Lecture 7 – Computer Hardware

Karl R. Wilcox
Karl@cs.rhul.ac.uk

Royal Holloway University of London

Objectives

- In this class we will discuss:
  - The hardware components of a computer system
  - How the computer executes instructions
  - The hardware and instructions that are particularly concerned with I/O processing

Royal Holloway University of London

Operating System

- **Exploits the hardware resources of one or more processors**
- **Provides a set of services to system users**
- **Manages secondary memory and I/O devices**

Royal Holloway University of London

Basic Elements

- Processor
- Main Memory
  - referred to as real memory or primary memory
  - volatile
- I/O modules
  - secondary memory devices
  - communications equipment
  - terminals
- System bus
  - communication among processors, memory, and I/O modules

Top-Level Components

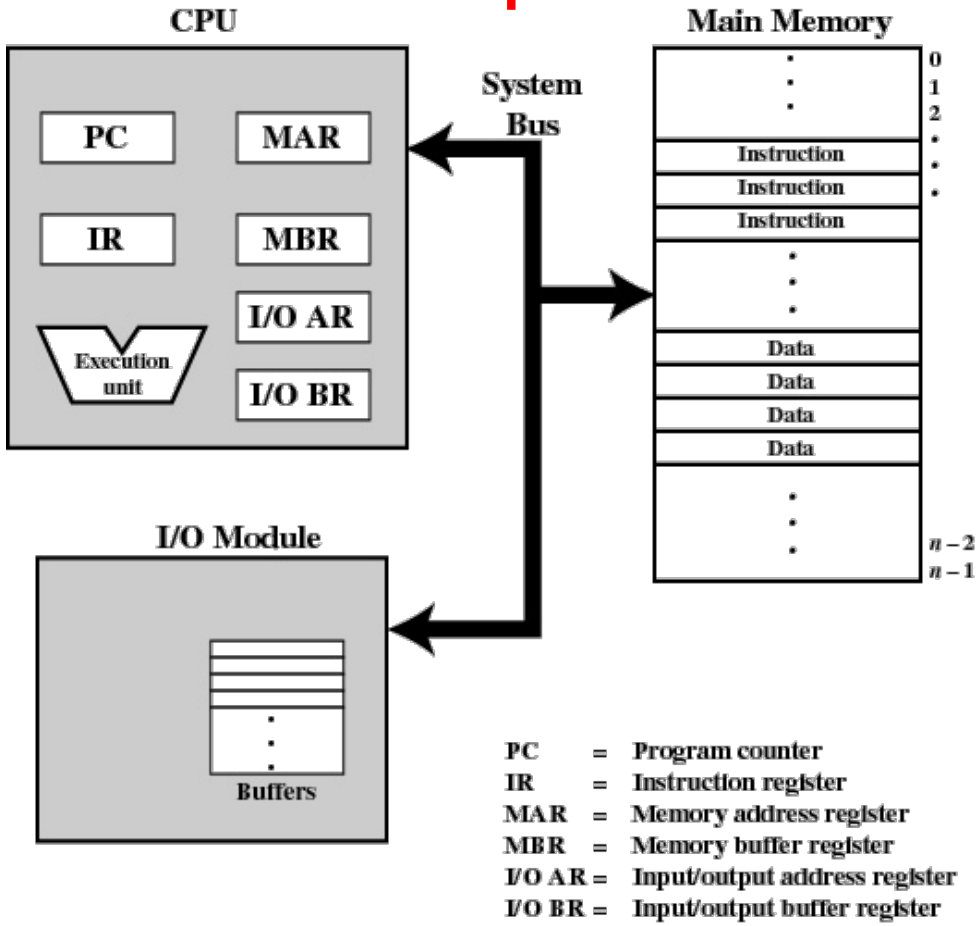


Figure 1.1 Computer Components: Top-Level View

Instruction Cycle

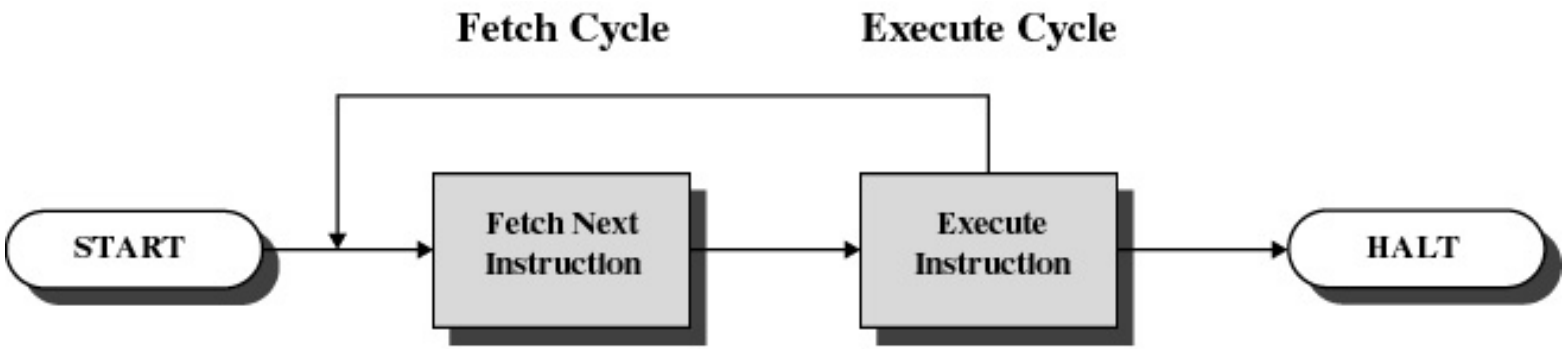


Figure 1.2 Basic Instruction Cycle

Royal Holloway University of London

Instruction Fetch and Execute

- The processor fetches the instruction from memory
- Program counter (PC) holds address of the instruction to be fetched next
- Program counter is incremented after each fetch

Instruction Register

- Fetched instruction is placed in the instruction register
- Types of instructions
  - Processor-memory
    - transfer data between processor and memory
  - Processor-I/O
    - data transferred to or from a peripheral device
  - Data processing
    - arithmetic or logic operation on data
  - Control
    - alter sequence of execution

Royal Holloway University of London

Direct Memory Access (DMA)

- I/O exchanges occur directly with memory
- Processor grants I/O module authority to read from or write to memory
- Relieves the processor responsibility for the exchange
- Processor is free to do other things

Royal Holloway University of London

Interrupts

- An interruption of the normal sequence of execution
- Improves processing efficiency
- Allows the processor to execute other instructions while an I/O operation is in progress
- A suspension of a process caused by an event external to that process and performed in such a way that the process can be resumed

Royal Holloway University of London

Classes of Interrupts

- **Program**
  - **arithmetic overflow**
  - **division by zero**
  - **execute illegal instruction**
  - **reference outside user's memory space**
- **Timer**
- **I/O**
- **Hardware failure**

Royal Holloway University of London

Interrupt Handler

- A program that determines nature of the interrupt and performs whatever actions are needed
- Control is transferred to this program
- Generally part of the operating system

Interrupt Cycle

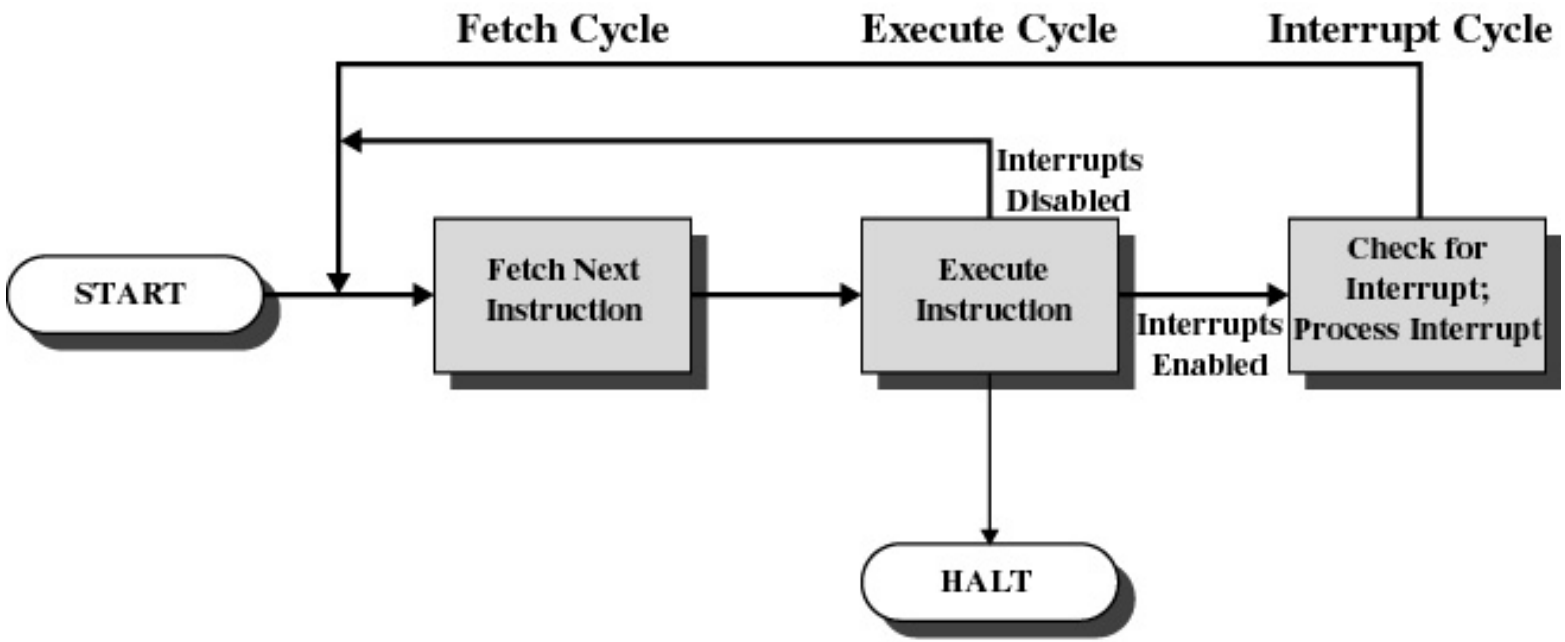
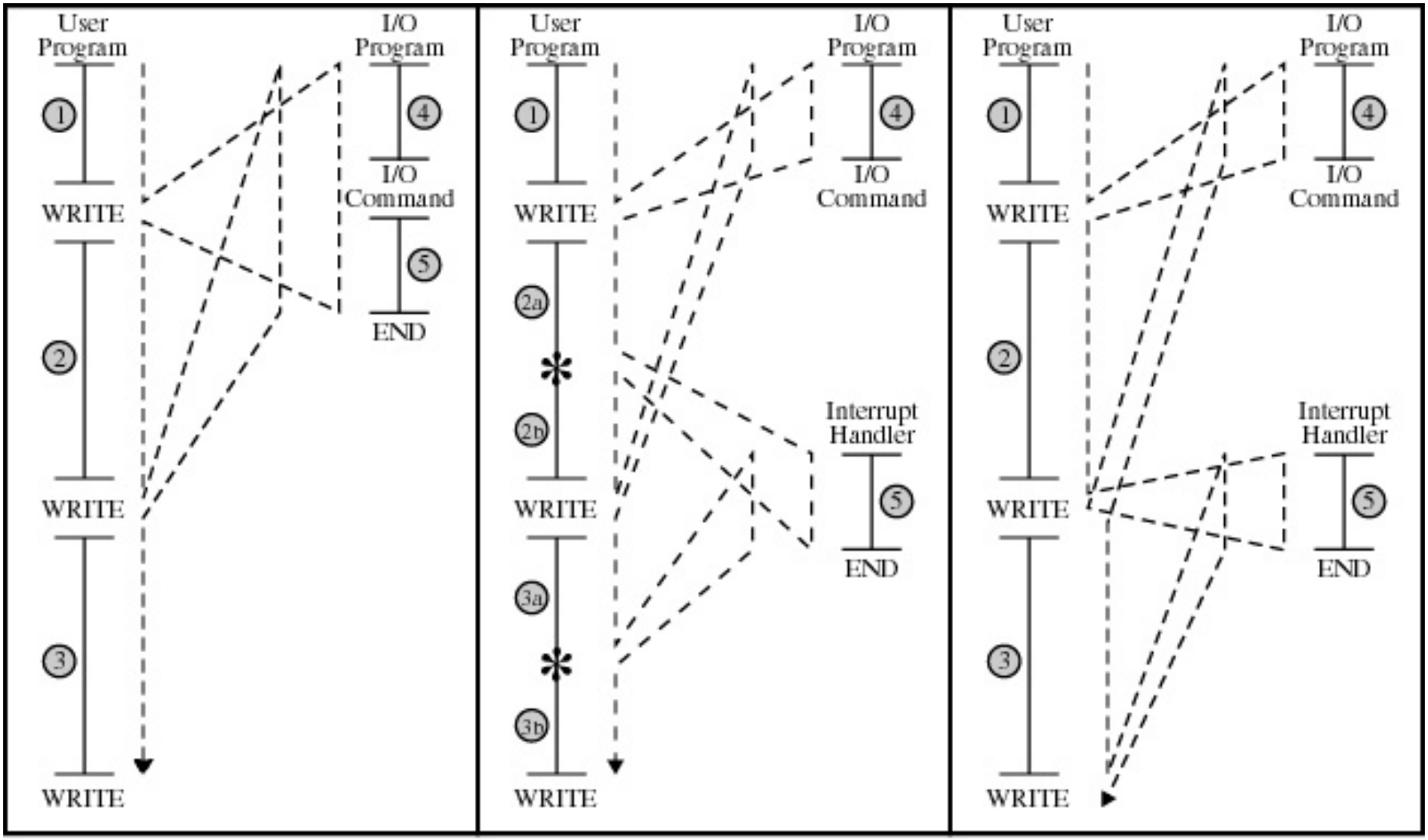


Figure 1.7 Instruction Cycle with Interrupts

Royal Holloway University of London

Interrupt Cycle

- **Processor checks for interrupts**
- **If no interrupts fetch the next instruction for the current program**
- **If an interrupt is pending, suspend execution of the current program, and execute the interrupt handler**



(a) No interrupts (b) Interrupts; short I/O wait (c) Interrupts; long I/O wait

Figure 1.5 Program Flow of Control Without and With Interrupts

Multiple Interrupts

- **Disable interrupts while an interrupt is being processed**
  - **Processor ignores any new interrupt request signals**

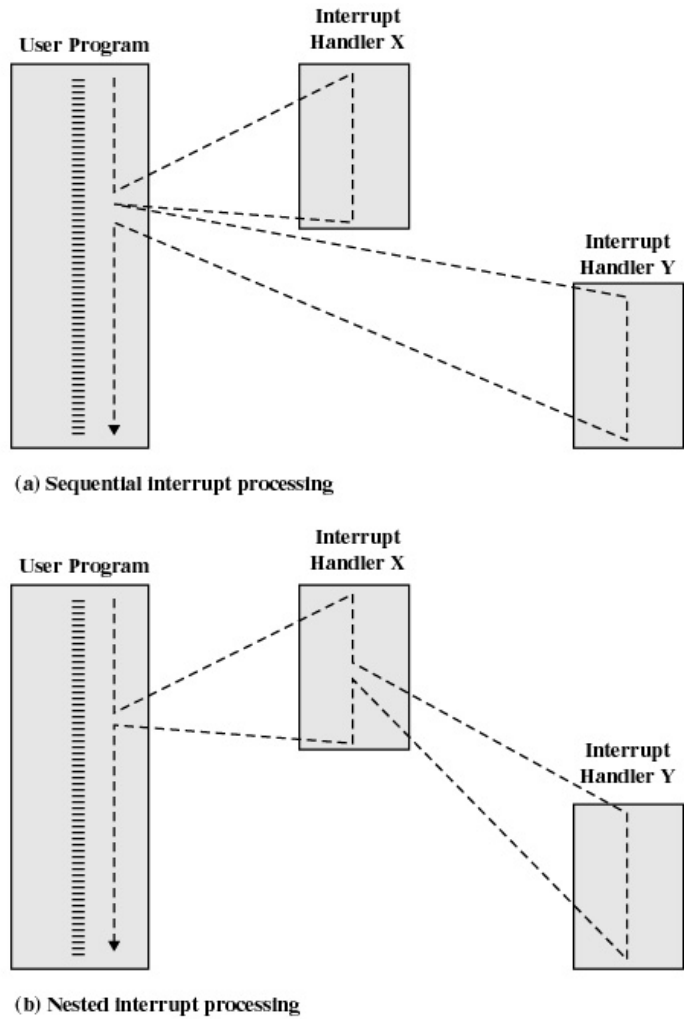


Figure 1.12 Transfer of Control with Multiple Interrupts

Royal Holloway University of London

Multiple Interrupts Sequential Order

- **Disable interrupts so processor can complete task**
- **Interrupts remain pending until the processor enables interrupts**
- **After interrupt handler routine completes, the processor checks for additional interrupts**

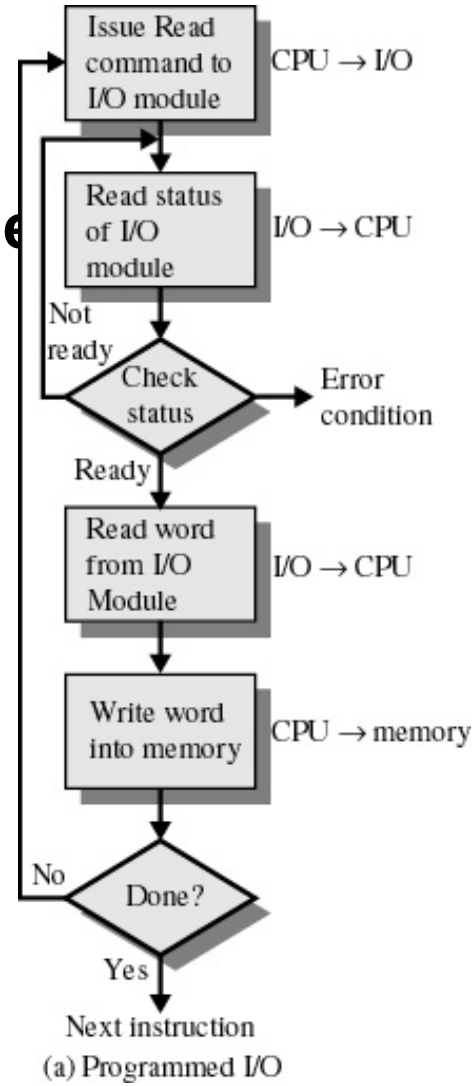
Royal Holloway University of London

Multiple Interrupts
Priorities

- **Higher priority interrupts cause lower-priority interrupts to wait**
- **Causes a lower-priority interrupt handler to be interrupted**
- **Example when input arrives from communication line, it needs to be absorbed quickly to make room for more input**

Programmed I/O

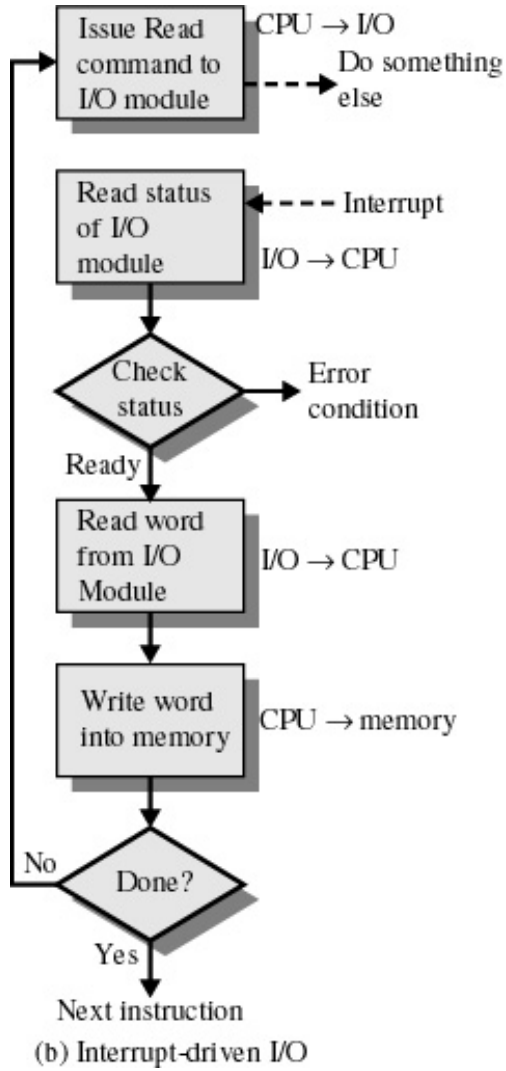
- I/O module performs the action, not the processor
- Sets appropriate bits in the I/O status register
- No interrupts occur
- Processor checks status until operation is complete



(a) Programmed I/O

Interrupt-Driven I/O

- **Processor is interrupted when I/O module ready to exchange data**
- **Processor is free to do other work**
- **No needless waiting**
- **Consumes a lot of processor time because every word read or written passes through the processor**

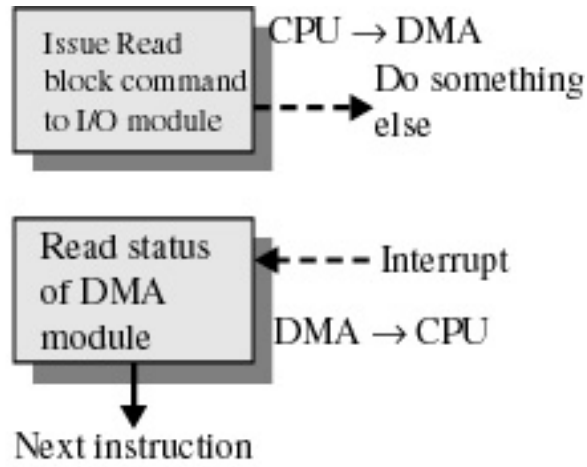


(b) Interrupt-driven I/O

Royal Holloway University of London

Direct Memory Access

- **Transfers a block of data directly to or from memory**
- **An interrupt is sent when the task is complete**
- **The processor is only involved at the beginning and end of the transfer**



(c) Direct memory access

Royal Holloway University of London

Summary

- **We have covered**
  - **Basic components of a computer**
  - **How computers execute instructions**
  - **What are interrupts**
  - **How computers manage interrupts**
  - **What is DMA**

Royal Holloway University of London

Next Lecture

- **On Monday, BLT1**
- **We will talk in more detail about I/O**
- **Lecture Notes: http://www.cs.rhul.ac.uk/~karl**