

Kertausluento 2 (lu04, lu05, lu06)
Aliohjelmien toteutus
Suoritin, väylä, tiedon esitys

Tyypit, Parametrit
Aktivointitietue (AT)
AT-pino
Käskeyjen suoritusyksi
Suorittimen tilat
Poikkeukset ja keskeytykset
Kokonaisluvut
Liukuluvut
Muut tiedot

26.3.2010

Copyright Teemu Kerola 2010

1

Luento 4
Aliohjelmien toteutus

Tyypit
Parametrit
Aktivointitietue (AT)
AT-pino
Rekursio

26.3.2010

Copyright Teemu Kerola 2010

2

Aliohjelmatyypit

- Korkean tason ohjelmointikielen käsitteet
 - aliohjelma, proseduuri
 - parametrit
 - funktio
 - parametrit, paluuarvo
 - metodi
 - parametrit, ehkä paluuarvo
- Konekielen tason vastaavat käsitteet
 - aliohjelma
 - parametrit ja paluuarvo(t)

26.3.2010

Copyright Teemu Kerola 2010

3

Parametrit ja paluuarvo

- Muodolliset parametrit
 - määritelty aliohjelmassa Tulosta (int x, y)
ohjelmointihetkellä Laske(int x): int
 - tietty järjestys ja tyyppi
 - paluuarvot
 - käsittely hyvin samalla tavalla kuin parametreillekin
- Todelliset parametrit ja paluuarvo
 - todelliset parametrit sijoite- Tulosta (5, apu);
taan muodollisten parametrien x = Laske(y+234);
paikalle kutsuhetkellä suoritusaikana
 - paluuarvo saadaan paluuhetkellä ja sitä
käytetään kuten mitä tahansa arvoa

26.3.2010

Copyright Teemu Kerola 2010

4

Parametryypit

- Arvoparametri
 - välitetään parametrin arvo (eli sen kopio) kutsuhetkellä
 - arvo voidaan lukea
 - alkuperäistä arvoa ei voi muuttaa, mutta arvon kopiota voi muuttaa
- Viiteparametri
 - välitetään parametrin osoite
 - arvo ja osoite voidaan lukea, arvoa voi muuttaa
- Nimiparametri
 - välitetään parametrin nimi
 - nimi (merkkijono) kuvataan arvoksi kutsuhetkellä
 - semantiikka määräytyy vasta kutsuhetkellä

```
Swap(i, k);      Swap(i, T[i]);
```

```
tmp = i;
i = T[i];
T[i] = tmp; # ?
```

26.3.2010

Copyright Teemu Kerola 2010

5

Aliohjelmien toteutuksen osat

- Paluuosoite
 - kutsukohtaa seuraava käskyn osoite
- Parametrien välitys
- Paluarvon välitys
- Paikalliset muuttujat
- Rekistereiden allokointi (varaus)
 - kutsuvalla ohjelman osalla voi olla käytössä rekistereitä, joiden arvon halutaan säilyä!
 - pääohjelma, toinen aliohjelma, sama aliohjelma, metodi, ...
 - käytettyjen rekistereiden arvot pitää aluksi tallettaa muistiin ja lopuksi palauttaa ennalleen

26.3.2010

Copyright Teemu Kerola 2010

6

Aktivointitietue

(activation record,
activation frame)

```
int funcA (int x,y);
```

- Aliohjelman toteutusmuoto (ttk-91)
 - funktion paluuarvo (tai kaikki paluuarvot)
 - kaikkien (sisäänmeno- ja ulostulo-) parametrien arvot
 - paluusoite
 - kutsukohdan aktivointitietue **FP**
 - kaikki paikalliset muuttujat ja tietorakenteet
 - aliohjelman ajaksi talletettujen rekistereiden alkuperäiset arvot

26.3.2010 Copyright Teemu Kerola 2010 7

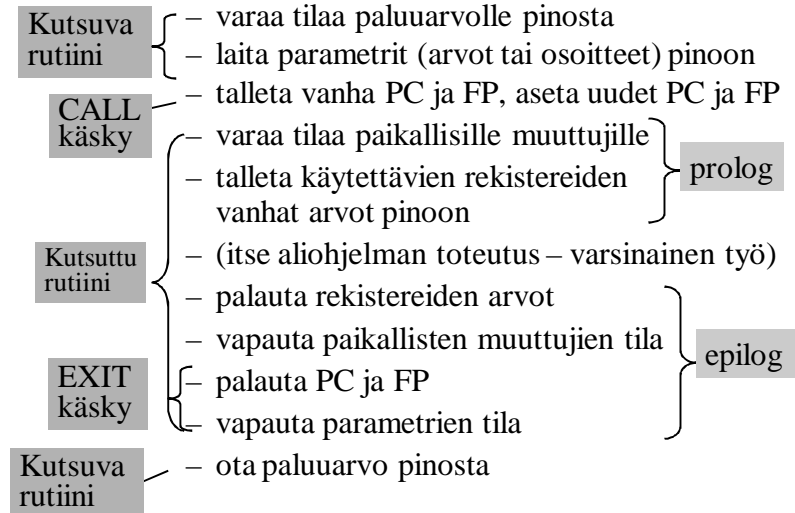
Aktivointitietueiden hallinta

- Aktivointitietueet (AT) varataan ja vapautetaan dynaamisesti (suoritusaikana) pinosta (muistista)
 - SP (=R6) osoittaa pinon pinnalle
- Aktivointitietuepino
 - FP (R7) osoittaa voimassa olevan AT:n sovittuun kohtaan (ttk-91: vanhan FP:n osoite)
- Pinossa olevaa AT:tä rakennetaan ja puretaan käskyillä:
 - PUSH, POP, PUSHR, POPR
 - CALL, EXIT (SVC, IRET)

26.3.2010 Copyright Teemu Kerola 2010 8

Aliohjelman käytön toteutus

- Toteutus jaettu eri yksiköille



26.3.2010

Copyright Teemu Kerola 2010

9

KJ-palvelun kutsu (proseduraalisesti) (kesk.)

- Samalla tavalla kuin aliohjelman kutsu
 - CALL käskyn asemesta SVC
- Tila paluuarvolle?
- Parametrit pinoon vai rekistereissä?
- SVC kutsu
 - Kutsuttavan rutiinin numero operandina
- IRET paluu
- Paluuarvo (OK, virhe) pois pinosta tarkistusta varten

fOK = ReadBlock (fp, 64)

```

...
PUSH SP, =0 ;paluuarvo
PUSH SP, =FileBuffer
PUSH SP, CharCnt
PUSH SP, FilePtr

SVC SP, =ReadFile

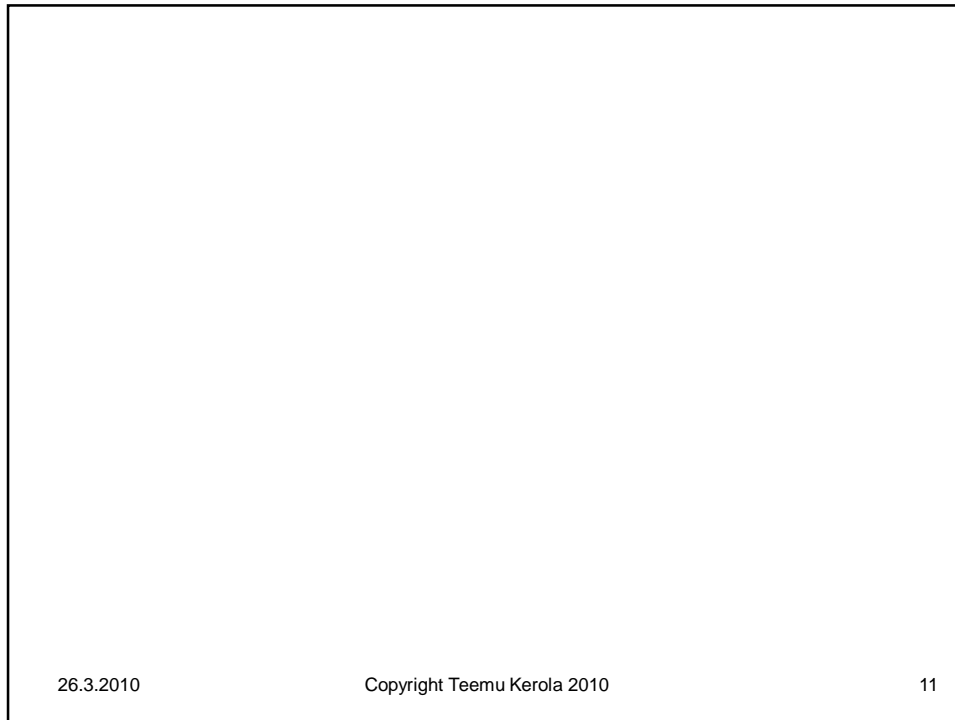
POP SP, R1
JNZER R1, FileTrouble
...

```

26.3.2010

Copyright Teemu Kerola 2010

10



26.3.2010

Copyright Teemu Kerola 2010

11

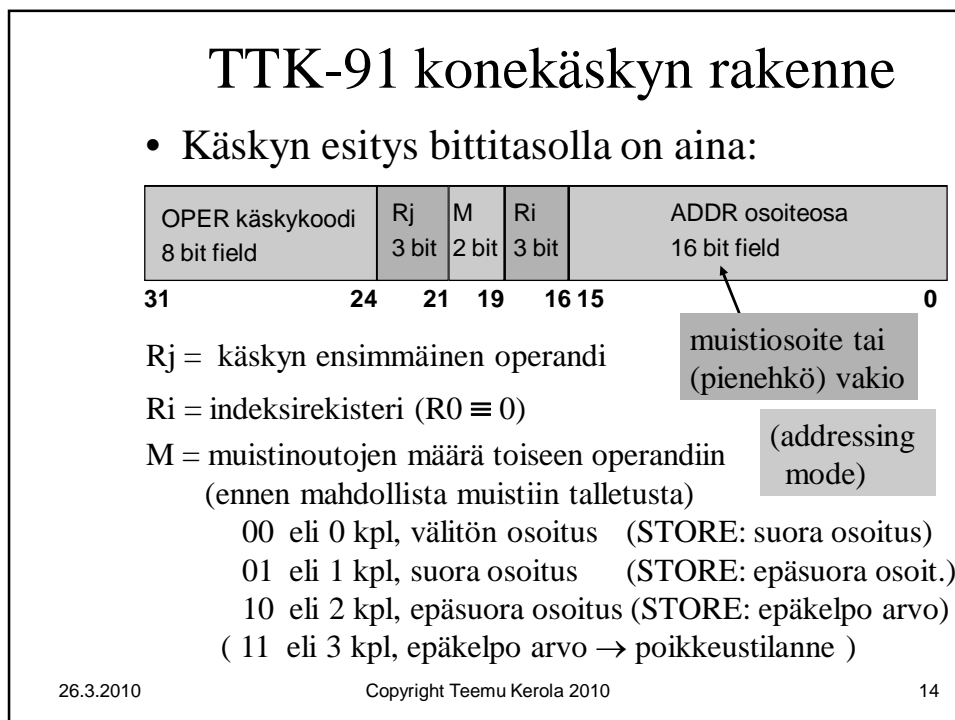
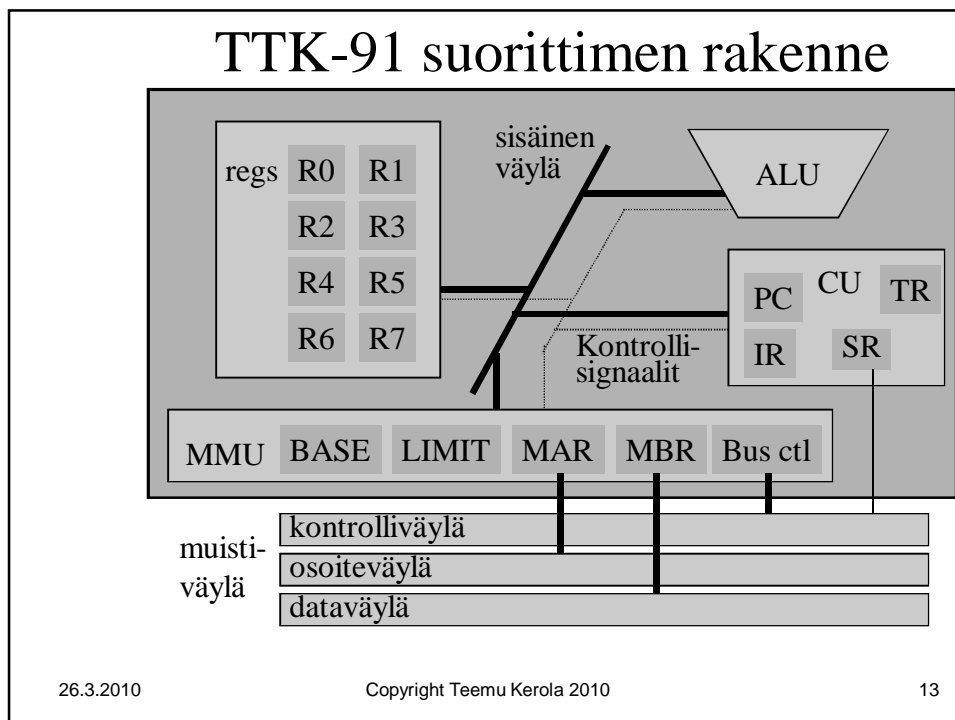
Luento 5
Suoritin ja väylä

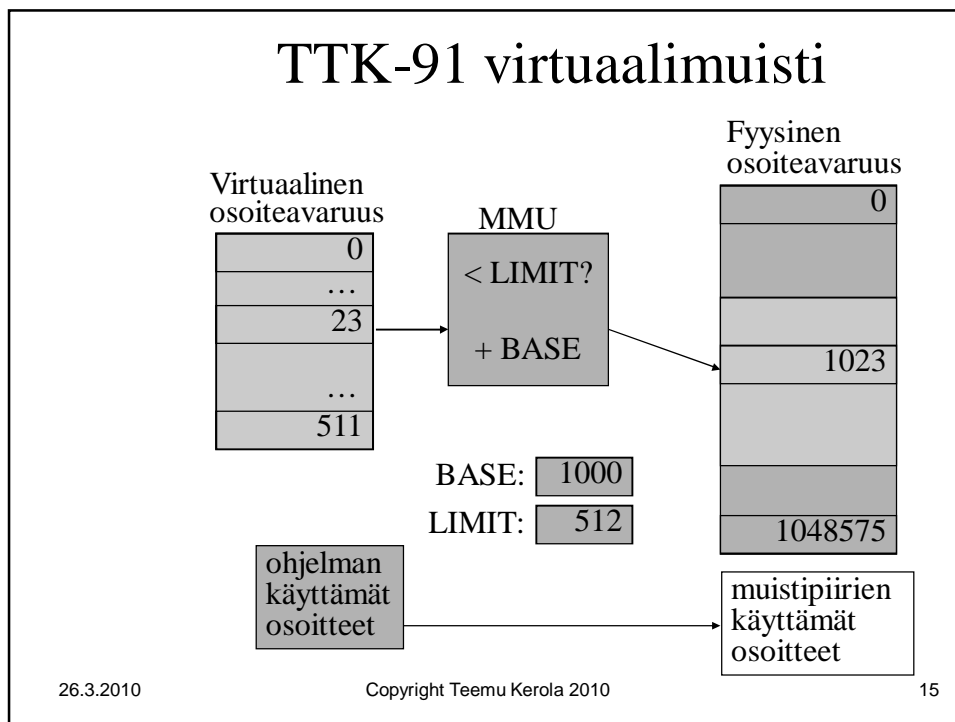
Suorittimen rakenne
Väylän rakenne
Käskyjen suoritussykli
Suorittimen tilat
Poikkeukset ja keskeytykset
TTK-91:n ja simulaattorin rakenne

26.3.2010

Copyright Teemu Kerola 2010

12





Virtuaalimuistin osoitteenmuunnos menetelmiä

- 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, ...

Lisää tietoa? → käyttöjärjestelmä kurssit

Lisää tietoa? → käyttöjärjestelmä kurssit

26.3.2010 Copyright Teemu Kerola 2010 16

Keskeytysten käsittely

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

26.3.2010

Copyright Teemu Kerola 2010

17

Keskeytyskäsittelijä

(kesk.)

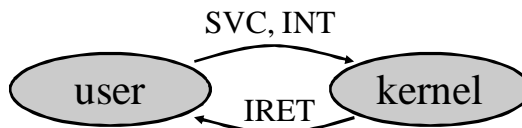
- Osa käyttäjärjestelmää
- Ennen keskeytyskäsittelijään hyppäämistä asetetaan suoritin ja MMU etuoikeutettuun käyttäjärjestelmätilaan `(supervisor state)`
 - SR:n bitti P on päällä => etuoikeutettu tila eli (P = Priviledged) käyttäjärjestelmä tila
 - käyttäjärjestelmätilassa saa viitata mihin tahansa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

26.3.2010

Copyright Teemu Kerola 2010

18

Suorittimen tilan muuttaminen



- Käyttäjätila → etuoikeutettu tila
 - keskeytys tai suora KJ:n palvelupyynnö (SVC käsky)
 - keskeytyskäsitteijä 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

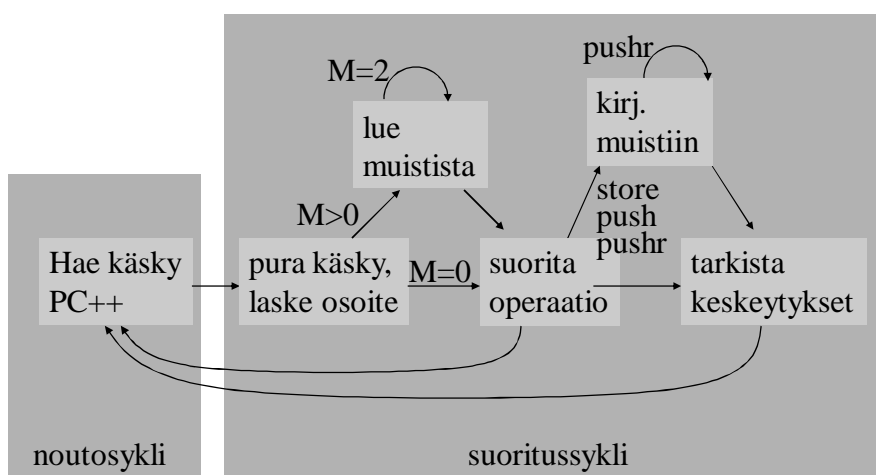
26.3.2010

Copyright Teemu Kerola 2010

19

TTK-91 Nouto- ja suoritussykli vielä vähän tarkemmin

(kesk.)

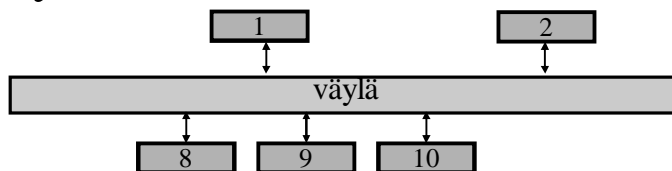


26.3.2010

Copyright Teemu Kerola 2010

20

Väylät



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

Lisää tietoa?



Tietokoneen rakenne -kurssi

26.3.2010

Copyright Teemu Kerola 2010

21

TTK-91 koneen simulaattori

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

Ks. Processor.java Tietokoneen koodissa

26.3.2010

Copyright Teemu Kerola 2010

22

TTK-91 käsken suoritusyksi

hae käsky simuloitusta muistista $IR = \text{mem}[PC]$
 pura käsky osiin (OPER, Rj, M, Ri, ADDR) ja
 laske osoiteosan arvo TR (ADDR tai $\text{regs}[Ri] + \text{ADDR}$)
 $ADDR = IR \bmod 32768$ $TR = \text{regs}[Ri] + \text{ADDR}$
 tee tarvittava määrä (M) operandin
 hakuja muistista rekisteriin TR $TR = \text{mem}[TR]$
 valitse aliohjelma operaatiokoodin (OPER) perusteella
 if (opcodeOK[OPER] = FALSE) then SR.U = 1;
 simuloi konekäskyn suorituksen muutokset
 rekistereihin (R0...R7, SR, PC, MAR, MBR)
 $ADD Rj, M ADDR(Ri) \Rightarrow \text{regs}[Rj] += TR;$
 lopeta suoritus jos SVC tai keskeytys $SR.O = \dots$

26.3.2010

Copyright Teemu Kerola 2010

23

26.3.2010

Copyright Teemu Kerola 2010

24

Luento 6

Tiedon esitysmuodot

Lukujärjestelmät
Kokonaisluvut
Liukuluvut
Merkit, merkkijonot
Totuusarvot
Kuvat, äänet, hajut(?)

26.3.2010

Copyright Teemu Kerola 2010

25

Tiedon tyypit

- Kommunikointi ihmisen kanssa
 - kuva, ääni, merkit, ...
- Laitteiston sisäinen talletus
 - kuvaformaattit, ääniformaatit, pakkausstandardit, ...
 - kokonaisluvut, liukuluvut, merkit, merkistöt
 - ohjelmat
- Suorittimen omana lajinaan ymmärtämät tyypit
 - on olemassa konekäskyjä tälle tietotyypille
 - kokonaisluvut
 - liukuluvut (useimmat suorittimet nykyään)
 - totuusarvot (jotkut suorittimet)
 - merkit (jotkut suorittimet)
 - konekäskyt

26.3.2010

Copyright Teemu Kerola 2010

26

Tiedon esitys

- Kysymys: miten esittää eri tyyppisiä tietoja?
- Vastaus: koodataan ne biteiksi
 - kaikki tieto on koneessa bitteinä
- Kaikelle käsitellylle tiedolle on omat koodausmenetelmänsä
 - kaikkia koodausmenetelmiä ei ole standardoitu
 - samalla tietotyypille voi olla useita koodausmenetelmiä
 - kokonaisluvut, liukuluvut, merkit, merkkijonot, kuvat, ...
 - ongelma: ymmärtävätkö koneet toisiaan?
 - tiedon esitysmuotoa voidaan joutua muuttamaan, kun tietoa siirretään koneelta toiselle

26.3.2010

Copyright Teemu Kerola 2010

27

Suorittimen ymmärtämä tieto

- Kaikki tieto koneessa on koodattuna biteiksi
- Muistissa voidaan esittää kaikki tieto millä tahansa sovitulla esitystavalla (koodauksella)
- Suoritin osaa tehdä operaatioita joillakin esitystavoilla koodatuille tiedoille
 - kokonaisluvut ja liukuluvut (lähes aina)
 - totuusarvot, merkit ja merkkijonot (joskus)
 - kuvat ja äänet (ei yleensä ellei erikoistunut suoritin)
 - hajut (ei vielä)
- Muiden tietojen käsittely tapahtuu ohjelmallisesti
 - esim. merkkejä (merkkien esitysmuotoa) voidaan käsitellä kokonaislukuoperaatioilla ja aliohjelmilla

TTK-91:
kokonaisluvut

26.3.2010

Copyright Teemu Kerola 2010

28

Big vs. Little Endian

- Miten monitavuiset arvot talletetaan?

Sanan osoite: 0x1200

0x1200 0x1201 0x1202 0x1203

talleta 0x11223344 ??

tavuosoitteet

Big-Endian: eniten merkittävä tavu pienimpään osoitteeseen

0x11 0x22 0x33 0x44

0x1200 0x1201 0x1202 0x1203

Little-Endian: vähiten merkittävä tavu pienimpään osoitteeseen

0x44 0x33 0x22 0x11

26.3.2010 Copyright Teemu Kerola 2010 29

Negatiiviset luvut

- Etumerkkibitti erikseen
- Yhden komplementtiresitys
- Kahden** komplementtiresitys
- Vakiolisäys
 - Esim lisää 127 ($=2^8 - 1$)
 - yleensä: $2^{\text{bittilkm}} - 1$
 - Talleta etumerkittömänä

arvo talletus

+57 = 0011 1001

sign bit = MSB = most significant bit

luku -57 = 1011 1001 talletusmuoto

-57 = 1100 0110

"sign" bit

-57 = 1100 0111

"sign" bit

+1

-57 = 0100 0110

-57 + 127 = 70

+57 = 1011 1000

+57 + 127 = 184

26.3.2010 Copyright Teemu Kerola 2010 30

IEEE 32-bit FP Standard (kesk.)

“+”
sign

“15”
exponent

“0.1875” = “0.0011”
mantissa or significand

$$\frac{1}{8} = 0.1250$$

$$\frac{1}{16} = \frac{0.0625}{0.1875}$$

• 23 bittiä mantissalle, siten että ...

1) Binääripiste (.) on heti ensimmäisen bitin jälkeen

2) Mantissa on normalisoitu: vasemmanpuolimmainen bitti on 1

3) Vasemmanpuolimmaista (eniten merkitsevä) bittiä (1) ei talleteta (implied bit, piilobitti)

mantissa eksponentti

0.0011 “15”

1.1000 “12”

1000 “12”

24 bitin mantissa!

26.3.2010
Copyright Teemu Kerola 2010
31

UCS ja Unicode

- UCS - Universal Character Set
- Samat merkit, eri standardit
- 2 tavua eli 16 bittiä per merkki
 - 65536 merkkiä koko maailmassa käytössä oleville n. 200000 symbolille
- Kontrollimerkit
 - 0x0000-001F and 0x0080-009F
 - 0x007F = DELETE, 0x0020 = SPACE
- UCS:ssä myös 8-bittiset koodi ”rivit”
 - eri alueille tai tarkoituksiin (zone) omat 8-bittiset koodinsa

26.3.2010
Copyright Teemu Kerola 2010
32

Merkkijonot

- Yleensä peräkkäin talletettu joukko tavuja
- Lisäksi tarvitsee jollain tavalla koodata merkkijonon pituus
 - laitetaan loppuun erikoismerkki, tai
 - C-kieli: `'\0'` = `0x00`
 - toteutetaan tietueena

20	"Ei yleensä nyt enää!"
----	------------------------

pituus merkkijono
 - ei omia konekäskyjä, manipulointi aliohjelmilla
 - kokonaisluku- ja bittimanipulointikäskyt
 - joissakin koneissa `"strcpy"` ja `"strcmp"` konekäskyt

26.3.2010

Copyright Teemu Kerola 2010

33

Maku, haju, tunto ja muu data

(kesk.)

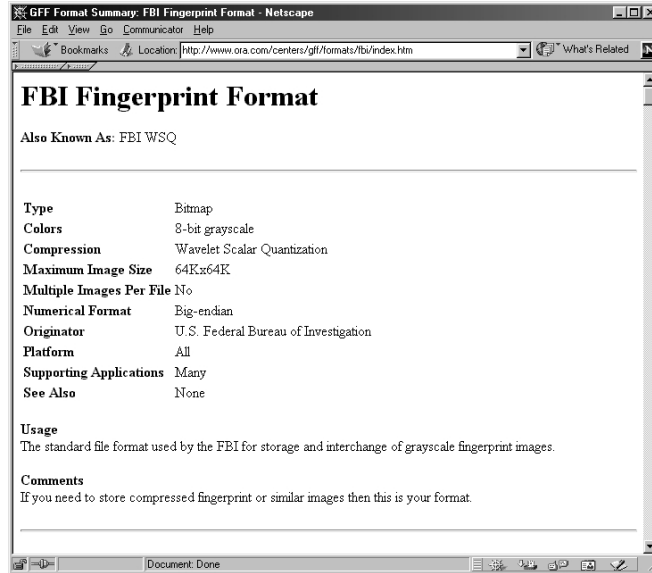
- Tähtien kirkkaus, hajut, veneen tyyppi, tunteen palo,
- Toteutus sovelluskohtaisesti, ei vielä yleisiä standardeja
 - kokonaisluvut (diskreetti data)
 - liukuluvut (jatkuva data)
- Ei omia konekäskyjä, manipulointi omilla aliohjelmilla

26.3.2010

Copyright Teemu Kerola 2010

34

-- Luennon 6 loppu --



26.3.2010

Copyright Teemu Kerola 2003

35