

Arvude esitusviisid



1. loeng,
15. veebruar 2008

©Toomas Plank, 2008

Sissejuhatus

- Kümnersüsteem
- Kahendsüsteem
- Kaheksandsüsteem
- Kuueteistkümnersüsteem

Arvuti arhitektuur FKEF.02.143



Kümnersüsteemi arvud *decimal number*

- Olgu meil n -kohaline vektor
 $D = d_{n-1} \dots d_2 d_1 d_0$,
 kus d_i on täisarv vahemikus $0 \dots 9$ ($0 \leq i \leq n-1$)
- Selle arvu võime kirjutada kujul
 $d_{n-1} \times 10^{n-1} + \dots + d_2 \times 10^2 + d_1 \times 10^1 + d_0 \times 10^0$
- Ta on vahemikus $0 \dots 10^n - 1$
- Näide $328\,245 =$
 $3 \times 10^5 + 2 \times 10^4 + 8 \times 10^3 + 2 \times 10^2 + 4 \times 10^1 + 5 \times 10^0$
- Näide $328,245 =$
 $3 \times 10^2 + 2 \times 10^1 + 8 \times 10^0 + 2 \times 10^{-1} + 4 \times 10^{-2} + 5 \times 10^{-3}$

Arvuti arhitektuur FKEF.02.143



Kahendsüsteemi arvud *binary number*

- Olgu meil n -bitine vektor
 $B = b_{n-1} \dots b_2 b_1 b_0$,
kus b_i on kas 0 või 1 ($0 \leq i \leq n-1$)
- Selle arvu võime kirjutada kümnendsüsteemis kujul
 $b_{n-1} \times 2^{n-1} + \dots + b_2 \times 2^2 + b_1 \times 2^1 + b_0 \times 2^0$
- Ta on vahemikus 0 ... $2^n - 1$
- Näide 101 101 =
 $1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 45$
- Näide 101,001 =
 $1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} = 5,125$

Arvuti arhitektuur FKFE.02.143

Kaheksandsüsteemi arvud *octal number*

- Olgu meil n -kohaline vektor
 $D = d_{n-1} \dots d_2 d_1 d_0$,
kus d_i on täisarv vahemikus 0 ... 7 ($0 \leq i \leq n-1$)
- Selle arvu võime kirjutada kujul
 $d_{n-1} \times 8^{n-1} + \dots + d_2 \times 8^2 + d_1 \times 8^1 + d_0 \times 8^0$
- Ta on vahemikus 0 ... $8^n - 1$
- Näide 327 245 =
 $3 \times 8^5 + 2 \times 8^4 + 7 \times 8^3 + 2 \times 8^2 + 4 \times 8^1 + 5 \times 8^0 = 110\,245$
- Näide 327,245 =
 $3 \times 8^2 + 2 \times 8^1 + 7 \times 8^0 + 2 \times 8^{-1} + 4 \times 8^{-2} + 5 \times 8^{-3} =$
215,322265625

Arvuti arhitektuur FKFE.02.143

Kuueteistkümnendsüsteemi arvud (*hexadecimal number*)

- Olgu meil n -kohaline vektor
 $H = h_{n-1} \dots h_2 h_1 h_0$,
kus h_i on arv vahemikus 0 ... F ($0 \leq i \leq n-1$)
- Selle arvu võime kirjutada kujul
 $h_{n-1} \times 16^{n-1} + \dots + h_2 \times 16^2 + h_1 \times 16^1 + h_0 \times 16^0$
- Ta on vahemikus 0 ... $16^n - 1$
- Näide 3B 82F5 =
 $3 \times 16^5 + 11 \times 16^4 + 8 \times 16^3 + 2 \times 16^2 + 15 \times 16^1 + 5 \times 16^0 =$
3 900 149
- Näide 3B8,2F5 =
 $3 \times 16^2 + 11 \times 16^1 + 8 \times 16^0 + 2 \times 16^{-1} + 15 \times 16^{-2} + 5 \times 16^{-3} =$
952,184814453125

Arvuti arhitektuur FKFE.02.143

Võrdlustabel

DEC	BIN	OCT	HEX
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	6
7	0111	07	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Arvuti arhitektuur FKEE.02.143

Teisendus kümnendsüsteemist kahendsüsteemi

Täisosa

- Jaga kümnendsüsteemi arvu täisosa 2-ga
- Kui
 - jagus täpselt, kirjuta paremalt järgmiseks bitiks 0
 - tekis murdosa, kirjuta paremalt järgmiseks bitiks 1
- Jagatise täisosa korda samme 1 ja 2 kuni jagatise täisosa on 0

Murdosa

- Korruta kümnendsüsteemi arvu murdosa 2-ga
- Kui tulemus
 - <1, kirjuta vasakult järgmiseks bitiks 0
 - > 1, kirjuta vasakult järgmiseks bitiks 1 ja lahuta jäägist 1
- Tulemiga korda samme 1 ja 2 kuni korrutise murdosaosa on 0

Arvuti arhitektuur FKEE.02.143

Näide

Teisendame arvu 201,24 täisosa:

- $201 / 2 = 100 + \frac{1}{2} \rightarrow 1$
- $100 / 2 = 50 \rightarrow 0$
- $50 / 2 = 25 \rightarrow 0$
- $25 / 2 = 12 + \frac{1}{2} \rightarrow 1$
- $12 / 2 = 6 \rightarrow 0$
- $6 / 2 = 3 \rightarrow 0$
- $3 / 2 = 1 + \frac{1}{2} \rightarrow 1$
- $1 / 2 = 0 + \frac{1}{2} \rightarrow 1$

Saime täisosaks 1100 1001

Arvuti arhitektuur FKEE.02.143

Näide

Teisendame arvu 201,24 murdosa:

- $0,24 \times 2 = 0,48$ à 0
- $0,48 \times 2 = 0,96$ à 0
- $0,96 \times 2 = 1,92$ à 1
- $0,92 \times 2 = 1,84$ à 1
- $0,84 \times 2 = 1,68$ à 1
- $0,68 \times 2 = 1,36$ à 1
- $0,36 \times 2 = 0,72$ à 0
- $0,72 \times 2 = 1,44$ à 1
- $0,44 \times 2 = 0,88$ à 0
- $0,88 \times 2 = 1,76$ à 1
- $0,76 \times 2 = 1,52$ à 1
- $0,52 \times 2 = 1,04$ à 1

Saime murdosaks 0011 1101 0111 ...
 Seega kogu arv (1100 1001, 0011 1101 01 ...) ₂
 Kuueteistkümnendsüsteemis (C9,3D7 ...) ₁₆



Arvuti arhitektuur FKFE.02.143

Teisendus kümnendsüsteemist kuueteistkümnendsüsteemi

Täisososa

1. Jaga kümnendsüsteemi arvu täisososa 16-ga
2. Kui
 - jagus täpselt, kirjuta paremalt järgmiseks arvuks 0
 - tekkis murdosa, kirjuta paremalt järgmiseks arvuks jääk
3. Jagatise täisosaga korda samme 1 ja 2 kuni jagatise täisososa on 0

Murdosa

1. Korruta kümnendsüsteemi arvu murdosa 16-ga
2. Lahuta tulemusest korruptise täisososa ...
3. ... ja kirjuta see vastusesse vasakult järgmiseks arvaks
4. Tulemiga korda samme 1 ... 3 kuni korruptise murdosaosa on 0



Arvuti arhitektuur FKFE.02.143

Märgiga kahendarvud

- Kasutatakse nelja süsteemi
 - Märgi ja väärtuse süsteem (*sign-and-magnitude*)
 - Ühe täiend (*1's-complement*)
 - Kahe täiend (*2's-complement*)
 - Nihkega süsteem (*Excess or biased representation*)
- Kolmes esimeses süsteemis on kõige vasakpoolsem bit
 - 0 positiivsete arvude korral
 - 1 negatiivsete arvude korral
- Positiivsed arvud on kolmes esimeses süsteemis ühtemoodi kirja pandud
- Negatiivsed arvud on eri süsteemides erinevad



Arvuti arhitektuur FKFE.02.143

Märgi ja väärtuse süsteem *sign-and-magnitude*

- Positiivne ja negatiivne arv erinevad ainult kõige vasakpoolsema biti poolest
 - +5 à 0101
 - -5 à 1101

Arvuti arhitektuur FKFE.02.143



Ühe täiend *1's-complement*

- Negatiivne arv saadakse positiivse arvu kõigi bittide invertteerimisel
 - +5 à 0101
 - -5 à 1010
- Negatiivse arvu saame, kui lahutame vastava positiivse arvu arvust $2^n - 1$, kus n on bittide arv

Arvuti arhitektuur FKFE.02.143



Kahe täiend *2's-complement*

- Negatiivne arv saadakse positiivse arvu kõigi bittide invertteerimisel ja saadud tulemusele ühe liitmisel
 - +5 à 0101
 - -5 à $1010 + 1$ à 1011
- Negatiivse arvu saame, kui lahutame vastava positiivse arvu arvust 2^n , kus n on bittide arv

Arvuti arhitektuur FKFE.02.143



Nihkega esitus Excess or biased representation

- Siin nihutatakse kahe täiendina saadud arvu etteantud väärtuse võrra
- Tulemusena saame kõige väiksema arvu esituseks 0000
- Ülejäänud arvud on kasvavas järjestuses
- Näide: nihkega 8
 - +5 à 5 + 8 à 13 à 1101
 - 5 à -5 + 8 à 3 à 0011
 - 8 à -8 + 8 à 0 à 0000

Arvuti arhitektuur FKFF.02.143

Märgiga kahendarvud

BIN	Märk + väärtus	1 täiend	2 täiend	Nihkega 8
0111	+7	+7	+7	-1
0110	+6	+6	+6	-2
0101	+5	+5	+5	-3
0100	+4	+4	+4	-4
0011	+3	+3	+3	-5
0010	+2	+2	+2	-6
0001	+1	+1	+1	-7
0000	+0	+0	+0	-8
1000	-0	-7	-8	0
1001	-1	-6	-7	1
1010	-2	-5	-6	2
1011	-3	-4	-5	3
1100	-4	-3	-4	4
1101	-5	-2	-3	5
1110	-6	-1	-2	6
1111	-7	-0	-1	7

Arvuti arhitektuur FKFF.02.143

Positiivsete kahendarvude liitmine

	0
+	0
	0

	0
+	1
	1

	1
+	0
	1

Summa on 0
Ülekanne on 1

	1
+	1
	1 0

Arvuti arhitektuur FKFF.02.143

Positiivsete kahendarvude liitmine

	0000000110	6
+	0000000100	4
	0000001010	10
	0000101101	45
+	0001011011	91
	0010001000	136
	0100101100	300
+	0011010011	211
	0111111111	511

Arvuti arhitektuur FKFF.02.143

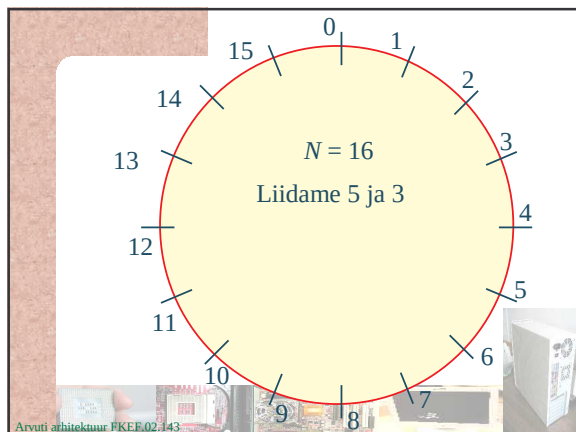
Märgiga kahendarvude liitmine-lahutamine

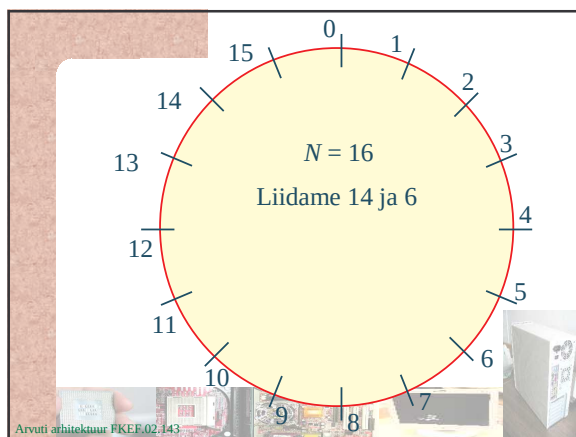
Kirjeldasime kolme süsteemi

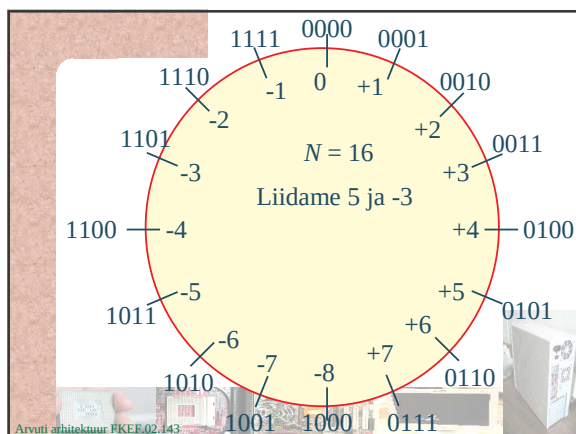
- Märgi ja väärtuse süsteem (*sign-and-magnitude*)
 - Kõige lihtsamini kirjeldatav
 - Kõige keerulisem arvutada
- Ühe täiend (*1's-complement*)
- Kahe täiend (*2's-complement*)
 - Kõige keerulisemalt kirjeldatav
 - Kõige lihtsam arvutada
 - Arvutamise aluseks mod N

Arvuti arhitektuur FKFF.02.143

Arvuti arhitektuur FKFF.02.143







Näited

	0000000101	5
+	1111111101	-3
1	0000000010	2
	0000110010	50
+	1110001000	-120
	1110111010	-70

Arvuti arhitektuur FKFF.02.143

Resümee

- Kahe arvu **liitmisel** tuleb liita nende n-bitised versioonid ja ignoreerida kõige suurema biti (*most significant bit*) ülekande signaali
 - Vastus on korrektne, kui ta on vahemikus $-2^{n-1} \dots 2^{n-1} - 1$
- Ühe arvu **lahutamisel** teisest tuleb
 - leida lahutatava arvu 2-täiend
 - liita nüüd nende arvude n-bitised versioonid ja ignoreerida kõige suurema biti (*most significant bit*) ülekande signaali
 - Vastus on korrektne, kui ta on vahemikus $-2^{n-1} \dots 2^{n-1} - 1$

Arvuti arhitektuur FKFF.02.143

Bittide arvu suurendamine

- Kuidas teisendada 2-täiend arve väiksema bittide arvuga arvust suurema bittide arvuga arvuks ilma väärtust muutmata?
 - +5 à 0101 à 0000 0101
 - -5 à 1011 à 1111 1011
- ehk siis tuleb märgi bitti kopeerida vasakule niimitu korda kui tarvis (et saada soovitud bittide arvuga arvu)
- Inglise keeles nimetatakse seda tegevust *sign extension*

Arvuti arhitektuur FKFF.02.143

Liitmine lahutamine 1-täiend arvude korral

- Siin on liitmine-lahutamine veidi komplitseeritud – ülekande signaali ei saa ignoreerida
- Vastusele tuleb liita ülekande signaali väärtus, siis on vastus õige

Arvuti arhitektuur FKFE.02.143



Ületäitumine

- Vastus on korrektne, kui ta on vahemikus $-2^{n-1} \dots 2^{n-1} - 1$
- Kui $n = 4$, siis on see vahemik $-8 \dots +7$

	0110	6
+	0100	4
0	1010	-6
	1010	-6
+	1100	-4
1	0110	6

Arvuti arhitektuur FKFE.02.143

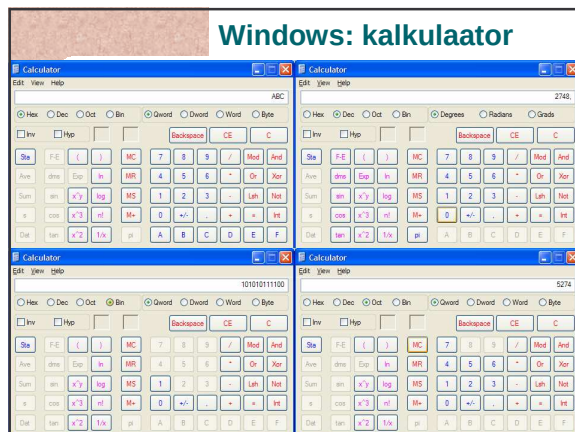


Ületäitumine

- Ületäitumine saab tekkida ainult samamärgiliste arvude liitmisel
- Ülekande signaali väärtus pole piisav kriteerium ületäitumise avastamiseks
- **Korrektne kriteerium:** kui liidetavad on samamärgilised, siis ületäitumine leiab aset siis ja ainult siis kui vastuse märk erineb liidetavate märgist

Arvuti arhitektuur FKFE.02.143





Windows: kalkulaator

Kasutatud kirjandus

- Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer organization 5th edition (2002) 805 p.
- M J Murdocca, V P Heuring, Computer Architecture and Organization: An Integrated Approach (2007) 544 p.



Arvuti arhitektuur FKEE.02.143
