Paavo Räisänen

Ohjelmoijan binaarialgebra ja heksaluvut

www.ohjelmoimaan.net

Sisällysluettelo:

1 Binaarijärjestelmä eli 2-järjestelmä

1.1: Positiiviset binaariluvut

Binaarijärjestelmä (voidaan ilmaista myös binäärijärjestelmä) on lukujärjestelmä, jonka kantaluku on 2, tosin sanoen numeron paikka kertoo mistä kakkosen potenssista on kysymys
vrt. 10-järjestelmä (desimaaliluvut), jossa numeron paikka kertoo, missä 10 potenssin kohdassa numero lukua on. 8 bittinen binaariluku voi olla esim. muotoa 10110010. Binaariluku muodostuu aina vain ykkösistä ja nollista. Tietokoneen prosessori ymmärtää lopulta vain binaarilukuja, joten binaarilukuja käytetään paljon erityisesti koneenläheisessä ohjelmoinnissa. Binääriluvutkin ovat lopulta vain merkintä osoittamassa sähkövarauksia. Itse tietokone ei käsittele numeroita vaan bitti tarkoittaa sähkövarausta, joka joko on varattu, eli 1, tai sähkövarausta ei ole, eli 0. Esimerkiksi tavussa, jossa on 8 bittiä, on 8 sähkövarausta, jotka voivat olla joko varattuja, tai varaamattomia.

Huomioitavaa on, että 8:lla bitillä (yksi tavu) voidaan ilmaista luvut 0 – 255, eli sillä voidaan osoittaa korkeintaan 256:een eri osoitteeseen.

Binaariluku saa painoarvot kahden potensseina oikealta vasemmalle, eli muodostuu lukujoukko:
$1=2^0$, $2=2^1$, $4=2^2$, $8=2^3$, $16=2^4$, $32=2^5$, $64=2^6$, $128=2^7$, $256=2^8$, $512=2^9$,$1024=2^{10}$, $2048=2^{11}$ ...jne.

2:n potenssi	***Painoarvot***
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024
11	2048
12	4096

2:n potenssi	*Painoarvot*
13	8192
14	16 384
15	32 768
16	65 536

Esimerkki. 1.1.1: Muunnetaan luku binaariluku 10011011 desimaaliluvuksi (kymmenjärjestelmän luvuksi).

Luetaan oikealta vasemmalle. Ykkosiä on yksi, samoin kakkosia, nelosia on nolla jne. eli alekkain lueteltuna:

1 : 2^7 *1 =128
0 : 2^6 *0 =0
0 : 2^5 *0 =0
1 : 2^4 *1 =16
1 : 2^3 *1 =8
0 : 2^2 *0 =0
1 : 2^1*1 =2
1 : 2^0 *1 =1

Lasketaan yhteen 1+2+0+8+16+0+0+128 = 155

Luku 10011011 on siis desimaalilukuna 155.

Esimerkki 1.1.2: Muunnetaan luku 1612 binaariluvuksi.

Desimaaliluku muutetaan binaariluvuksi sen painoarvojen, eli kahden potenssien apuna käyttäen seuraavasti:

 1612
<u>-1024</u>
= 588

eli 2:n potensseja 10 on 1.

Jatketaan laskutoimitusta jäljelle jäävällä luvulla 588.

 588
<u>-512</u>
= 76

eli 2:n potensseja 9 on 1. Jatketaan:

76
<u>-256</u>
menisi miinuksen puolelle, eli 2:n potensseja 8=0. Jatketaan:

76
<u>-128</u>
menisi miinuksen puolelle, eli 2:n potensseja 7=0.

76
<u>-64</u>
=12

2:n potensseja 6 on 1.
12
<u>-32</u>
miinuksen puolelle

2:n potensseja 5 on 0.

12
<u>-16</u>
miinuksen puolelle

2:n potensseja 4 on 0.

12
<u>- 8</u>
4

2:n potensseja 3 on 1.

4
<u>- 4</u>
0

eli 2:n potensseja 2 on 1;

lasku meni tasan, joten seuraavat kahden potenssit 1 (painoarvo 2) ja 0 (painoarvo 1=) ovat nollia.

Luku 1612 binäärilukuna on siis 11001001100.

Helpoiten binaari- , heksa- ja desimaalilukuja (eli kymmenjärjestelmän lukuja) muuntaa laskimen avulla, mutta joskus voi olla tarpeen osata muuntaa niitä myös

paperilla laskemalla. Binaari- ja heksalukujen toimintaperiaate täytyisi jokaisen vähän pidemmälle ehtineen ohjelmoijan osata.
Positiivisia binaarilukuja käsiteltäessä edessä olevia nollia ei tarvitse merkitä.
Bitti (b , esim. Mb megabittiä) on yksi nolla tai ykkönen, ja tavu (B, esim. MB megatavu, englanniksi byte, käytetään Suomessa myös merkintää t, eli esim. kt tai Mt) on 8 bittiä.
Binaariluvun perässä käytetään kirjoittaessa kirjainta B osoittamassa, että kyse on binaariluvussa. Siten merkinnät 100 (desimaaliluku) ja 100B (binaariluku) ovat eri asia.

Tietokonetekniikassa etuliitteet kilo, mega, giga, tera viittaavat tavujen määrään.
Tietokoneteknikassa 1 kt (kilotavu, kB, tai kt) on $\mathbf{2^{10}}$ eli 1024 B. Vastaavasti:
MB tai Mt= $\mathbf{10^{20}}$ tavua, eli 1048576 tavua (tuhat kilotavua).
GB tai Gt = $\mathbf{10^{30}}$ tavua, eli miljoona kilotavua.
TB tai Tt= $\mathbf{10^{40}}$ tavua, eli miljardi kilotavua.

Modeemien ja verkkokorttien nopeudet ilmoitetaan yleensä bitteinä sekunnissa (kb/s tai Mb/s).

Bittijonoja käsiteltäessä luvun oikeanpuoleisinta bittiä kutsutaan vähiten merkitseväksi bitiksi, eli lyhenteellä LSB. Vastaavasti vasemmanpuoleisinta bittiä kutsutaan lyhenteellä MSB, eniten merkitsevä bitti.
Pitkiä bittijonoja käsiteltäessä bitit ryhmitellään oikealta alkaen neljän ryhmiin.
Esim. 10 1101 1001 0011
Binaariluvun voi kertoa kahdella lisäämällä oikealle (LSB päähän) nollan.

1.2: Negatiiviset binaariluvut komplementtilukuina

Binaariluvuissa ei ole erikseen + ja – merkkiä. Joskus harvoin käytetään jotain bittiä merkkibittinä osoittamassa, onko luku positiivinen vai negatiivinen. Tällaisilla laskeminen on kuitenkin hyvin vaikeaa. Niinpä yleensä käytetään komplementtilukuja.
Kun yksi tavu voi sisältää positivisena desimaalilukuna arvot 0...+255, voi komplementtinen tavu sisältää arvot -128...0...+127. Luku ikäänkuin pyörähtää ympäri. Jos arvoon +127 lisätään 1, arvoksi tulee -128.
Komplemennitlukujen perässä ei käytetä merkintä B, vaan on erikseen kerrottava, että käytetään 2:n komplementtilukua. Komplemettiluvuilla voi esittää sekä positiivisia, että negatiivisia lukuja. Sama numerosarja voi tarkoittaa eri lukua binaari- tai komplementtilukuna.

Binaarilukuja voi muuttaa negatiiviksi komplementtiluvuiksi muuttamalla nollat ykkösiksi ja ykköset nolliksi, ja lisäämällä lukuun yksi.

Esimerkki 1.2.1: kahdeksanbittisten binaarilukujen muuttaminen komplementtiluvuiksi.

des. +1   00000001   kompl: 11111111 (11111110 +1) des. -3
des. +2   00000010   kompl. 11111110 (11111101 +1) des. -2
des +3   00000011   kompl. 11111101 (11111100 +1) des. -3
nolla on kummassakin sama 00000000

Muutettaessa toisinpäin tehdään täysin samoin. Muutetaan nollat ykkosiksi, ja lisätään yksi.

Jos käytetään MSB bittiä komplementtiluvuilla, sen arvo 1 kertoo, että kyseessä on negatiivinen arvo, ja 0 tarkoittaa puolestaan positiivista.

Jos tiedetään, että kyse on komplementtiluvusta, luvun edessä olevia ykkösiä tai nollia voidaan poistaa, kunhan vähintään yksi ykkönen tai nolla jätetään.

Kun käytetään 2:n koplementtilukuja, on aina tiedettävä etukäteen kuinka monta bittiä luvussa on, jotta tietää lukualueen.
2:n komplementtiluku on helpoin muodostaa esim. desimaaliluvusta tekemällä ensin positiivinen vastaava binaariluku, ja muuttamalla se 2.n komplementtiluvuksi (esim. 2 , -2). Negatiivisesta 2:n komplementtiluvusta (”1”:llä alkavasta 2:n komplementtiluvusta) ei voi suoraan päätellä sen lukuarvoa. Helpoin keino tutkia arvo on muuttaa se ensin vastaavaksi positiiviseksi luvuksi.

2 Binaarilukujen loogiset operaatiot

2.1: Yleistä

Binaariluvuille käytetään paljon erilaisia bittioperaatioita. Loogisia operaatioita (AND, OR) käytetään paljon ehtolauseissa kaikessa ohjelmoinnissa. Tässä oppaassa käsittelen, miten loogisilla operaatioilla operoidaan binaarilukuja. Menetelmiä voi joutua käyttämään esim. tietokonegrafiikassa. Loogisten operaatioiden (propositiologiikka, lauselogiikka, Boolen logiikka) matemaattisena esitystapana voidaan käyttää Boolen algebraa.

2.2: AND operaatio

AND operaatio tarkoittaa, että kummankin ehdon täytyy toteutua, eli olla yksi. Jos arvo on eri, tai nolla ei toteudu.

Esimerkki 2.2.1:

```
      11010101
AND  01110011
  =  01010001
```

2.3: OR operaatio

OR operaatio tarkoittaa, että kumpikin, tai jompikumpi ehdoista toteutuu, eli on yksi.

Esimerkki 2.3.1:

```
     10100110
OR   01101011
  =  11101111
```

2.4: XOR operaatio

XOR toteutuu, jos kumpikin verrattava bitti ovat erisuuria, eli toinen on nolla, ja toinen on yksi.

Esimerkki 2.4.1:

```
      11010010
XOR  10110110
  =  01100100
```

2.5: NOT operaatio

Kääntää binaariluvusta nollat ykkösiksi ja ykköset nolliksi.

Operoitava luku: 11010110
tulos NOT operaation jälkeen: 00101001

3 Heksaluvut

3.1: Yleistä

Heksalukuja käytetään paljon varsinkin koneenläheisessä ohjelmoinnissa, koska niitä on helppo muuttaa binaariluvuiksi, ja päinvastoin. Heksajärjestelmän kantaluku on 16, joka on 2:n potenssi
eli $\mathbf{2^4}$.
Heksajärjestelmää voidaan pitää binaarijärjestelmän lyhennysmuotona. Jokainen heksanumero voidaan esittää neljällä binaariluvun numerolla. Kun merkistö on 16 numeroinen, tarvitaan apuna kuutta kirjainta. Heksajärjestelmän numerot ovat 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F. Heksaluvuissa käytetään kirjoitettessa perässä kirjainta H, esim. 6EH tai 12H.

Taulukko heksa-, binaari ja desimaalilukujen vastaavuudesta

Heksaluku	***Binaariluku***	***Desimaaliluku***
0	*0000*	*0*
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15
10	10000	16

3.2: Heksaluvut binaariluvuiksi

Heksaluvut pystyy muuttamaan binaariluvuiksi korvaamalla numero binaarinumerosarjalla.

Esimerkki 3.1.1:

2EH = numeroa 2 vastaa binaarinumerosarja 0010, numeroa E vastaa 1110, joten binaariluku on
0010 1110.

3.3: Binaariluku heksaluvuksi

Binaariluvun saa helposti heksaluvuksi jakamalla lukun oikealta lukien neljännumeron sarjoihin, ja muuttamalla luvut sitten heksaluvuiksi taulukon avulla (tai laskemalla).

Esimerkki 3.2:

Binaariluku: 101 0010 . Jaetaan osiin: 101 on sama kuin 0101 eli heksanumero 5. 0010 on heksanumerona 2, eli koko luku on 52H.

Esimerkki 3.3:

Binaariluku 11 1101 0011. Jaetaan osiin: 11 on sama kuin 0011, eli heksanumerona 3 (muistathan: etunollia saa binaarilukuun lisätä ja poistaa).
1101 on heksana D ja 0011 on heksana 3, eli koko luku on 3D3H.

3.4: Heksaluku desimaaliluvuksi ja päinvastoin

Heksaluvut saa muutettua helposti desimaaliluvuksi (kymmenjärjestelmäksi) samalla tekniikalla kuin binaarilukujen muutos kerrotaan esimerkissä 1.1.1.

Esimerkki 3.4.1: Heksaluku desimaaliluvuksi

Heksaluku 14E6H = $1*16^3 + 4*16^2 + 14*16^1 + 6*16^0 = 1*4096 + 4*256 + 14*16 + 6*1 = 4096 + 1024 + 224 + 6 = 5350$

eli heksaluku 14E6H on desimaalilukuna 5350.

Desimaaliluvun saa muutettua heksaluvuksi vastaavasti kuin binaariluvuksi esimerkissä 1.1.2.

Esimerkki 3.4.2: Desimaaliluku heksaluvuksi

Desimaaliluku 4868. Painoarvot 4096, 256, 16, 1.

Laskettaessa 4868 – 4096 = 772 , (menee yhden kerran jakolaskussa 4368/ 4096)
772 / 256 = 3 kokonaista. Jakojäännös (772-3*256) = 4
4 / 16 laskun tuloksena ei yhtään kokonaista
4 / 1 = 4

eli heksaluku on 1304H.

Edellinen lasku taulukossa esiteltynä (desimaaliluku 4868):

Painoarvot	*Laskutoimitus*	*Kerroin*	*Kerroin heksana*	*Jakojäännös*
4096	4868 / 4096	1	1	772
256	772 / 256	3	3	4
16	4 / 16	0	0	4
1	4 / 1	4	4	0

Saadaan sama luku 1304H.

Esimerkki 3.4.3: Desimaaliluku heksaluvuksi

Desimaaliluku 2607 . Painoarvot 256, 16, 1.

Laskettaessa 2607 / 256 menee 10 kokonaista, eli numero A.
Jakojäännös 2607 - 10*256 = 47
47 / 16 menee 2 kokonaista, jakojäännös 15
15 / 1 = 1, eli viimeinen numero F (des. 15)

eli heksaluku on A2FH.

Edellinen lasku taulukossa esiteltynä (desimaaliluku 2607).

Painoarvo	*Laskutoimitus*	*Kerroin*	*Kerroin heksana*	*Jakojäännös*
256	2607 / 256	10	A	47
16	47 / 16	2	2	15
1	15 / 1	15	F	0

Eli tulos sama A2FH.

Painoarvoja, eli 16:sta potensseja: 1, 16, 256, 4096, 65536, 1 048 576, 16 777 216.

16:sta potenssi	***Painoarvo***
0	1
1	16
2	256
3	4096
4	65536
5	1 048 576
6	16 777 216

4 Lähteet

Tietotekniikka osa 1: Digitaalitekniikka osa A: Pekka Rantala
MSX-assembler ja -konekieli: Ian Sinclair
Flash 8 & ActionScript: Pasi Manninen, Jarno Marttila
Wikipedia: http://fi.wikipedia.org/wiki/Boolen_algebra
Wikipedia: http://fi.wikipedia.org/wiki/Lausekalkyyli