CHAPTER

9

I/O SYSTEM

TRUONG DAI HOC KHOA HOC TU NHIEN
KHTN
TP. HO CHI MINH

cdio 4.0

KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

fit@hcmus

What will you learn?

- I/O devices
- I/O System Characteristics
- I/O Modules
- I/O Register Mapping
- I/O Data transfer
- I/O Command
- Life cycle of an I/O request
- I/O Bus
- Typical x86 PC I/O System

I/O devices

- Can be typify by:
  - Behavior: input, output, storage
  - Partner: human / machine
  - Data rate: bytes/sec, transfer/sec

- Character & Block devices

The device interface gives the illusion that devices support the same API – character stream and block access

application/user:	*read character from device*	naming, protection, read,write
operating system:	*character & block API*	hardware specific PIO, interrupt handling, or DMA
hardware:	*keyboard, mouse, etc.*	

I/O Modules

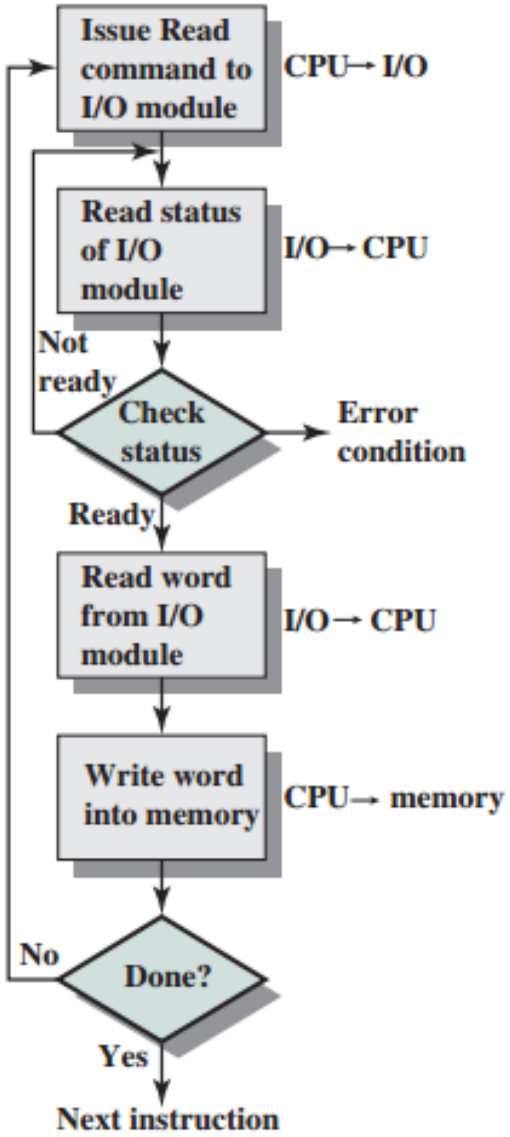
- Interface to the processor and memory via the system bus or control switch
- Interface to one or more peripheral devices

I/O Register Mapping

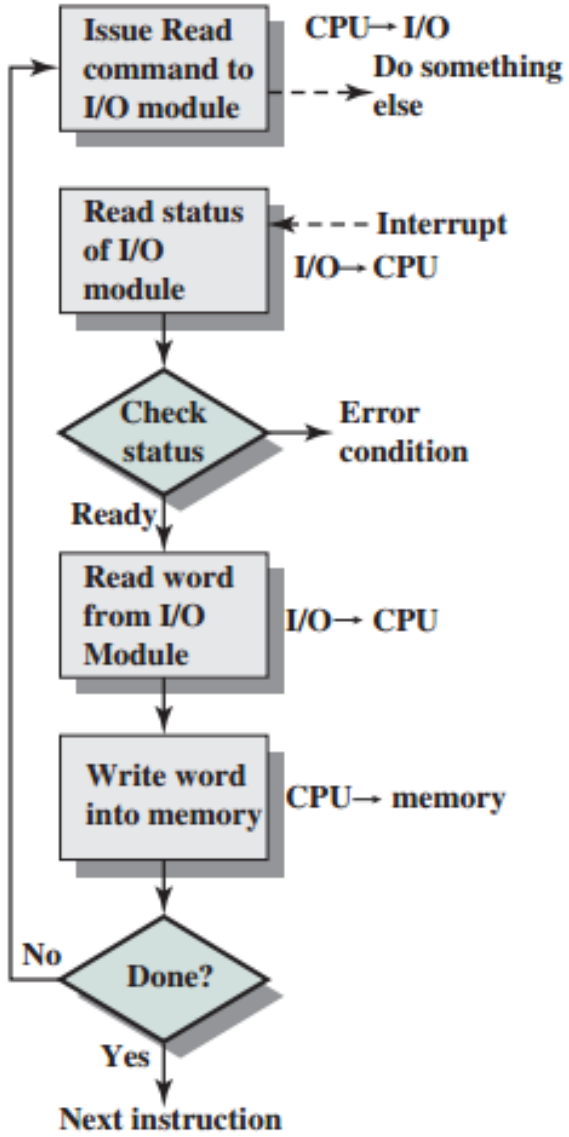
- Memory-mapped I/O
  - Registers are addressed in same space as memory
  - Address decoder distinguishes between them
  - OS uses address translation mechanism to make them only accessible to kernel
- Isolated I/O
  - Separate instructions to access I/O registers
  - Can only be executed in kernel mode

I/O Data Transfer

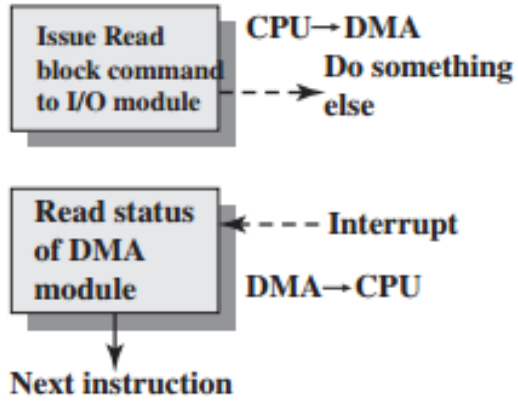
- Data transfer between CPU and I/O devices can be handled in generally three types of modes:
  - Programed I/O
  - Interrupt Driven I/O
  - Direct Memory Access



(a) Programmed I/O



(b) Interrupt-driven I/O



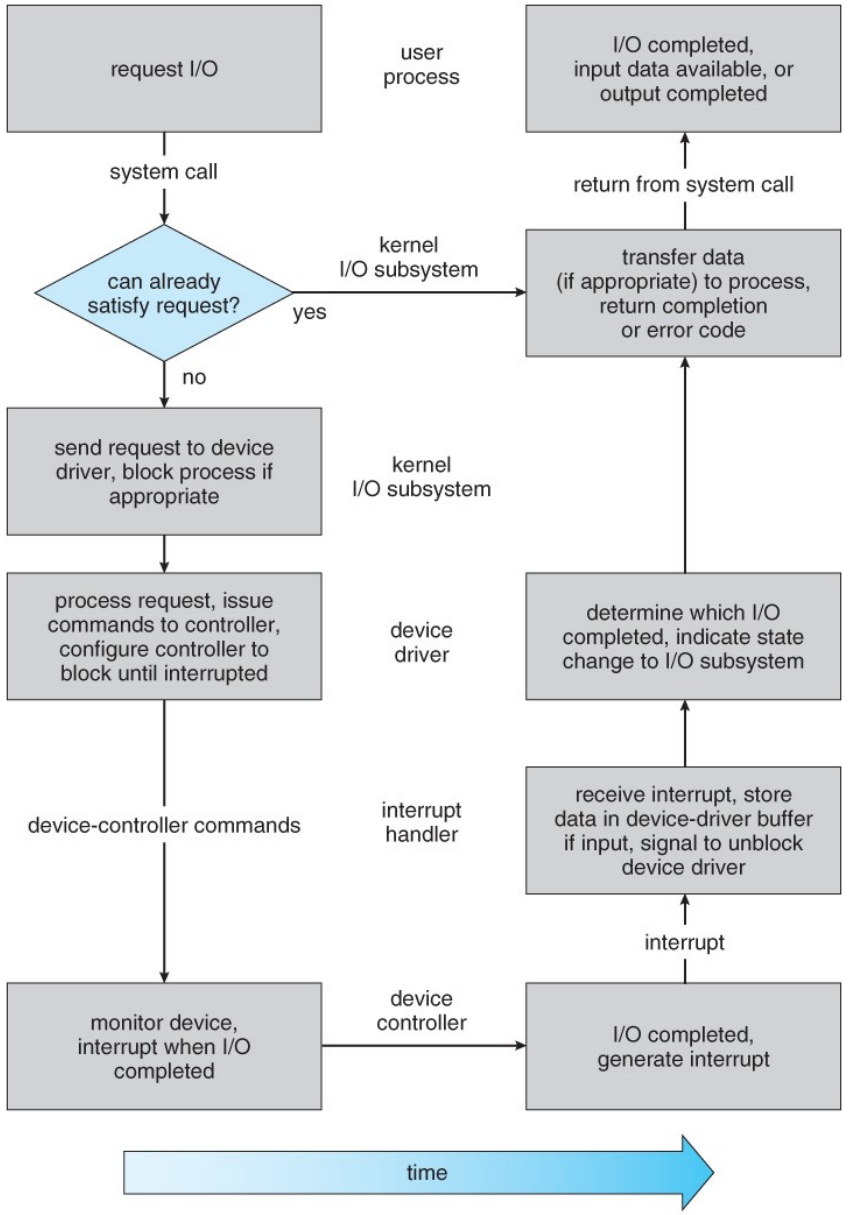
(c) Direct memory access

I/O Commands

- I/O devices are managed by I/O Controller hardware
- The processor issues an address, specifying I/O module and device, and an I/O command. The commands are:
  - Control
  - Test
  - Read
  - Write

Life cycle of an I/O request

- Users request data using file names, which must ultimately be mapped to specific blocks of data from a specific device managed by a specific device driver.
- UNIX uses a *mount table* to map filename prefixes (e.g. /usr) to specific mounted devices
- UNIX uses special *device files,* usually located in /dev, to represent and access physical devices directly



I/O Bus Types

- Processor-Memory buses
  - Short, high speed
  - Design is matched to memory organization
- I/O buses
  - Longer, allowing multiple connections
  - Connect to processor-memory bus through a bridge

I/O Bus Example

	Firewire	USB 2.0	PCI Express	Serial ATA	Serial Attached SCSI
Intended use	External	External	Internal	Internal	External
Devices per channel	63	127	1	1	4
Data width	4	2	2/lane	4	4
Peak bandwidth	50MB/s or 100MB/s	0.2MB/s, 1.5MB/s, or 60MB/s	250MB/s/lane 1×, 2×, 4×, 8×, 16×, 32×	300MB/s	300MB/s
Hot pluggable	Yes	Yes	Depends	Yes	Yes
Max length	4.5m	5m	0.5m	1m	8m
Standard	IEEE 1394	USB Implementers Forum	PCI-SIG	SATA-IO	INCITS TC T10

Typical X86 PC I/O System