

**Computer System
Overall Structure
Ch 1-7**

Review
Overall Picture
Refresh Computer Organization I
(TiTo)

12.9.2000 Copyright Teemu Kerola 2000 1

Computer System

- Data movement, storage, and processing
Figs 1.3, 1.4
- Control
Figs 1.5, 1.6 Figs 3.2, 3.3, 3.9
- System and I/O Buses
- Internal and external memories
- Input/Output systems
- Operating Systems support

12.9.2000 Copyright Teemu Kerola 2000 2

System & I/O Buses

- Bus configurations Fig 3.18
- Local (internal, memory) bus (sisäinen väylä)
 - inside CPU chip
 - connects CPU to cache
- System bus (systeemiväylä)
 - connects CPU to memory
- I/O bus (I/O väylä)
 - connects CPU & memory to I/O devices
- Implementation details later on

12.9.2000

Copyright Teemu Kerola 2000

3

Internal and External Memories

- Memory hierarchy (muistihierarkia)
Fig 4.1
 - Registers, L1 Cache, L2 Cache
 - Main memory, Disk cache
 - Disk, Optical, Tape
 - File server (local, via LAN)
 - Remote server (via WWW?)
- Storage capacity vs. access time (saantiaika)

Fig 4.3 [Stal96]

12.9.2000


Copyright Teemu Kerola 2000

4

Teemu's Cheesecake (5)

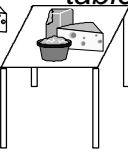
Register, on-chip cache, memory, disk, and tape speeds relative to times locating cheese for the cheese cake you are baking...

hand



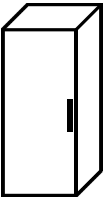
0.5 sec
(register)

table




1 sec
(cache)

refridgerator



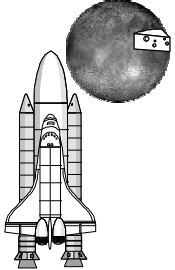
10 sec
(memory)

moon



12 days
(disk)

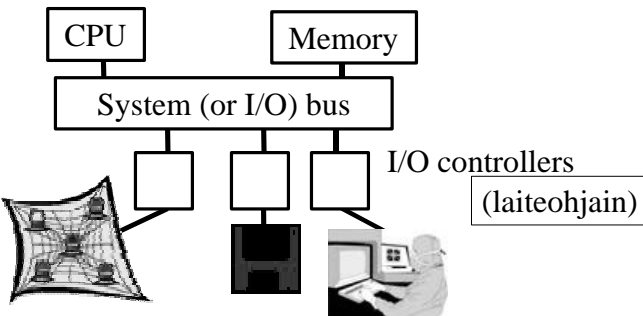
Europa (Jupiter)



4 years
(tape)

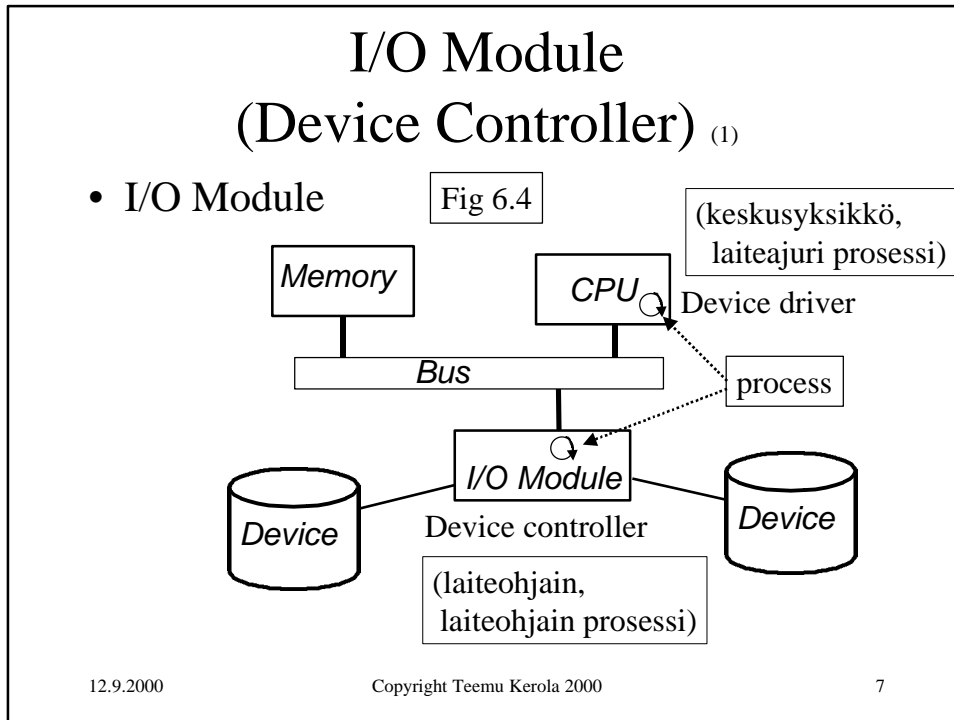
12.9.2000
Copyright Teemu Kerola 2000
5

Input/Output Systems (3)



- Three categories
 - I/O with people Video display, joy-stick, ...
 - I/O with machines CD, disk, ...
 - Communication Ethernet, token ring, ...

12.9.2000
Copyright Teemu Kerola 2000
6



Direct vs. Interrupt-driven I/O ⁽²⁾

- Direct, I.e., programmed I/O (suora I/O)
 - CPU controls I/O directly
 - CPU spins (waits) while I/O device works
 - I/O device transfers one word at a time

- Interrupt-driven I/O (keskeyttävä I/O)
 - CPU gives one I/O command, does a process switch, and continues with some other work
 - when I/O is done, I/O controller interrupts the CPU, and original process is made ready to run again

12.9.2000 Copyright Teemu Kerola 2000 8

Direct vs. Interrupt-driven I/O (contd) ⁽²⁾

- Direct Memory Access (DMA)
 - I/O controller can directly access memory
 - o/w access only to “data registers”
 - interrupt CPU only after (a big) block transfer
- I/O channels and I/O processors
 - I/O controller is smart
 - I/O controller manages complete I/O jobs
 - each with many DMA transfers?
 - many I/O jobs in queue at a time?

12.9.2000

Copyright Teemu Kerola 2000

9

Memory-Mapped I/O ⁽³⁾

(muistiinkuvattu I/O)

- Each device controlled via device registers
 - data, status, control
- Device registers are addressed similarly as memory
 - with normal read/write instructions
(vs. specific machine instructions for I/O)
 - device controller acts also as a memory card
- Device registers are physically located in the device controller which recognises certain memory addresses belonging to it

(laiterekisterit)

12.9.2000

Copyright Teemu Kerola 2000

10

SCSI - Small Computer System Interconnect ⁽³⁾

- Parallel data interface
 - 8,16, or 32 parallel data lines (wires)
 - 9 control lines
- Max 7 devices
- Arbitration
 - select who can use
 - the one with the highest priority wins
 - priority = SCSI id selected for the device

12.9.2000

Copyright Teemu Kerola 2000

11

Operating Systems Support

- User/computer interface (käyttöliittymä)
Fig 7.1
- Resource manager (resurssien hallinta)
Fig 7.2
- Process manager (prosessien hallinta)
Fig 7.8
(prosessin tilat)
- Process Control Block (PCB) (prosessin kontrollilohko)
Fig 7.9

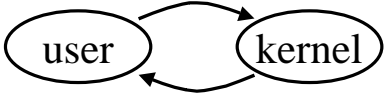
12.9.2000

Copyright Teemu Kerola 2000

12

Processor States

(suorittimen tilat)

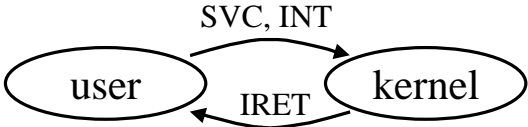


- **User mode (normal mode)** (käyttäjätila)
 - can use only non-privileged instructions
 - can access only memory in user-space

- **Kernel mode (privileged mode)** (etuoikeutettu tila)
 - can use all machine instructions, including privileged instructions (etuoikeutetut konekäskyt)
 - can access all memory, including kernel memory (KJ:n ytimen omat muistialueet)

12.9.2000 Copyright Teemu Kerola 2000 13

Changing Processor Mode

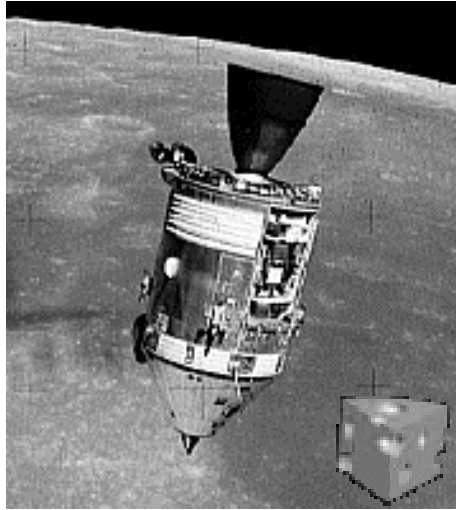


- **User mode → kernel mode**
 - interrupt or explicit SVC instruction
 - interrupt handler checks for rights to change mode (keskeytykskäsitteijä)

- **Kernel mode → user mode**
 - privileged machine instruction
 - return from interrupt (e.g., IRET)
 - returns control & restores previous mode

12.9.2000 Copyright Teemu Kerola 2000 14

-- End of Chapter 1-7: Intro --



12.9.2000

Copyright Teemu Kerola 2000

15