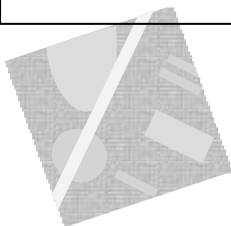
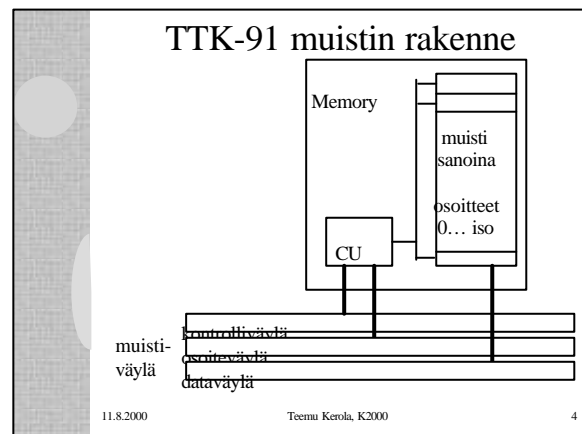
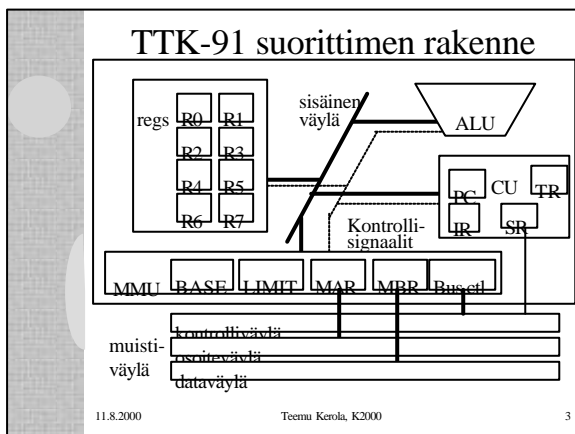
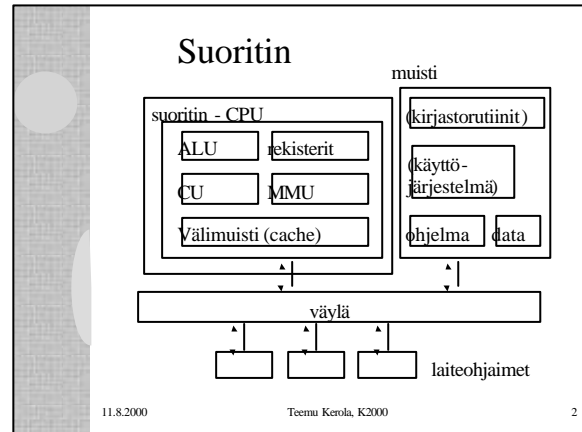


## Jakso 5 Suoritin ja väylä



Suorittimen rakenne  
 Väylän rakenne  
 Käskyjen suoritusyksi  
 Poikkeukset ja keskeytykset

11.8.2000 Teemu Kerola, K2000 1



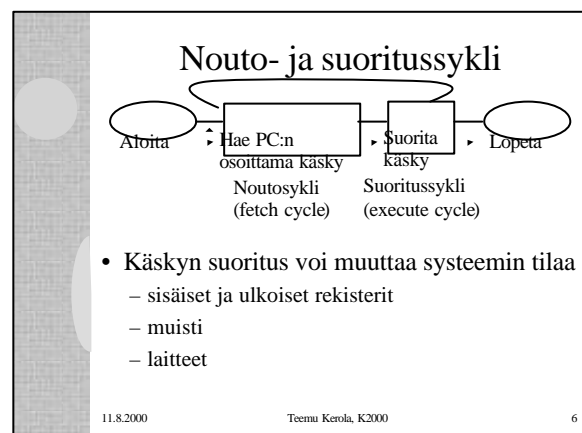
### Käskyjen nouto- ja suoritusyksi<sup>(5)</sup>

- Hae PC:n osoittama konekäsky muistista
  - lisää samalla PC:n arvoa yhdellä
- Suorita konekäsky
  - jos (ehdollinen) hyppykäsky, niin PC:n arvo voi vielä muuttua

Suoritin ei näe mitään suurempia kokonaisuuksia kuin konekäskyjä!

Suoritin ei tiedä mitään ohjelmista!

11.8.2000 Teemu Kerola, K2000 5



### TTK-91 konekäskyn rakenne

- Käskyn esitys bittitasolla on aina:

OPER käskykoodi 8 bit field	Rj 3 bit	M 2 bit	Ri 3 bit	ADDR osoiteosa 16 bit field
--------------------------------	-------------	------------	-------------	--------------------------------

31 24 21 19 16 15 0

Rj = käskyn ensimmäinen operandi  
 Ri = indeksirekisteri (R0<sup>o</sup> 0)  
 M = muistinoutojen määrä toiseen operandiin (ennen mahdollista muistiin talletusta) (addressing mode)  
 00 eli 0 kpl, rekisteri tai välitön osoitus  
 01 eli 1 kpl, suora osoitus  
 10 eli 2 kpl, epäsuora osoitus  
 ( 11 eli 3 kpl, epäkelpo arvo → poikkeustilanne )

11.8.2000 Teemu Kerola, K2000 7

### Nouto- ja suoritussykli tarkemmin <sup>(5)</sup>

- Noutovaihe
  - muistista MBR:n kautta IR:ään
  - Lisää 1 PC:hen
- Käskyn purku ja muistiosoitteen (EA) lasku
  - OPER, Rj, M, Ri, ADDR
  - $TR \leftarrow (Ri) + ADDR$  (tai pelkkä ADDR)
- Operandin nouto
  - muistista MBR:n kautta TR:ään (0-2 krt ?)
- ALU operaatio
  - tulos rekisteriin R0-R7 tai TR:ään
- Muistiin talletus
  - muistiin MBR:n kautta

11.8.2000 Teemu Kerola, K2000 8

### Käskyn noutovaihe <sup>(4)</sup>

ks. TTK-91 suorittimen rakennekuva

- Vie PC:n arvo MAR:iin
- Aseta muistin kontrollisignaali väylälle asentoon "lue"
- Odota kunnes muistiväylä vapautuu ja muistipiiri toimittaa väylän kautta uuden arvon MBR:ään
- Siirrä konekäsky MBR:stä IR:ään

11.8.2000 Teemu Kerola, K2000 9

### Käskyn purku ja muistiosoitteen laskemis -vaihe

OPER käskykoodi 8 bit field	Rj 3 bit	M 2 bit	Ri 3 bit	ADDR osoiteosa 16 bit field
--------------------------------	-------------	------------	-------------	--------------------------------

31 24 21 19 16 15 0

- Purku automaattisesti langoitettuna IR:stä
- Muistiosoitteen lasku, tulos TR:ään
  - jos Ri=0, niin ADDR
  - muutoin laske (Ri)+ADDR
    - ALU suorittaa laskutoimituksen
  - Effective Address (EA)

11.8.2000 Teemu Kerola, K2000 10

### Operandin luku vaihe <sup>(4)</sup>

ks. TTK-91 suorittimen rakennekuva

- Vie muistiosoitte MAR:iin
- Aseta muistin kontrollisignaali väylälle asentoon "lue"
- Odota kunnes muistiväylä vapautuu ja muistipiiri toimittaa väylän kautta uuden arvon MBR:ään
- Siirrä sana MBR:stä TR:ään
  - tai suoraan johonkin laiterekisteriin (R0-R7)

11.8.2000 Teemu Kerola, K2000 11

### ALU operaatio -vaihe <sup>(10)</sup>

- Lähtötilanne
  - käsky haettu ja purettu osiin IR:ssä
  - 1. operandi rekisterissä (R0, ..., R7)
  - 2. operandi TR:ssä
- Käskyn suoritus ALU:ssa
  - vie operandit sisäistä väylää pitkin ALU:un
  - anna ALU:lle sopiva ohjaussignaali
    - add, mul, copyLeft, comp, ...
  - odota, että tulos valmis
  - tulos siirretään joko TR:ään ja/tai SR:ään

Tässä tapahtuu tietokoneen tekemä työ, kaikki muu on hallintoa

11.8.2000 Teemu Kerola, K2000 12

### Tuloksen muistiin kirjoitus - vaihe <sup>(4)</sup>

ks. TTK-91 suorittimen rakennekuva

- Vie muistiosoite MAR:iin
- Vie kirjoitettava sana MBR:ään
- Aseta kontrollisignaali väylälle asentoon "kirjoita muistiin"
- Odota kunnes muistiväylä vapautuu, sana siirretään muistiin väylää pitkin, ja väylän kontrollisignaali kertovat muistiinkirjoittamisen tapahtuneen

11.8.2000 Teemu Kerola, K2000 13

### TTK-91 Nouto- ja suoritusyksi vähän tarkemmin

11.8.2000 Teemu Kerola, K2000 14

### MMU:n toiminta <sup>(2)</sup>

ks. TTK-91 suorittimen rakennekuva

- Ohjelman käyttämät muistiosoitteet (VA) ovat näennäisiä, välillä 0 ... LIMIT-1
  - ne eivät ole samoja osoitteita kuin keskusmuisti käyttää
- MAR:iin menevä arvo VA ei käytetä suoraan, vaan se tarkistetaan ja muokataan ensin
  - Tarkista, onko VA ∈ [0, LIMIT-1]. Jos ei ole, niin aseta bitti M SR:ssä päälle, ja lopeta käskyn suoritus
  - Lisää VA:han BASE ja laita tämä arvo (PA) MAR:iin

VA = virtual address, PA = physical address = BASE + VA

11.8.2000 Teemu Kerola, K2000 15

### TTK-91 virtuaalimuisti

11.8.2000 Teemu Kerola, K2000 16

### Virtuaalimuistin menetelmiä <sup>(4)</sup>

- Kanta- ja rajarekisteriin perustuva
  - base ja limit rekisterit
- Sivuttava
  - sivutaulut
  - virtuaaliavaruus jaettu saman kokoiisiin sivuihin
- Segmentoiva
  - virtuaaliavaruus jaettu erillisiin eri kokoiisiin segmentteihin
    - koodi segmentti, data segmentti, ...

Lisää tietoa? → käyttöjärjestelmäkurssit

11.8.2000 Teemu Kerola, K2000 17

### Keskeytystilanteet <sup>(3)</sup>

- Mikä tahansa tilanne, jonka käsittely vaatii poikkeuksen käskyjen normaaliin suoritusjärjestykseen
- Rakkaalla lapsella on monta nimeä:
  - poikkeus, keskeytys, virhetilanne, trappi, ...
  - exception, interrupt, fault, trap, failure, ....
- Jatkossa yleisnimi keskeytys tarkoittaa kaikkia näitä eri tapauksia tai tyyppisiä

11.8.2000 Teemu Kerola, K2000 18

### Keskeytysten käsittely <sup>(4)</sup>

- Jokainen mahdollinen keskeytystyyppi on ennalta tunnettu
- Jokaiselle keskeytystyypille on oma käyttöjärjestelmän tuntema keskeytyskäsittelyrutiini interrupt handler
- Käskyn suorituksen jälkeen tarkistetaan keskeytysten olemassaolo SR:stä ja haaraututaan keskeytyskäsittelijään tarvittaessa
  - joskus keskeytykset on estetty (SR:n bitti D)
  - paluu käsittelijästä "return-from-interrupt" käskyllä
- "Yllättävä aliohjelmakutsu"

11.8.2000 Teemu Kerola, K2000 19

### Keskeytystyyppejä <sup>(14)</sup>

- Käskyn aiheuttamat virhetilanteet
  - virheellinen käskyn tai datan osoite
  - tuntematon käsky (opcode)
  - nollalla jako
  - kokonaisluvun tai liukuluvun yli/alivuoto
  - käytetty osoite ei ole muistissa (MMU)
- Käskyn aiheuttamat muut poikkeustilanteet
  - SVC käsky
  - I/O kone käsky
  - trace keskeytys
- Ulkoapäin (muualta kuin CPU:lta) tulleet
  - kellolaitekeskeytys (esim. joka 10 ms)
  - laitekeskeytys (esim. levy I/O valmis)
  - laitteistovirhe (esim. virhe väylän tiedonsiirrossa)

11.8.2000 Teemu Kerola, K2000 20

### Keskeytyskäsittelijä

- Osa käyttöjärjestelmää
- Ennen käsittelijän aloittamista asetetaan suoritin (bitti P SR:ssä) ja MMU (BASE=0, LIMIT="hyvin iso") käyttöjärjestelmätilaan (supervisor state)
  - käyttöjärjestelmätilassa saa viitata mihin tahansa kohtaan muistia
  - käyttöjärjestelmätilassa saa käyttää kaikkia konekäskyjä
- Käsittelijästä paluun yhteydessä MMU:n tila ja prosessorin tila asetetaan ennalleen

11.8.2000 Teemu Kerola, K2000 21

### Prossessorin tilat <sup>(6)</sup>

user

kernel

- Käyttäjätila** (user mode, normal mode)
  - voi käyttää vain tavallisia käskyjä
  - voi viitata vain käyttäjän omaan muistiavaruuteen (MMU valvoo)
- Etuoikeutettu tila tai (KJ:n) ytimen tila** (kernel mode, privileged mode)
  - voi käyttää kaikkia konekäskyjä, myös etuoikeutettuja
  - voi viitata kaikkialle muistiin, myös käyttöjärjestelmän ytimeen (kernel)
    - voi käyttää (myös) suoria muistiosoitteita (PA)

11.8.2000 Teemu Kerola, K2000 22

### Prossessorin tilan muuttaminen <sup>(6)</sup>

user

kernel

SVC, INT → kernel  
IRET → user

- Käyttäjätila → etuoikeutettu tila
  - keskeytys tai suora KJ:n palvelupyyntö (SVC käsky)
  - keskeytyskäsittelijä tarkistaa onko keskeytys vaihtoon (interrupt handler)
- Etuoikeutettu tila → käyttäjätila
  - etuoikeutettu konekäsky "return from interrupt handler" esim. IRET (Pentium II)
  - palauttaa kontrollin ja prosessorin tilan keskeytyneeseen kohtaan sen hetkiseen tilaan

11.8.2000 Teemu Kerola, K2000 23

### TTK-91 Nouto- ja suoritusyksi vielä vähän tarkemmin

11.8.2000 Teemu Kerola, K2000 24

### Väylät (5)

- Tiedon siirtoa varten laitteistossa
- Yksi käyttäjä kerrallaan
- Toteutettu johdinkimppuina
- Eri tasoilla
  - suorittimen sisällä ”sisäinen väylä” (internal bus)
  - muistiväylä suorittimen ja muistin välillä (memory bus)
  - I/O-väylä muistiväylän ja I/O-laitteiden välillä (I/O bus)
- Useita eri tapoja yhdistellä edellä olevia

11.8.2000
Teemu Kerola, K2000
25

### Väylähierarkia

Typical Pentium II system

Fig. 3-50 [Tane99]

11.8.2000
Teemu Kerola, K2000
26

### Väylät (5)

- Kullakin laitteella oma osoite
- Yksi lähettää, kaikki kuulevat
- Paljon erilaisia
- Lähellä prosessoria olevat ovat nopeampia

Lisää tietoa?
➔
Tietokoneen rakenne kurssi

11.8.2000
Teemu Kerola, K2000
27

### TTK-91 koneen KOKSI simulaattori (6)

- Tavallinen Pascalilla kirjoitettu ohjelma
- TTK-91 koneen osat tietorakenteina
  - rekisterit, MMU, CU, muisti
- Simuloi käskyjen suoritussykliä käsky kerrallaan
- Toteuttaa TTK-91 koneen käyttäjärjestelmän osat osana tavallista ohjelmaa
  - assembler kääntäjä, lataaja, debugger, kesk. käsittelijät
- Graafinen käyttöliittymä

11.8.2000
Teemu Kerola, K2000
28

### TTK-91 käskyn suoritussykli (6)

hae käsky simuloidusta muistista IR = mem[PC]

pura käsky osiin (OPER, Rj, M, Ri, ADDR) ja laske osat

$ADDR = IR \% 65536$       $TR = regs[Ri] + ADDR$

tee tarvittava määrä (M) operandin hakuja muistista rekisteriin TR TR = mem[TR]

valitse a

if (opcodeOK[OPER] = FALSE) then SR.U = 1; simuloi konekäskyn suorituksen muutokset rekistereihin

$ADDRi, M ADDR(Rj) \Rightarrow [regs[Ri] += TR;$

lopetä suoritus jos SVC tai keskeytys SR.O = ...

11.8.2000
Teemu Kerola, K2000
29

### -- Jakson 5 loppu --

Figure 3.12 Instruction Cycle State Diagram, With Interrupts [Stal99]

11.8.2000
Teemu Kerola, K2000
30