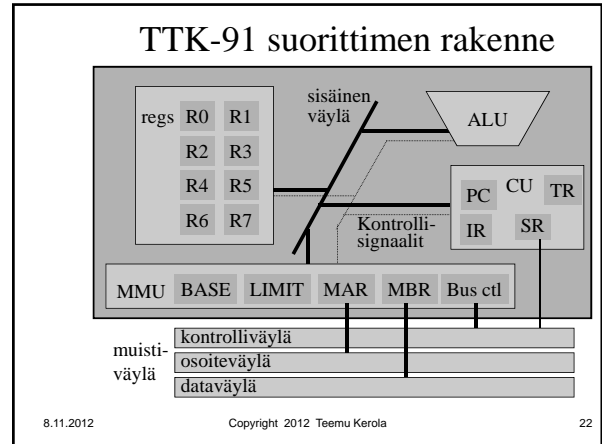


### Luento 4 (verkkoluento 5) Suoritin ja väylä

Käskyjen suoritusyksi  
Suorittimen tilat  
Poikkeukset ja keskeytykset

8.11.2012 Copyright 2012 Teemu Kerola 1



### Käskyn suorittaminen

- Titokone näyttää vaiheet simuloimalla
- Store R4, @10(R1) ; R1 =20, R4=15
  - Käskyn nouto ja ohjelmalaskurin kasvatus:
    - PC -> MAR, mem -> MBR, (odota), MBR -> IR
    - PC+1 -> PC
    - PC -> ALU1, 1-> ALU2, +, (odota), ALU -> PC (??)
  - Käskyn suoritus:
    - R1 -> ALU1, 10 -> ALU2, +, (odota), ALU-> TR
    - TMP -> MAR, mem -> MBR, (odota), MBR -> MAR, R4 -> MBR, MBR -> mem, (odota)
  - Huom: yhteensä 3 muistiviitettä

8.11.2012 Copyright 2012 Teemu Kerola 3

### 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 (0 ≡ ei käytetty)  
M = muistinoutojen määrä toiseen operandiin (ennen mahdollista muistiin talletusta)  
00 eli 0 kpl, välitön osoitus (STORE: suora osoitus)  
01 eli 1 kpl, suora osoitus (STORE: epäsuora osoit.)  
10 eli 2 kpl, epäsuora osoitus (STORE: epäkelvo arvo)  
11 eli 3 kpl, epäkelvo arvo → poikkeustilanne

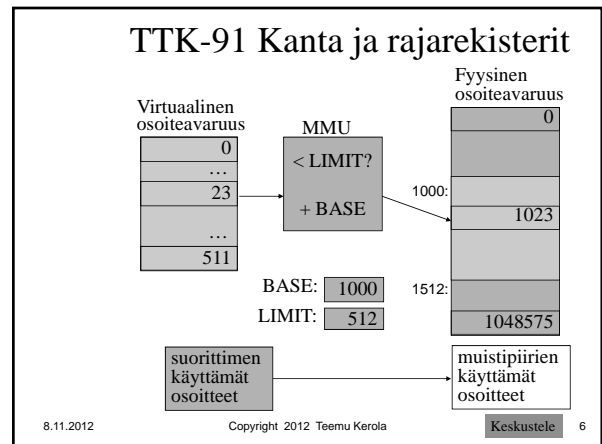
8.11.2012 Copyright 2012 Teemu Kerola Keskustelee 4

### Käskyn purku ja tehollisen muisti-osoitteen (EA) laskemisvaihe

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
				(bitti:) 0



- Purku automaattisesti langoitettuna käskyrekisteristä (IR)
- Muistiosoitteen lasku, tulos tilapäisrekisteriin (TR)
  - jos Ri=0, niin TR ← ADDR
  - muutoin TR ← (Ri)+ADDR
    - ALU suorittaa laskutoimituksen
    - jos ADDR = 0, niin TR ← (Ri)
  - Effective Address (EA) on nyt TR:ssä
- EA:ta voi käyttää datana sellaisenaan tai muistiosoitteena

8.11.2012 Copyright 2012 Teemu Kerola 5



## Virtuaalimuistin osoitteenmuunnosmenetelmiä

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

8.11.2012 Copyright 2012 Teemu Kerola 7

## Keskeytysten käsittely

- Jokainen mahdollinen keskeytystyyppi on ennalta tunnettu, eli *mitään todella yllättävää ei tapahdu*
- Jokaiselle keskeytystyypille on oma käyttäjärjestelmän tuntema **keskeytyskäsitteilyrutiini** **interrupt handler**
- Jokaisen käskyn suorituksen jälkeen tarkistetaan keskeytysten olemassaolo SR:stä ja haaraututaan keskeytyskäsitteilyrutiiniin tarvittaessa
  - joskus keskeytykset on estetty (ttk-91:ssä SR:n bitti D)
  - paluu käsitteilyrutiinista ”return-from-interrupt-handler” käskyllä (esim. IRET, tms)
- ”Yllättävä aliohjelmakutsu”

8.11.2012 Copyright 2012 Teemu Kerola 8

## Keskeytyskäsitteily

- Osa käyttäjärjestelmää
- Ennen keskeytyskäsitteilyrutiiniin hypäämistä asetetaan suoritin ja MMU etuoikeutettuun käyttäjärjestelmätilaan (supervisor state, kernel st.)
  - SR:n bitti P on päällä => etuoikeutettu tila eli (P = Priviledged) käyttöjärjestelmä tila
  - käyttäjärjestelmätilassa saa viitata mihin tahansa kohtaan muistia (MMU: BASE=0, LIMIT=”hyvin iso”)
  - käyttäjärjestelmätilassa saa käyttää kaikkia konekäskyjä (esim. IRET tai ClearCache)
- Käsitteilyrutiinista paluun yhteydessä MMU:n tila ja suorittimen tila (bitti P) asetetaan ennalleen

8.11.2012 Copyright 2012 Teemu Kerola 9

## Suorittimen tilan muuttaminen

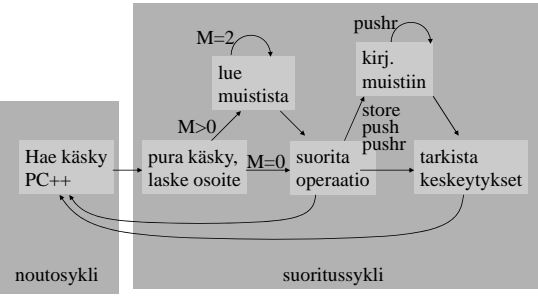


- Käyttäjätila → etuoikeutettu tila
  - keskeytys tai suora KJ:n palvelupyyntö (SVC käsky)
  - keskeytyskäsitteilyrutiini tarkistaa onko (oliko) oikeutta tilan vaihtoon (interrupt handler)
- Etuoikeutettu tila → käyttäjätila
  - etuoikeutettu konekäsky ”return from interrupt handler” esim. IRET (Pentium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

Keskustele 10

8.11.2012 Copyright 2012 Teemu Kerola

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



8.11.2012 Copyright 2012 Teemu Kerola Keskustele 11

## Titokone - TTK-91 koneen simulaattori

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

Ks. Processor.java Titokoneen koodissa:  
titokone.jar/fi/lu/cs/titokone/Processor.java  
(<http://www.cs.helsinki.fi/group/nodes/kurssit/tito/Processor.java.txt>)

8.11.2012 Copyright 2012 Teemu Kerola 12

### TTK-91 käskyn suoritusykli

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)  
 $ADDR = IR \text{ mod } 32768$      $TR = \text{regs}[Ri] + ADDR$

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)  
 $ADD Rj, M \text{ ADDR}(Ri) \Rightarrow \text{regs}[Rj] += TR;$   
 lopeta suoritus jos SVC tai keskeytys     $SR.O = \dots$

Simulaattori C:llä: <http://www.cs.helsinki.fi/group/nodes/kurssit/tito/simu/simu.c>

8.11.2012    Copyright 2012 Teemu Kerola    13

### Väylät

- Kullakin laitteella oma osoite
- Yksi lähettää, kaikki kuulevat, vain ”oikea” laite vastaanottaa
- Paljon erilaisia
- Lähellä suoritinta

Lisää tietoa? Tietokoneen rakenne -kurssi

8.11.2012    Copyright 2012 Teemu Kerola    14

### Väylähierarkia

Tyypillinen Pentium II systeemin emolevy

omalla lastulla, tason 1 välimuistin kanssa

PCI to SCSI bridge

Fig. 3-50 [Tane99]

8.11.2012    Copyright 2012 Teemu Kerola    Keskustele 15

-- Luennon 4 loppu --

ESKO, 1960 ensimmäinen Suomessa rakennettu tietokone, vanhentunut jo valmistuessaan, 20 yhteenlaskua per sek.

- Ohjelmakoodi luettiin reikänauhoilta (10 kpl)
- Aliohjelmakutsu toteutettu kontrollin siirrolla toiseen reikänauhanlukijaan

8.11.2012    Copyright 2012 Teemu Kerola    16