

Jakso 5

Suoritin ja väylä

Suorittimen rakenne

Väylän rakenne

Käskeyjen suoritusyksi

Poikkeukset ja keskeytykset

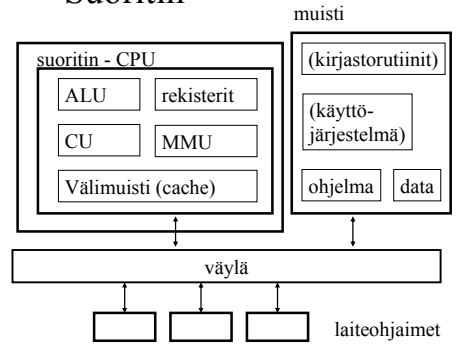
TTK-91:n ja KOKSI:n rakenne

17/05/2004

Copyright Teemu Kerola, K2003

1

Suoritin

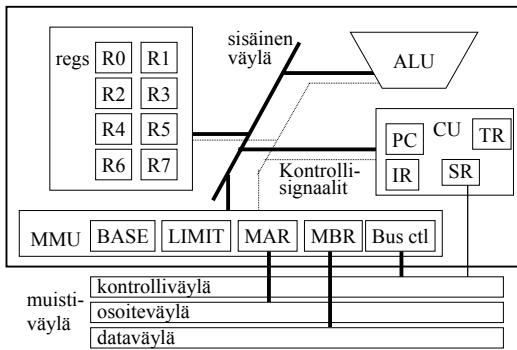


17/05/2004

Copyright Teemu Kerola, K2003

2

TTK-91-suorittimen rakenne

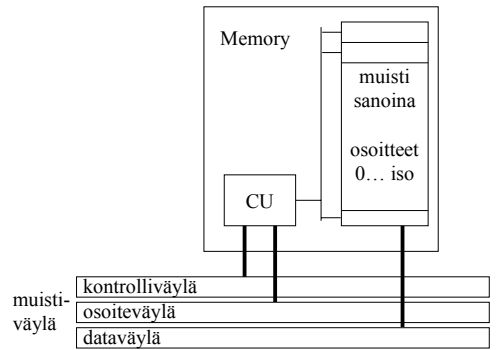


17/05/2004

Copyright Teemu Kerola, K2003

3

TTK-91 muistin rakenne



17/05/2004

Copyright Teemu Kerola, K2003

4

Käskeyjen nouto- ja suoritusyksi

- 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!

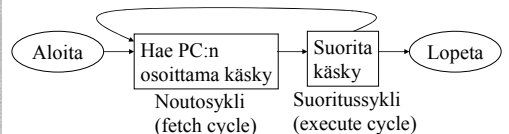
Suoritin ei tiedä mitään ohjelmista!

17/05/2004

Copyright Teemu Kerola, K2003

5

Nouto- ja suoritusyksi



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

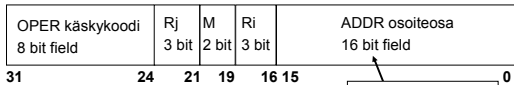
17/05/2004

Copyright Teemu Kerola, K2003

6

TTK-91-konekäskyn rakenne

- Käskyn esitys bittitasolla on aina:



Rj = käskyn ensimmäinen operandi

Ri = indeksirekisteri ($R0 \equiv 0$)

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

- 00 eli 0 kpl, rekisteri tai välitön osoitus (STORE: suoraosoitus)
- 01 eli 1 kpl, suora osoitus (STORE: epäsuoraosoitus)
- 10 eli 2 kpl, epäsuora osoitus (STORE: epäkelpo arvo)
- 11 eli 3 kpl, epäkelpo arvo → poikkeustilanne

muistiosoite tai (pienehkö) vakio

(addressing mode)

17/05/2004

Copyright Teemu Kerola, K2003

7

Nouto- ja suoritussykli tarkemmin ⁽⁵⁾

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

ks. TTK-91 suorittimen rakennekuva

Ei kaikilla käskyillä

Ei kaikilla käskyillä

17/05/2004

Copyright Teemu Kerola, K2003

8

Käskyn noutovaihe ⁽⁴⁾

ks. TTK-91 suorittimen rakennekuva

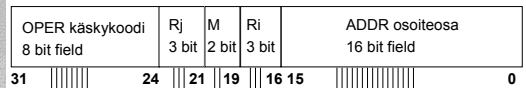
- Vie PC:n arvo MAR:iin
- Aseta muistin käsittelysignaali kontrolliväylälle asentoon "lue"
- Odoti kunnes muistiväylä vapautuu ja muistipiiri toimittaa väylän kautta uuden arvon MBR:ään
- Siirrä konekäsky MBR:stä IR:ään

17/05/2004

Copyright Teemu Kerola, K2003

9

Käskyn purku ja tehollisen muistiosoitteen (EA) laskemisvaihe



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

17/05/2004

Copyright Teemu Kerola, K2003

10

Operandin luku vaihe ⁽⁴⁾

ks. TTK-91 suorittimen rakennekuva

- Vie muistiosoite MAR:iin
- Aseta kontrollisignaali väylälle asentoon "lue muistista"
 - Varaa ensin kontrolliväylä signaalilla "varaa väylä"
- Odoti kunnes muistiväylä vapautuu ja muistipiiri toimittaa väylän kautta uuden arvon MBR:ään
 - vapautaa kontrolliväylä
- Siirrä sana MBR:stä TR:ään
 - (tai suoraan johonkin laiterekisteriin (R0-R7))

17/05/2004

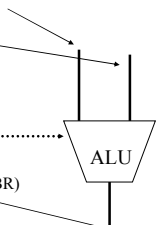
Copyright Teemu Kerola, K2003

11

ALU-operaation suoritussvaihe ⁽¹⁰⁾

ks. TTK-91 suorittimen rakennekuva

- Lähtötilanne
 - käsky haettu ja purettu osiin IR:stä
 - 1. operandi rekisterissä ($R0, \dots, R7$)
 - 2. operandi TR:ssä
- Käskyn suoritus ALU:ssa
 - vie operandit sisäistä väylää pitkin ALU:un
 - anna ALU:lle sopiva ohjaussignaali
 - add, mul, shl, not, comp, ...
 - odota, että tulos valmis
 - talleta tulos rekisteriin (R0-R7, TR, PC, MBR) ja/tai SR:ään



Tässä tapahtuu tietokoneen tekemä työ, kaikki muu on hallintoa

17/05/2004

Copyright Teemu Kerola, K2003

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 sana siirretään muistiin väylää pitkin, ja väylän kontrollisignaalit kertovat muistiinkirjoittamisen tapahtuneen

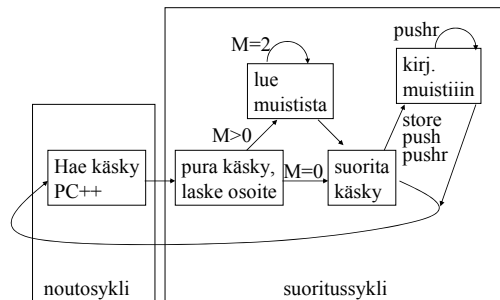


17/05/2004

Copyright Teemu Kerola, K2003

13

TTK-91: Nouto- ja suoritussykli vähän tarkemmin



PUSHR, POPR erikoistapauksia: aika monimutkaisia

17/05/2004

Copyright Teemu Kerola, K2003

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

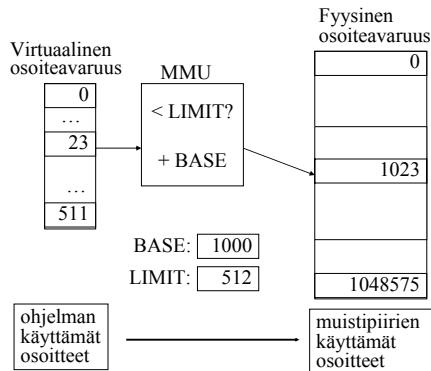
VA = virtual address, PA = physical address = BASE+VA

17/05/2004

Copyright Teemu Kerola, K2003

15

TTK-91- virtuaalimuisti



17/05/2004

Copyright Teemu Kerola, K2003

16

Virtuaalimuistin menetelmiä (4)

- Kanta- ja rajarekisteriin perustuva
 - base ja limit rekisterit (esim. TTK-91, 8086, ...)
- Sivuttava
 - sivutaulut
 - virtuaaliavaruus jaettu saman kokoisiin sivuihin
- Segmentoiva
 - virtuaaliavaruus jaettu ohjelman mukaan erillisiin eri kokoisiin segmentteihin
 - koodi segmentti, data segmentti, ...



17/05/2004

Copyright Teemu Kerola, K2003

17

Keskeytystilanteet (3)

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

17/05/2004

Copyright Teemu Kerola, K2003

18

Keskeytysten käsittely ⁽⁴⁾

- Jokainen mahdollinen keskeytystyyppi on ennalta tunnettu
- Jokaiselle keskeytystyypille on oma käyttöjärjestelmän tuntema keskeytyskäsittelyrutiini interrupt handler
- Käselyn 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 (IRET)
- ”Yllättävä aliohjelmakutsu”

17/05/2004

Copyright Teemu Kerola, K2003

19

Keskeytystyyppejä ⁽³⁾

- **Käskyn aiheuttamat virhetilanteet**
- **Käskyn aiheuttamat muut poikkeustilanteet**
 - kyseessä ei siis ole virhetilanne, vaan haluttu käyttäytyminen
 - tilanne vaatii erikoistoimenpiteen, jonka toteutus on tehty keskeytyskäsittelyn kaltaiseksi
- **Ulkoapäin (muualta kuin CPU:lta) tulleisiin signaaleihin reagointi**

17/05/2004

Copyright Teemu Kerola, K2003

20

Käskyn aiheuttamat virhetilanteet ⁽⁵⁾

- Virheellinen käselyn tai datan osoite
- Tuntematon käsky (opcode)
- Nollalla jako
- Kokonaisluvun tai liukuluvun yli/alivuoto
- Käytetty osoite ei ole muistissa (MMU)

17/05/2004

Copyright Teemu Kerola, K2003

21

Käskyn aiheuttamat muut poikkeustilanteet

- SVC- käsky
- I/O- konekäsky
- Trace- keskeytys
- Käyttäjän määrittelemä keskeytys
 - esim. Javan operaatioiden throw/catch tai try/catch toteutus

17/05/2004

Copyright Teemu Kerola, K2003

22

Ulkoapäin (muualta kuin suorittimelta) tulleet keskeytykset ⁽³⁾

- Kellolaitekeskeytys (esim. joka 10 ms)
- Laitekeskeytys (esim. levy I/O valmis)
- Laitteistovirhe (esim. virhe väylän tiedonsiirrossa)

17/05/2004

Copyright Teemu Kerola, K2003

23

Keskeytyskäsittelijä

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

17/05/2004

Copyright Teemu Kerola, K2003

24

Suorittimen tilat

(6)



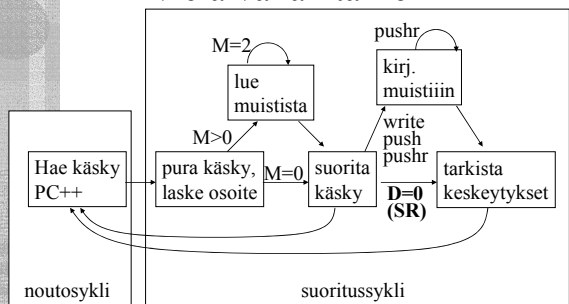
- Käyttäjätila (user mode, normal mode)
 - voi käyttää vain tavallisia käskyjä
 - voi viitata vain käyttäjän omaan muistiavaruuteen (MMU valvoo)
- Etuoikeutettu tila tai (KJ:n) ytimen tila (kernel mode, privileged mode)
 - voi käyttää kaikkia konekäskyjä, myös etuoikeutettuja (esim. clear_cache, iret)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

Suorittimen tilan muuttaminen (6)



- Käyttäjätila → etuoikeutettu tila
 - keskeytys tai suora KJ:n palvelupyynnö (SVC käsky)
 - keskeytyskäsitteijä tarkistaa onko oikeutta tilan vaihtoon (interrupt handler)
- Etuoikeutettu tila → käyttäjätila
 - etuoikeutettu konekäsky "return from interrupt handler" esim. IRET (Pentium II)
 - palauttaa kontrollin keskeytyneeseen kohtaan ja suorittimen tilan keskeytystä edeltäneeseen tilaan

TTK-91 Nouto- ja suoritusyksi vielä vähän tarkemmin



Väylät (5)

- Tiedon siirtoa varten laitteistossa
- Yksi kirjoittaja 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



(internal bus)

(memory bus)

(I/O bus)

Väylähierarkia

Tyypillinen Pentium II systeemin emolevy

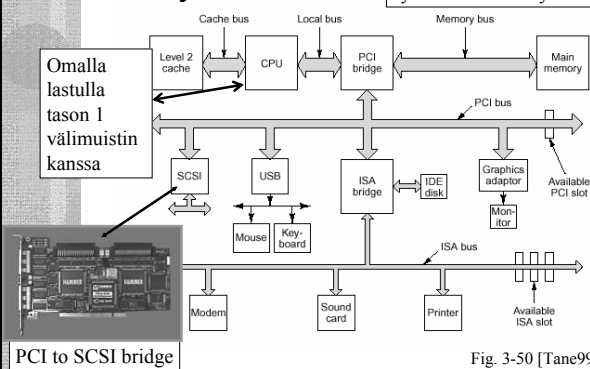
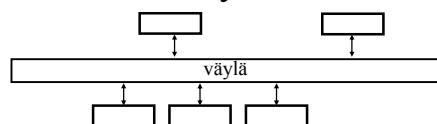


Fig. 3-50 [Tane99]

Väylät (5)



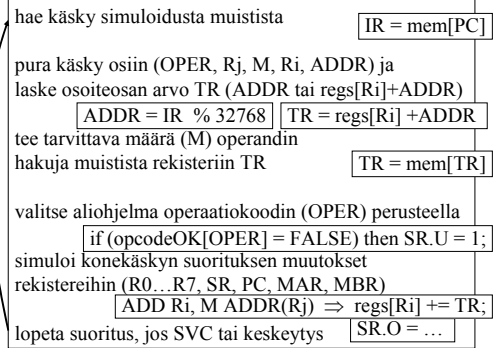
- Kullakin laitteella oma osoite
- Yksi lähettää, kaikki kuulevat, vain 'oikea' laite vastaanottaa
- Paljon erilaisia
- Lähellä suoritinta olevat ovat nopeampia

Lisää tietoa? Tietokoneen rakenne kurssi

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

TTK-91-käskyn suoritussykli (5)



-- Jakson 5 loppu --

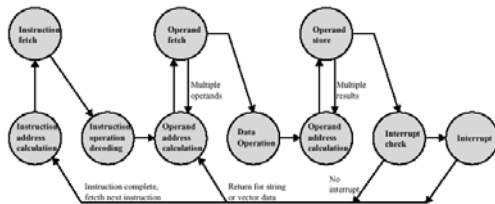


Figure 3.12 Instruction Cycle State Diagram, With Interrupts

[Stal99]