DIGITAALTEHNIKA

1. Arvusüsteemid.

1.1. Kümnendsüsteem.

Sümbolite arv e süsteemi alus p=10. Sümbolid on 0;1;2;3;4;5;6;7;8;9. Iga number paikneb arvus kindlal positsioonil e kindlas järgus. Järkude kaalud vasakul pool koma on 10^0;10^1;10^2 ... ja paremal pool koma 10^{-1};10^{-2};10^{-3}...

nt $593{,}74_{10} = \mathit{3}*\mathbf{10^0}+\mathit{9}*\mathbf{10^1}+\mathit{5}*\mathbf{10^2}+\mathit{7}*\mathbf{10^{-1}}+\mathit{4}*\mathbf{10^{-2}}$

Tumedalt trükitud – järkude kaalud.
Kaldkirjas – kaalude arvulised kordajad.

1.2. Kahendsüsteem.

Sümbolite arv e süsteemi alus p=2. Sümbolid on 0 ja 1. Järkude kaalud vasakul pool koma on 2^0;2^1;2^2;2^3 ... ja paremal pool koma 2^{-1};2^{-2};2^{-3} ...

nt $10011{,}001_2 = 1*2^0+1*2^1+0*2$

1.3. Kaheksandsüsteem.

Sümbolite arv e süsteemi alus p=8. Sümbolid on 0;1;2;3;4;5;6;7. Järkude kaalud vasakul pool koma on 8^0;8^1;8^2 ... ning paremal pool koma 8^{-1};8^{-2};8^{-3} ...

nt $253{,}12_8 = 3*8^0+5*8^1+2*8^2+1*8^{-1}+2*8^{-2} = 3 + 40 + 128 + 1/8 + 2/64 = 171\ 5/32 = 171{,}15625$

1.4. Kuueteistkümnendsüsteem.

Sümbolite arv e süsteemi alus p=16. Sümbolid on 0;1;2;3;4;5;6;7;8;9;A_{10};B_{11};C_{12};D_{13};E_{14};F_{15}.

nt $2A9{,}EC3_{16} = (9*16^0+10*16^1+2*16^2+14*16^{-1}+12*16^{-2}+3*16^{-3})_{10}$

1.5. Kahendkodeeritud kümnendsüsteem 8421 (BCD – Binary Coded Decimal).

8	4	2	1
2^3	2^2	2^1	2^0

Mitmekohaline arv kodeeritakse kümnendkoodis kuid iga selle number esitatakse kahendkoodis.

nt $9371{,}654_{10}$ = 1001 0011 0111 0001,0110 0101 0100_{8421}

1.6. Kahendkodeeritud kümnendsüsteemid 2421 ja liiaga 3.

	8421	2421	Liiaga 3
0	0000	0000	0011
1	0001	0001	0100
2	0010	0010	0101
3	0011	0011	0110
4	0100	0100	0111
5	0101	1011	1000
6	0110	1100	1001
7	0111	1101	1010
8	1000	1110	1011
9	1001	1111	1110

Koodide 2421 ja liiaga 3 teineteist 9-ni täiendavate arvude (0 ja 9;1 ja 8;2 ja 7 jne) koodid on teineteise inversioonid.<

1.7. Arvu teisendamine kaheksandsüsteemist kahendsüsteemi.

Iga number tuleb kirjutada kolmejärgulise kahendarvuna.

nt $523{,}41_8 = \underline{101}010\underline{011}{,}100\underline{001}_2$

1.8. Arvu teisendamine kahendsüsteemist kaheksandsüsteemist.

nt 010 111 011 100,011 110 = $2734{,}36_8$

1.9. Arvu teisendamine kuuteistkümnendsüsteemist kahendsüsteemi.

Igale arvujärgule vastab kahendsüsteemis neli järku.

nt $B5D{,}E2C_{16}$ = 1011 0101 1101,1110 0010 1100_2

1.10.Arvu teisendamine kahendsüsteemist kuueteistkümnendsüsteemi.

nt 0011 1110 1010 1000, 1100 1110= $3EA8{,}CE_{16}$

1.11.Arvu teisendamine kümnendsüsteemist kahend-, kaheksand- ja kuuteistkümnendsüsteemi.

Täisarvu teisendamiseks jagatakse seda süsteemi alusega ja jääk kirjutatakse kõrvale.

nt 55 $10 \rightarrow 2$

55 / 2	1
27 / 2	1
13 / 2	1
6 / 2	0
3 / 2	1
1	1

Vanemad järgud on allpool ja arvud kirjutatakse vastusesse vasakult paremale alates vanimast järgust.
$55_{10} = 110111_2$
Murdosa teisendamiseks korrutatakse seda süsteemi alusega ja korrutise täis osa eraldatakse.

1.58 $10 \rightarrow 2$

0.58*2 = 1.	16
0.16*2 = 0.	32
0.32*2 = 0.	64
0.64*2 = 1.	28
0.28*2 = 0.	56
0.56*2 = 1.	12
0.12*2 = 0.	24

$0.58_{10} = 0{,}1001010_2$

<u>Ülesanne</u>

Teisendada 753,21 kaheksand ja kuueteistkümnendsüsteemi.

$753{,}21_{10} = 361{,}1534_8$

753 / 8	1
94 / 8	6
11 / 8	3
0,21*8 = 1	68
0,68*8 = 5	44
0,44*8 = 3	52
0,52*8 = 4	16

$753{,}21_{10} = 1F2{,}35C2_{16}$

753 / 16 = 47	1
47 / 16 = 2	F
	2
0.21*16 = 3	36
0.36*16 = 5	76
0.76*16 = C	16
0.16*16 = 2	56

<u>Kodused ülesanded</u>

2.	D0,B		16	→ 10
3.	507,52		8	→ 10
4.	11000011,001		2	→ 10
5.	2570,326	8	→ 2	
6.	39,6CA	16	→ 2	
7.	196,82		10	→ 2
8.	524,318			
a.	10 → 8421			
b.	10 → 2421			
c.	10 → Liiaga 3			
9.	1010000,1	2	→ 8	
10.	653,35		10	→ 8
11.	101011,101001	2	→ 16	
12.	538,7		10	→ 16

<u>Vastused</u>

1. 208,6875
2. 327,78125
3. 195,125
4. 010101111000,011010110
5. 00111001,011011001010
6. 11000100,1101
7. a
   7.1. 010100100100,001100011000
   7.2. 101100100100,001100011110
   7.3. 100001010111,011001001011
8. 120,4
9. 1215,2631
10. 2B,A4
11. 21A,B333

1.12. Aritmeetilised operatsioonid kahendsüsteemis.

1.12.1. Positiivsete arvude liitmine.

N1 = 001011
N2 = 001101

```
 1111
 001011
 001101
--------
 011000
```

1.12.2. Algebraline liitmine pöördkoodis.

Et saada õiget tulemust positiivse ja negatiivse arvu liitmisel, tuleb negatiivne arv viia pöördkoodi. Selleks tuleb inverteerida kõik arvu järgud v.a. märgi järk.

Näide 1

N_1 = 011010
N_2 = 101000
$N_{2pöörd}$ = 110111

$N_1 + N_2$ = 010010

Märgi järgust tekkiv ülekanne liidetakse juurde noorimale järgule.
Kui tulemus on positiivne, siis pole saadud vastust enam teisendada vaja.

Näide 2

N_1 = 111010
N_2 = 001000
$N_{1pöörd}$ = 100101

$N_1 + N_2$ = 110010

Kui liitmise tulemus on negatiivne, tuleb see lõpliku vastuse saamiseks viia pöördkoodist otsekoodi. Selleks tuleb inverteerida kõik arvu järgud v.a. märgi järk.

1.12.3. Algebraline liitmine täiendkoodis.

Negatiivse arvu täiendkoodi viimiseks inverteeritakse kõik arvu järgud v.a. märgi järk ja noorimale järgule liidetakse 1.

Näide 1

N_2 = 101000
N_1 = 011010
$N_{2täiend}$ = 111001
$N_1 = N_2$ = 010010

Täiendkoodis ei ole vaja arvestada märgi järgust tekkivat ülekannet.

Näide 2

$N_1 = \underline{1}11010$
$N_2 = \underline{0}01000$
$N_{1täiend} = \underline{1}00101$
$N_1 + N_2 = \underline{1}01110 \Rightarrow \underline{1}10010$

Kui liitmise tulemus on negatiivne, siis tuleb see lõpliku vastuse saamiseks viia täiendkoodist otsekoodi. Selleks tuleb inverteerida kõik arvu järgud v.a. märgi järk ja noorimale järgule liidetakse 1.

Kodune ülesanne

Liita N_1 ja N_2 pöörd- ja täiendkoodis.

$N_1 = \underline{1}10111$
$N_2 = \underline{0}10001$

2. Loogikafunktsioonid.

2.1. Loogikafunktsioon ja loogikaseade. Boole`i algebra.

Loogika algebra e Boole`i algebra on matemaatilise loogika üks osa ja seda nimetatakse lause arvutuseks. Kui lause on tõene, siis tähistatakse seda numbriga 1, ja kui lause on väär, siis numbriga 0. Muutujat, mille väärtus võib olla 0 või 1, nimetatakse kahendmuutujaks. Nulli nimetatakse loogiliseks nulliks ja ühte loogiliseks üheks. Sõltumatuid muutujaid (sisendeid) nimetatakse argumentideks ja neist sõltuvaid muutujaid (väljundeid) nimetatakse funktsioonideks. Loogikafunktsiooni kõik argumendid on loogilised muutujad, millel on kaks väärtust – 0 või 1. Funktsioone, mis võivad omandada väärtusi 0 või 1, nimetatakse loogikafunktsioonideks. Seadmed, mis formeerivad loogikafunktsioone, nimetatakse loogika e digitaalseadmeteks.
Kahendkoodil sisestamis- ja väljastamisviiside järgi jaotatakse loogikaseadmed:

1. Jadatoimega, kus üks takt sisaldab ainult ühe biti ja ühe biti kaupa saadakse ka väljundsignaal.
2. Rööptoimega, kus kõik bitid sisestatakse korraga ja saadakse ka rööpväljunditest korraga.
3. Segatoimega, kus rööpinfo muudetakse jadainfoks või vastupidi.

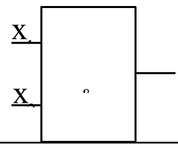
Tööpõhimõtte järgi jaotatakse loogikaseadmed:

1. Kombinatsioonseadmed (mäluta), kus väljundsignaal on määratud ainult antud hetkel sisendites toimivate signaalidega ja ei sõltu seadme eelmistest olekutest. Nt summaator.
2. Järjestikseadmed (mäluga), kus väljundsignaal sõltub nii momendil sisendites toimivatest signaalidest kui ka seadme eelmistest olekutest. Nt loendur.

2.2. Ühe argumendi loogikafunktsioonid.

n argumendi korral on argumentide kombinatsioonide arv 2^n ja funktsioonide arv 2^{2^n}. Ühe argumendi korral, kui n = 1, on argumentide kombinatsioonide arv 2 ja funktsioonide arv 4. Kui n = 2, on vastavad arvud 4 ja 16. Kui n = 3, siis 8 ja 256.

2.3. Kahe argumendi loogikafunktsioonid.

Funk. Nr	Funktsiooni nimetus	Argumentide kombinatsiooonid x_1 0 0 1 1 x_2 0 1 0 1	Funktsiooni selgitus	Funktsiooni matemaatiline valem	Loogikaelemendi tähis
F_0	Konstant null	0 0 0 0	Väljundis signaal 0	$f_0 = 0$	
F_1	NING	0 0 0 1	Väljundis on 1, kui mõlemas sisendis on 1	$f_1 = x_1 – x_2$	
F_2	x_2 keeld	0 0 1 0	Väljund võrdub x_1 kui $x_2 = 0$. Signaali $x_2 = 1$ puhul on väljundis alati 0 sõltumata x_1-st		
F_3	x_1 kordus	0 0 1 1			
F_4	x_1 keeld	0 1 0 0			
F_5	x_2 kordus	0 1 0 1			
F_6	VÕI	0 1 1 0	Väljundis on signaal 1 ainult siis kui sisendid on erinevad.	$f_6 = x1 \oplus x2$	
F_7	Loogikaline liitmine e VÕI	0 1 1 1	Väljundis on signaal 1 siis kui kas või ühes on 1		
F_8	Pierce'e tehe e diijunktsiooni ümber	1 0 0 0	Väljundis on signaal 0, kui kas või ühes on sisendis on 1	$f_8 = x_1 \downarrow x_2$	
F_9	Ekvivalentsus e samaväärsus	1 0 0 1	Väljundis on signaal 1 ainult siis, kui sisendites on ühesugused väärtustes	$f_9 = x_1 + x_2 + \overline{x_1}\,\overline{x_2}$	
F_{10}	x_2 inversioon e x_2 eitus e EI	1 0 1 0	Väljundis on signaal 1, kui $x_2 = 0$ ja signaal 0, kui $x_2 = 1$	$f_{10} = \overline{x_2}$	
F_{11}	x_1 implikatsioon	1 0 1 1	Väljundis on signaal 0 ainult siis kui $x_2 = 1$ ja $x_1 = 0$	$f_{11} = x_1 + \overline{x_2}$	
F_{12}	x_1 inversioon e x_1 eitus	1 1 0 0		$f_{10} = \overline{x_2}$	

F_{13}	x_2 implikatsioon	1 1 0 1	Väljundis on signaal 0 ainult siis, kui $x_1 = 1$ ja $x_2 = 0$	$f_{13} = \overline{x_1} + x_2$	
F_{14}	Schefferi tehe e NING-EI	1 1 1 0	Väljundis on signaal 0 kui kõikides sisendites on signaal 1	$f_{14} = \overline{x_1 * x_2}$	
F_{15}	Konstant 1	1 1 1 1	Väljundis on alati signaal 1	$f_{15} = 1$	

2.4. Loogikaseadused.

2.4 Loogikaseadused

Aksioomide põhjalk tuletatakse peamised loogikaseadused.
Domineerimisseadus 1. Suvalise muutujate hulga konjuktsioon on null (tühimik), kui kas või ainult üks muutujatest võrdub nulliga.

$$0 \cdot A \cdot B \cdot C = 0$$

Domineerimisseadus 2. Suvalise muutujate hulga disjunktsioon on üks (universaalhulk), kui kas või üks muutujatest võrdub ühega.

$$1 + A + B + C = 1$$

Idempotentsus- ehk samaväärsusseadus (kehtib ka kolme ja enama muutuja kohta).
Argumendi loogiline korrutamine või liitmine iseendaga ei muuda tulemi väärtust.

$$a \cdot a = a$$
$$a + a = a$$

Eituse eitamise seadus. Argumendi väärtus tema kahekordsel eitamisel ei muutu.

$$\bar{\bar{a}} = a$$

Komplementaarsus-ehk täiendiseadus. Argumendi ja tema eituse ehk täiendi loogiline korrutis on null, loogiline summa üks.

$$a \cdot \bar{a} = 0$$
$$a + \bar{a} = 1$$

Kommutatiivsusseadus. Argumentide järjekorda loogikatehetes võib muuta.

$$a \cdot b = b \cdot a$$
$$a + b = b + a$$

Assotsiatiivsusseadus. Mitme argumendi loogilist korrutamist ja liitmist võib sooritada suvalises järjekorras või samaaegselt.

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c = a \cdot b \cdot c$$
$$a \cdot (b + c) = (a + b) \cdot (a + c)$$

Distributiivsusseadus (sulgude avamise seadus). Argumentide loogilist summat võib loogiliselt korrutada argumendiga a või korrutada esmalt kõiki argumente a-ga ning seejärel need korrutised loogiliselt liita. Argumentide loogilisele korrutisele võib liita argumend a või esmalt liita loogiliselt kõikidele argumentidele a ning seejärel need summad loogiliselt korrutada. Kui esimene teisendus vastab sulgude avamisele arvude algebra, siis teine on rakendav üksnes loogikaalgebras.

$$a\cdot(b+c)=a\cdot b+a\cdot c$$
$$a+b\cdot c=(a+b)\cdot(a+c)$$

Absorbtsiooni- ehk neeldusseadused:

$$a\cdot(a+b)=a$$
$$a+a\cdot b+a\cdot c+\ldots(a+w)=a$$

$$a\cdot(\bar{a}+b)=a\cdot b$$
$$a+\bar{a}\cdot b=a+b$$

Kleepimisseadus

$$a\cdot b+a\cdot\bar{b}=a$$
$$(a+b)\cdot(a+\bar{b})=a$$

Üldised kleepimisseadused.

$$(a+b)\cdot(\bar{a}+c)=a\cdot c+\bar{a}\cdot b$$
$$a\cdot b+\bar{a}+c+b\cdot c=a\cdot b+\bar{a}\cdot c$$
$$(a+b)\cdot(\bar{a}+c)\cdot(b+c)=(a+b)\cdot(\bar{a}+c)$$

Ülesanne.

Tõestada neelduvus- ja kleepimisseadused.

a(a*b) = a+ab = a(1+b)=a*1=a
a*(a kriipsuga + b) = a*a kriipsuga + a*b = 0 + a*b = a*b
a + a kriipsuga* b = (a+a kriipsuga) * (a +b) = 1 * (a + b) = a + b
a*b+a*b kriipsuga = a(b+b kriipsuga) = a * 1 = a
(a+b)*(a+b kriipsuga) = a + b*b kriipsuga = a + 0 = a

Näidata olekutabeli abil de morgani seaduste kehtivust kolme argumendi korral.

A	***B***	***C***	$\overline{a \cdot b \cdot c}$	$\bar{A}$	$\bar{B}$	$\bar{C}$	$\bar{A}+\bar{B}+\bar{C}$
0	0	0	1	1	1	1	1
0	0	1	1	1	1	0	1
0	1	0	1	1	0	1	1
0	1	1	1	1	0	0	1
1	0	0	1	0	1	1	1
1	0	1	1	0	1	0	1
1	1	0	1	0	0	1	1
1	1	1	0	0	0	0	0

A	***B***	***C***	(a+b+c) kriipsuga	***A kriipsuga***	***B kriipsuga***	***C kriipsuga***	***A kriipsuga * b kriipsuga * c kriipsuga***
0	0	0	1	1	1	1	1
0	0	1	0	1	1	0	0
0	1	0	0	1	0	1	0
0	1	1	0	1	0	0	0
1	0	0	0	0	1	1	0
1	0	1	0	0	1	0	0
1	1	0	0	0	0	1	0
1	1	1	0	0	0	0	0

Ülesanne 4.4

Lihtsustada.
a) A + A*B kriipsuga + A kriipsuga = A(1 + B kriipsuga)+ A kriipsuga = A + A kriipsuga = 1

Ülesandeid

$$\overline{A\cdot B+\overline{A}\cdot\overline{B}}+A\cdot\overline{B}=\overline{A}\cdot B+A\cdot\overline{B}=A\cdot\overline{B}$$

$$a\cdot b+a(b+1)+\overline{a+b}=a\cdot b+a+\overline{a}\cdot\overline{b}=a(b+1)+\overline{a}\cdot\overline{b}=a+\overline{a}\cdot\overline{b}=(a+\overline{a})\cdot(a+\overline{b})=a+\overline{b}$$

$$X=(A+\overline{B})\cdot(A+B\cdot C)=(A+\overline{B})\cdot(A+B)\cdot(A+C)$$
$$(A+B\cdot\overline{B})\cdot(A+C)=A(A+C)=A+A\cdot C=A(1+C)=A$$

$$X=(A+B)(A+C)+\overline{C}=A+B+C=A+B\cdot C+\overline{C}=A+(\overline{C}+B)(\overline{C}+C)=A+B+\overline{C}$$

$$X=\overline{\overline{\overline{\overline{A\cdot B}\cdot\overline{A\cdot C}}}\cdot B}\cdot\overline{\overline{\overline{A\cdot B}\cdot\overline{A\cdot C}}}+\overline{B}=$$
$$=\overline{A\cdot B}\cdot\overline{A\cdot C}+\overline{B}=(\overline{A}+\overline{B})\cdot(\overline{A}+\overline{C})+\overline{B}=\overline{A}+\overline{B}\cdot\overline{C}+\overline{B}=\overline{A}+\overline{B}(\overline{C}+1)=\overline{A}+\overline{B}=\overline{A\cdot B}$$

Kirjutada olekutabeli järgi funktsioon X, minimeerida ja joonestada skeem.
$$X=\overline{A}\cdot\overline{B}\cdot\overline{C}+\overline{A}\cdot\overline{B}\cdot C+\overline{A}\cdot B\cdot\overline{C}+\overline{A}\cdot B\cdot C+A\cdot B\cdot\overline{C}=$$
$$=\overline{A}\cdot\overline{B}(\overline{C}+C)+\overline{A}\cdot B(\overline{C}+C)+A\cdot B\cdot\overline{C}=\overline{A}\cdot\overline{B}+\overline{A}\cdot B+A\cdot B\cdot\overline{C}=$$
$$=\overline{A}(\overline{B}+B)+A\cdot B\cdot\overline{C}=\overline{A}+A\cdot B\cdot\overline{C}=(\overline{A}+B\cdot\overline{C})(\overline{A}+A)=\overline{A}+B\cdot\overline{C}$$

Example 4.3, 4.6,
de-morgani abil:
$$\overline{x1+x2+\overline{x1}\cdot\overline{x2}\cdot\overline{x3}+x1\cdot\overline{x2}\cdot x3}$$

2.5. Funktsionaalselt täielikud süsteemid.

Loogikaelementide süsteemi, mis realiseerib kõiki kahe argumendi funktsioone nimetatakse funktsionaalselt täielikuks süsteemiks. Nt NING, VÕI ja EI. Minimaalne funktsionaalselt täielik süsteem on selline, millest ükskõik millise elemendi väljajätmine muudab süsteemi mittetäielikuks. Nt VÕI-EI või NING-EI.

Kui on tarvis realiseerida skeem VÕI-EI abil, siis on tavaliselt parem kasutada funktsiooni summade korrutise kujul.

Kui on tarvis realiseerida skeem NING-EI abil, siis on tavaliselt parem kasutada korrutiste summa kujul.