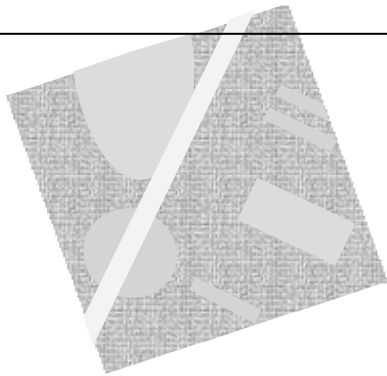


Jakso 5

Suoritin ja väylä



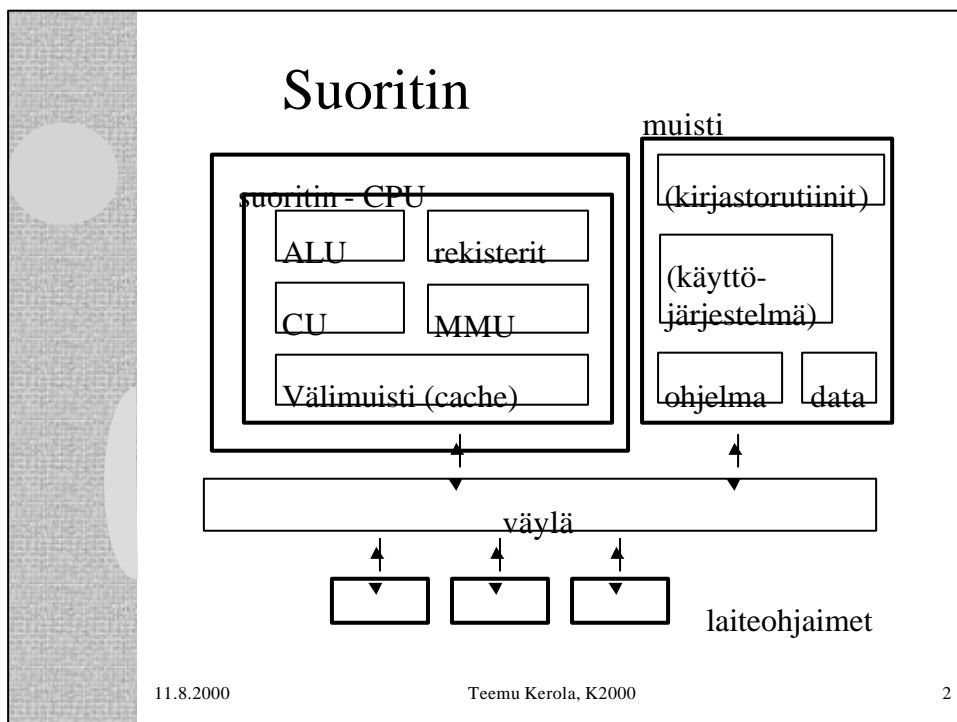
Suorittimen rakenne

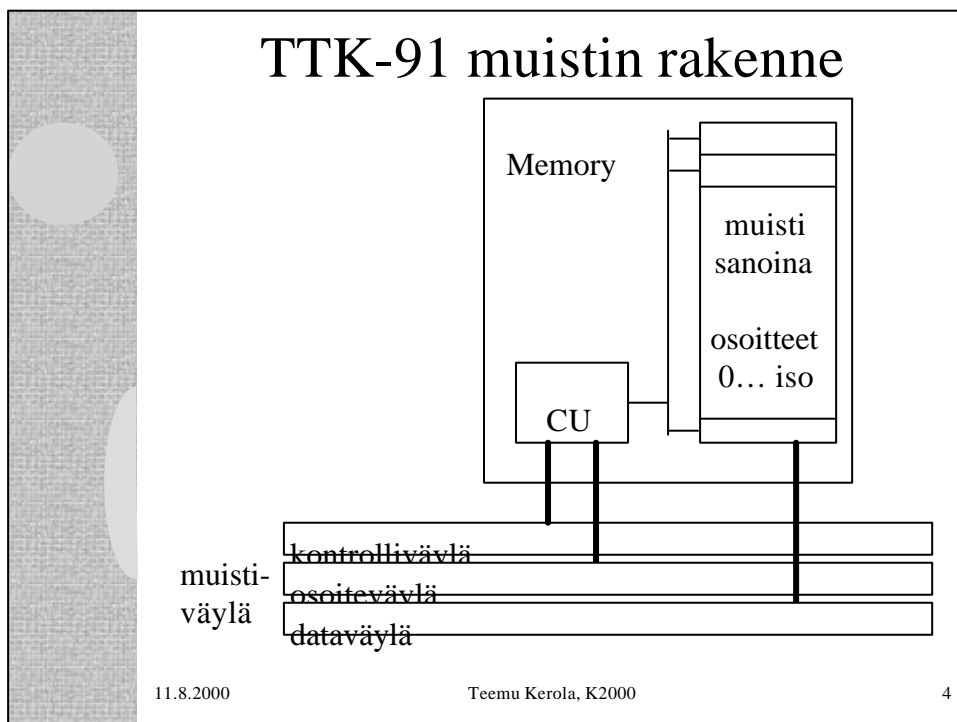
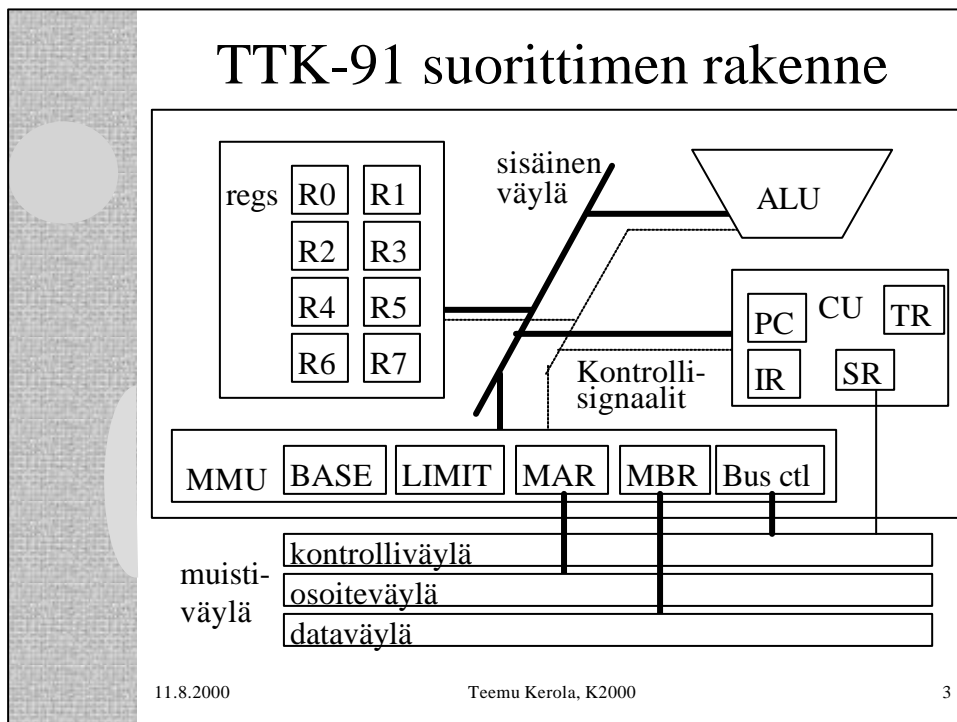
Väylän rakenne

Käskyjen suoritusyksi

Poikkeukset ja keskeytykset

11.8.2000
Teemu Kerola, K2000
1





Käskyjen nouto- ja suoritusyksi (5)

- 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

Nouto- ja suoritusyksi

```

graph LR
  Aloita([Aloita]) --> Hae[Hae PC:n osoittama käsky]
  Hae --> Suorita[Suorita käsky]
  Suorita --> Lopeta([Lopeta])
  Suorita --> Hae
  
```

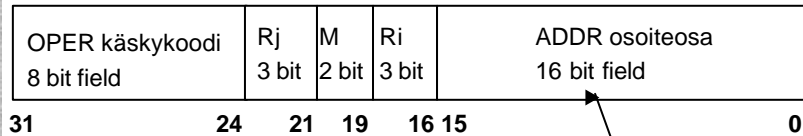
Noutosykli (fetch cycle)
Suoritusyksi (execute cycle)

- Käskyn suoritus voi muuttaa systeemin tilaa
 - sisäiset ja ulkoiset rekisterit
 - muisti
 - laitteet

11.8.2000 Teemu Kerola, K2000 6

TTK-91 konekäskyn rakenne

- Käskyn esitys bittitasolla on aina:



Rj = käskyn ensimmäinen operandi

Ri = indeksirekisteri (R0^o 0)

M = muistinoutojen määrä toiseen operandiin
(ennen mahdollista muistiin talletusta)

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)

muistiosoite tai
(pienekkö) vakio

(addressing
mode)

11.8.2000

Teemu Kerola, K2000

7

Nouto- ja suoritusyksi tarkemmin ⁽⁵⁾

- 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

ks. TTK-91
suorittimen
rakennokuva

Ei kaikilla käskyillä

Ei kaikilla käskyillä

11.8.2000

Teemu Kerola, K2000

8

Käskyn noutovaihe ⁽⁴⁾

ks. TTK-91 suorittimen rakennekuva

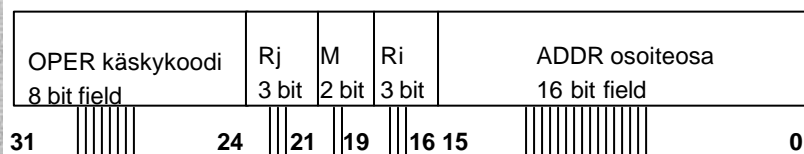
- Vie PC:n arvo MAR:iin
- Aseta muistin kontrollisignaalit väylälle asentoon "lue"
- Odota kunnes muistiväylä vapautuu ja muistipiiri toimittaa väylän kautta uuden arvon MBR:ään
- Siirrä konekäsky MBR:stä IR:ään

11.8.2000

Teemu Kerola, K2000

9

Käskyn purku ja muistiosoitteen laskemis -vaihe



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

11.8.2000

Teemu Kerola, K2000

10

Operandin luku vaihe ⁽⁴⁾

ks. TTK-91 suorittimen rakennekuva

- Vie muistiosoite MAR:iin
- Aseta muistin kontrollisignaalit 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 ⁽¹⁰⁾

ks. TTK-91 suorittimen rakennekuva

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

Tässä tapahtuu tietokoneen tekemä työ,
kaikki muu on hallintoa

11.8.2000

Teemu Kerola, K2000

12

Tuloksen muistiin kirjoitus - vaihe ⁽⁴⁾

ks. TTK-91 suorittimen rakennekuva

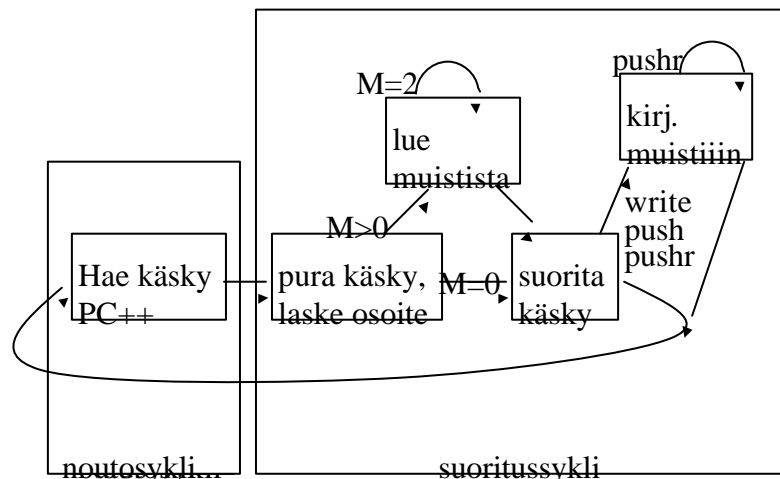
- Vie muistiosoite MAR:iin
- Vie kirjoitettava sana MBR:ään
- Aseta kontrollisignaalit väylälle asentoon ”kirjoita muistiin”
- Odota kunnes muistiväylä vapautuu, sana siirretään muistiin väylää pitkin, ja väylän kontrollisignaalit kertovat muistiinkirjoittamisen tapahtuneen

11.8.2000

Teemu Kerola, K2000

13

TTK-91 Nouto- ja suoritussykli vähän tarkemmin



11.8.2000

Teemu Kerola, K2000

14

MMU:n toiminta (2)

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 $VA \in [0, \text{LIMIT}-1]$.
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

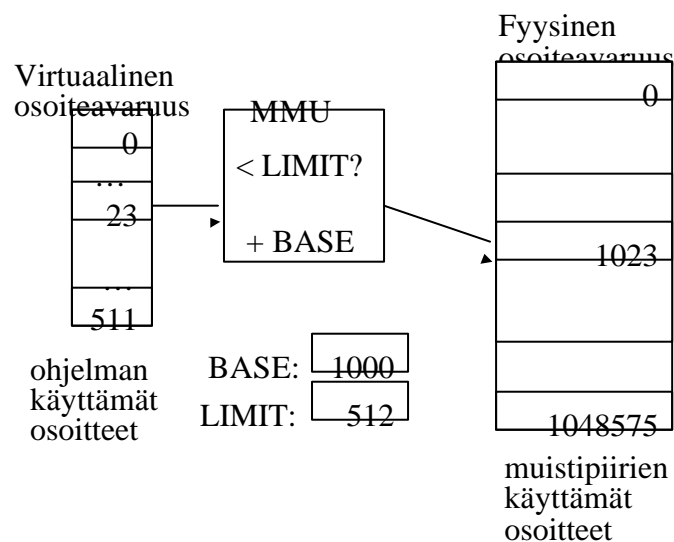
VA = virtual address, PA = physical address = BASE+VA

11.8.2000

Teemu Kerola, K2000

15

TTK-91 virtuaalimuisti



11.8.2000

Teemu Kerola, K2000

16

Virtuaalimuistin menetelmiä ⁽⁴⁾

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

Lisää
tietoa?



käyttö-
järjestelmä
kurssit

11.8.2000

Teemu Kerola, K2000

17

Keskeytystilanteet ⁽³⁾

- 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 ⁽⁴⁾

- Jokainen mahdollinen keskeytystyyppi on ennalta tunnettu
- Jokaiselle keskeytystyypille on oma käyttäjärjestelmän tuntema keskeytyskäsitteilyrutiini interrupt handler
- Käsken suorituksen jälkeen tarkistetaan keskeytysten olemassaolo SR:stä ja haaraututaan keskeytyskäsitteilyrutiiniin tarvittaessa
 - joskus keskeytykset on estetty (SR:n bitti D)
 - paluu käsitteilyrutiiniin ”return-from-interrupt” käskyllä
- ”Yllättävä aliohjelmakutsu”

11.8.2000

Teemu Kerola, K2000

19

Keskeytystyyppejä ⁽¹⁴⁾

- Käsken aiheuttamat virhetilanteet
 - virheellinen käsken tai datan osoite
 - tuntematon käsky (opcode)
 - nollalla jako
 - kokonaisluvun tai liukuluvun yli/alivuoto
 - käytetty osoite ei ole muistissa (MMU)
- Käsken aiheuttamat muut poikkeustilanteet
 - SVC käsky
 - I/O konekäsky
 - 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

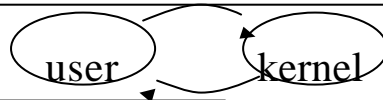
- Osa käyttöjärjestelmää
- Ennen käsittelijän aloittamista asetetaan suoritin (bitti P SR:ssä) ja MMU (BASE=0, LIMIT="hyvin iso") käyttöjärjestelmätilaan.
(supervisor state)
 - käyttöjärjestelmätilassa saa viitata mihin tahansa kohtaan muistia
 - käyttöjärjestelmätilassa saa käyttää kaikkia konekäskyjä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

11.8.2000

Teemu Kerola, K2000

21

Prossessorin tilat ⁽⁶⁾



- 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

Proessorin tilan muuttaminen ⁽⁶⁾


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

11.8.2000
Teemu Kerola, K2000
23

TTK-91 Nouto- ja suoritussykli 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

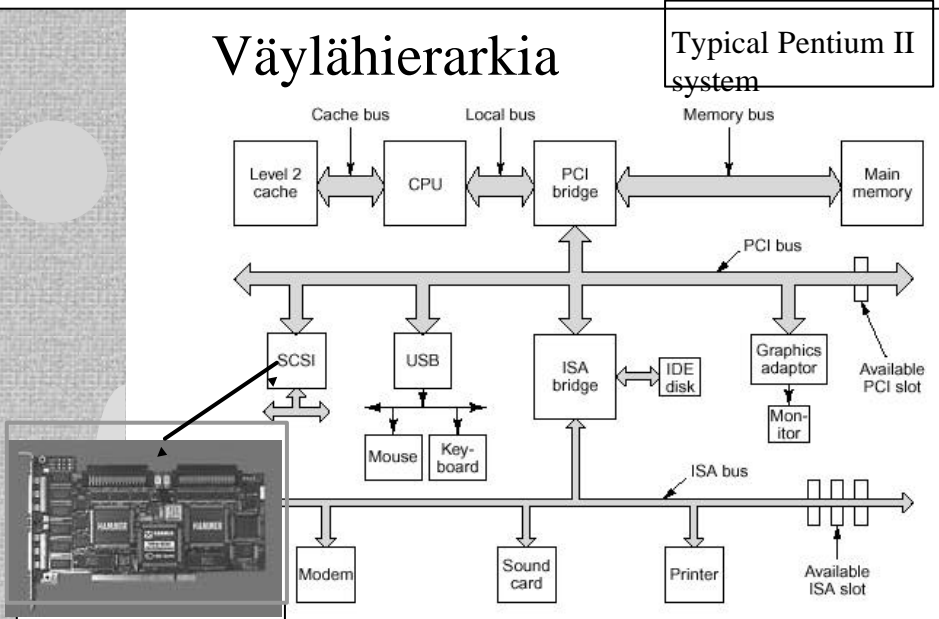
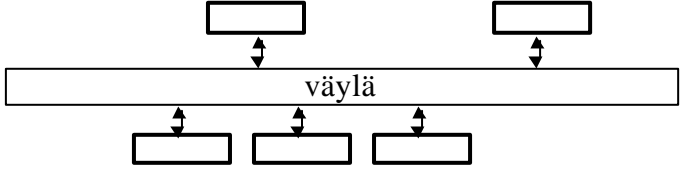


Fig. 3-50 [Tane99]

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 käyttöjärjestelmän osat osana tavallista ohjelmaa
 - assembler kääntäjä, lataaja, debugger, kesk. käsittelijät
- Graafinen käyttöliittymä

11.8.2000 Teemu Kerola, K2000 28

TTK-91 käskyn suoritusyksi (5)

hae käsky simuloidusta muistista IR = mem[PC]
 pura käsky osiin (OPER, Rj, M, Ri, ADDR) ja
 laske osoiteosan arvo TR (ADDR tai regs[Ri]+ADDR)
ADDR = IR % 65536 TR = regs[Ri] + ADDR
 tee tarvittava määrä (M) operandin
 hakuja muistista rekisteriin TR TR = mem[TR]
 valitse aliohjelma operaatiokoodin (OPER) perusteella
if (opcodeOK[OPER] = FALSE) then SR.U = 1;
 simuloi konekäskyn suorituksen muutokset
 rekistereihin (R0...R7, SR, PC, MAR, MBR)
ADD Ri, M ADDR(Rj) ⇒ regs[Ri] += TR;
 lopeta 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 [Sta199]

11.8.2000
Teemu Kerola, K2000
30