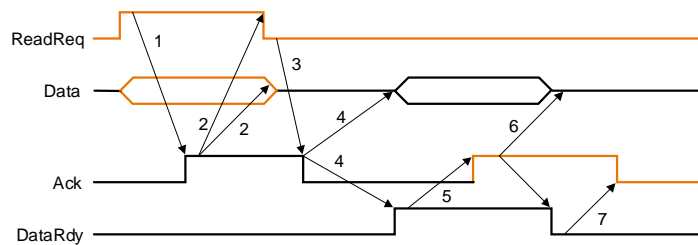
12

Synchronous & Asynchronous Buses

- Asynchronous Buses
 - Not clocked
 - Can be lengthened
 - Use a handshaking protocol
- Synchronous buses typically faster than asynchronous

13

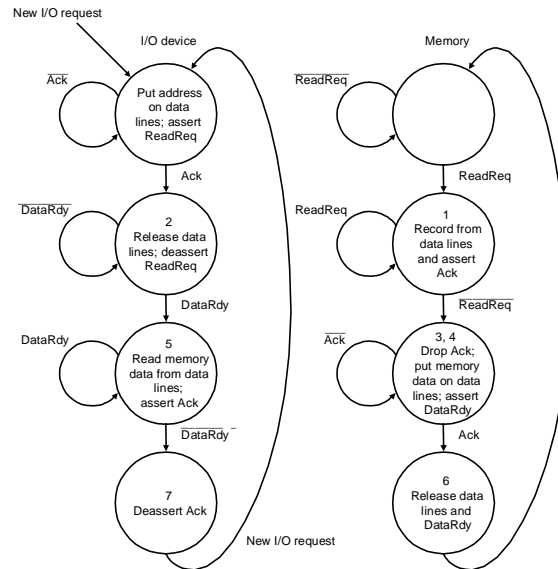
Handshaking Protocol for an Asynchronous Bus



1. When memory sees ReadReq line, it reads the address from the Data bus and raises ACK to indicate it has been seen
2. I/O device sees ACK and releases ReadReq and Data lines
3. Memory sees ReadReq is low and drops the ACK signal
4. This step starts when memory has the data ready. It places data on the Data line and raises DataRdy
5. I/O device sees DataRdy, reads the data, and then raises ACK
6. The memory sees ACK, and drops DataRdy and releases the data lines
7. I/O device sees DataRdy go low and drops the ACK line

14

Finite State Machine for implementing the control for the handshaking protocol



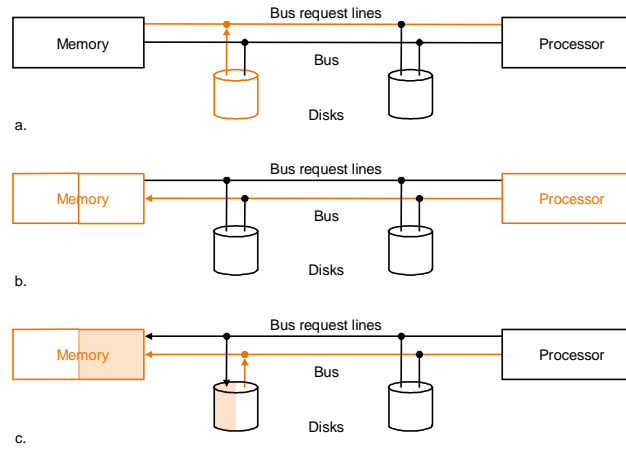
15

Obtaining Access to the Bus

- How is the bus reserved by one of the devices that wants to use it to communicate?
- Single Bus Master: the **CPU**
 - Advantage: simple
 - Disadvantage: processor involved in every bus transaction
- Multiple bus masters
- Need Bus Arbitration:
 - daisy chain arbitration (not very fair)
 - centralized arbitration (requires an arbiter), e.g., PCI
 - self selection, e.g., NuBus used in Macintosh
 - collision detection, e.g., Ethernet

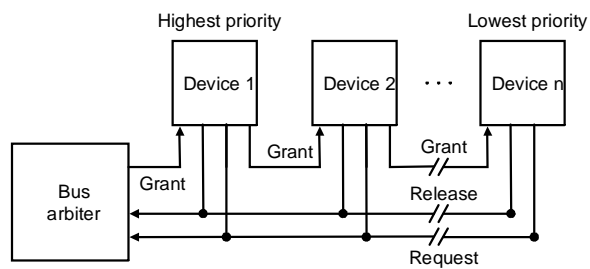
16

A bus transaction with a single master (the processor)



17

Daisy-chain Bus Arbitration



18

Organization of the I/O system in the Apple Macintosh 7200 (PCI backplane, SCSI for slower devices)

