

Luento 9

Järjestelmän ulkoinen muisti


Muistihierarkia
Kiintolevyt
Muut pyörivät levyt
I/O:n toteutus

19.4.2004 Teemu Kerola, Copyright 2003 1

Muistihierarkia (4)

ks. Fig 4.1 [Stal03]
(ks. Fig 4.1 [Stal99])

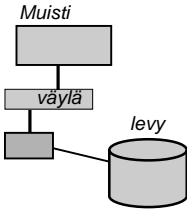
- Ulkoinen muisti on halvempaa toteuttaa per tavu
- Ulkoinen muisti on hyvin paljon hitaampaa kuin sisäinen muisti
- Aika/tila optimointi
 - suuret tietomäärät täytyy (kannattaa) kustannussyistä pitää ulkoisessa muistissa
 - pienet tietomäärät täytyy (kannattaa) tehokkuussyistä pitää sisäisessä muistissa
- Kaiken viitatus tiedot tulee suoritusaikana olla sisäisessä muistissa!



19.4.2004 Teemu Kerola, Copyright 2003 2

Virtuaalimuisti

- Osa muistihierarkiaa
- Vastaus ongelmaan
 - miten tehdä suoritusaikaisesta muistista ”yhtä suuri” kuin levymuisti ja ”yhtä nopea” kuin keskusmuisti?
- Kaksitasoinen:
 - keskusmuistissa kulloinkin käytössä olevat alueet
 - levyllä kaikki tiedot
 - kopiointi tarvittaessa



19.4.2004 Teemu Kerola, Copyright 2003 3

Virtuaalimuistin toteutus (4)

- Toteutustavat
 - kanta- ja rajarekisterit
 - sivutus
 - (segmentointi ja sivuttava segmentointi)
- Pääosa toteutuksesta ohjelmistotasolla
- Laitteistotuki
 - MMU – muistinhallintayksikkö
 - nopeuttaa viitatus muistipaikan todellisen osoitteen laskentaa
 - osoitetta ei tarvitse laskea usealla konekäskyllä, kun MMU tekee sen laitteistotasolla
 - rakenne ja toiminta vaihtelee virtuaalimuistin toteutustavan mukaan

Lisää tietoa? Tietokoneen järj.-rakenne Käyttöjärj.-I ja II

19.4.2004 Teemu Kerola, Copyright 2003 4

19.4.2004 Teemu Kerola, Copyright 2003 5

Tiedostojärjestelmä (5)

- KJ:n osa, hallitsee kaikkia tiedostoja
- Valvoo oikeuksia tiedostoa avattaessa
- Muuntaa tiedostonimet fyysisiksi osoitteiksi
- Ylläpitää taulukoita, joista näkee mitä kohtaa mistäkin tiedostosta kukin prosessi on käsittelemässä
- Tiedostojärjestelmä lukee ja kirjoittaa tiedostoja suurina kerralla käsiteltävinä lohkoina (0.5-8 KB?)
 - käyttäjätason prosessit käsittelevät tiedostoja tavuittain eikä niiden tarvitse tietää tiedoston todellista fyysistä rakennetta (KJ:n laiteajuri huolehtii siitä)

19.4.2004 Teemu Kerola, Copyright 2003 6

Levymuisti (2)

ks. Fig 6.5 [Stal03]
(ks. Fig 5.4 [Stal99])

- Levykkö
 - pyörii nopeasti (koko ajan?)
 - luku/kirjoituspäätt liikkuvat kaikki yntä aikaa?
 - monta levyä
- Levyypinta
 - 2 per levy (tai 1)
 - ura
 - sektori: pienin kerralla osoitettavissa oleva alue
 - sylinteri: päällekkäin olevat urat, luku/kirj. pää samalla kohtaa

150 MB-181GB/levykkö
~ 3600-10800 rpm
~ 1-16 levyä/levykkö
~ 2000-3000 uraa/pinta
~ 20-100 sektoria/ura
~ 0.5-8KB/sektori
~ 1-32 uraa/sylinteri

19.4.2004 Teemu Kerola, Copyright 2003 7

Levymuistin saantiaika (2)

- Tiedon osoite: levyypinta + ura + sektori
 - laiteajuri etsii KJ-taulukoista loogisen osoitteen perusteella
- Saantiaika:
 - hakumarren siirtoaika (seek time)
Esim: aver 6.3 ms, min-max 2-15 ms?
 - odota kunnes sektori kohdalla (rotational delay)
Esim: pyörähdysviive kun 3600 rpm: 8.33 ms (keskim. puolen kierroksen aika)
 - siirrä sektorin verran tietoa (data transfer time)
Esim: pyör.aika / sekt. lkm = 0.42 ms

ks. Fig 6.2 [Stal03]
(ks. Fig 5.3 [Stal99])

19.4.2004 Teemu Kerola, Copyright 2003 8

Tiedoston talletus levyille (2)

- Tiedosto koostuu useista lohkoista
 - lohko per sektori (lohko per usea sektori?)
- Levyn hakemistossa on tieto kunkin tiedoston käyttämistä lohkoista
 - luetaan lohkot annetussa järjestyksessä

19.4.2004 Teemu Kerola, Copyright 2003 9

Levyn käyttö

- Virtuaalimuistin tukimuistina
- Tiedostojen talletukseen
- Virtuaalimuistin voi toteuttaa tiedostojärjestelmän päälle (sitä käyttäen), tai päinvastoin!

19.4.2004 Teemu Kerola, Copyright 2003 10

DOS-levykkeen rakenne (2)

- FAT – File Allocation Table
 - kertoo, mitkä sektorit ovat vapaana
 - kertoo, mitkä sektorit ovat käytössä millekin tiedostolle
 - kiinteä paikka levykkeellä, 2 kopiota
- Hakemisto
 - erikoistyyppinen tiedosto
 - sisältää hakemistoalkion joka tiedostolle
 - nimi, tyyppi, koko, muutos pvm ja kellonaika
 - attribuutit (invisible, read-only, ...)
 - linkki ensimmäiseen sektoriin (FAT ja itse tiedosto)

19.4.2004 Teemu Kerola, Copyright 2003 11

FAT - File Allocation Table (DOS)

Eksa	E	10 ¹⁸
Peta	P	10 ¹⁵
Tera	T	10 ¹²
Giga	G	10 ⁹
Mega	M	10 ⁶
Kilo	k	10 ³

DOS levykkeet: 1.44MB, lohko 512 B, 2.9k lohkoa
entä: 1 GB, lohko 64 kB, 64k lohkoa OK?

19.4.2004 Teemu Kerola, Copyright 2003 12

Mikä on hyvä levylohkon koko?

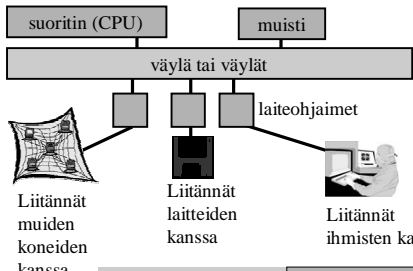
- Mihin käyttöön?
- Videokuvan talletus/playback?
 - 1 minuutti
 - 12.3 MB
- lohko 4kB? osoitetaulu?
- epäsuorat viittaukset?
- riittääkö levyn/väylän/ohjaimien nopeus?
- entä jos 1 tunti? riittääkö levyn kapasiteetti?



19.4.2004 Teemu Kerola, Copyright 2003 13

19.4.2004 Teemu Kerola, Copyright 2003 14

Laitteiden liittäminen järjestelmään

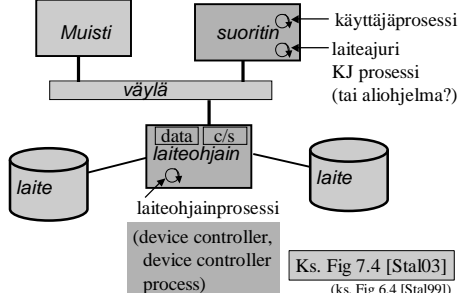


Liitännät muiden koneiden kanssa
Liitännät laitteiden kanssa
Liitännät ihmisten kanssa

Kaikki samalla tavalla: ks. Fig. 2-29 [Tane99]

19.4.2004 Teemu Kerola, Copyright 2003 15

Laiteohjain (I/O Moduuli)



Muisti suoritin
väylä
laite
laite
laiteohjain (device controller, device controller process)

käyttäjäprosessi
laiteajuri
KJ prosessi (tai aliohjelma?)

Ks. Fig 7.4 [Stal03]
(ks. Fig 6.4 [Stal99])

19.4.2004 Teemu Kerola, Copyright 2003 16

Laitteiden käytön toteutus ⁽⁵⁾

ks. laiteohjainkuva (ed. kalvo)

- Käyttäjöohjelma kutsuu käyttöjärjestelmän laiteajuria tekemään I/O:n. Laiteajuri suoritetaan samalla suorittimella kuin käyttäjöohjelmakin.
- Laiteajuri ohjaa laitteen toimintaa laitteen laiteohjaimella olevien kontrollirekisterien (muistialue "c") avulla
- Laiteajuri voi lukea laitteen tilatietoa laiteohjaimella olevien statusrekisterien (muistialue "s") avulla
- Laiteajuri voi lukea (kirjoittaa) laitteen lukemaa (laitteelle kirjoitettavaa) tietoa laiteohjaimella olevien datarekistereiden (muistialue "data") avulla
- Kontrolli-, status- ja datarekisteri kolmikko muodostaa "I/O portin" suorittimen näkökulmasta

19.4.2004 Teemu Kerola, Copyright 2003 17

Laiteohjaimen rekistereihin viittaaminen ⁽⁵⁾

- Ongelma: miten suorittimella suorittavan laiteajuri viittaa eri kortilla oleviin rekistereihin?
- Ratkaisu 1: omat I/O-konekäskyt tätä tarkoitusta varten
 - käskyssä annetaan laiteohjaimen identifikaatio ja laiterekisterin nro (oma I/O osoitevaraus)
 - vaikea laajentaa käyttöä uusiin laitteisiin, joilla "laiterekisterit" voivat olla hyvinkin erilaisia
 - suorittimen konekäskyjä ei voi muuttaa

x86: IN, OUT KOKSI: IN, OUT
INS, OUTs port_out [Tane87]

ks. Minix esimerkin port_out [Tane87]

19.4.2004 Teemu Kerola, Copyright 2003 18

Ratkaisu 2: muistiinkuvattu I/O ⁽⁵⁾

ks. laiteohjainkuva

- Laiteajuri lukee/kirjoittaa laiteohjaimella olevia rekistereitä (data, status/kontrolli) tavallisilla muistin luku/kirjoitus käskyillä
 - ei tarvita erillisiä I/O-konekäskyjä! `load R1,=DiskRd`
`store R2, DiskCtr`
 - laiteohjaimella olevat "laiterekisterit" ovat samanlaista viitattavaa muistia kuin "normaali muisti"
 - muistisoitteen ensimmäiset bitit (ei siis käskykoodi) valitsevat, mille laitteelle (vai tavallisen muistiin) viittaus kohdistuu
 - voidaan käyttää rinnan I/O `DiskCtr EQU 0x80000001` (laiterekistereihin voi siis viitata sekä I/O-käskyillä että muistiinkuvattun I/O:n avulla)

esim. Intelin arkkitehtuurit

19.4.2004 Teemu Kerola, Copyright 2003 19

I/O tyypit ⁽²⁾

ks. laiteohjainkuva

- Suora I/O:** laiteajuri odottaa tiukassa silmukassa, kunnes laiteohjaimen statusrekisteri ilmoittaa I/O-pyyntön valmistuneen (direct I/O)
 - laiteajuri siirtää tietoa muistin ja datarekisterin välillä
- Epäsuora I/O:** I/O:n odotusaikana suorittimella suoritetaan jotain muuta ohjelmaa (indirect I/O interrupt driven I/O)
 - Kun I/O-pyyntö valmistuu, laite keskeytyksen (laitekeskeytyks, I/O interrupt) suorittimelle, joka (jonkin ajan kuluttua) jatkaa kesken jäänyttä I/O-pyyntöä esittänyttä ohjelmaa.
 - laiteajuri siirtää tietoa muistin ja datarekisterin välillä

19.4.2004 Teemu Kerola, Copyright 2003 20

I/O tyypit (jatkoa) ⁽⁴⁾

ks. laiteohjainkuva

- DMA - Direct Memory Access**
 - älykkäämpi laiteohjain
 - laiteohjain voi suoraan kopioida tiedot keskusmuistiin
 - laiteajurin ei tarvitse laiterekistereitä käyttäen siirtää tietoa muistin ja datarekisterin välillä
 - laiteohjain tekee paljon suuremman määrän työtä itsenäisesti (kuin epäsuorassa I/O:ssa) ennen suorittimelle annettavaa laitekeskeytystä

19.4.2004 Teemu Kerola, Copyright 2003 21

Tiedostopalvelin

- (Lähi)verkossa oleva palvelin orig. tiedosto
- Käytettäessä tiedoston (osien) kopio on muistissa (ja ehkä myös paikallisella levyllä) tiedoston kopio?

19.4.2004 Teemu Kerola, Copyright 2003 22

Tiedostopalvelin ⁽⁴⁾

- Käytetään kuten paikallista levyä oman systeemin KJ:n liitospalikan avulla
- Paljon hitaampi kuin paikallinen levy
- Tiedostovälimuistit (muistipuskurit tai levypuskurit) nopea (file cache) toimintaa käytännössä
 - omassa järjestelmässä
 - palvelimella
 - 50% oman järjestelmän keskusmuistista voi olla varattu tiedostovälimuistille
 - tiedon päivitys tiedoston kirjoituksen yhteydessä?

Lisää tietoa? → KJ kurssit

19.4.2004 Teemu Kerola, Copyright 2003 23

19.4.2004 Teemu Kerola, Copyright 2003 24

Esimerkki: kirjoittimen laiteajuri ttk-91 koneelle ⁽⁵⁾

- Laitteella voi tulostaa kokonaislukuja yksi kerrallaan
- Muistiinkuvattu I/O, suora I/O
- Laiteportti
 - kontrollirekisteri muistipaikka 1048576 = 0x80000
 - tilarekisteri muistipaikka 1048577 = 0x80001
 - datarekisteri muistipaikka 1048578 = 0x80002
- Laiteajuri Print toimii etuoikeutetussa tilassa
- Kutsu:


```
PUSH SP, =0 ; space for return value
PUSH SP, X ; parameter to print
SVC SP, =Print ; returns Success/Failure
POP SP, R1
JNZER R1, TakeCareOfTrouble
```

19.4.2004

Teemu Kerola, Copyright 2003

25

Esim: laiteajurin toteutus ⁽¹²⁾

ks. laiteohjainkuva

```
ptrCtrl DC 1048576 ; control register address
ptrStat DC 1048577 ; status register address
ptrData DC 1048578 ; Print PUSH R1, ptrData ; save regs
retVal EQU -3 LOAD R1, ptrData(FP)
parData EQU -2 STORE R1, @ptrData ; data to print

Oleta: SVC:n ja IRET:n
toteutus samalla tavalla
kuin CALL ja EXIT
LOAD R1, =0
STORE R1, @ptrStat ; init (clear) state register
LOAD R1, =1
STORE R1, @ptrCtrl ; give command to print

Wait LOAD R1, @PtrStat ; check state register
JNZER R1, Done
JUMP Wait ; wait until I/O done

Done LOAD R1, =0 ; return "Success"
STORE R1, retVal(FP)
POPR SP ; recover regs
IRET SP, =1
```

ptrCtrl → 1
ptrStat → 1
ptrData → 200

See: driver.k91

19.4.2004

Teemu Kerola, Copyright 2003

26

Erilaisia levyjä

- Kiintolevy
- ZIP levyke
- Levyke
- CD-ROM, CD-R, CD-RW
- DVD
- ...?...

19.4.2004

Teemu Kerola, Copyright 2003

27

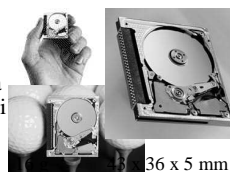
19.4.2004

Teemu Kerola, Copyright 2003

28

Kiintolevy

- Kiinteä, ei vaihdettavissa oleva levy (hermeettisesti suljettu kotelo)
- Tila: 0.150-181 GB (v. 2000)
- Hakuaika: 5-15 ms
- 1-10 levyä
- Pyörimisnopeus: 4500-10800 rpm
- Siirtonopeus: 5-50 MB/sec



36 x 5 mm



19.4.2004

Teemu Kerola, Copyright 2003

29

Zip & Jaz levykeasemat

- Vaihdettava levyke
- Tila: 0.1-2 GB
- Hakuaika: 10-30 ms
- Pyörimisnopeus: 3000-5400 rpm
- Siirtonopeus: 1-6 MB/sec



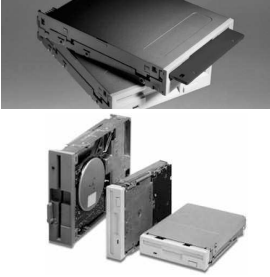
19.4.2004

Teemu Kerola, Copyright 2003

30

Levykeasema


- Vaihdeettava levyke
- Tila: 1.44 MB
- Haku aika 90 ms
- Pyörimisnopeus 300 rpm
- Siirtonopeus 0.05 MB/sec



19.4.2004 Teemu Kerola, Copyright 2003 31

CD - Compact Disc

- Vaihdeettava levyke
- CD-R (Recordable)
- CD-RW (Rewritable)
- Yksi pitkä spiraalimainen "ura"
- Tila: 650 MB
- Haku aika 90 ms
- Pyörimisnopeus 200-9000 rpm
- Siirtonopeus 0.1-2 MB/sec



19.4.2004 Teemu Kerola, Copyright 2003 32

DVD - Digital Versatile Disk

- Vaihdeettava levyke
- DVD-ROM
- DVD-R (Recordable)
- DVD-RAM
 - kuten tavallinen kovalevy
- Tila: 4.7-17 GB
- Haku aika 100-180 ms
- Pyörimisnopeus 2000-8000 rpm
- Siirtonopeus 2-8 MB/sec
 - hitaampi kuin kovalevy



19.4.2004 Teemu Kerola, Copyright 2003 33

