- **Negatiivsete kahendarvude esitus**

Kahendarvude negatiivsete väärtuste esitamisel kasutatakse põhiliselt kolme esitusvarianti:

- märk-ja-(abs.)väärtus (sign-and-magnitude)
- täiendkood (two's compement)
- pöördkood (one's complement)

- **Märk-ja-väärtus kood**

Formaadi esimene bitt näitab arvu märki - "0" näitab positiivset ja "1" negatiivset arvu. Märgile järgneb arvu absoluutväärtus.
Järgnevas on 8-bitises formaadis kujutatud väärtused $+55_{10}$ ja -55_{10}.

+/-	a_6	a_5	a_4	a_3	a_2	a_1	a_0
0	0	1	1	0	1	1	1
1	0	1	1	0	1	1	1

Arvude esitusdiapasoon n-bitise formaadi korral: $-(2^{n-1} -1)$ kuni $+(2^{n-1} -1)$

Aritmeetikatehete sooritamisel tuleb arvestada esitusviisi omapära.
Arvude liitmise/lahutamise puhul on vaja arvestada, et:

- samamärgiliste arvude väärtused liituvad;
- erimärgiliste arvude korral on vaja väärtusi võrrelda; kui väärtused on võrdsed on resultaat võrdne nulliga; kui väärtused on erinevad, on vaja lahutada suuremast väiksem ja absoluutväärtuselt suurema suurema arvu märk jääb tulemuse märgiks;
- lahutamise korral on eelnevalt vaja muuta teise operandi märk.

<u>Näide:</u>
Formaadi pikkus: 8 bitti.
$A=19_{10} = 00010011_2$
$B=14_{10} = 00001110_2$
Liitmine A+B:

$$\begin{array}{r} 00010011_2 \\ \underline{+\ 00001110_2} \\ 00100001_2 \end{array}$$

Lahutamine A-B:
$00010011_2 - 00001110_2$
Lahutamine realiseerub kui negatiivse arvu liitmine:
$00010011_2 + 10001110_2$

Kuna $0010011_2 > 0001110_2$, tuleb lahutada $0010011_2 - 0001110_2 = 0000101_2$.
Tulemuse märk on positiivne ja tulemus ise: 00000101_2 .

Korrutamise/jagamise puhul määratakse tulemuse märk aritmeetika reeglitega (samamärgilised operandid - positiivne, erimärgilised - negatiivne); tulemuse väärtus saadakse tavalise kahendtehtega.

<u>Näide:</u>

Formaadi pikkus: 8 bitti.
$A=19_{10} = 00010011_2$
$B=-6_{10} = 10000110_2$

Korrutise A•B märk on negatiivne (erimärgilised), absoluutväärtus: 1110010_2 .
Tulemus: $1\ 1110010_2$.

- **Täiendkood (täiend aluse suhtes)**

Arvude esitamise aluseks täiendkoodis, samuti täiendkoodis esitatud arvude aritmeetika teoreetiliseks aluseks on nn. moodularvutus, mille korral arvu A väärtus mooduliga m ehk A (mod m) esitatakse positiivse jäägina a, mis tekib arvu A jagamisel mooduli m väärtusega. Teisisõnu kehtib seos (A-a)/m = täisarv.

Rakendades seda erijuhul, kus $A < 0$ & $|A| < m$, kusjuures $a > 0$, jõuame seosteni:
$(A - a) / m = -1$ ehk
$A - a = - m$ ehk
$a = m + A$ ehk
$a = m - |A|$.
See viimane ongi tuntud nn. täiendkoodi valemina negatiivsete arvude jaoks.

Täiendkood on rakendatav suvalises positsioonilises arvusüsteemis.

<u>Näide:</u>
Täiendkood 10-süsteemis

Olgu mooduli väärtus m = 10 000
Sellisel juhul:

-0153	täiendkoodis	⇒	9847
+0176	täiendkoodis	⇒	0176
			(1)0023

Arvude summa leitakse vastavate täiendkoodide summana, kusjuures vanemast järgust väljaleviv ülekanna "läheb kaduma" tänu moodularvutusele (m=10 000)

Analoogiliselt:

+0153	täiendkoodis	⇒	0153	
-0176	täiendkoodis	⇒	9824	
			9977	(ehk -0023 täiendkood)

Järgnevalt rakendame **täiendkoodi kahendsüsteemis (two's complement)**, kus mooduli sobivaks väärtuseks on 2^k, kusjuures väärtus k määrab arvude esitusdiapasooni.

Murdarvude x korral on arvude esitusdiapasoon $(-1) \leq x \leq (1-2^{-m})$ ja moodul m väärtus on $2^1 = 2$.

Täiendkood valem negatiivsete murdarvude jaoks on seega $x_t = 2 - |x|$.
Positiivsete murdarvude korral mooduli 2 rakendamine ei muuda arvu väärtust.
Seega jäävad täiendkoodi rakendamisel positiivsed arvud kahendkujul diapasooni:

0,000….00 (minimaalne, võrdub 0-ga) kuni 0,111….11 (maksimaalne, võrdub $1-2^{-m}$). Negatiivsete murdarvude täiendkoodid jäävad vahemikku 1,111….11 (minimaalne, võrdub -2^{-m}) kuni 1,000….00 (maksimaalne, võrdub -1).

Negatiivse murdarvu täiendkoodi leidmiseks on kolm võrdväärset võimalust, mis annavad kõik samasuguse tulemuse:

- rakendada täiendkoodi valemit $x_t = 2 - | x |$

Näide:

$x = - 0{,}63_{10} = - 0{,}101000010_2$

$x_t = 2 - | x | = 10{,}000000000_2 - | 0{,}101000010_2 | = 1{,}010111110_2$

- leida teisendatava arvu pöördkood ja liita 1 selle nooremasse järku

$x = - 0{,}63_{10} = - 0{,}101000010_2$

$x_p = 1{,}010111101_2$

$x_t = 1{,}010111101_2 + 0{,}000000001_2 = 1{,}010111110_2$

- teisendada arvu absoluutväärtust nii, et nooremad järgud samaks kuni esimese "1"-ni (kaasa arvatud), vanemad järgud teisendada pöördkoodi.

Märgime, et täiendkoodis esitatud arvu täiendkoodina saame uuesti arvu absoluutväärtuse, kuna $(x_t)_t = 2 - | x_t | = 2 - | 2 - | x | | = | x |$.

Näiteid täiendkoodi kasutamisest arvude liitmisel.

Näide 1: kaks positiivset arvu:

	0,0101	(5/16)
+	0,1001	(9/16)
	0,1110	(14/16)

Näide 2: kaks negatiivset arvu

	1,1010	(-6/16)
+	1,1001	(-7/16)
(1)	1,0011	(-13/16)

NB! Viimases näites ei arvestata märgijärgust väljalevivat "1"-te, kuna arvutused toimuvad mooduliga m=2.

Märgime, et mõlemas eeltoodud näites on ülekannete väärtused vanemast väärtusjärgust märgijärku ja märgijärgust välja võrdsed (esimeses näites mõlemad võrdsed "0"-ga, teises näites võrdsed "1"-ga). See viitab asjaolule, et saadav tulemus ei ole ületäitunud ehk teisisõnu, ei ole väljaspool arvude esitusdiapasooni $(-1) \leq x \leq (1-2^{-m})$.

Näide 3: kaks negatiivset arvu

	1,0110	(-10/16)
+	1,0010	(-14/16)
(1)	0,1000	(-24/16)

Ülekanne märgijärku võrdub "0"-ga, ülekanne märgijärgust välja võrdub "1"-ga. Tegemist on negatiivse ületäitumisega ehk tulemus ületab diapasooni negatiivse piiri (-1).

Näide 4: kaks positiivset arvu:

	0,1101	(13/16)
+	0,1001	(9/16)
(0)	1,0110	(+22/16)

Ülekanne märgijärku võrdub "1"-ga, ülekanne märgijärgust välja võrdub "0"-ga. Tegemist on positiivse ületäitumisega ehk tulemus ületab diapasooni positiivse piiri $(1- 2^{-m})$.

Näide 5: erimärgilised arvud

	0,0101	(5/16)
+	1,0010	(-14/16)
	1,0111	(-9/16)

Mõlemad ülekanded (märgijärku ja sealt välja) võrduvad "0"-ga. Ületäitumist pole.

Näide 6: erimärgilised arvud

	1,1011	(-5/16)
+	0,1110	14/16)
(1)	0,1001	(9/16)

Mõlemad ülekanded (märgijärku ja sealt välja) võrduvad "1"-ga. Ületäitumist pole. Märgijärgust väljaleviv "1" jääb arvestamata, kuna moodul m=2.

Kokkuvõttes: kui liitmisel mõlemad tekkivad ülekanded (märgijärku ja sealt välja) on võrdsed (1 või 0), siis tulemus pole ületäitunud. Kui nimetatud ülekanded on erinevad, on tegemist kas negatiivse või positiivse ületäitumisega.

Ületäitumise tekkimist on võimalik avastada ka loogilise märgianalüüsi abil. Erinevate märkidega liidetavate korral on ületäitumine välistatud. Samamärgiliste liidetavate korral peab tulemuse märk olema kokkulangev liidetavate omaga, vastasel korral on tegemist ületäitumisega.

- **Täiendkood kahendtäisarvude puhul**

Olgu meil tegemist järgmise 8-bitise arvuformaadiga ja näitena selles kujutatud kümnendarvudega $+55_{10}$ ja -55_{10} .

+/-	a_6	a_5	a_4	a_3	a_2	a_1	a_0
0	0	1	1	0	1	1	1
1	1	0	0	1	0	0	1

Kasutatav mooduli väärtus on $m = 2^8$.
Täisarvude esitusdiapasoon on $(-2^7) \leq x \leq (2^7 -1)$, kokku 2^8 kahendvektorit.

Täiendkoodi valem negatiivsete arvude jaoks: $x_t = 2^8 - | x |$

Näide 1:
$A = -19_{10}$
$+19 = 00010011_2$
$-19_{tk} = 11101101_{tk}$
$B = +36_{10}$
$+36 = 00100100_2$

Liitmine A+B annab tulemuseks:

-19	11101101_{tk}
+36	00100100_2
+17	(1) 00010001_2

Selle arvuformaadi korral on nii positiivsete arvud kui ka nende kujutised diapasoonis $0 \le x \le 127$ (märgibitt võrdne "0"-ga). Negatiivsed arvud on diapasoonis $-128 \le x \le -1$, nende kujutised $2^8 - |x|$ aga diapasoonis $+128 \le x \le +255$.

Näide 2:
Realiseerida täiendkoodi kasutamisega tehe:
$-39_{10} + (-51)_{10}$

$+39_{10} = 00100111_2$
$-39_{tk} = 11011001_{tk}$

$+51_{10} = 00110011_2$
$-51_{tk} = 11001101_{tk}$

-39_{tk} =	11011001_{tk}
-51_{tk} =	11001101_{tk}
-90_{tk} =	(1) 10100110_{tk} = $- 01011010_2 = -(64 + 16 + 8 + 2) = -90_{10}$

- **Laiendatud (modifitseeritud) täiendkood**

Tavalise täiendkoodi puhul on tülikas ületäitumise fikseerimine. See nõuab kahe ülekande (märgibitti sisse ja sealt välja) analüüsi ning ei võimalda ületäitumise tekkimist selgitada ainult liitmise tulemusvektori põhjal. Et tulemuse ületäitumist avastada ainult resultaatvektorit analüüsides on otstarbekas sisse tuua nn. "dubleeritud" märgibitt ehk teisisõnu - suurendada esitatavate arvude diapasooni ja lülitada sellesse ka ületäitunud arvud.
Kahendmurdarvude korral tähendab see mooduli m = 4 rakendamist, kusjuures kaks täisosa järku a_1 ja a_0 annavad infot nii arvu märgi kui ka ületäitumise kohta. Järk a_1 on endiselt märgijärgu rollis ("1" näitab negatiivset ja "0" positiivset arvu). Järk a_0 näitab aga ületäitumist juhul kui ta on erinev arvu märgijärgust a_1.
Seega näitavad laiendatud täiendkoodi täisosa järgud arvu kohta järgnevat infot:
00, …. - normaalsed (mitteületäitunud) positiivsed arvud
01, …. - ületäitunud positiivsed arvud (arvud vahemikus $(+1) \le x \le (2-2^{-m})$)
11, …. - normaalsed (mitteületäitunud) negatiivsed arvud
10, …. - ületäitunud negatiivsed arvud (täiendkoodis esitatud arvud vahemikus $(-1+2^{-m}) \le x \le (-2)$)

Näited:

- kaks negatiivset arvu:

	11,0110	(-10/16)
+	11,0010	(-14/16)
(1)	10,1000	(-24/16)

Tulemus on ületäitunud negatiivne arv, ülekanne järku a_2 jääb arvestamata, kuna rakendub moodul m=4.

- kaks erimärgilist arvu:

	11,1011	(-5/16)
+	00,1110	(14/16)
(1)	00,1001	(9/16)

Tulemus on normaalne positiivne arv, ülekanne järku a_2 jääb arvestamata, kuna rakendub moodul m=4.

Laiendatud täiendkoodi rakendamine täisarvude korral (8-bitise formaadi näitel). Järgnevas on esitatud dubleeritud märgibitiga formaadis kümnendarvud $+55_{10}$ ja -55_{10}.

+/-	+/-	a_5	a_4	a_3	a_2	a_1	a_0
0	0	1	1	0	1	1	1
1	1	0	0	1	0	0	1

8-bitise formaadi korral (2 märgibitti + 6 väärtusbitti) on arvude esitusdiapasoon jagunenud selliselt:
0…63 - normaalsed positiivsed arvud (märgibitid = 00)
64 … 127 - ületäitunud positiivsed arvud (märgibitid = 01)
128 … 191 - ületäitunud negatiivsed arvud (märgibitid = 10)
192 … 255 - normaalsed negatiivsed arvud (märgibitid = 11)

Näited:
$-21_{10} + 57_{10}$

-21		1	1	1	0	1	0	1	1
+57		0	0	1	1	1	0	0	1
+36	(1)	0	0	1	0	0	1	0	0

Vastus: $+36_{10}$

$-21_{10} - 57_{10}$

-21		1	1	1	0	1	0	1	1
-57		1	1	0	0	0	1	1	1
-78	(1)	1	0	1	1	0	0	1	0

Vastus: -78_{10} (ületäitumine)

$+21_{10} - 57_{10}$

+21		0	0	0	1	0	1	0	1
-57		1	1	0	0	0	1	1	1
-36	(1)	1	1	0	1	1	1	0	0

Vastus: -36_{10}

- **Pöördkoodi kasutamine negatiivsete arvude esitamiseks (one's complement)**

Sarnaselt täiendkoodiga on negatiivseid arve võimalik esitada ka nn. pöördkoodi kaudu, kus negatiivse arvu kujutiseks on positiivse (absoluut)väärtuse pöördväärtus, asendadess "1"-d "0"-dega ja vastupidi.
Murdarvude korral on pöördkoodi valemiks $x_p = 2 - 2^{-m} - |x|$ ehk $x_p = x_t - 2^{-m}$. Seega on pöördkood erinev täiendkoodist vaid nooremasse järku juurdeliidetava väärtuse "1" võrra.
Kahe pöördkoodis esitatud arvu liitmisel tekib järgnev summa:

$x_{p\Sigma} = 2 - 2^{-m} - |x_1| + 2 - 2^{-m} - |x_2| = 2+2-|x_1+x_2|-2*2^{-m}$,

mis erineb korrektsest summast $2-|x_1+x_2|-2^{-m}$ nooremas järgus, kuhu on vaja juurde liita 2^{-m}. Liitmiseks vajalik "1" tekib ülekandena märgibittide liitmisest (nn. ringülekanne, mida kasutataksegi selle tekkimisel korrektsiooniks).
Märgime, et võimaliku ringülekande tulemusena suureneb oluliselt (kuni 2 korda) liitmisoperatsiooni ajaline kestvus, kuna vajalik on halvimal juhul kahekordne ülekande "läbilevimine" läbi kõigi summaatori järkude.

<u>Näited:</u>
(-38/64) + (-24/64) = - 0,100110 + (-0,011000) = (-62/64)

	1	0	1	1	0	0	1
	1	1	0	0	1	1	1
(1)	1	0	0	0	0	0	0
→							+1
	1	0	0	0	0	0	1

(-38/64) + (24/64) = - 0,100110 + (0,011000) = (-14/64)

1	0	1	1	0	0	1
0	0	1	1	0	0	0
1	1	1	0	0	0	1

(38/64) + (-24/64) = 0,100110 + (-0,011000) = (+14/64)

	0	1	0	0	1	1	0
	1	1	0	0	1	1	1
1	0	0	0	1	1	0	1
→							+1
	0	0	0	1	1	1	0

Täisarvude korral avaldub pöördkood valemina $x_p = 2^n - 1 - |x|$ ning kahe negatiivse arvu summa $x_{p\Sigma} = 2^n - 1 - |x_1| + 2^n - 1 - |x_2| = 2^n + 2^n - |x_1+x_2| - 1 - 1$. Vajalik korrektsioon nooremasse järku "+1" saadakse endiselt ringülekandena.

Näide:
Formaadi pikkus 7 bitti (märk + 6 väärtusbitti)
Realiseeritav tehe:
$-22_{10} + 28_{10}$

Järgu kaal		64	32	16	8	4	2	1
-22		1	1	0	1	0	0	1
+28		0	0	1	1	1	0	0
	(1)	0	0	0	0	1	0	1
Ringülekanne	→							+1
Tulemus (+6)						1	1	0

Laiendatud (modifitseeritud) pöördkood võimaldab analoogiliselt laiendatud täiendkoodiga avastada ületäitumise dubleeritud märgibittide kasutamise abil.

Murdarvude korral on laiendatud pöördkoodi valemiks $x_p = 4 - 2^{-m} - |x|$ ning ta erineb arvväärtuselt laiendatud täiendkoodist väärtuse 2^{-m} võrra: $x_p = x_t - 2^{-m}$.

- **Täiend- ja pöördkoodis esitatud arvude nihutamine**

Üldjuhul kahendarvude vasakule nihutamine ühe järgu võrra vastab arvu arvväärtuse korrutamisele 2-ga. Nihutamine k järgu võrra tähendab arvu korrutamise 2^k-ga.
Analoogiliselt on paremale nihutamine seotud jagamisele 2-ga (kui nihutatakse ühe järgu võrra) või 2^k-ga (kui nihe toimub k järgu võrra).
Täiendkoodis esitatud arvu $x_t = 2 - |x|$ nihutamine vasakule ühe biti võrra annab tulemuseks $2 \bullet x_t = (2)+2 - |2 \bullet x|$. Tänu mooduli m=2 rakendamisele ei arvestata märgibitist väljalevivat "1"-te.
Täiendkoodis esitatud arvu $x_t = 2 - |x|$ nihutamine paremale ühe biti võrra annab tulemuseks $0{,}5 \bullet x_t = 1 - |x/2|$. Vasakult tühjaks jääv bitt tuleb täita väärtusega "1", et saada õiget tulemust $2 - |x/2|$.
Pöördkoodi $x_p = 2 - 2^{-m} - |x|$ korral on vasakule nihutamise tulemuseks
$2*x_p = (2)+2 - |2 \bullet x| - 2 \bullet 2^{-m}$, mistõttu märgibitist väljaleviv "1" tuleb ringülekandena nooremasse bitti juurde liita.
Pöördkoodi $x_p = 2 - 2^{-m} - |x|$ paremale nihutamisel saame $0{,}5*x_p = 1 - 2^{-m-1} - |0{,}5*x|$, mida võib avaldada kui $1 - 2^{-m} + 2^{-m-1} - |0{,}5*x|$ ja mis nõuab märgibitti väärtuse "1" sissenihutamist õige tulemuse saamiseks.
Analoogilised on nihete sooritamise reeglid ja korrektsioonide vajadused laiendatud täiend- ja pöördkoodi korral.

Näide:
Antud $A=+36_{10}$ ja $B=-6_{10}$. Leida (1/4)*A + 8*B. Formaadi pikkus 7 bitti.
Pöördkood.

A=+36	0	1	0	0	1	0	0
B=-6	1	1	1	1	0	0	1
1/4*A = +9	0	0	0	1	0	0	1
8*B = -48	1	0	0	1	1	1	1
Tulemus = -39	1	0	1	1	0	0	0
Väärtus otsekoodis |-39|	-	1	0	0	1	1	1