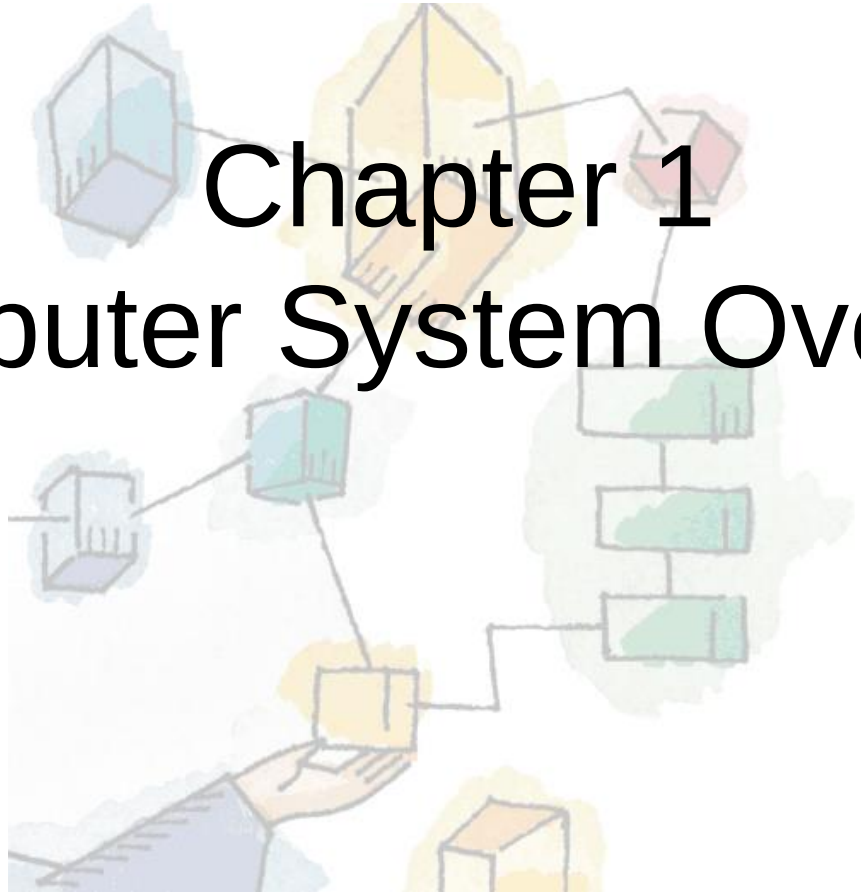
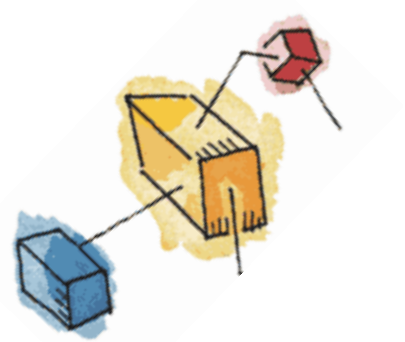
Operating Systems:
Internals and Design Principles, 6/E
William Stallings



Chapter 1
Computer System Overview

Patricia Roy
Manatee Community College, Venice, FL

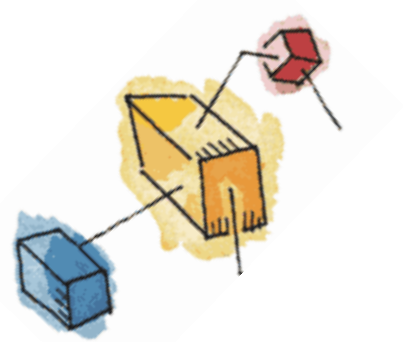


Operating System

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

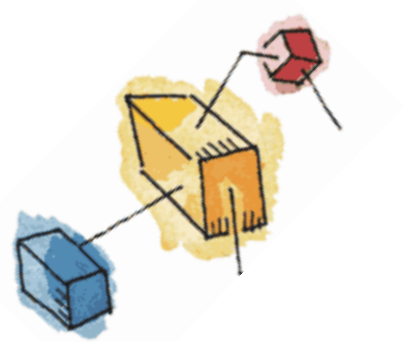
Basic Elements

- Processor
  - Two internal registers
    - Memory address resister (MAR)
      - Specifies the address for the next read or write
    - Memory buffer register (MBR)
      - Contains data written into memory or receives data read from memory



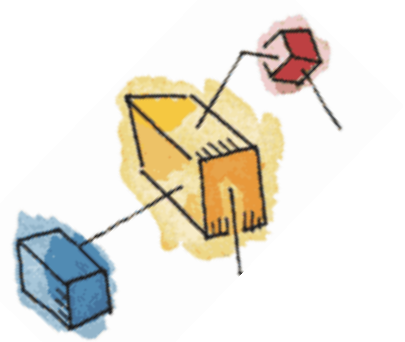
Basic Elements

- Processor
  - I/O address register
  - I/O buffer register



Basic Elements

- Main Memory
  - Volatile
  - Referred to as real memory or primary memory



Basic Elements

- I/O Modules
  - Secondary Memory Devices
  - Communications equipment
  - Terminals
- System bus
  - Communication among processors, main memory, and I/O modules

Computer Components: Top-Level View

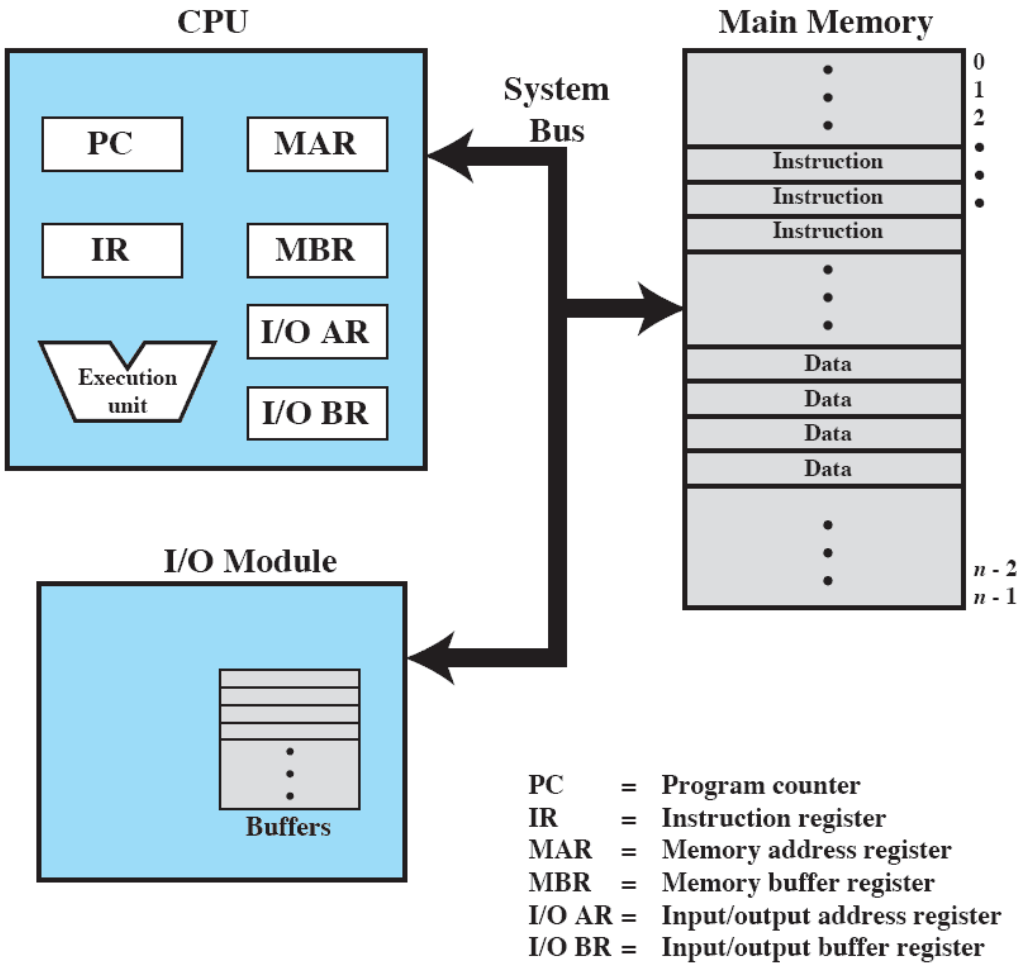
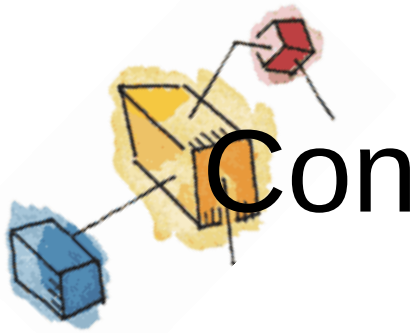
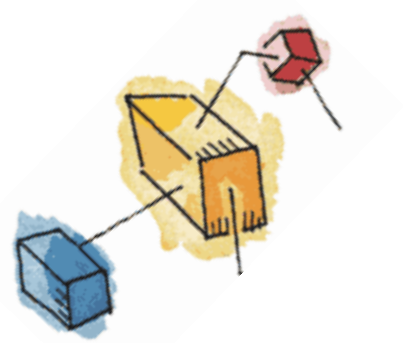


Figure 1.1 Computer Components: Top-Level View



Control and Status Registers

- Program counter (PC)
  - Contains the address of an instruction to be fetched
- Instruction register (IR)
  - Contains the instruction most recently fetched
- Program status word (PSW)
  - Contains status information



Instruction Execution

- Two steps
  - Processor reads (fetches) instructions from memory
  - Processor executes each instruction

Basic Instruction Cycle

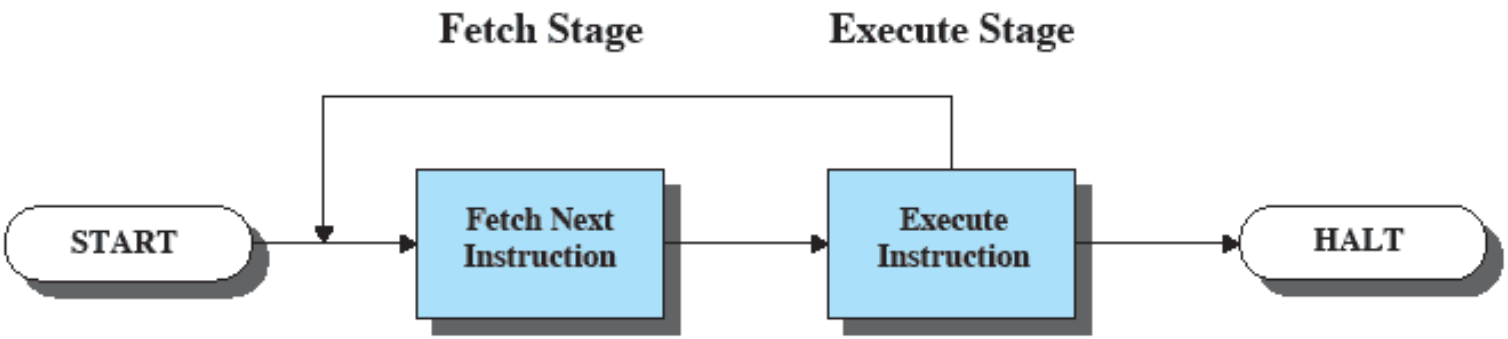
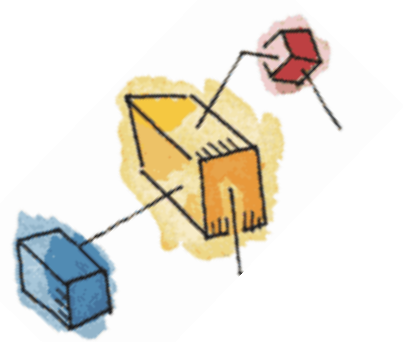


Figure 1.2 Basic Instruction Cycle

Instruction Fetch and Execute

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

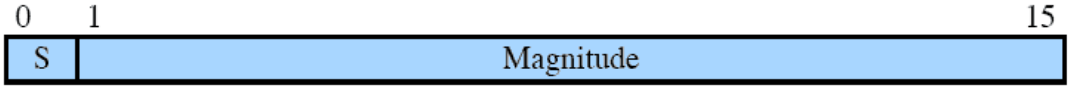


Instruction Register

- Fetched instruction loaded into instruction register
- Categories
  - Processor-memory, processor-I/O, data processing, control

Characteristics of a Hypothetical Machine

(a) Instruction format



(b) Integer format

Program counter (PC) = Address of instruction
Instruction register (IR) = Instruction being executed
Accumulator (AC) = Temporary storage

(c) Internal CPU registers

0001 = Load AC from memory
0010 = Store AC to memory
0101 = Add to AC from memory

(d) Partial list of opcodes

Figure 1.3 Characteristics of a Hypothetical Machine

Example of Program Execution

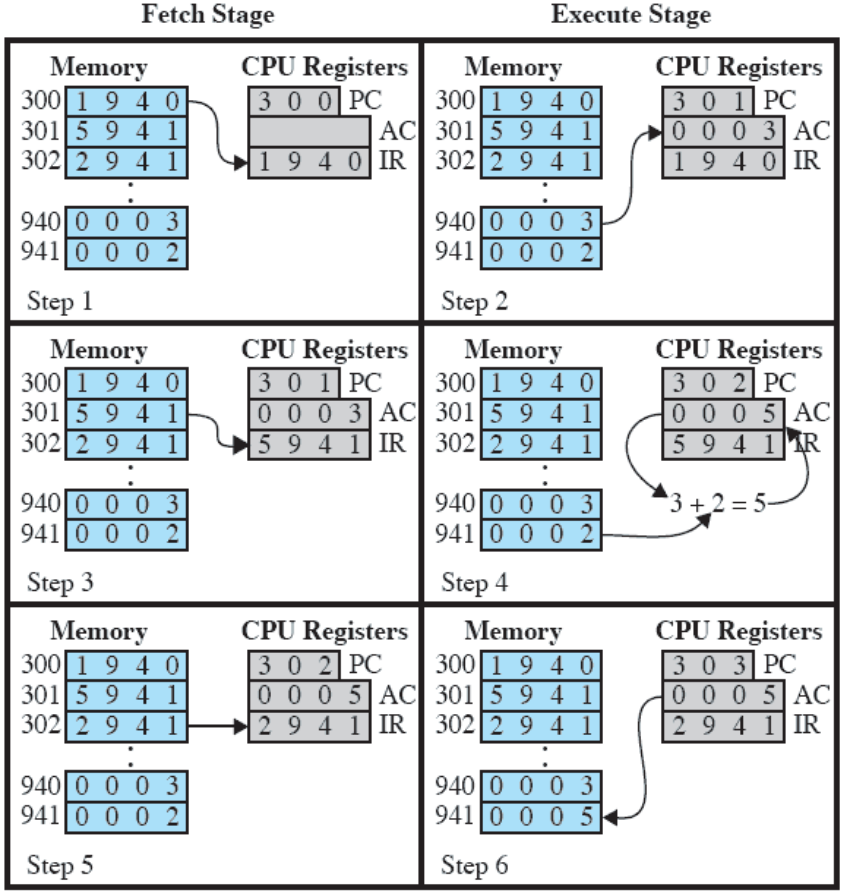
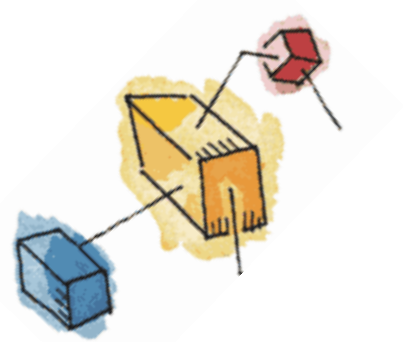


Figure 1.4 Example of Program Execution
(contents of memory and registers in hexadecimal)



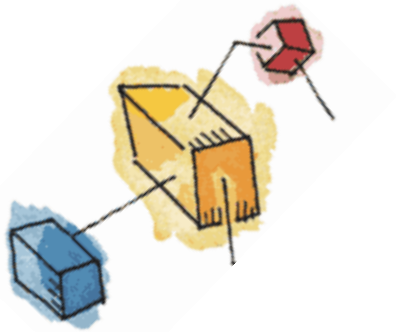
Interrupts

- Interrupt the normal sequencing of the processor
- Most I/O devices are slower than the processor
  - Processor must pause to wait for device

Classes of Interrupts

Table 1.1 Classes of Interrupts

Program	Generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, and reference outside a user's allowed memory space.
Timer	Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.
I/O	Generated by an I/O controller, to signal normal completion of an operation or to signal a variety of error conditions.
Hardware failure	Generated by a failure, such as power failure or memory parity error.



Interrupt Stage

- Processor checks for interrupts
- If interrupt
  - Suspend execution of program
  - Execute interrupt-handler routine

Transfer of Control via Interrupts

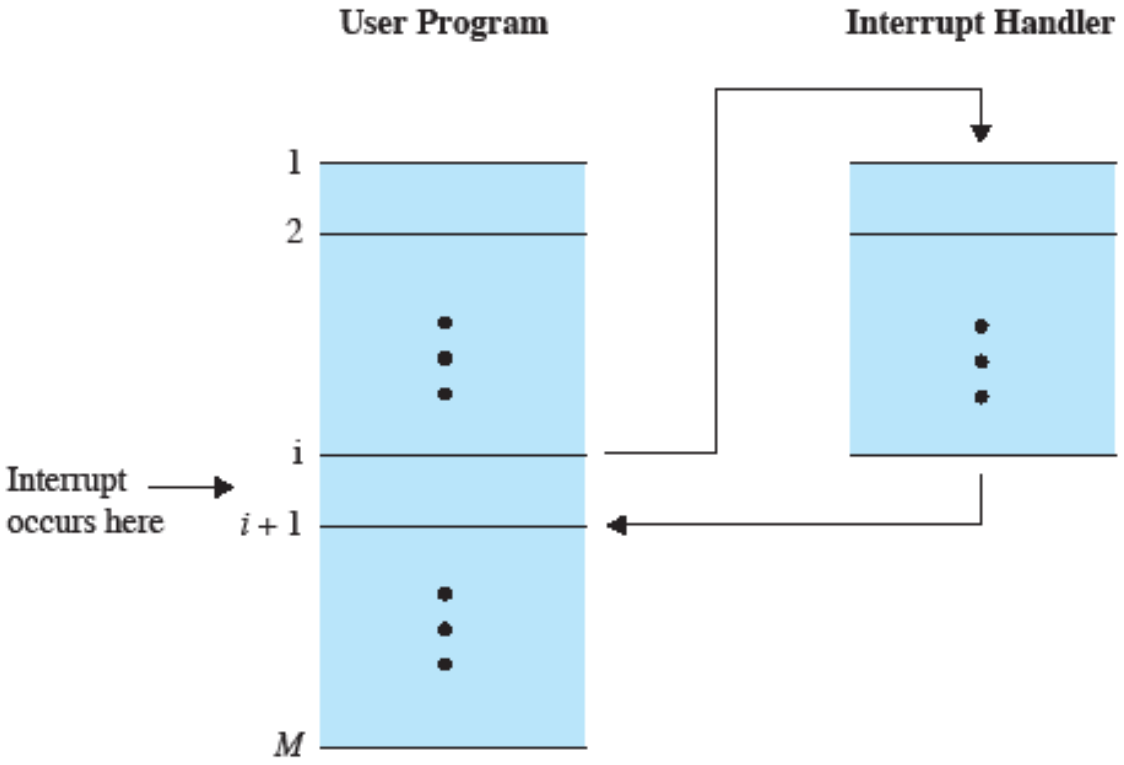


Figure 1.6 Transfer of Control via Interrupts

Instruction Cycle with Interrupts

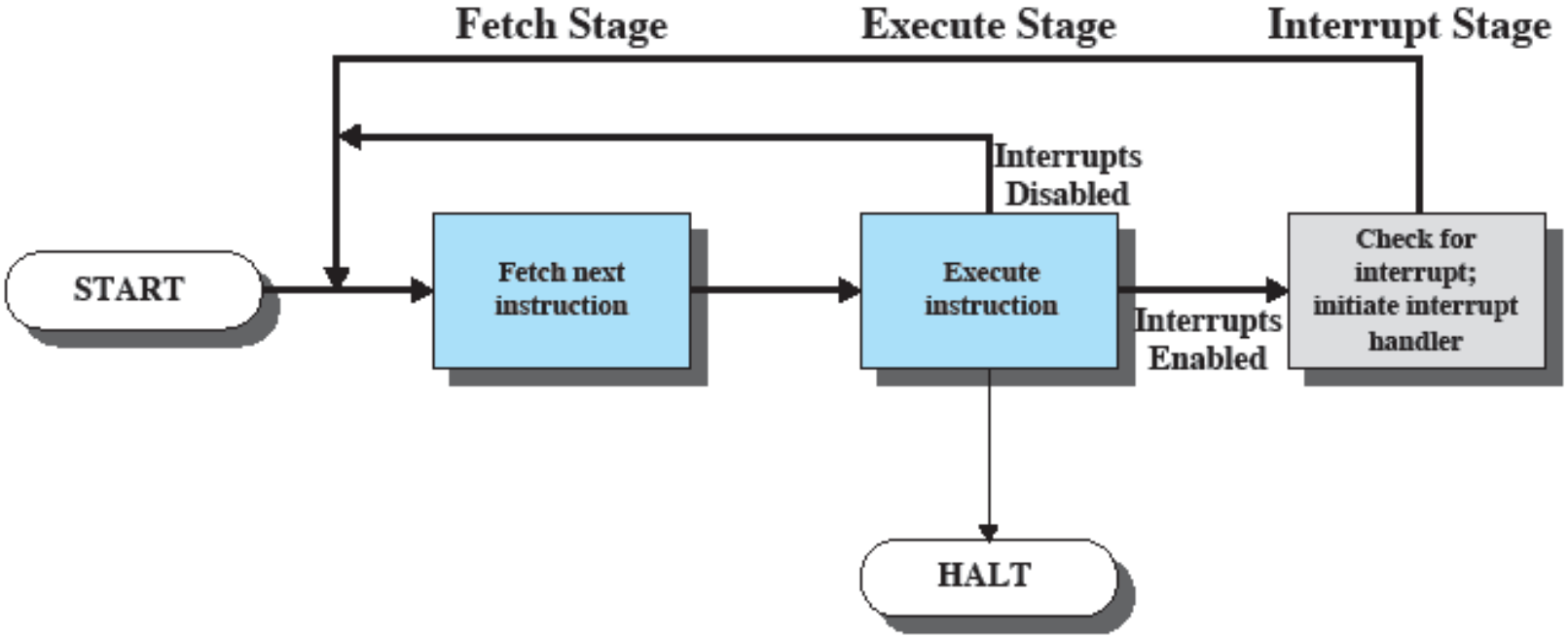
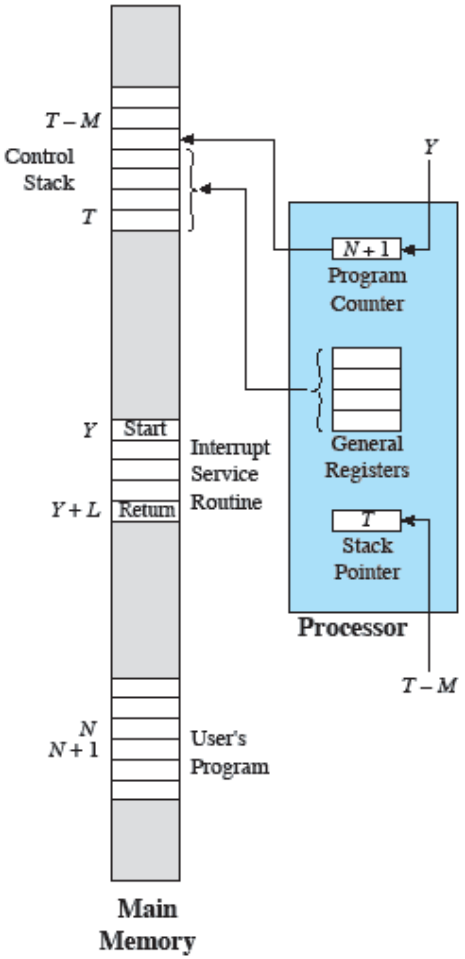


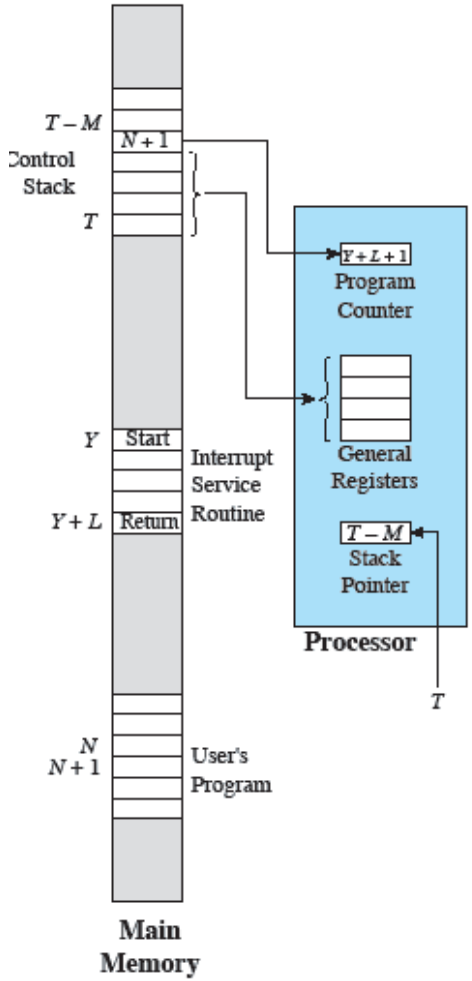
Figure 1.7 Instruction Cycle with Interrupts

Changes in Memory and Registers for an Interrupt



(a) Interrupt occurs after instruction at location N

Changes in Memory and Registers for an Interrupt



(b) Return from interrupt