

LUENTO 1

## 582497 Käyttöjärjestelmät, 8 op

Tiina Niklander

Helsingin yliopisto  
Tietojenkäsittelytieteen laitos

## Kurssin asema

- *Hajautettujen järjestelmien ja tietoliikenteen erikoistumislinjan* maisteriopintojen pakollinen kurssi
- Esitietoina edellytetään:
  - Kandidaatin tutkinnon opinnot, ainakin
    - Tietokoneen toiminta
    - Rinnakkaisohjelmointi
- Edeltäjiä:
  - Tietokone ja käyttöjärjestelmä
  - Käyttöjärjestelmät I ja II

## Siirtymävaiheen vastaavuudet

2004 tutk. vaatimukset	2005 tutk. vaatimukset
Tito (2 ov)	Tito (4 op)
KJ-I (2 ov)	KJ (8 op)
KJ-II (2 ov)	
HJ (3 ov)	HJ (4 op)

## Kurssin tavoite

- Ymmärtää KJ:n perusrakenteet ja toteutusperiaatteet sekä käyttäjän että toteuttajan näkökulmasta
- Tuntee yleisperiaatteiden lisäksi myös vaihtoehtoisia algoritmeja ja toteutuksia
- Esimerkeinä  
Unix = Linux, Windows XP ja Solaris
- Kurssin jälkeen:
  - Osaa kuvata käyttöjärjestelmän osat, niiden toiminnan ja vuorovaikutuksen
  - Pystyy tarvittaessa vaikka lähdekoodin avulla selvittämään miten tietty käyttöjärjestelmä toteuttaa jonkin tietyn toiminnon

## Kurssikirja

- Kurssikirja
  - [Stal05] William Stallings, *Operating Systems, 5th. ed.*, Prentice-Hall, 2005
- Oheislukemistoa
  - [Tane01] A.S. Tanenbaum, *Modern Operating Systems, 2nd. ed.*, Prentice-Hall, 2001.
  - [DDC04] H.M. Deitel, P.J. Deitel, D.R. Choffnes, *Operating Systems, 3rd ed.*, Prentice-Hall 2004
  - Mikä tahansa kirja, jonka otsikossa maagiset avainsanat *Operating Systems*

## Kurssin rakenne:

- Osio 1: Prosessit ja säikeet
  - Myös käyttöjärjestelmän rakenne ja tehtävät
- Osio 2: Muistinhallinta
  - Erityisesti virtuaalimuisti
  - Myös vuorottaminen
- Osio 3: Siirräntä ja tiedostojärjestelmät
- Osio 4: Hajautus ja tietoturva

## Kurssin suorittaminen

### Tapa 1: luentokurssi

- Ilmoittaudu kurssin harjoitusryhmään
- Seuraa luentoja
- Tee viikoittaiset harjoitukset
  - osallistu tavallisiin laskareihin
- Osallistu opintopiiriin ja tee ryhmätöitä
- Osallistu kurssikuulusteluihin
- Kerää riittävästi pisteitä harjoituksista, ryhmätöistä ja kokeesta

## Kurssin suorittaminen

### Tapa 2: erilliskuulustelu

- Ilmoittaudu ja osallistu erilliskokeeseen
- Erilliskokeen koealue määriteltä kurssikuvauksessa
- Koe max 60 pistettä

*Kurssin läpäisyyn tarvitset vähintään 30 pistettä*

*HUOM: Vasta keväällä 2007*

## Ryhmätöitä (opintopiirit)

- Kolmeen ensimmäiseen osioon liittyy ryhmätöitä
- Näistä saa pisteitä 1-4
- Ryhmissä on 2-4 opiskelijaa
- Ryhmät sovitaan harjoituksissa, niistä voi ilmoittaa myös sähköpostitse luennoijalle
- Ryhmätöitä voivat olla
  - Raportin kirjoittamista (selvitä miten jokin toimii tai on toteutettu)
  - Ohjelmointia (toteuta jokin toiminnallisuus)
  - Jotain muuta (simulaattori, yms.)

## WWW Informaatio

- Kurssin kotisivu  
<http://www.cs.helsinki.fi/u/niklande/opetus/kj/>
- Uutisryhmä [hy.opiskelu.tkl.kj](mailto:hy.opiskelu.tkl.kj)
- Kertauskysymyksiä (vanhalta KJ-II kurssilta)  
<https://www.cs.helsinki.fi/i/kerola/kj2/kertaus/pract.html>
- CSMoodle <http://moodle.cs.helsinki.fi>

LUENTO 1

## Yleistä käyttöjärjestelmistä ja laitteistokertaus

## Sisältö

- Käyttöjärjestelmän palvelut
- Käyttöjärjestelmän keskeisiä osa-alueita
  - Prosessien hallinta
  - Resurssien hallinta
  - Muistin hallinta
  - Tiedostojärjestelmä
  - Siirräntäjärjestelmä (I/O)
- Laitteistosta

## Tietokonejärjestelmä (=laitteisto + ohjelmisto)

- Sovellus saa laitteiston käyttöönsä KJ:n avustuksella
  - CPU ja muisti
  - oheislaitteet
- KJ tarjoaa laitteiston käytössä tarvittavat 'palvelunsa' sovellukselle
  - prosessien hallinta
  - muistinhallinta
  - siirräntäjärjestelmä
  - tiedostojärjestelmä
- KJ on tärkein ja yleensä monimutkaisin laitteiston suorittamista ohjelmistoista

## Kerrosmalli

- Peruskäyttäjä näkee vain sovellukset
  - Niiden alla on varusohjelmia ja käyttöjärjestelmä
- Sovellusohjelmoija näkee ja käyttää palvelurajapintaa (esim. funktiokirjastona)
- Käyttöjärjestelmä hallinnoi laitteistoja sovellusten puolesta

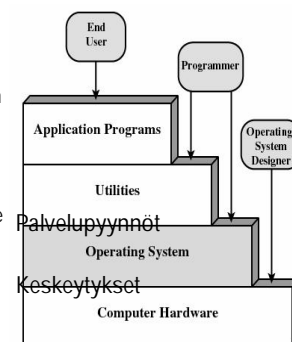


Figure 2.1 Layers and Views of a Computer System

## KJ:n tehtäviä ja palveluja

- KJ on ohjelmisto, joka huolehtii sovellusten suorittamisesta prosessorissa
  - antaa laitteiston sovelluksen käyttöön
  - huolehtii, että sovellus ei yksin valloita koko laitteistoa
  - tarjoaa turvallisen suoritusympäristön
- KJ on palveluliittymä sovelluksen ja laitteiston välillä
  - laitteistopiirteiden hallinta sovelluksen ulkopuolella
  - sovellus esittää tarpeensa palvelupyynnöin
    - siirtyminen KJ:n palveluun keskeytysmekanismin kautta
- KJ optimoi laitteiston käyttöä hyvän suorituskyvyn saamiseksi

## KJ:n tarjoamia palveluja (1)

- Käyttöliittymä (user interface)
  - Ikkunointiympäristö, komentotulkki (shell)
- Ohjelmointiympäristö (utility programs)
  - editorit, kääntäjät, linkittäjät, debuggerit, ...
- Ohjelmien suorittaminen
  - prosessien käynnistäminen
    - Kirjanpito, lataaminen muistiin ...
  - prosessien vuorottaminen
    - CPU:n käyttövuorot, resurssien varaus ja käyttö
  - prosessien tappaminen
    - resurssien vapauttaminen

## KJ:n tarjoamia palveluja (2)

- Muistinhallinta
  - varaa muistitilaa sovellukselle
- Siirräntä
  - sovellus pyytää palvelua KJ:ltä
  - KJ 'komentaa' I/O-ohjaimia töihin
- Tiedostojen hallinta
  - kirjanpito tdstoista, hakemistot
  - tdstosta lukeminen / tallettaminen
- Suojaus
  - muistin suojaus ja tdstojen käyttöoikeudet
  - resurssien varaamisen kilpailutilanteet ja ristiriidat

## KJ:n tarjoamia palveluja (3)

- Virhetilanteiden hallinta
  - laitteistovirheet
  - ohjelmistovirheet
  - resurssipula
- Virheistä toipuminen
  - palauttaa status tietoa sovellukselle
  - uudelleenyhtymiset
  - prosessin tappaminen

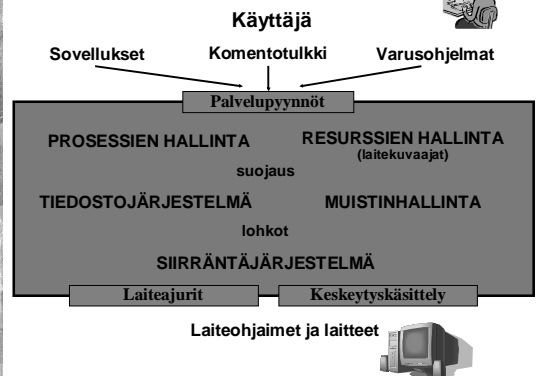
**Väläkysymys: Mitä yleistermi resurssi tarkoittaa?**

## KJ:n tarjoamia palveluja <sup>(4)</sup>

- Kirjanpito / tilinpito (accounting)
  - tilastointi resurssien käytöstä
  - suorituskyvyn seuranta (esim. vastausaika)
  - järjestelmäparametrien optimointi hyvän suorituskyvyn saamiseksi
  - koneen käyttäjien laskuttaminen

Kirjanpito voi antaa vihjeitä siitä, kuinka KJ:tä voitaisiin kehittää edelleen

## Keskeiset KJ:n osa-alueet



## Keskeiset KJ:n osa-alueet

- Keskeisimmät osa-alueet (kirjan jaottelu)
  - 1) Prosessin käsite
  - 2) Muistinhallinta
  - 3) Tietoturva ja suojaukset
  - 4) Vuorottaminen ja resurssien hallinta
  - 5) Järjestelmän hierarkkinen rakenne

virt. edellinen kuva

## 1) Prosessi

= Suoritettavaksi otettu ohjelma

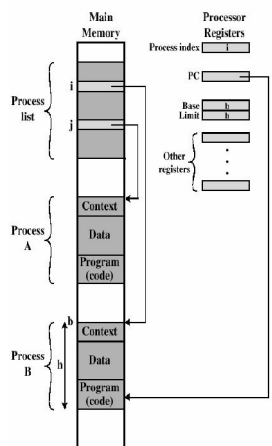
- koodi muistissa (voi olla yhteiskäytössä)
- oma data-alue ja pino muistissa (muuttujat)
- kaikki ei välttämättä yhtäaikaan muistissa

+ KJ:n ylläpitämät hallinnolliset rakenteet

- prosessin kuvaaja PCB (Process Control Block)
  - CPU:n dataa prosessin suorittamisesta
    - prosessorin rekistereiden arvot
  - KJ:n dataa prosessin hallitsemiseksi
    - tunnistus, omistaja, prioriteetti ..
    - tietoja prosessin varaamasta muistista, tdsista, ...
    - prosessin tila (esim. odottaa siirännän valmistumista)
    - ...

## Eräs toteutus

- KJ:llä prosessilista, jossa viitteet prosessin kuvaajiin (**context**)
- CPU:ssa rekisteri, jossa suoritettavan prosessin numero
- Prosessinvaihto: CPU A:lta B:lle
  - KJ talletti rekistereiden arvot A:n kuvaajaan
  - KJ larsi B:n kuvaajasta arvot rekistereihin



## 2) Muistinhallinta

- Suoritusaikainen tallennus
  - prosessit (ohjelmat+data) keskusmuistissa (primary memory)
- Pysyvä tallennus
  - tiedostot (ohjelmat+data) 'tuki muistissa' (secondary memory)
- Tilan allokointi
  - KJ huolehtii automaattisesti
  - kirjanpito vapaista muistialueista / levylohkoista
  - kirjanpito varatuista muistialueista / levylohkoista
  - tarvittaessa KJ käyttää levyä muistin jatkeena (virtuaalimuisti)
- Suojaus ja käyttöoikeudet
  - prosessien eristäminen toisistaan
  - silti tuettava modulaarista ohjelmointia
    - koodin / datan yhteiskäyttö sallittua
  - muistinsuojaus, tdsien käyttöoikeudet

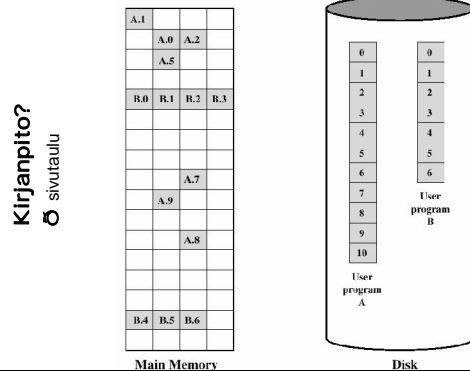
## Muistinhallinta

### Avainkäsite virtuaalimuisti

- Ei ota kantaa todellisen muistin määrään tai minne ohjelma muistissa sijoitetaan
  - ohjelman muistiavaruus voi olla suurempi kuin fyysinen muistiavaruus (ts. todellinen muistin määrä)
- Suoritusaikana muistissa tarvittava osa koodista/datasta, loput levyllä
  - MMU huomaa puuttumisen
  - KJ lataa muistiin
- Ei näy sovelluksen ohjelmoijalle

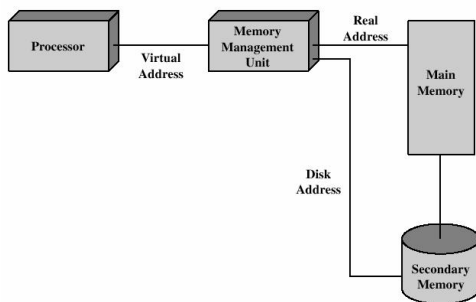
## Virtuaalimuisti

Kuva 2.9



## Virtuaalimuisti

Kuva 2.10



## Virtuaalimuisti

- Ohjelma käyttää virtuaaliosoitteita, ts. osoitteita ohjelman alun suhteen
- Laitteisto käyttää aina fyysisiä (todellisia) osoitteita
- MMU tekee osoitemuunnoksen aina suoritusaikana muistiin viitattaessa
- Jos viitattu osa ei ole keskusmuistissa
  - MMU aiheuttaa keskeytyksen
  - jos muistissa ei ole vapaata tilaa, KJ vapauttaa jonkin alueen
  - KJ hakee levyllä puuttuvan osan muistiin

## 3) Tietoturva ja suojaus

- Käyttäjien tunnistus
  - Vain tunnuksen & salasanan tuntevat käyttäjät saavat ottaa istunnon koneeseen
  - Prosessilla aina omistaja
- Resurssien käyttöoikeus
  - Prosessi käyttää resurssia vain omistajan luvalla
    - tdstoihin liittyy omistaja ja käyttöoikeudet
    - vain omistaja voi muuttaa käyttöoikeuksia
  - Ohjelmat ja data suojattava toisilta ohjelmilta
    - erityisen tärkeää on suojata KJ sovelluksilta
    - MMU ja ajaon aikainen osoitemuunnos
  - Resurssien yhteiskäyttö silti sallittava

## 4) Resurssien hallinta

- Resurssi?
  - CPU, muisti, tdsto, I/O-laite ...
  - CPU:n allokointi = vuorottaminen (skedulointi)
- Milloin?
  - Prosessia käynnistettäessä, suoritettaessa, tapettaessa
  - KJ:n päätöksillä, prosessin (palvelu)pyyntöjen perusteella
- Vastausaika
  - Interaktiivisuus vs. tausta-ajo (eräajo)
  - KJ:n palveluprosessi vs. sovellus
- Tasapuolisuus
  - Samanlaisille prosesseille samantasoinen palvelu
- Tehokkuus
  - Maksimoi läpimenoaste, minimoi vastausajat
  - Palvele mahd. useaa käyttäjää (sovellusta)

## Vuorottamisen perusideoita

- Suoritukseen otetuille prosesseille
- **READY-jono (short term queue)**
  - vuorottaja valitsee seuraavaksi suoritettavan prosessin (jonon ensimmäinen)
  - Round-Robin: uusi työ jonon loppuun ja CPU:lla pois tuleva työ jonon loppuun
- Suoritettavaksi ottamista odottaville prosesseille oma jono (long term queue)
  - ei liian monta prosessia yhtäaikaa READY-jonoon (moniajoaste)
- Kullakin tapahtumalla omat odotusjononsa
  - I/O-laitteet, semaforit, ajastimet, ...

## Milloin?

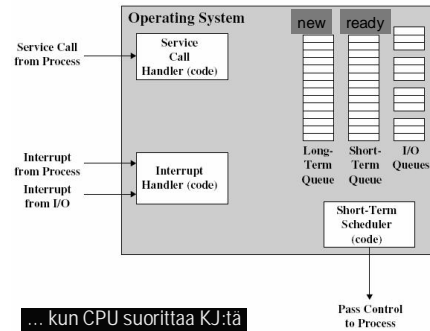


Figure 2.11 Key Elements of an Operating System for Multiprogramming

## 5) Hierarkkinen rakenne

- KJ muodostuu useista erillisistä tasoista
- Hierarkian ansiosta kokonaisuus jakaantuu helpommin hallittaviin osiin
  - suunnittelu, toteutus, testaus
- Kullakin tasolla oma osajoukko tehtävistä
  - ylemmillä tasoilla kehittyneimmät KJ:n palvelut
  - alemmilla tasoilla laiteläheisimmät toiminnot
- Taso tarjoaa palvelunsa ylemmälle tasolle, käyttää alemman/alempien tasojen palveluja
- Rajapinnat hyvin määriteltyjä
  - tason toteutusta voi muuttaa koskematta muihin tasoihin

## Laitetasot

Brown, Denning 1984

Level	Name	Objects	Example Operations
4	Interrupts	Interrupt-handling programs	Invoke, mask, unmask, retry
3	Procedures	Procedures, call stack	Mark stack, call, return
2	Instruction Set	Evaluation stack, micro-program interpreter, scalar and array data	Load, store, add, subtract, branch
1	Electronic circuits	Registers, gates, buses, etc.	Clear, transfer, activate, complement

Mitä laitteistopiirteitä tarvitaan KJ:n toteutuksessa?

## KJ:n laiteriippuvat tasot

Level	Name	Objects	Example Operations
7	Virtual Memory	Segments, pages	Read, write, fetch
6	Local secondary store	Blocks of data, device channels	Read, write, allocate, free
5	Primitive processes	Primitive process, semaphores, ready list	Suspend, resume, wait, signal P / V

Muistinhallinta: MMU:n rakenteen huomiointi  
 Laitteajurit: ohjaimien ja esim. levyn rakenteen huomiointi  
 Vuorottaja: rekistereiden talletus/palautus, synkronointiprimitiivit

## KJ:n laiteriippumattomat palvelut

Level	Name	Objects	Example Operations
13	Shell	User programming environment	Statements in shell language
12	User processes	User processes	Quit, kill, suspend, resume
11	Directories	Directories	Create, destroy, attach, detach, search, list
10	Devices	External devices, such as printer, displays and keyboards	Create, destroy, open, close, read, write
9	File system	Files	Create, destroy, open, close, read, write
8	Communications	Pipes	Create, destroy, open, close, read, write

## KJ-toteutuksen vaikeat kohdat

- Toimintojen synkronointi / ajoitus
  - joskus odotettava että jotain muuta tapahtuu ennenkuin voi jatkaa
  - keskeytysten oikea priorisointi
  - laitteistosiignaalit tai ohjelmien toisilleen välittämät tiedot eivät saa kadota tai kahdentua
- Poissulkeminen (Mutual exclusion)
  - eräitä resursseja voi käyttää vain yksi kerrallaan
    - esim. kirjoitin tai yhteinen tdsto / data
- Lukkiutuminen / Nälkiintyminen (Deadlock, Starvation)
  - vaikeaa havaita
  - odotettava resursseja, jotka ovat toisten hallussa
  - huono prioriteetti, ei saada ensinkään palvelua

kertaa RiO

## Poissulkemisongelma

- Mutex määrittely
- Ohjelmistoratkaisu
- Laitetuki ongelman ratkaisuun
- Lukko vai semafori? Milloin?
- Monitorit ja niiden toteutus
- Lukijat ja kirjoittajat (synkronointi)
- Miten tiedät, että ratkaisusi on oikein?

## Lukkiutumisongelma

- Lukkiutumisen havaitseminen
- Lukkiutumisen välttäminen
- Lukkiutumisen purkaminen
- Aterioivat filosofit

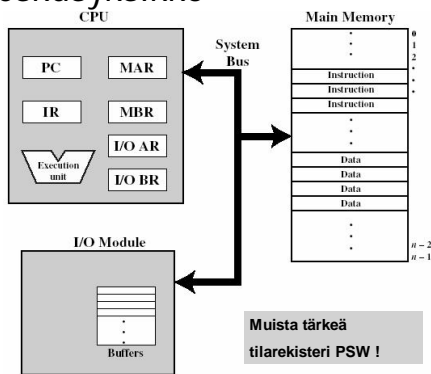
Samanaikaisuuden hallinta  
Ch 5, 6.1-6 [Stal05]

## Laitteistosta

Keskusyksikkö, Käskeysykli  
Keskeytys ja sen käsittely, Siirräntä  
Muistihierarkia  
Stallings, Luku 1  
= Tietokoneen Toiminta kurssin kertausta

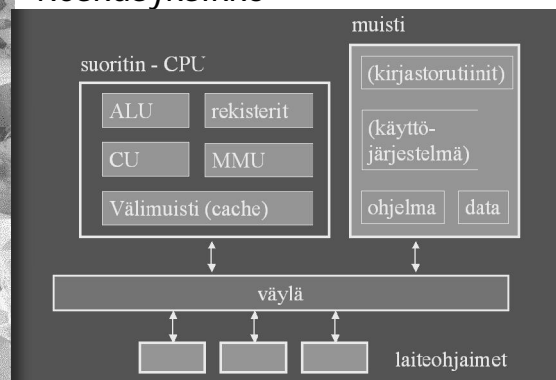
## Keskusyksikkö

Fig 1.1 [Stal05]



## Keskusyksikkö

ks. Tietokoneen Toiminta -kurssi



## Keskusyksikön osat

- Prosessori (CPU)
  - käskyjen suoritus
- (Keskus)muisti
  - tilaa ohjelmille ja niiden datalle (myös KJ on ohjelmat!)
- I/O ohjaimet (I/O kanavat, I/O prosessorit...)
  - laitteistoa, joka siirtää tietoa muistin ja oheislaitteiden välillä
    - syöttö- ja tulostuslaitteet
    - tallennuslaitteet
    - tietoliikennelaitteet
  - kommunikointi rekistereitä (I/O portteja) käyttäen
  - voivat sisältää omaa muistia (puskureita)
- Väylät
  - tiedonsiirto eri osien välillä

## Muistinhallintayksikkö MMU

### Memory Management Unit

- Muuttaa ohjelman sisäisen osoitteen fyysiseksi muistiosoitteeksi
  - ohjelma käyttää siirtymiä alkunsa suhteen
  - laitteisto käyttää fyysisiä osoitteita
- MAR, Memory Address Register
  - muistiosoitetta varten
- MBR, Memory Buffer Register
  - Muistista noudettavan muistipaikan sisältö
  - Muistiin kirjoitettava arvo

## Muistinhallintayksikkö MMU

- Kanta- ja rajarekisteriä käyttävä järjestelmä:
  - Base Register: ohjelman fyysinen alkuosoite
  - Limit Register: viimeinen kelvollinen fyysinen osoite tai viitattavissa olevan alueen pituus
- Virtuaalimuistijärjestelmä (esim.):
  - PTR, Page Table Register
    - Suoritettavan prosessin sivutaulun fyys. alkuosoite
  - TLB, Translation Lookaside Buffer
    - Viimeisimmissä osoitemuunnoksissa käytettyjä tietoja

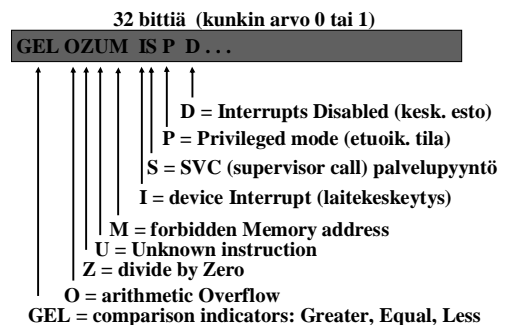
## Prossessorin rekistereitä

- Ohjaus- ja tilarekisterit (Control & Status Registers)
  - osa vain CPU:n sisäiseen käyttöön
  - osa vain KJ:n käyttöön
  - osa epäsuorasti ohjelmien viitattavissa
    - hyppykäsky muuttaa PC:n arvoa
    - vertailu asettaa tilarekisterin
    - ehdollinen hyppykäsky tutkii tilarekisteriä
- Yleiskäyttöiset rekisterit (User-visible registers)
  - ohjelmien käytettävissä (myös KJ:n!)
  - viitattavissa nimellä konekielen tasolla
  - datan ja osoitteiden tallettamiseen

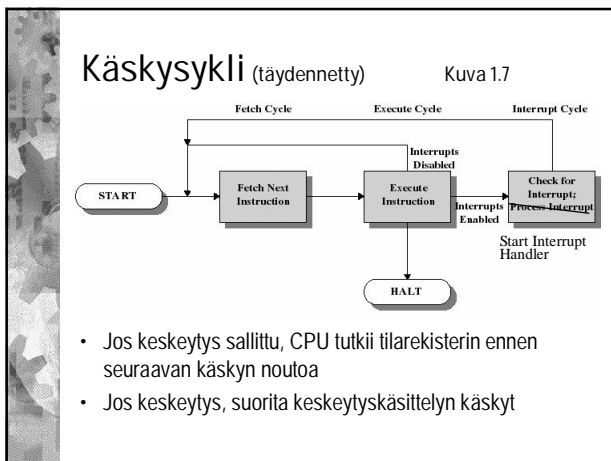
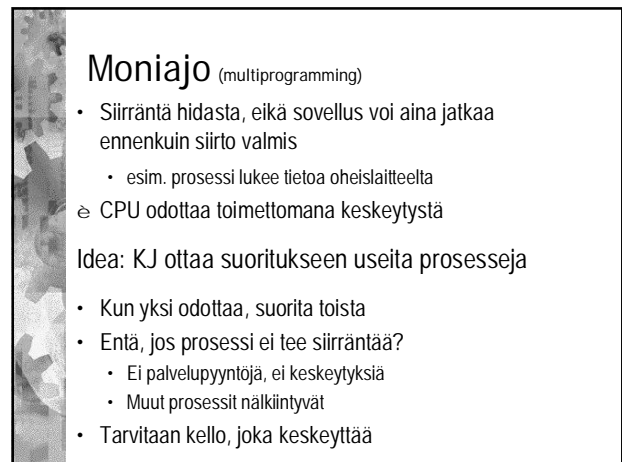
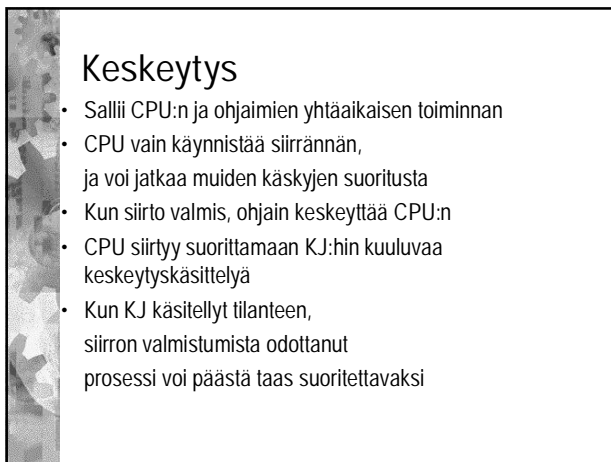
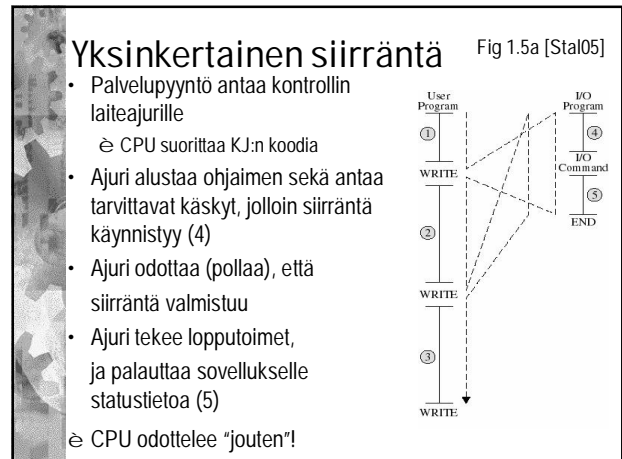
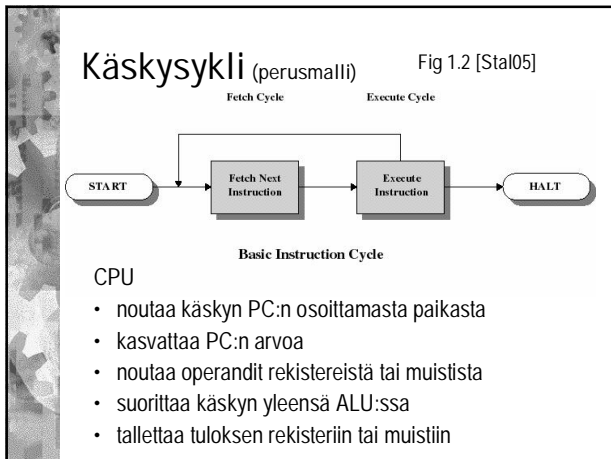
## Ohjaus- ja tilarekistereitä

- Käskyosoitin PC (Program Counter)
  - seuraavaksi suoritettavan käskyn virtuaaliosoite
- Käskyrekisteri IR (Instruction Register)
  - suoritettavaksi noudettu käsky
- Tilarekisteri PSW (Program Status Word)
  - Tietoa laitteiston tilasta ja toiminnan siihen aiheuttamista muutoksista
- lipukkeet (flags)
  - vertailujen tuloksille, virhetilanteille
  - keskeytyksille, keskeytyksien esto / salliminen
  - etuoikeutettu tila / käyttäjätila

## Esimerkki Tilarekisteristä







### Keskeytyks

Table 1.1    Classes of Interrupts

<b>Program</b>	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.
<b>Timer</b>	Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.
<b>I/O</b>	Generated by an I/O controller, to signal normal completion of an operation or to signal a variety of error conditions.
<b>Hardware failure</b>	Generated by a failure, such as power failure or memory parity error.

### Siirräntä ja keskeytys

- Palvelupyynnö siirtää kontrollin KJ:n laiteajurille
- Ajuri alustaa ohjaimen ja antaa siirtokäskyn (4)
- Kontrolli takaisin sovellukseen
- CPU voi suorittaa sovelluksen käskyjä samaan aikaan siirron kanssa (2a)
- Kun siirto valmis, ohjain keskeyttää (x)
- Kontrolli keskeytyskäsitteilylle, josta edelleen ajurille
- Ajuri tarkastaa miten siirrossa kävi, ja tekee tarvittavat lopputoimet (5)
- Kun keskeytys käsitelty, sovelluksen suoritus jatkuu (2b)

Fig 1.5b [Stal05]

### Keskeytyskäsitteily (Interrupt handler)

- KJ:n koodia, jonka laitteisto käynnistää keskeytyksen sattuessa
- Selvittää keskeytyksen syyn
- Käynnistää toimet tilanteen hoitamiseksi
  - Siirtyminen sopivaan käsitteilyrutiiniin
- Keskeytynyttä prosessia voitava jatkaa myöhemmin siitä mihin se jäi keskeytyksen sattuessa
  - CPU:n rekistereiden arvot talletettava muistiin
    - PC, PSW, muut ohjelman käyttämät rekisterit

### Keskeytyskäsitteily (kuva 1.10)

Kirja s. 21-25

- \* Etuoikeutettu tila vs. käyttäjätila
- \* Keskeytysten esto vs. salliminen

### Kuva 1.11

### Peräkkäinen käsittely

hyöty / haitta?

- Keskeytykset estetty käsittelyn aikana
  - PSW:n 'keskeytykset estetty' lipuke
- Uudet keskeytykset jäävät odottamaan
- Kun käsitelty, CPU tutkii taas keskeytyslipuketta

### Priorisoitu käsittely

Kirjoitin Tietoliikenne

hyöty / haitta?

- Alemman prioriteetin keskeytyskäsitteily jää kesken, jos tulee kiireellisempi keskeytys
  - Myös keskeytyskäsitteily voi keskeytyä!
- Esim. Tietoliikenneohjaimen keskeytys käsitellään välittömästi, jotta saadaan uutta tilaa saapuvalle datalle

## Muistihierarkia

Tan01 1.14

Typical access time		Typical capacity
1 nsec	Registers	<1 KB
2 nsec	Cache	1 MB
10 nsec	Main memory	64-512 MB
10 msec	Magnetic disk	5-50 GB
100 sec	Magnetic tape	20-100 GB

**Pentium 4 cache:**  
**8 KB dataalle, 12 KB koodille, ulkoinen 256 KB**

nano =  $10^{-9}$ , mikro =  $10^{-6}$ , milli =  $10^{-3}$

## Välimuisti (cache memory)

- Pieni, nopea muisti / rekisterijoukko CPU:n ja keskusmuistin välissä
- CPU:n osana ja / tai ulkopuolella
- Laitetasolla, ei näy ohjelmissa / KJ:ssä
- CPU tutkii ensin löytyykö viitatus fyys. muistipaikan sisältö välimuistista
- Jos ei löydy, CPU tuo välimuistista lohkon, joka sisältää myös viitatus muistipaikan

## Real Address

-> osoitemuunnos tehty MMU:ssa!

```

    graph TD
      START([START]) --> Receive[Receive address RA from CPU]
      Receive --> IsCache{Is block containing RA in cache?}
      IsCache -- No --> AccessMain[Access main memory for block containing RA]
      IsCache -- Yes --> Fetch[Fetch RA word and deliver to CPU]
      Fetch --> DONE([DONE])
      AccessMain --> Allocate[Allocate cache slot for main memory block]
      Allocate --> Parallel[Perform in parallel]
      Parallel --> Load[Load main memory block into cache slot]
      Parallel --> Deliver[Deliver RA word to CPU]
      Load --> DONE
      Deliver --> DONE
  
```

**Kuva 1.18**

## Osumatodennäköisyys (Hit Ratio)

= Todennäköisyys, että viitattu muistipaikka on välimuistissa

- $T_1$  = saantiaika välimuistista
- $T_2$  = saantiaika keskusmuistista
- $T_2 \gg T_1$
- Kun osumatodenn. lähellä arvoa 1, keskim. saantiaika lähellä  $T_1$ :tä

## Paikallisuus (locality of references)

Ajallinen ja alueellinen paikallisuus:

- Esim. silmukassa suoritetaan toistuvasti samaa käskyjoukkoa
- Tietystä osassa koodia käytetään tyypillisesti vain tiettyjä muuttujia (data)

⇒ Kun ohjelma viittaa tiettyyn muistipaikkaan (käsky tai data), on tn., että se viittaa pian samaan paikkaan uudelleen tai sen lähellä oleviin muistipaikkoihin

- Osumatodenn. helposti lähellä arvoa 1 jo pienellä välimuistilla

## Lohkopuskurit (block cache, disk cache)

- Keskusmuistialue, jonne puskuroidaan levyllä / levyllä siirrettävää dataa
  - ei siis erillinen laitteiston osa
- Kun tiettyihin tavuihin on viitattu, viitataan pian melko varmasti niitä seuraaviin tavuihin
- Jos haettavat tavut eivät löydy puskurista, siirtää KJ kokonaisen levylohkon levyllä puskurin
  - Ennaltanouto
- Paikallisuus!