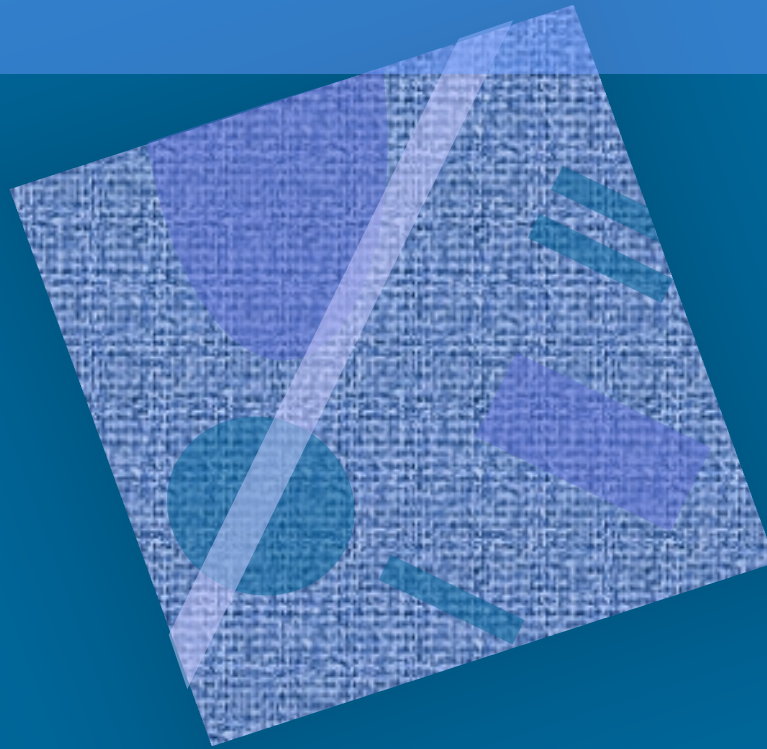


Tiedon esitysmuodot



Lukujärjestelmät
Kokonaisluvut, liukuluvut
Merkit, merkkijonot
Äänet, kuvat, muu tieto

Ohjelman esitysmuoto
Rakenteellinen tieto

Tiedon tyypit

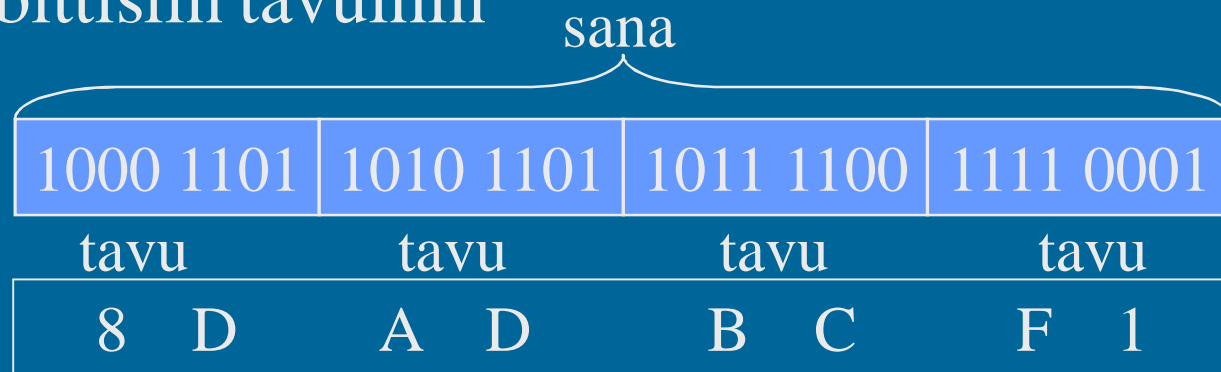
- Kommunikointi ihmisen kanssa
 - kuva, ääni, merkit, ...
- Laitteiston sisäinen talletus
 - kuvaformaattit, ääniformaatit, pakkausstandardit, ...
 - kokonaisluvut, liukuluvut, merkit, merkistöt
 - ohjelmat
- Suorittimen omana lajinaan ymmärtämät tyypit
 - on olemassa konekäskyjä tälle tietotyypille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

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
 - samalle tietotyypille voi olla useita koodausmenetelmiä
 - kokonaisluvut, liukuluvut, merkit, merkkijonot, kuvat, ...
 - ongelma: ymmärtävätkö koneet toisiaan?
 - tiedon esitysmuotoa voidaan joutua muuttamaan, kun tietoa siirretään koneelta toiselle

Tiedon esitys laitteistossa

- Kaikki tieto koneessa on binääribitteinä (0 tai 1)
 - binäärijärjestelmän numerot: 0, 1
 - helppo toteuttaa piireillä
 - helppo suunnitella logiikkaa Boolean algebran avulla
- Muisti jaettu tasapituisiin sanoihin (word)
 - sana = word = 32 bittiä (16 bittiä, 64 bittiä, ...)
- Yleensä sana on jaettu tasapituisiin 8-bittisiin tavuihin (byte)



Suorittimen ymmärtämä tieto

- Kaikki tieto koneessa on koodattuna biteiksi
- Muistissa voidaan esittää kaikki tieto millä tahansa sovitulla esitystavalla (koodauksella)
- Joillekin esitystavoille on omat konekäskyt (eli suoritin ymmärtää niitä)
 - Kokonaisluvut ja liukuluvut (lähes aina)
 - Totuusarvot, merkit ja merkkijonot (joskus)
 - Kuvat ja äänet (ei yleensä ellei erikoistunut suoritin)
- Muiden tietojen käsittely tapahtuu ohjelmallisesti (eli usea käsky tai aliohjelma tai metodi)
 - Esim. merkkejä (merkkien esitysmuotoa) voidaan käsitellä kokonaislukuoperaatioilla ja aliohjelmilla
 - Rationaaliluvut, 64-bittiset kokonaisluvut, isot taulukot, tietueet, oliot, sormenjäljet, äänet, kuvat, hajut, ...

TTK-91:
kokonaisluvut

Numeromuunnokset

- Binääriluvuista kymmenjärjestelmään

10110011 → 179

- Kymmenjärjestelmästä binääriluvuiksi

179 → 10110011

- Binääriluvusta heksadesimaaliluvuksi

10110011 → B3

- Heksadesimaaliluvusta binääriluvuksi

B3 → 10110011

Big vs. Little Endian

- Miten monitavuiset arvot talletetaan?



Negatiiviset kokonaisluvut

- Etumerkkibitti erikseen

arvo talletus
+57 = 0011 1001

- Yhden komplementtiesitys

sign bit = MSB
= most significant bit
luku -57 = 1011 1001 talletusmuoto

- **Kahden** komplementtiesitys

-57 = 1100 0110

“sign” bit

+1

-57 = 1100 0111

“sign” bit

- Vakiolisäys

– Esim. lisää 127 ($=2^7 - 1$)

• yleensä: $2^{\text{bittilkm}-1} - 1$

– Talleta etumerkittömänä

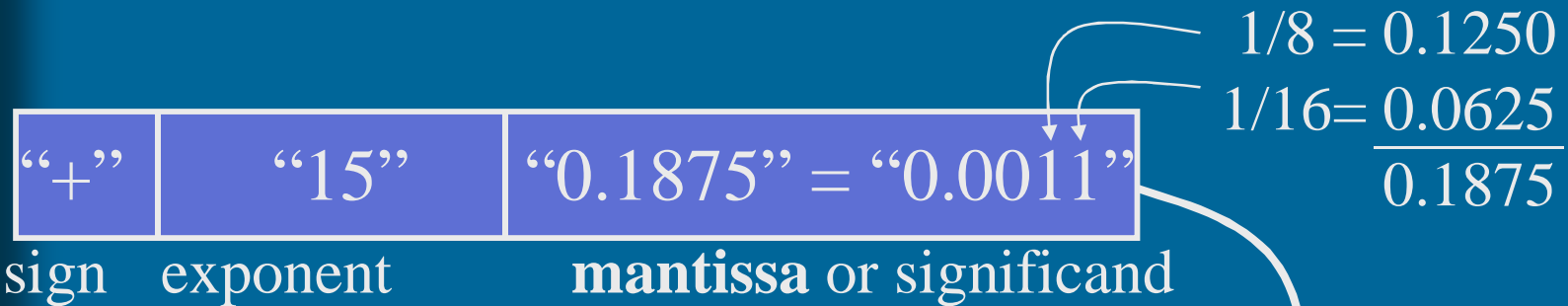
-57 = 0100 0110

-57 + 127 = 70

+57 = 1011 1000

+57 + 127 = 184

IEEE 32-bit FP Standard



- 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!

Miksi käytetään piilobittia?

UCS ja Unicode

- UCS - Universal Character Set
- Samat merkistöt, eri standardit
- 2 tavua eli 16 bittiä per merkki
 - 65536 merkkiä koko maailmassa käytössä oleville n. 200000 symbolille
- (32-bit UCS-4 sisältää myös kaikki kiinalaiset merkit)
- Kontrollimerkit
 - 0x0000-001F and 0x0080-009F
 - 0x007F = DELETE, 0x0020 = SPACE
- UCS:ssä myös 8-bittiset koodi ”rivit”
 - eri alueille tai tarkoitukseen (zone) omat 8-bittiset koodinsa, esim. UTF-8

Merkkijonot

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

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

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

Konekäskyjen esitysmuoto

- Konekohtainen, jokaisella omansa
- Käskyt ovat 1 tai useamman tavun mittaisia
 - SPARC, kaikki käskyt: 1 sana eli 4 tavua
 - ARM, kaikki käskyt: 1 sana eli 4 tavua
 - Pentium II: 1-16 tavua, paljon variaatioita
- Käskyillä on yksi tai useampi muoto, kussakin tietty määrä erilaisia kenttiä
 - opcode, Ri, Rj, Rk, osoitusmoodi
 - pitkä tai lyhyt vakio

TTK-91, kaikki käskyt: 1 sana, 1 muoto

Taulukkojen esitysmuoto

- Peräkkäisrakenteena, kuten esimerkit aikaisemmin
- Riveittäin tai sarakkeittain
- Ei omia konekäskyjä, manipulointi aliohjelmilla tai toistorakenteilla

Poikkeus: vektorisuorittimet, joilla

- vektorirekisterit (esim. 8 tai 64 liukulukua) tavallisten rekistereiden lisäksi
- Multimediakäskyt: $64b = 8 * 8b$ vektori (Intel MMX)
- omia konekäskyjä vektorioperaatioita varten
- Indeksoitu tiedonosoitusmoodi tukee 1-ulotteisten taulukoiden käyttöä

Tietueiden esitysmuoto

- Peräkkäisrakenteena
- Osoite on jonkin osoitinmuuttujan arvo
- Ei omia konekäskyjä, manipulointi aliohjelmilla tai kääntäjän generoimien vakiolisäysten avulla
- Indeksoitu tiedonosoitusmoodi tukee tietueiden käyttöä

Olioiden esitysmuoto

- Kuten tietueet, yleensä varattu keosta (heap)
- Useat olion kentistä sisältävät vuorostaan osoitteen keosta suoritusaikana varattuun toiseen olioon
- Metodit ovat aliohjelmien osoitteita
- Ei omia konekäskyjä, manipulointi aliohjelmilla

-- Luennon 5 loppu --

Intel 4004, 1971

- Faggin, Hoff, Mazor
- Ens. suoritin lastulla 3x4 mm, \$200
- 2300 transistoria
- 4 bitin sana
- Laskinta varten
- Sama laskentateho kuin Eniacilla (18000 tyhjiöputkea)

