Computer Systems Overview

Maurizio Pizzonia

slides adattate da

W. Stalling – Operating Systems: Internals and Design Principles

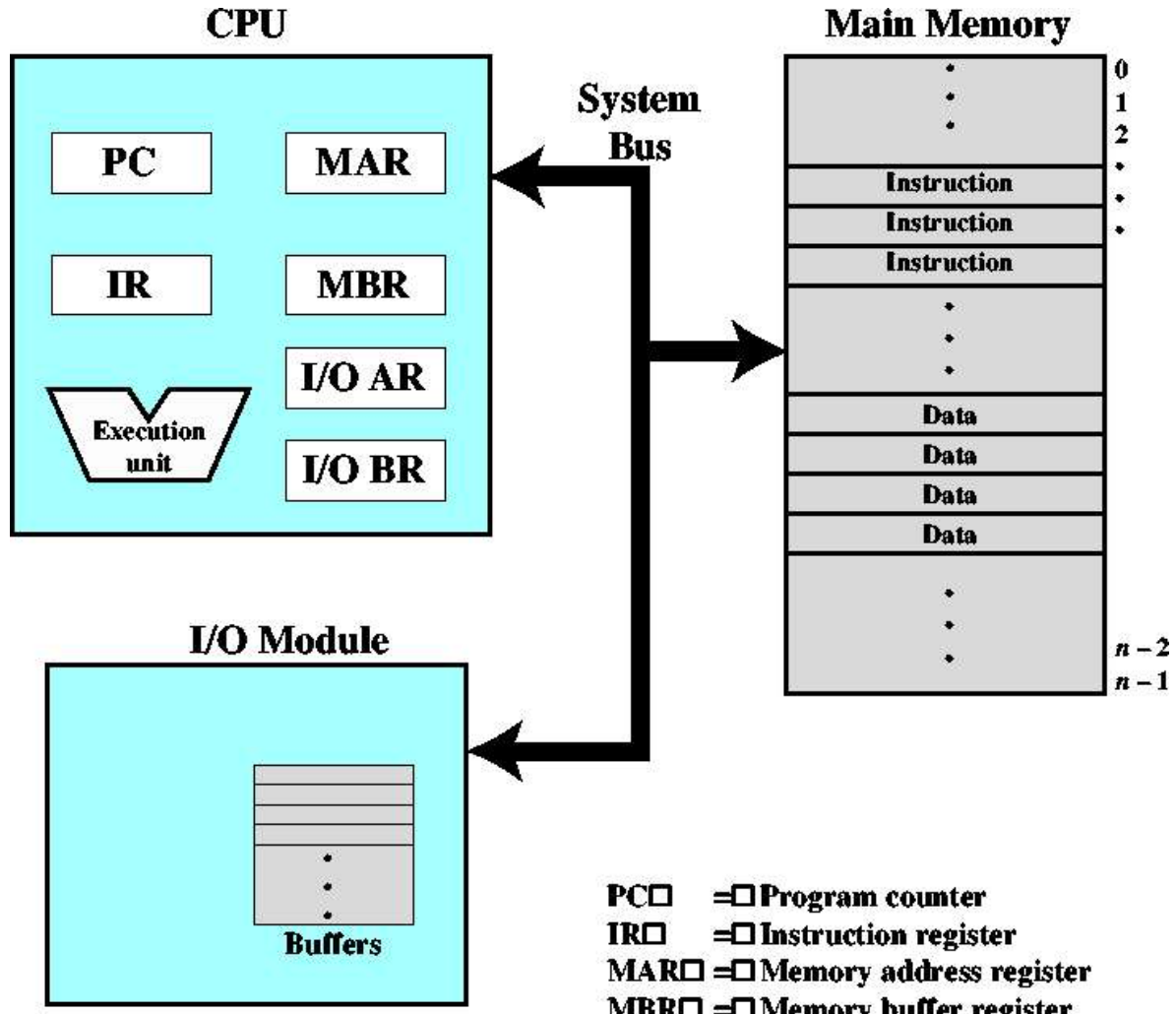
http://williamstallings.com/OS/OS5e.html

Basic Elements

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

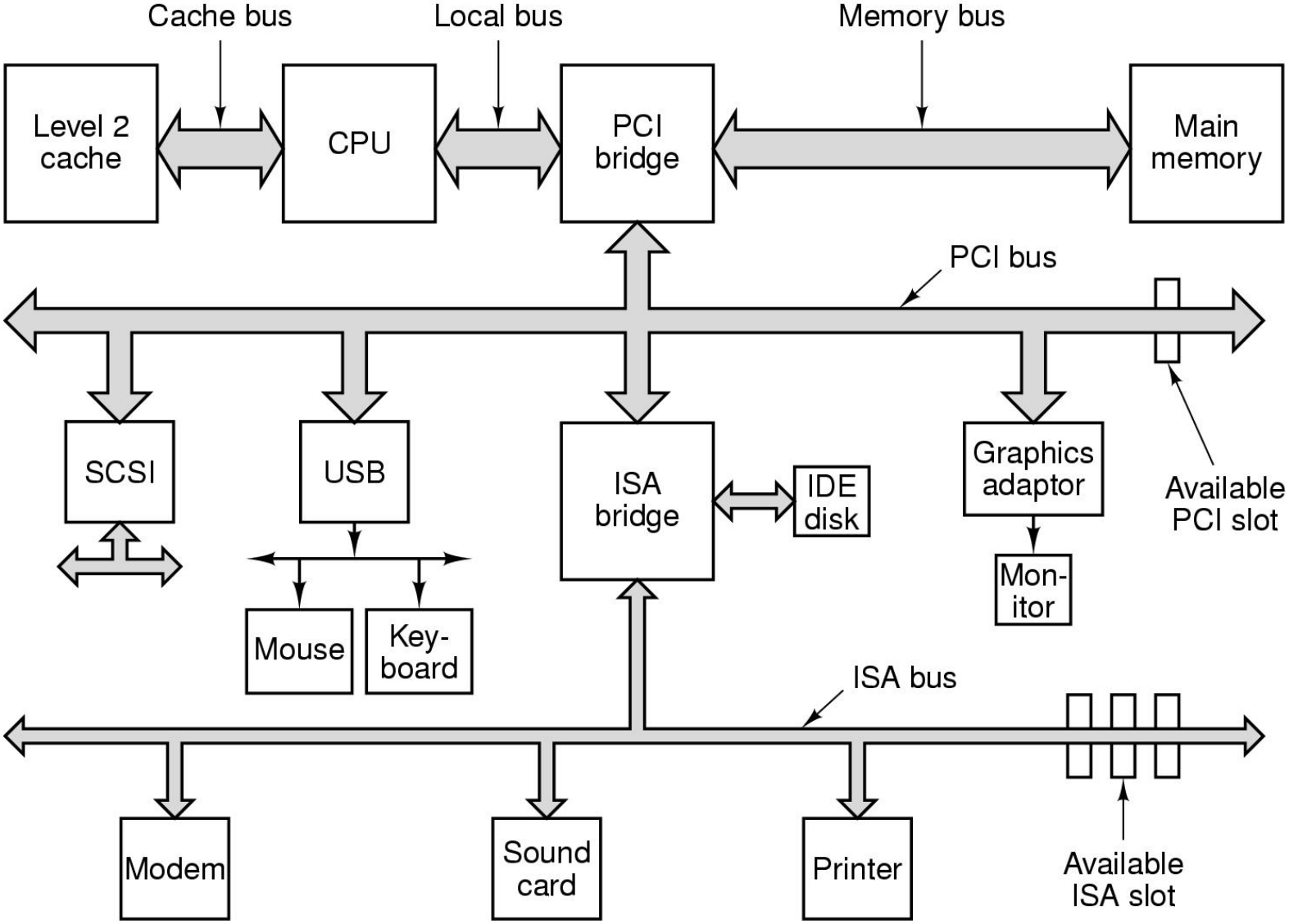
© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Basic Elements



© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

A (Simplified) Large Pentium System



© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Processor Registers

- User-visible registers
  - Enable programmer to minimize main-memory references by optimizing register use
- Control and status registers
  - Used by processor to control operating of the processor
  - Used by privileged operating-system routines to control the execution of programs

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

User-Visible Registers

- May be referenced by machine language
- Available to all programs - application programs and system programs
- Types of registers
  - Data
  - Address
    - Index
    - Segment pointer
    - Stack pointer

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

User-Visible Registers

- Address Registers
  - Index
    - Involves adding an index to a base value to get an address
  - Segment pointer
    - When memory is divided into segments, memory is referenced by a segment and an offset
  - Stack pointer
    - Points to top of stack

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Control and Status Registers

- Program Counter (PC, IP)
  - Contains the address of an instruction to be fetched
- Instruction Register (IR)
  - Contains the instruction most recently fetched
- Program Status Word (PSW)
  - Condition codes
  - Interrupt enable/disable
  - Supervisor/user mode

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Control and Status Registers

- Condition Codes or Flags
  - Bits set by the processor hardware as a result of operations
  - Examples
    - Positive result
    - Negative result
    - Zero
    - Overflow

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Instruction Execution

- Two steps
  - Processor reads instructions from memory
    - Fetches
  - Processor executes each instruction

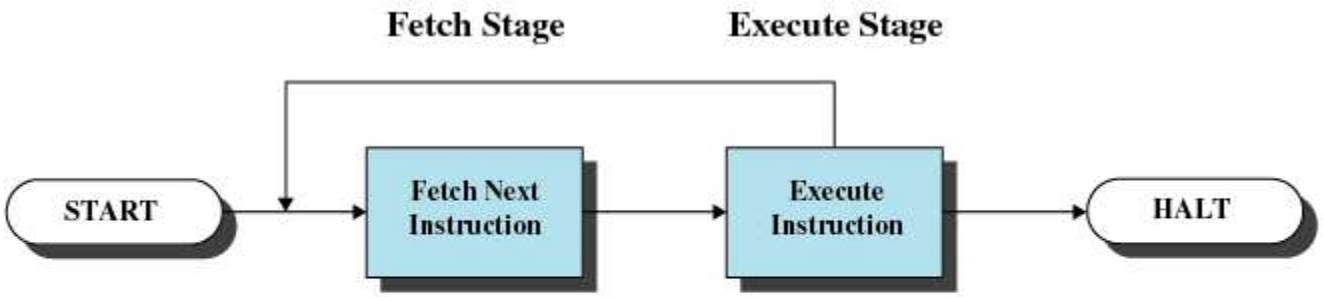


Figure 1.2 Basic Instruction Cycle

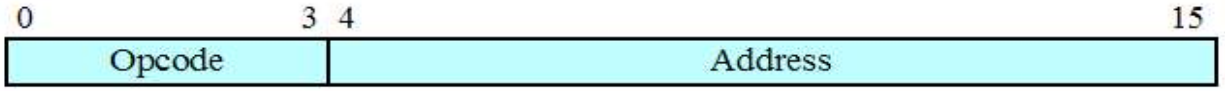
© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Instruction Categories

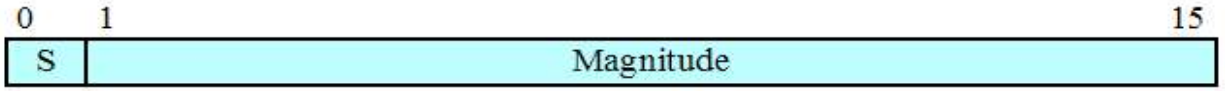
- Processor-memory
  - Transfer data between processor and memory
- Processor-I/O
  - Data transferred to or from a peripheral device
- Data processing
  - Arithmetic or logic operation on data
- Control
  - Alter sequence of execution

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

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

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Example of Program Execution

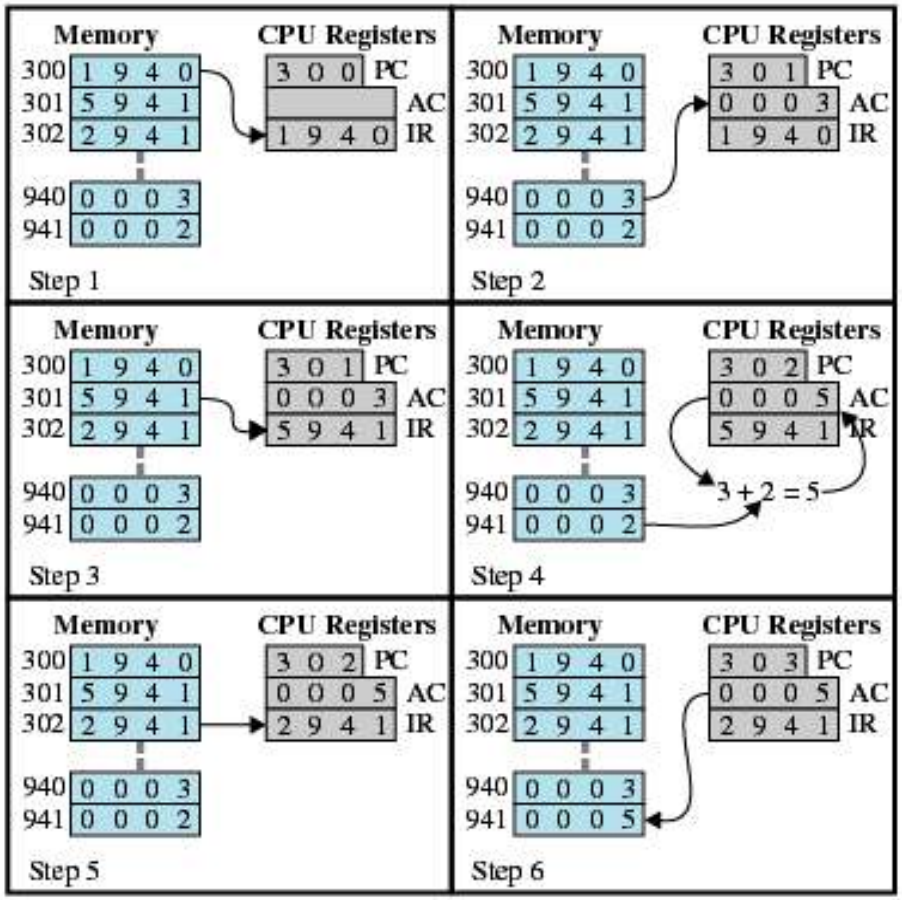


Figure 1.4 Example of Program Execution
(contents of memory and registers in hexadecimal)

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Procedure Calls

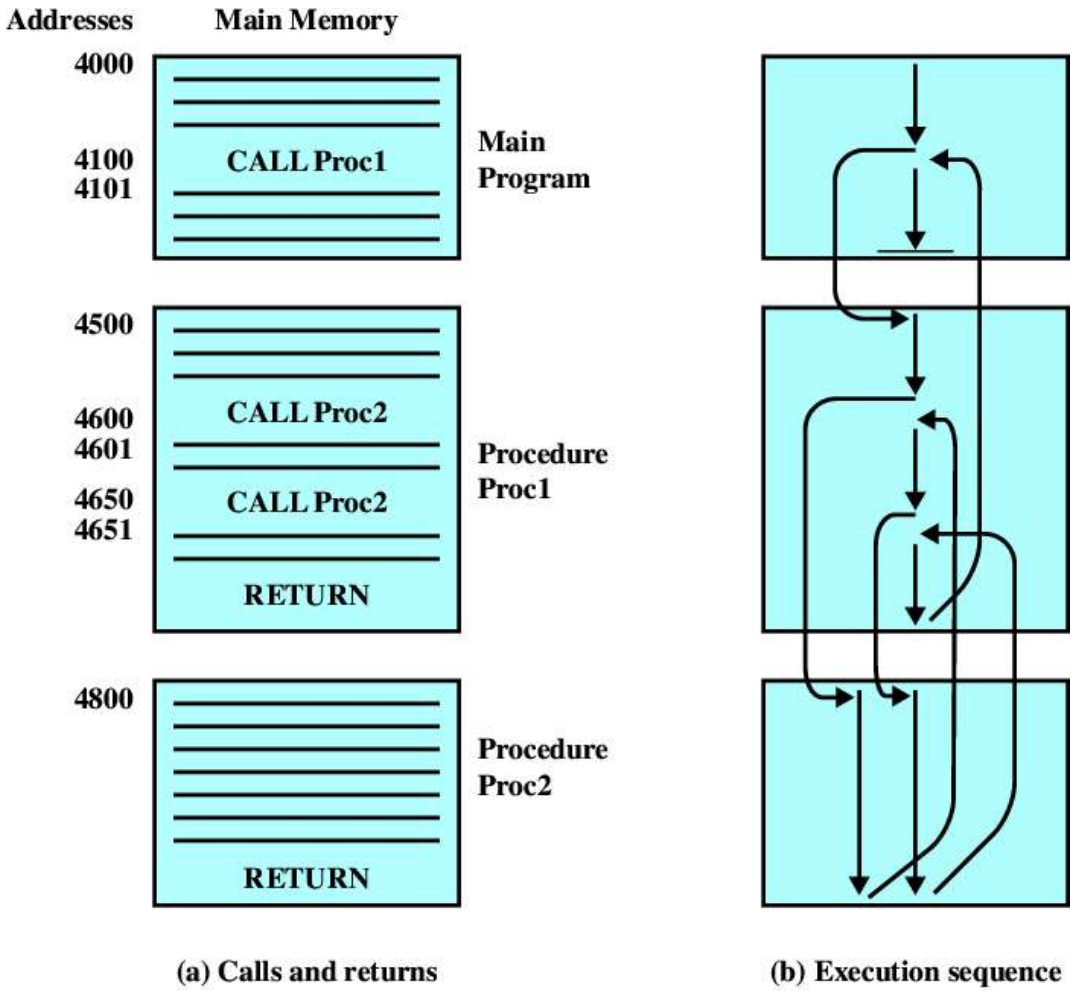
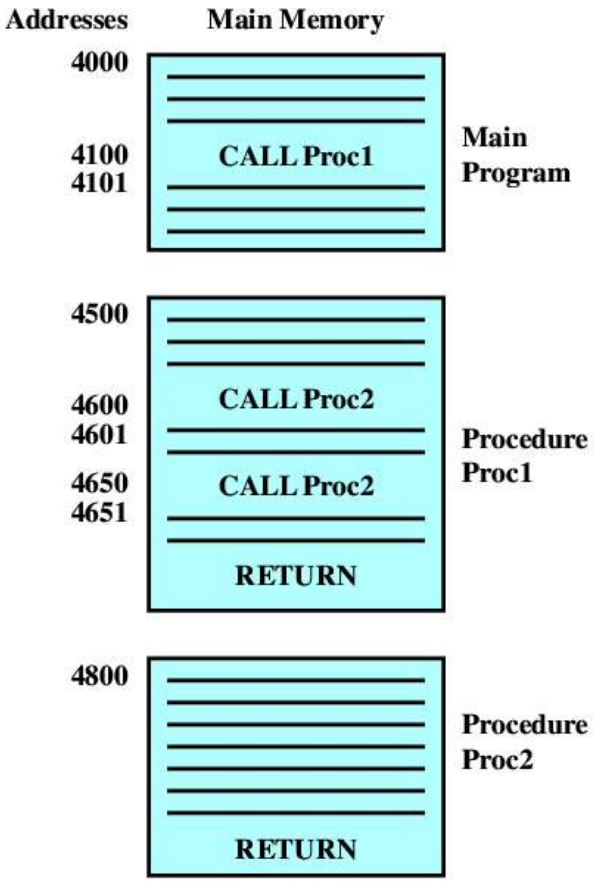
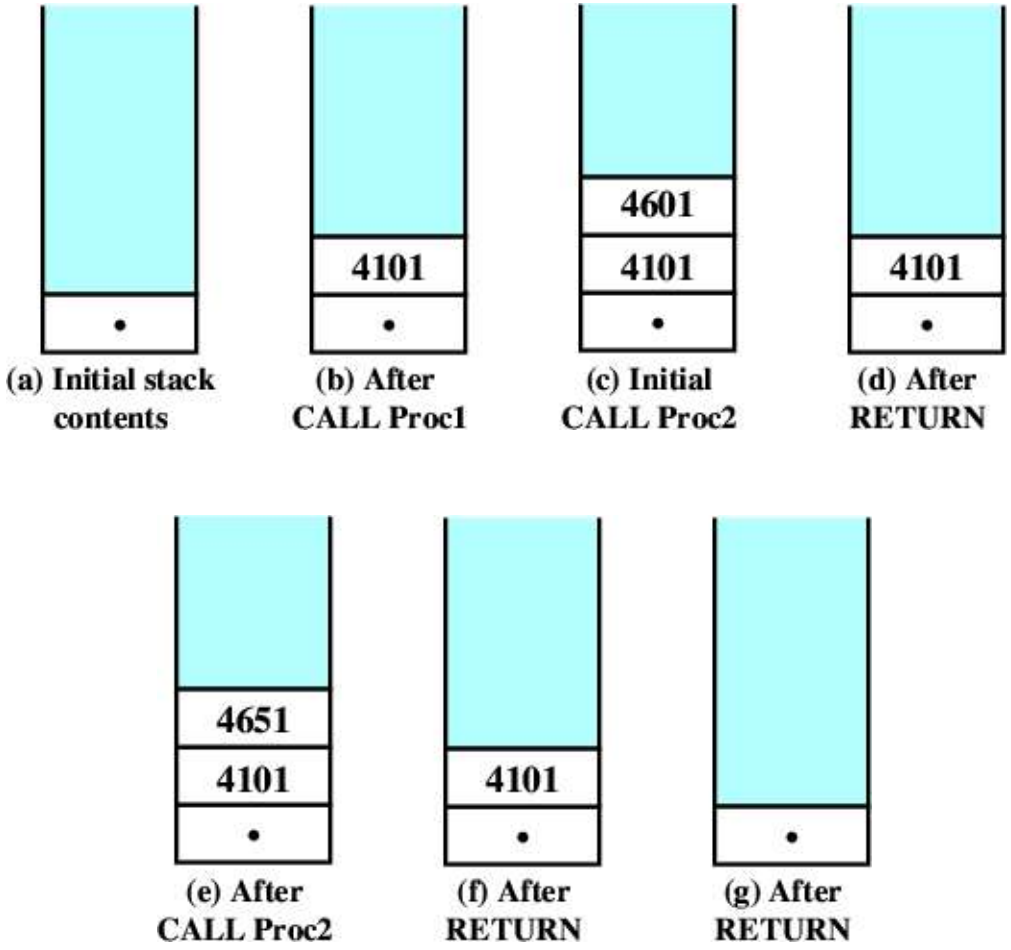


Figure 1.26 Nested Procedures

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

The Call Stack





© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

CPU Registers for the Stack

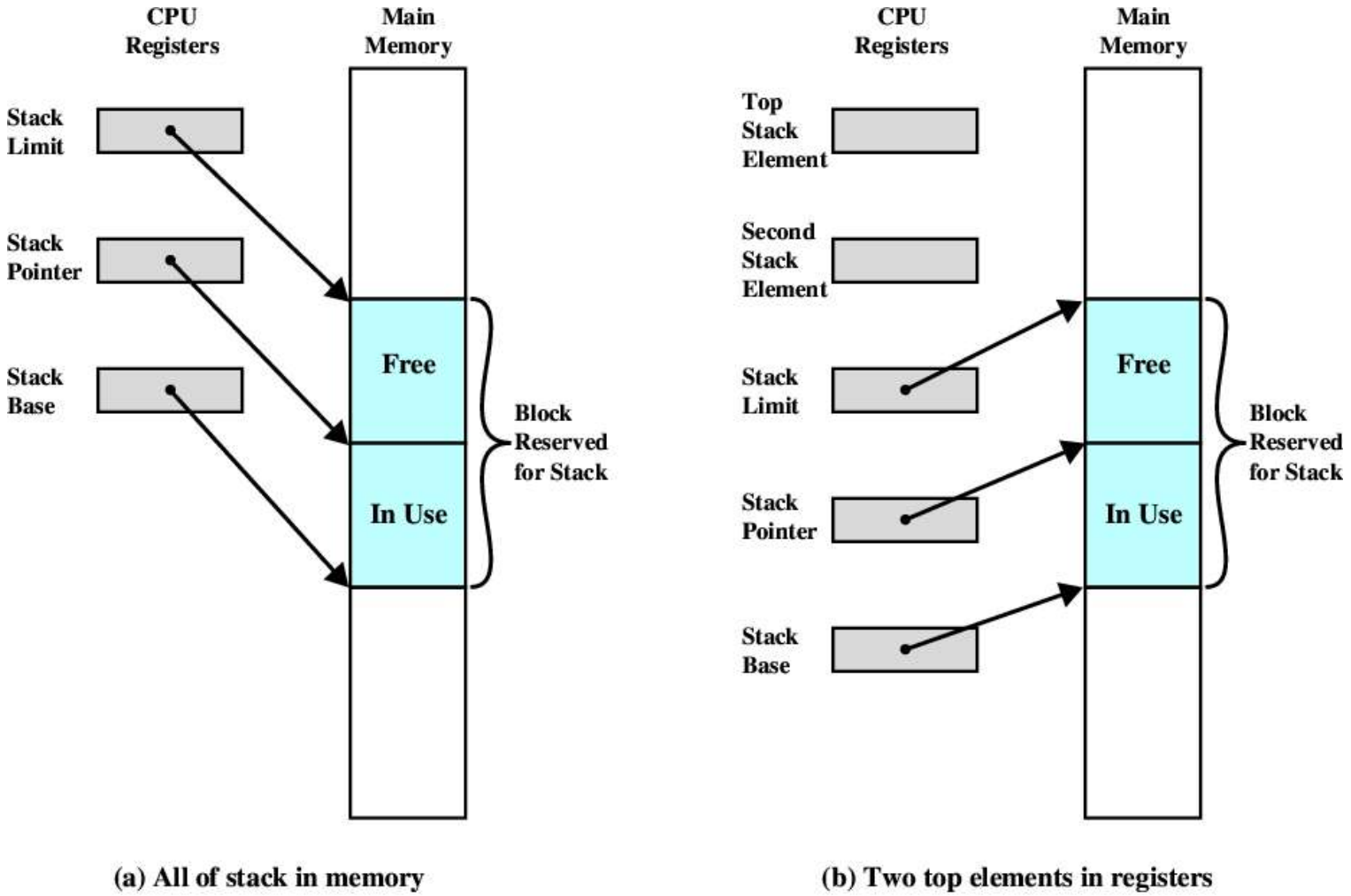


Figure 1.25 Typical Stack Organization

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Interrupts

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

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

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.

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Interrupt Handler

- Program to service a particular I/O device
- Generally part of the operating system

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Interrupts

- Suspends the normal sequence of execution

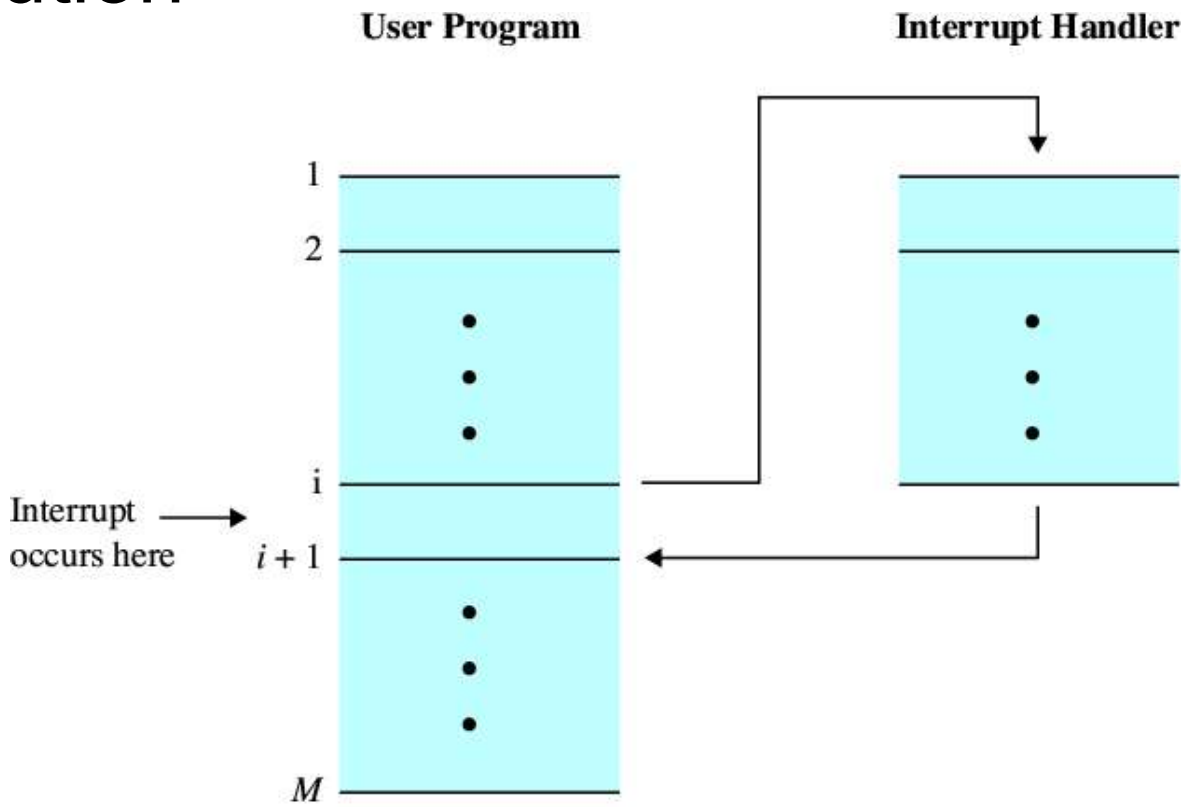


Figure 1.6 Transfer of Control via Interrupts

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Interrupt Cycle

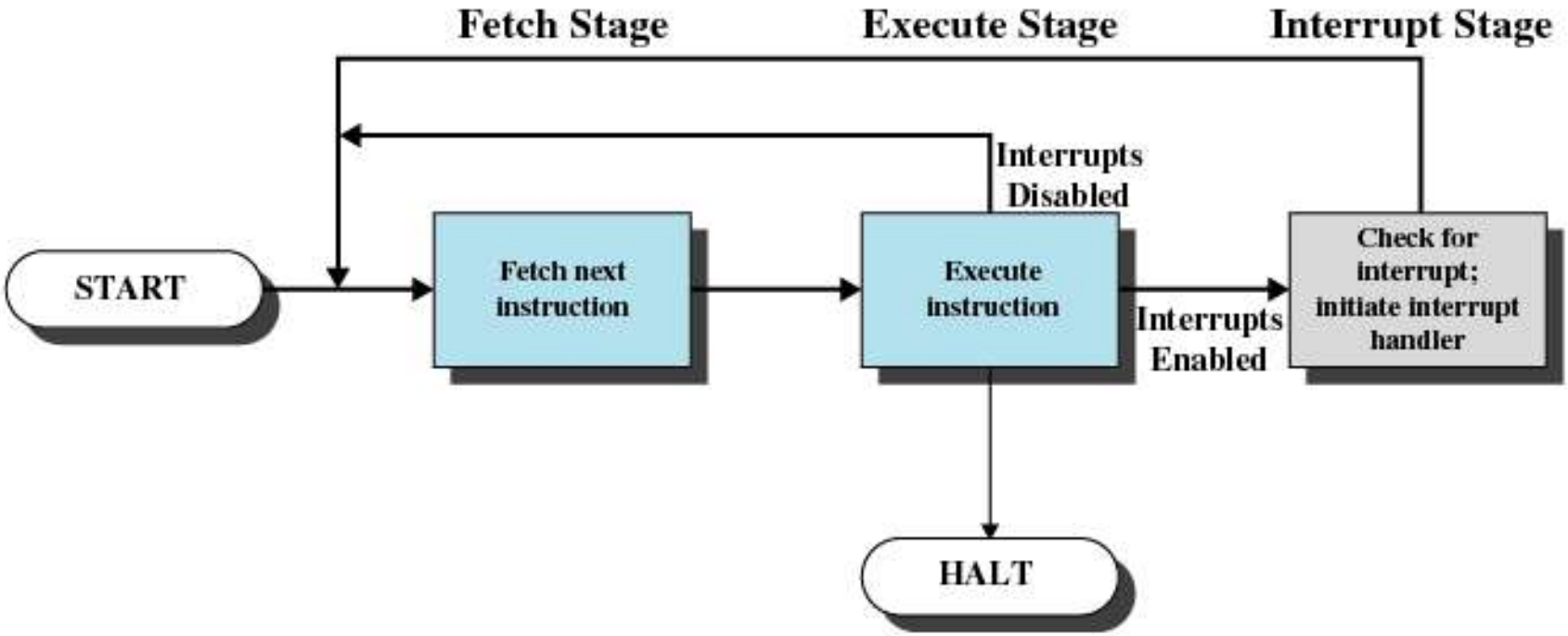


Figure 1.7 Instruction Cycle with Interrupts

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

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 routine

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Simple Interrupt Processing

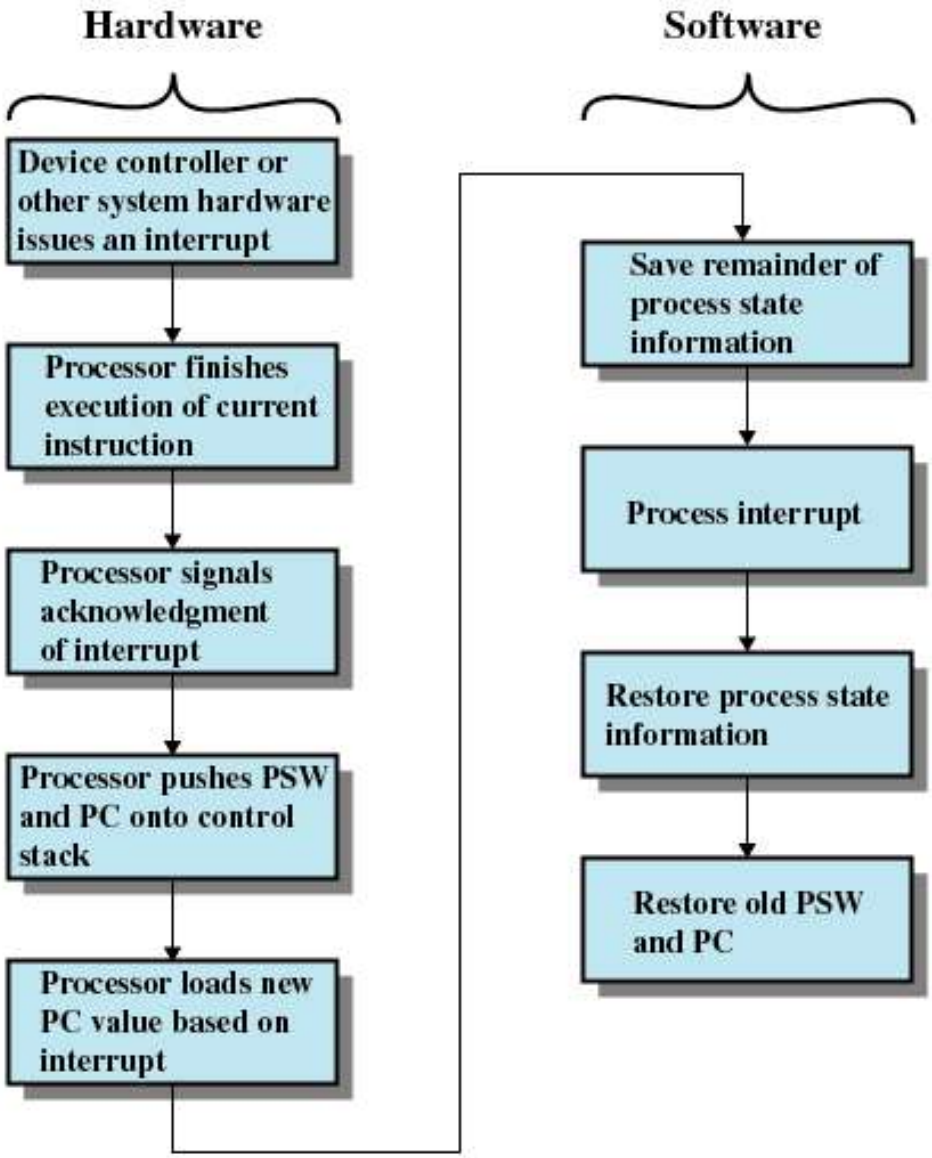


Figure 1.10 Simple Interrupt Processing

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Changes in Memory and Registers for an Interrupt

addresses grow

M=size of registers

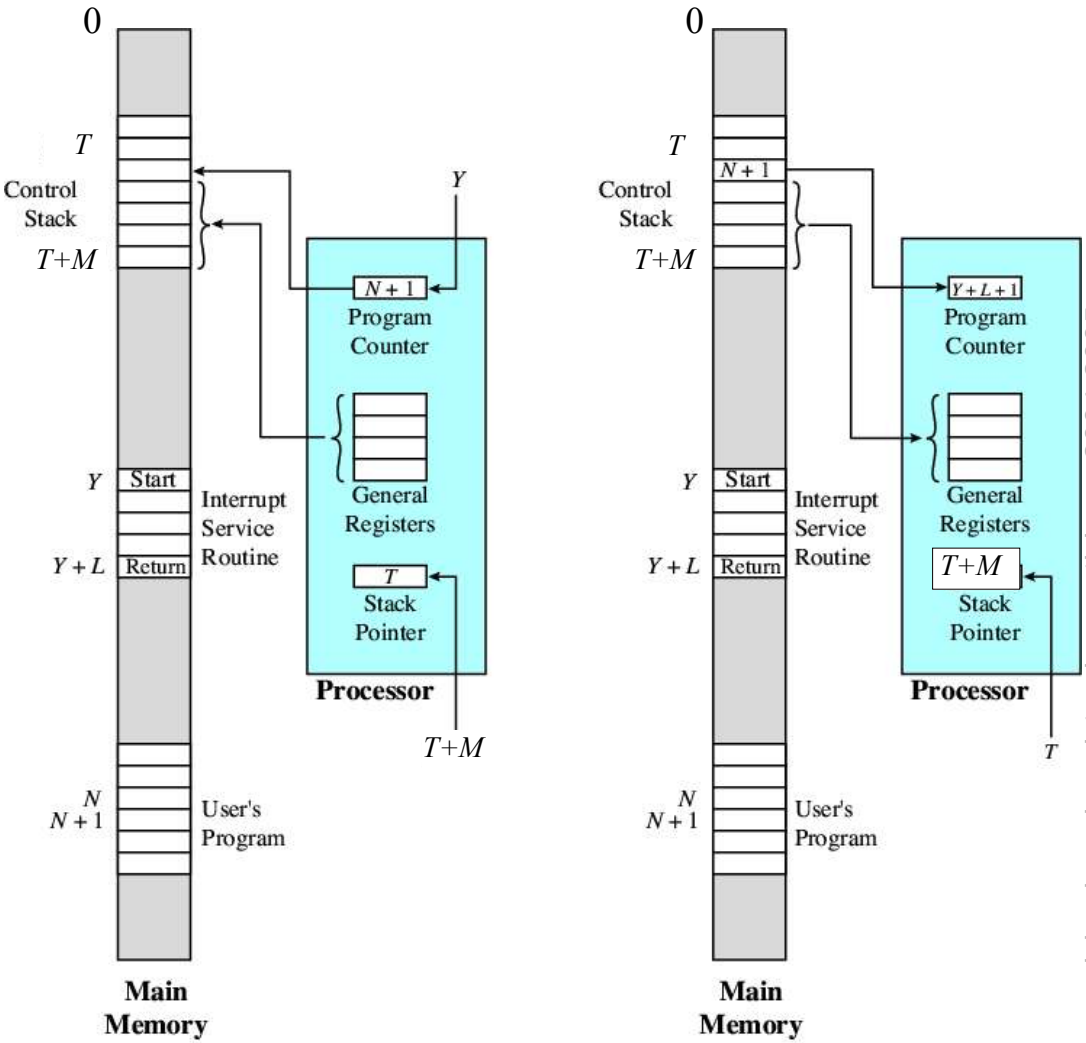
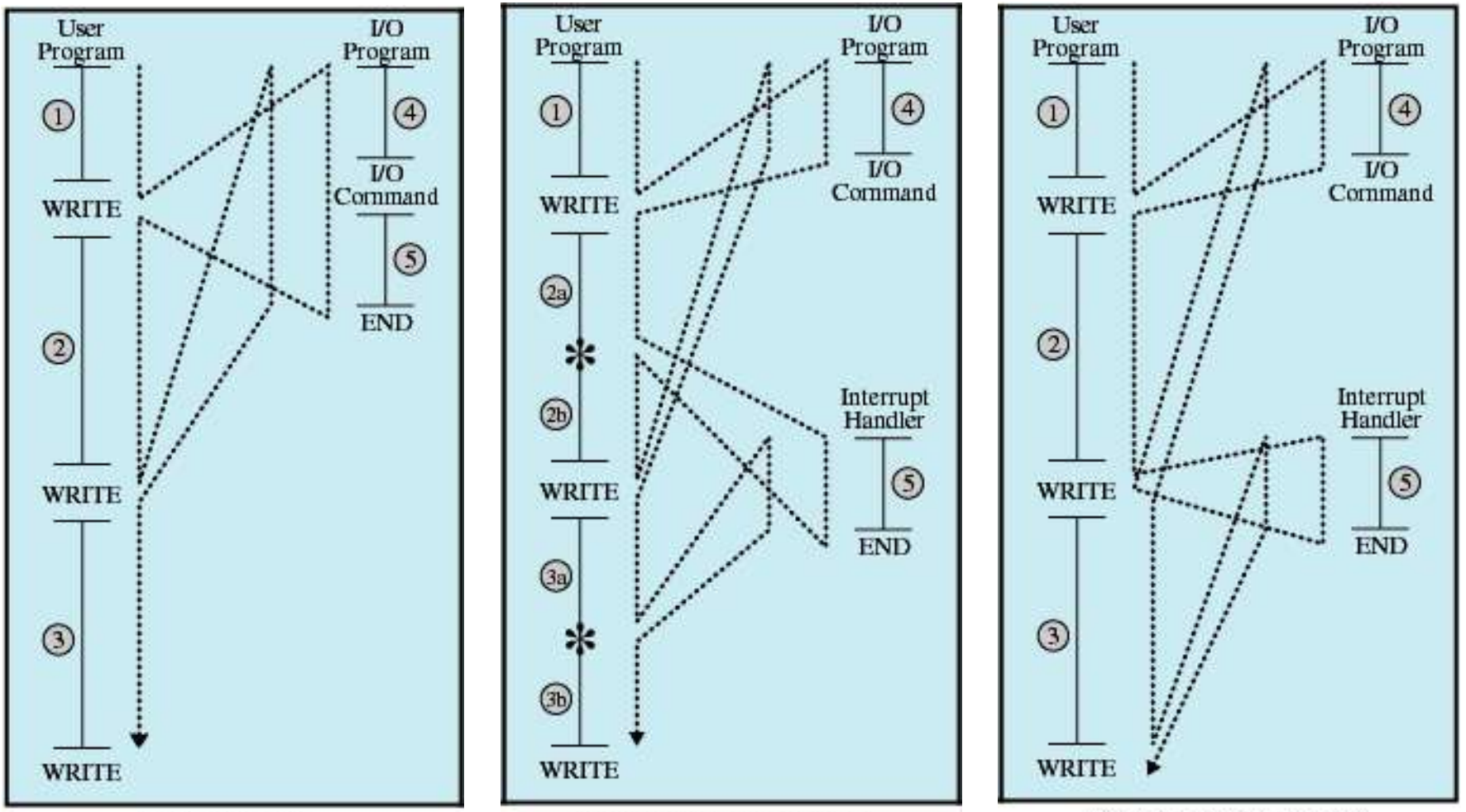


Figure 1.11 Changes in Memory and Registers for an Interrupt

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

I/O With and Without Interrupts



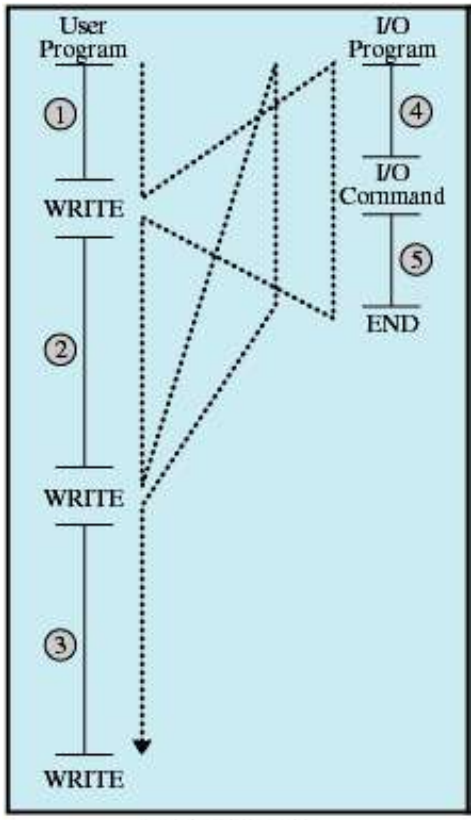
(a) No interrupts

(b) Interrupts; short I/O wait

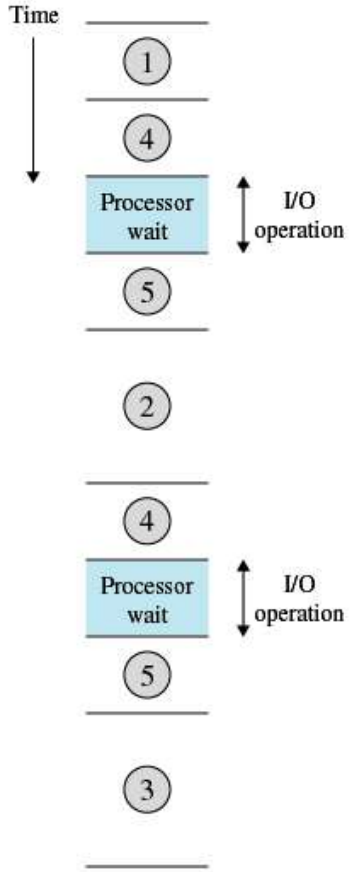
(c) Interrupts; long I/O wait

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

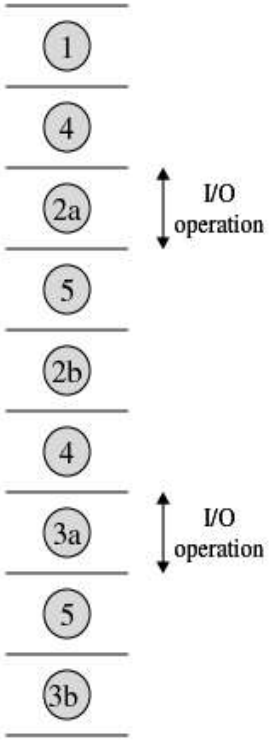
Timing Diagram Based on Short I/O Wait



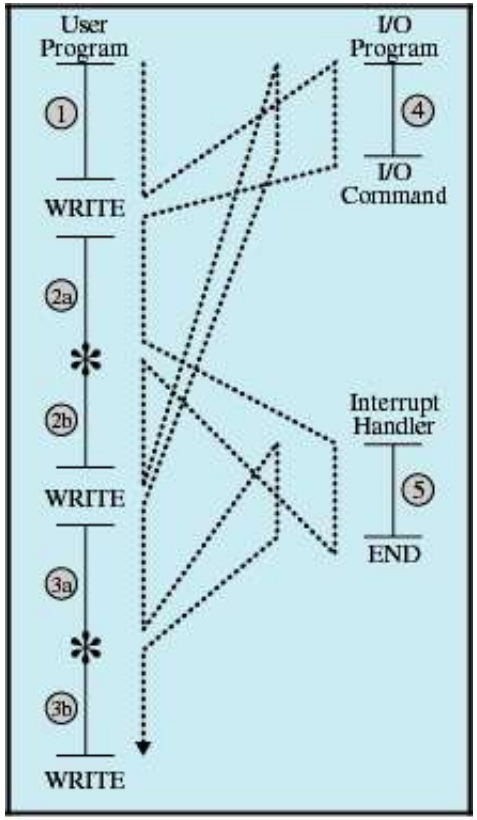
(a) No interrupts



(a) Without interrupts
(circled numbers refer
to numbers in Figure 1.5a)



(b) With interrupts
(circled numbers refer
to numbers in Figure 1.5b)

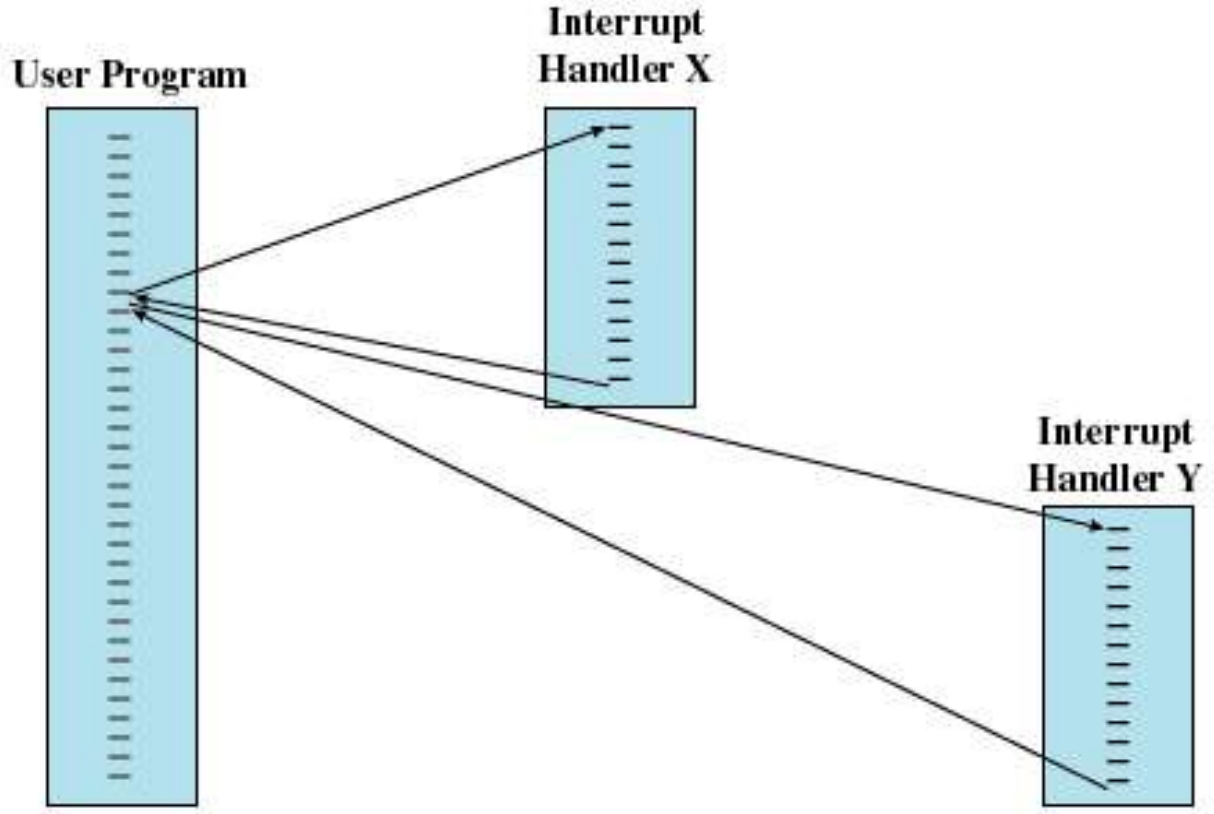


(b) Interrupts; short I/O wait

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Multiple Interrupts

- Disable interrupts while an interrupt is being processed

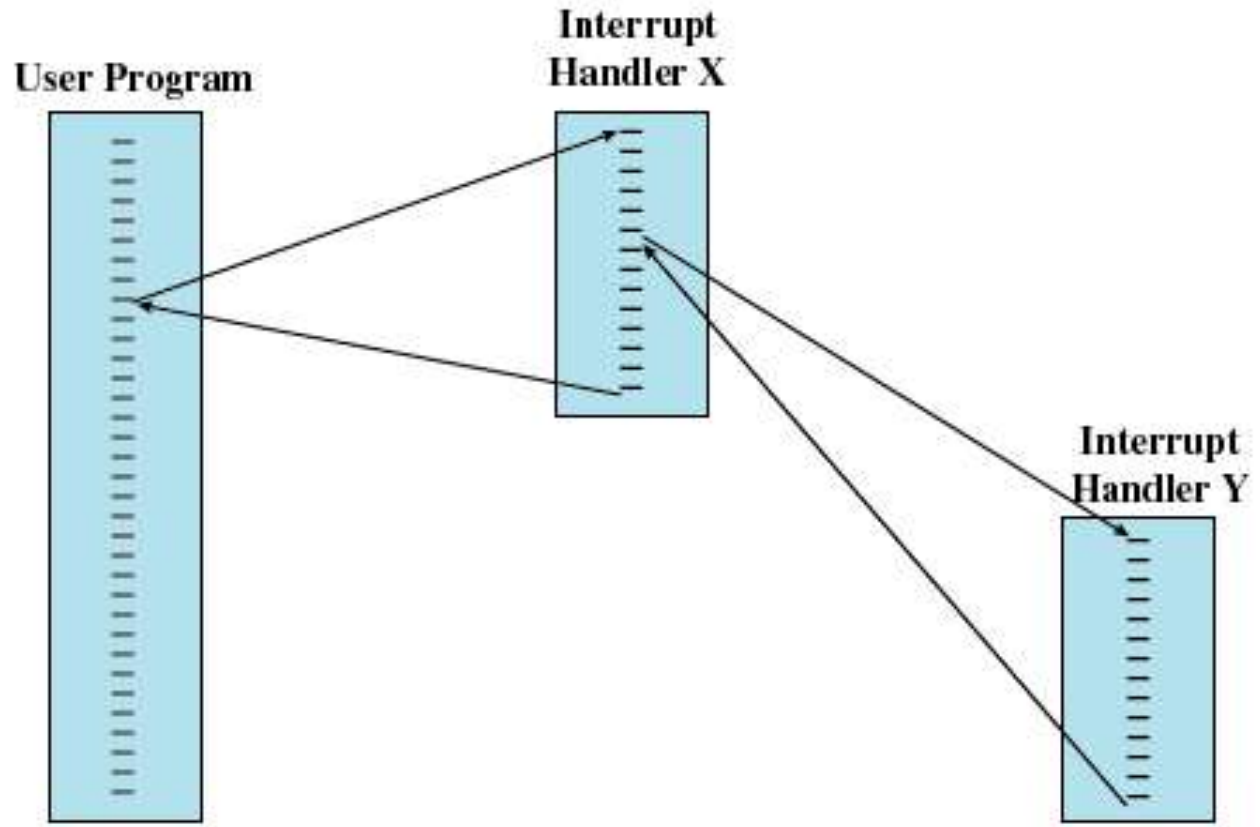


(a) Sequential interrupt processing

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Multiple Interrupts

- Define priorities for interrupts



(b) Nested interrupt processing

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Memory Hierarchy

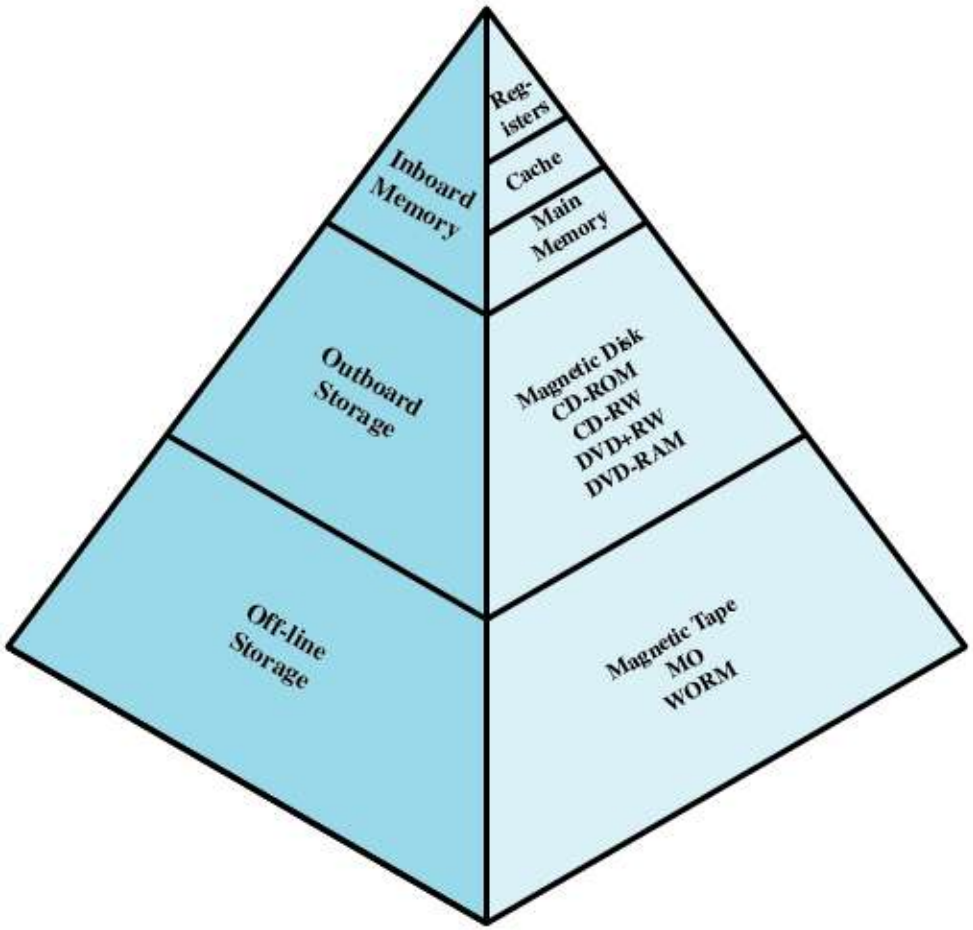


Figure 1.14 The Memory Hierarchy

- Faster access time, greater cost per bit
- Greater capacity
  - smaller cost per bit
  - slower access speed
- Based on Locality
  - temporal
  - spatial

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Secondary Memory

- a.k.a Auxiliary Memory, Mass Storage ore External Memory
- E.g. Disks, pen drives
- Nonvolatile
- Used to store program and data files
- Slow
- Cheap

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Disk Cache

- A portion of main memory used as a buffer to temporarily to hold data for the disk
- Disk writes are clustered
- Some data written out may be referenced again. The data are retrieved rapidly from the software cache instead of slowly from disk

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Cache Memory

- Contains a copy of a portion of main memory
- Processor first checks cache
- If not found in cache, the block of memory containing the needed information is moved to the cache and delivered to the processor
- Associative

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Cache Memory

- Invisible to operating system
- Increase the speed of memory
- Processor speed is faster than memory speed
- On board or on chip

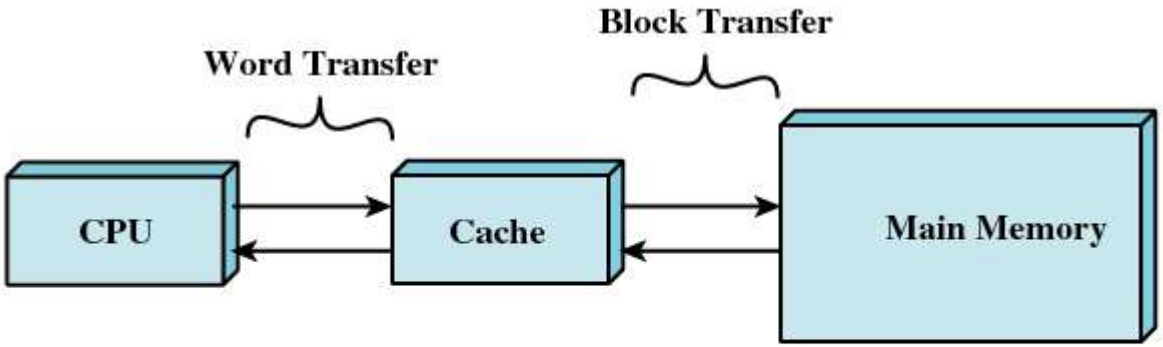


Figure 1.16 Cache and Main Memory

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Cache Memory

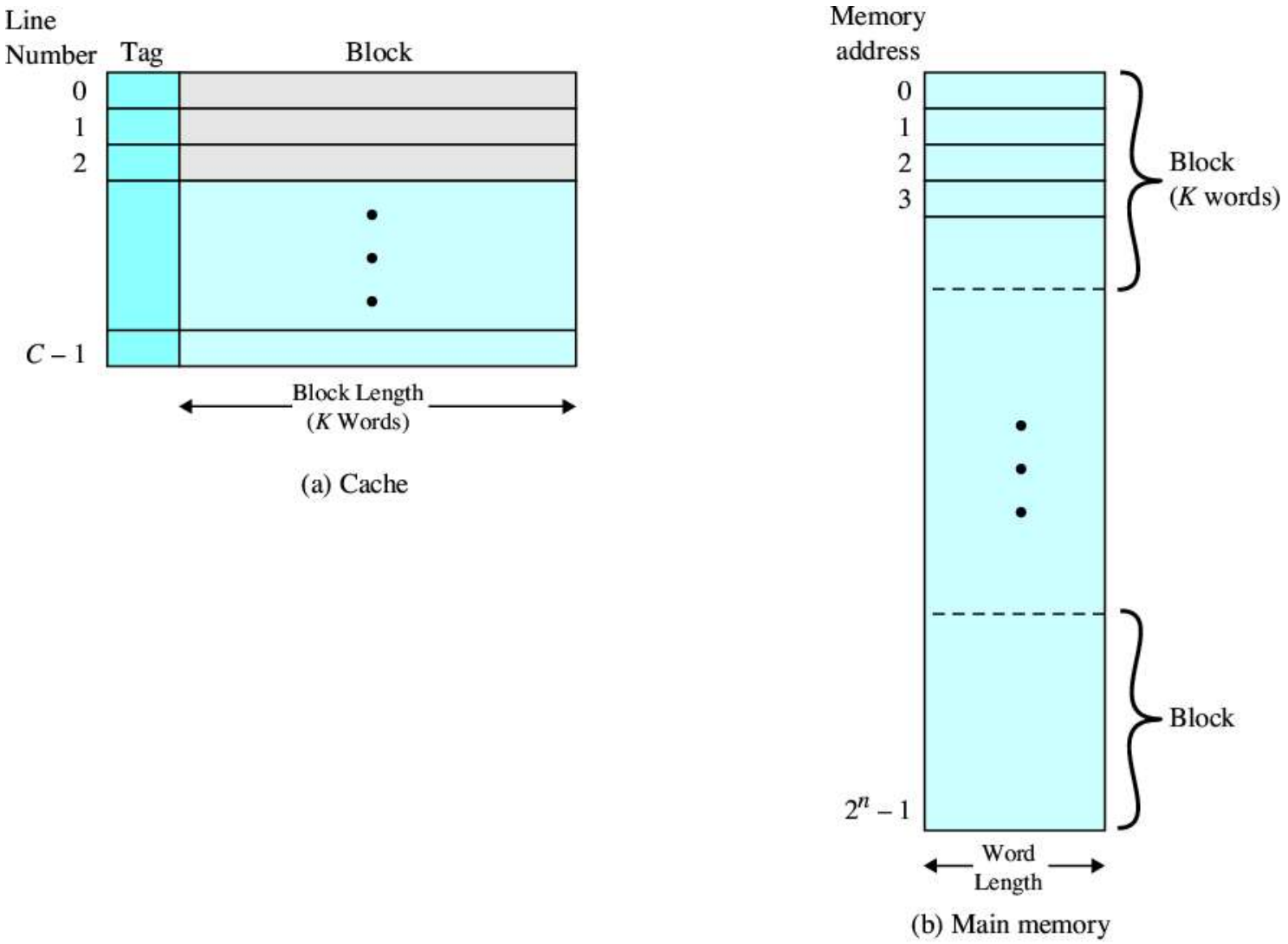


Figure 1.17 Cache/Main-Memory Structure

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Cache Memory

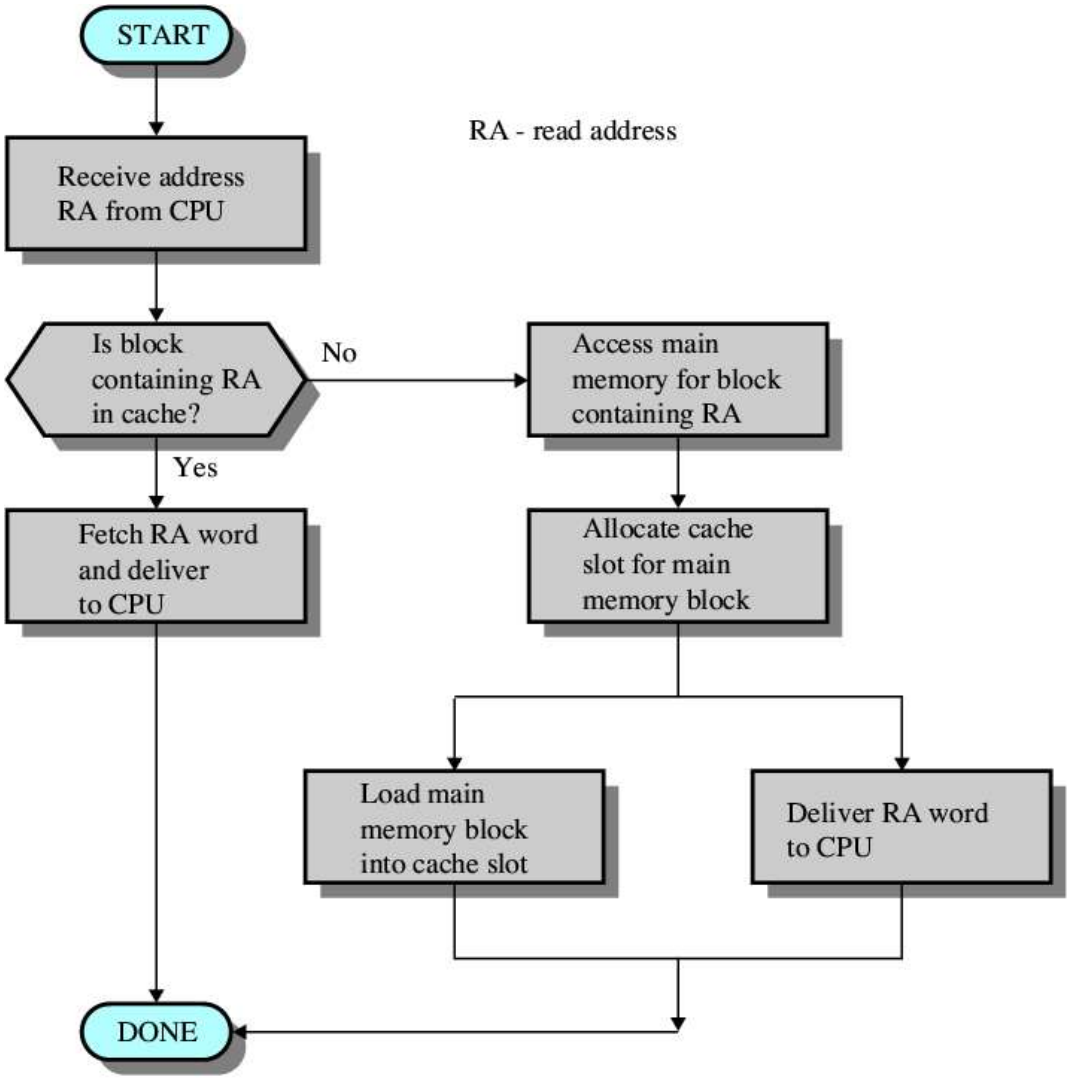
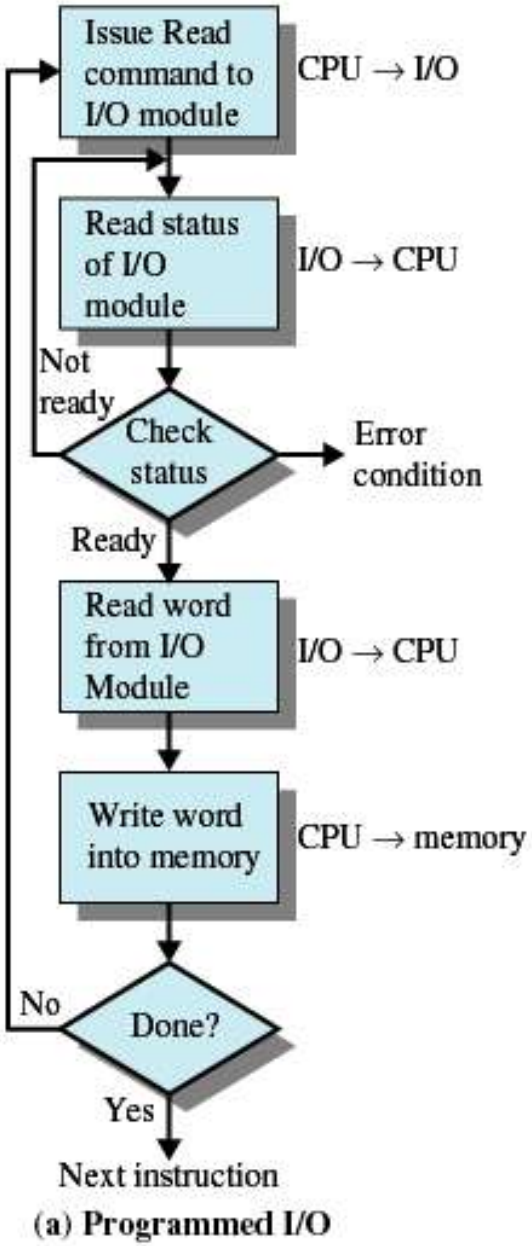


Figure 1.18 Cache Read Operation

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

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.k.a “busy waiting”

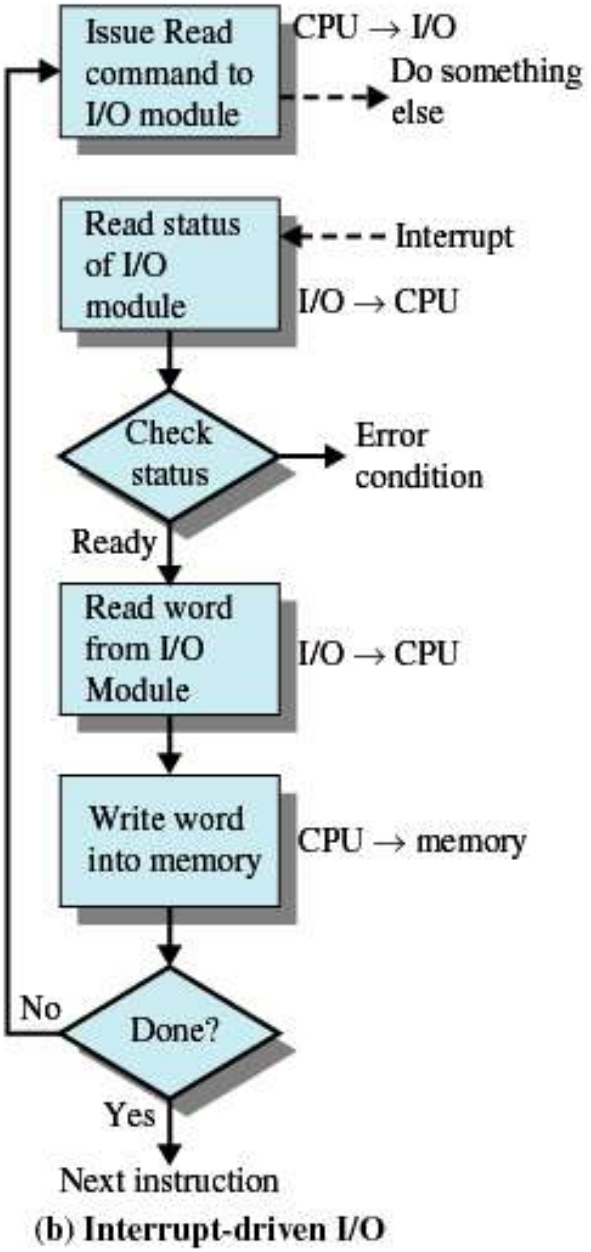


(a) Programmed I/O

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Interrupt-Driven I/O

- Processor is interrupted when I/O module ready to exchange data
- Processor saves context of program executing and begins executing interrupt-handler
- No needless waiting
- Consumes a lot of processor time because every word read or written passes through the processor

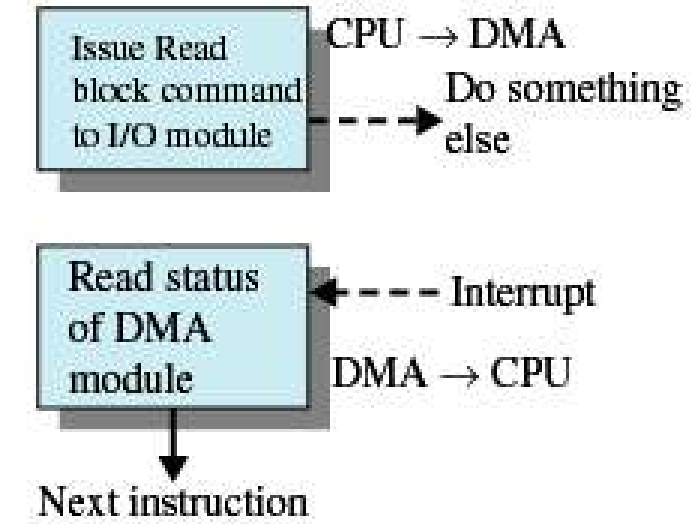


(b) Interrupt-driven I/O

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005

Direct Memory Access

- I/O to/from memory is performed by a special purpose chip (DMA controller)
- Moderated CPU slowdown
  - setup time
  - shared bus
- An interrupt is sent when the transfer is complete
- Processor continues with other work



(c) Direct memory access

© 2005 maurizio pizzonia - sistemi operativi a.a. 2004-2005