Zaštitno kodiranje signala

Laboratorijska vježba 0

SP

Brojevni sustavi, kodovi, osnovni i složeni logički sklopovi, logička algebra, komplementi, uvod u LogiSim

Student:

prezime	Ime	mat. broj
	Laboratorij	Datum

0. 0. BROJEVNI SUSTAVI, KODOVI, OSNOVNI I SLOŽENI LOGIČKI SKLOPOVI, LOGIČKA ALGEBRA, KOMPLEMENTI, UVOD U LOGISIM

0.1. 0.1. DE1.doc ()

0.2. 0.2. BROJEVNI SUSTAVI I KODOVI

0.2.1. 0.2.1. KREIRANJE BROJEVNOGA SUSTAVA

0.2.1.1. Porijeklo naziva nekih brojevnih sustava:

broj	grčki naziv
0	
1	mona (hena)
2	**binar (di)**
3	terna (tri)
4	tetra (quadri)
5	penta
6	hexa
7	hepta
8	**octa**
9	nona (ennea)

broj	grčki naziv
10	**deca**
11	undeca (hendeca)
12	dodeca
13	trideca
14	tetradeca
15	pentadeca
16	**hexadeca**
17	heptadeca
18	octadeca
19	enneadeca

broj	grčki naziv
20	icosa
30	triaconta
40	tetraconta
50	pentaconta
60	**hexaconta**
70	heptaconta
80	octaconta
90	enneaconta
100	hecto
10000	myria

U SVAKOME brojevnome sustavu, kada se iscrpe osnovne znamenke brojevnoga sustava, sljedeća znamenka je 10.

SUSTAV	BAZA	MOGUĆE ZNAMENKE	OZNAKA ZA DEKADSKI 10_{10}

Binarni	2	0, 1	1010_2	1010
Oktalni	8	0, 1, 2, 3, 4, 5, 6, 7	12_8	013
Dekadski	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	10_{10}	10
Heksadekadski	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F	A_{16}	0xA
Neki novi (trinarni)	**3**	**0, 1, 2**	**21_3**	**21**
Sexagesimal[1]	60	1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60		

0.2.2. 0.2.2. PRIMJERI BROJEVNIH SUSTAVA:

Moguće je jednostavno napraviti svoj brojevni sustav pa sljedeća tablica prikazuje tri takva, a zovu se npr. *pentarni* (5 osnovnih znamenaka), *nonarni* (9 osnovnih znamenaka) i *tridekadski* (13 osnovnih znamenka), ali oni imaju samo teorijsko značenje, jer nemaju svoj preslik ni primjenu u stvarnome svijetu.

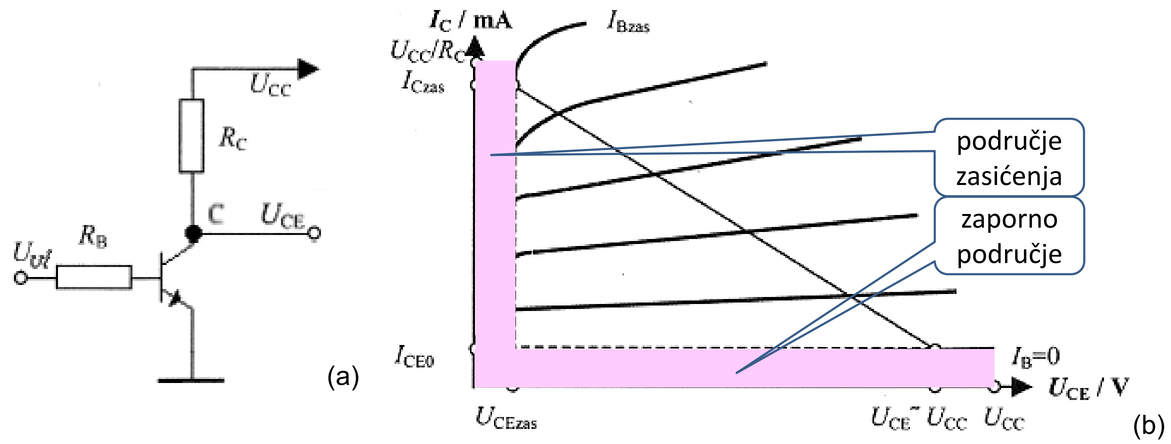
binarni	0	1	10	11	100	101	110	111	1000	1001	1010	1011	1100	1101	1110	1111	10000
neki novi baza = 5 *pentarni*	0	1	2	3	4	10	11	12	13	14	20	21	22	23	24	30	31
oktalni	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	20
neki novi baza = 9 *nonarni*	0	1	2	3	4	5	6	7	8	10	11	12	13	14	15	16	17
dekadski	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
neki novi baza = 13 *tridekadski*	0	1	2	3	4	5	6	7	8	9	10	11	12	10	11	12	13
heksa-dekadski	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10

Ovaj C++ program: (Sljedeća znamenka je 10), pokazuje da u bilo kojemu brojevnome sustavu - stvarnome ili izmišljenome, kada se pobroje sve osnovne znamenke toga sustava, sljedeće znamenka je 10 (čitaj: jedan-nula!) (vidi izvorni C++ kod).

0.2.2.1. Porijeklo naziva "hexdecimal system" u engleskome jeziku

Izraz "*Hexadecimal*" osmislili su 1963. godine u IBM. On je izveden iz grčke riječi *hexi* (šest) i latinske riječi *decem* (deset). Prikladniji izraz koristio bi latinsku riječ *sexa* (six), ali izraz *sexidecimal* ne zvuči previše pristojno.

0.2.2.2. Temelj digitalne tehnike



Slika 1.1. Radna područja tranzistora u polju izlaznih karakteristika

---

1 Prije 4000 godina Babilonci su razvili *sexagesimal* (latinski "šezdeset") brojevni sustav (baza 60). Svaka znamenka prenosi $\log_2 60 = 5.91$ informacijskih bitova. Na primjer, broj $4000_{10} = (1\ 6\ 40)_{60}$ (1 u 3600. stupcu, 6 u 60. stupcu i 40 u 1. stupcu).

0.3. 0.3. BROJEVNI SUSTAVI

0.3.1. 0.3.1. DEKADSKI BROJEVNI SUSTAV

U bilo kojemu brojevnome sustavu, prva znamenka je 0. Dekadski brojevni sustav ima deset znamenaka: 0, 1, 2, 3, 4, 5, 6, 7, 8 i 9, što znači da se svi brojevi od nula do devet mogu prikazati jednim simbolom (znamenkom). To su jednoznamenkasti brojevi. Najveći broj što se može napisati jednom znamenkom je devet. Broj deset nema posebnu oznaku pa se piše kombinacijom dviju znamenaka 1 i 0 (ili skraćeno 10 - čita se *deset* za razliku od ostalih brojevnih sustava gdje se ista kombinacija znamenaka čita "*jedan*" "*nula*"). Dvjema znamenkama mogu se napisati svi brojevi od deset do devedeset devet. To su dvoznamenkasti brojevi.

Za veće brojeve, potrebne su tri ili više znamenaka. Najveći broj što ga se može napisati s n znamenaka iznosi 10^n-1. Položaj znamenke u bilo kojemu broju naziva se *brojevno mjesto*. Svako brojevno mjesto ima svoju *vrijednost*, odnosno *težinu*. Težine brojevnih mjesta u dekadskome brojevnome sustavu mogu se prikazati kao *potencije broja deset* (ukupan broj osnovnih znamenaka u sustavu). Zato se kaže da je deset osnovica ili baza dekadskoga brojevnoga sustava. Najniže cjelobrojno mjesto ima težinu $10^0 = 1$.

Težine viših brojevnih mjesta iznose npr.: $10^1 = 10$, $10^2 = 100$, $10^3 = 1000$. Težine mjesta desno od dekadskoga zareza pišu se redom 10^{-1}, 10^{-2}, 10^{-3} itd.

0.3.1.1. Primjer: Težine mjesta desno od decimalnoga zareza

$10^2 = 100$
$10^1 = 10$
$10^0 = 1$
$10^{-1} = 0{,}1$
$10^{-2} = 0{,}01$

težine mjesta

$$234{,}56 = 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0 + 5 \times 10^{-1} + 6 \times 10^{-2}$$

… znamenka najnižega mjesta - najmanje težine LSD (*least significant digit*)
… decimalni zarez - točka (*decimal point*)
… znamenka najvišega mjesta - najveće težine MSD (*most significant digit*)

U dekadskome brojevnome sustavu brojevi se prikazuju nizom znamenaka što označavaju koeficijente kojima se množi baza sustava potencirana pripadnim težinskim mjestom. Pri pisanju brojeva pišu se samo koeficijenti (ne i potencirana baza), a težine mjesta određuju se prema položaju koeficijenata lijevo i desno od decimalnoga zareza. Opći prikaz broja u dekadskome brojevnome sustavu izgleda:

$$X = d_n\times10^n + d_{n-1}\times10^{n-1} + \ldots + d_2x\ 10^2 + d_1\times10^1 + d_0\times10^0 + d_{-1}\times10^{-1} + d_{-2}\times10^{-2} + \ldots$$
$$+ d_{-(m-1)}\times10^{-(m-1)}\times d^{-m}\times10^{-m}$$

Koeficijenti d, predstavljaju znamenke dekadskoga brojevnoga sustava.

0.3.2. 0.3.2. BINARNI BROJEVNI SUSTAV

Binarna matematika posebna je grana matematike, a bavi se brojevnim sustavom što sadrži samo dvije znamenke: 0 i 1. Čini se vrlo ograničavajućim koristiti samo dvije znamenje; Međutim, lako je stvoriti komponente što mogu razlikovati dvije naponske razine (a ne deset koliko bi trebalo za dekadski brojevni sustav). Komponente se jeftine, malih su dimenzija i male potrošnje snage.

U dekadskome brojevnome sustavu postoje: jedinice, desetice, stotice, ….

U binarnome sustavu postoje jedinice, dvojke, četvrtice itd. poput sljedećega prikaza:

2 (dvostruko):
2×2 (četvorostruko):
$2 \times 2 \times 2$ (osmerostruko):

jedinice

(polovina): 1/2
(četvrtina): $1/4=1/(2\times2)$
(osmina): $1/8=1/(2\times2\times2)$

1 1 0 1 , 1 0 1

$2 \times$ veći brojevi ←

→ $2 \times$ manji brojevi

Ovo znači:

$$1\times8 + 1\times4 + 0\times2 + 1\times1 + 1\times(1/2) + 0\times(1/4) + 1\times(1/8) = 13{,}625 \text{ u dekadskome.}$$

Brojevi se smještaju lijevo ili desno od decimalnoga zareza za naznačiti vrijednosti veće odnosno manje od 1.

Kao što se prije navelo, binaran brojevni sustav ima samo dvije znamenke: 0 i 1. Zato je za pisanje broja dva potrebno koristiti kombinaciju dviju binarnih znamenaka pa se u binarnome sustavu dekadski ekvivalent broju 2 piše kao 10. Najveći dvoznamenkasti broj u binarnome brojevnome sustavu je 11, što odgovara dekadskome broju 3, a najveći broj što ga se uopće može napisati s n znamenaka iznosi $2^n–1$.

Baza binarnoga brojevnoga sustava je 2. Težine cjelobrojnih mjesta u binarnome brojevnome sustavu su: $2^0 = 1$, $2^1 = 2$, $2^2 = 4$, $2^3 = 8$, …. Brojevna mjesta desno od binarnoga zareza imaju težine 2^{-1}, 2^{-2}, 2^{-3}, …. Za znamenke binarnoga brojevnoga sustava (binarne znamenke) obično se koristi naziv bit (*binary digit*). Opći prikaz broja u binarnome brojevnome sustavu izgleda:

$$A = b_m\times2^m + b_{m-1}\times2^{m-1} + \ldots + b_1\times2^1 + b_0\times2^0 + b^{-1}\times2^{-1} + \ldots + b_{-(n-1)}\times2^{-(n-1)} + b_{-n}\times2^{-n}$$

Koeficijenti b polinoma A, pripadaju znamenkama binarnoga brojevnoga sustava.

Dakle i u binarnome brojevnome sustavu brojevi se prikazuju nizom znamenaka koje označavaju koeficijente kojima se množi potencirana baza toga brojevnoga sustava potencijom što odgovara brojevnome mjestu. Isto kao i u dekadskome brojevnome sustavu, pišu se samo koeficijenti polinoma, a težine mjesta određuju se prema mjestu tih koeficijenata.

0.3.2.1. Primjer: Određivanje težina mjesta prema položaju koeficijenata

2^3 8
2^2 4
2^1 2 … težine mjesta
2^0 1
2^{-1} 0.5
2^{-2} 0,25

$$1011{,}01 = 1 \times 10^3 + 0 \times 10^2 + 1 \times 10^1 + 1 \times 10^0 + 0 \times 10^{-1} + 1 \times 10^{-2}$$

… znamenka najnižega mjesta - najmanje težine LSD (*least significant digit*)
… ~~decimalni~~ zarez - točka (~~*decimal*~~ *point*)
… znamenka najvišega mjesta - najveće težine MSD (*most significant digit*)

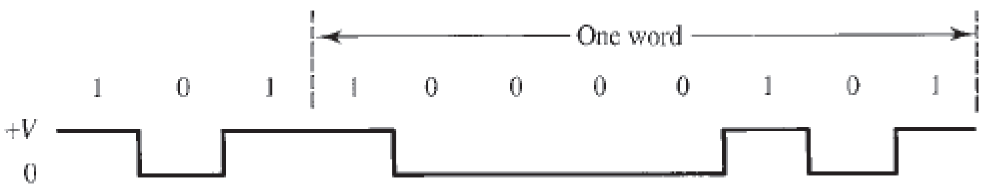
Tablica 1.1. Prikaz prvih 15 brojeva u binarnome i dekadskome brojevnom sustavu

Binaran broj	Dekadski broj	Binaran broj	Dekadski broj
0000	0	1000	8
0001	1	1001	9
0010	2	1010	10
0011	3	1011	11
0100	4	1100	12
0101	5	1101	13
0110	6	1110	14
0111	7	1111	15

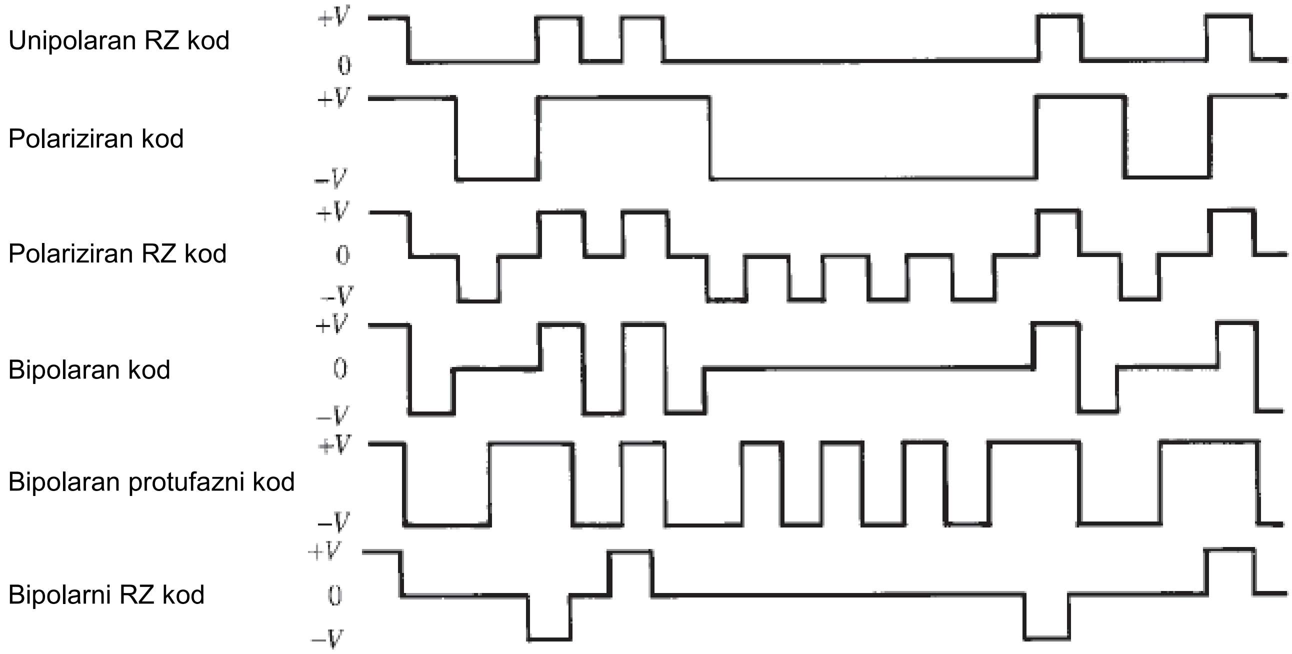
0.3.3. 0.3.3. BINARNI SIGNALI

Digitalan niz bitova

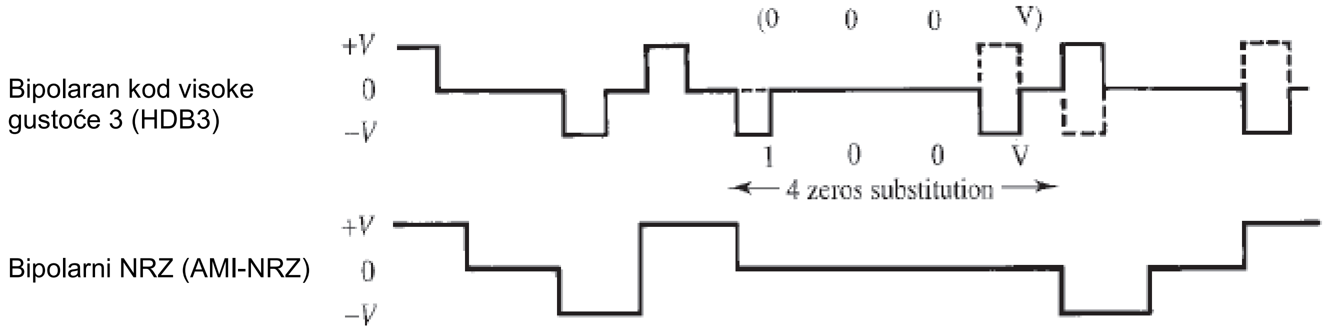
Unipolarni NRZ kod



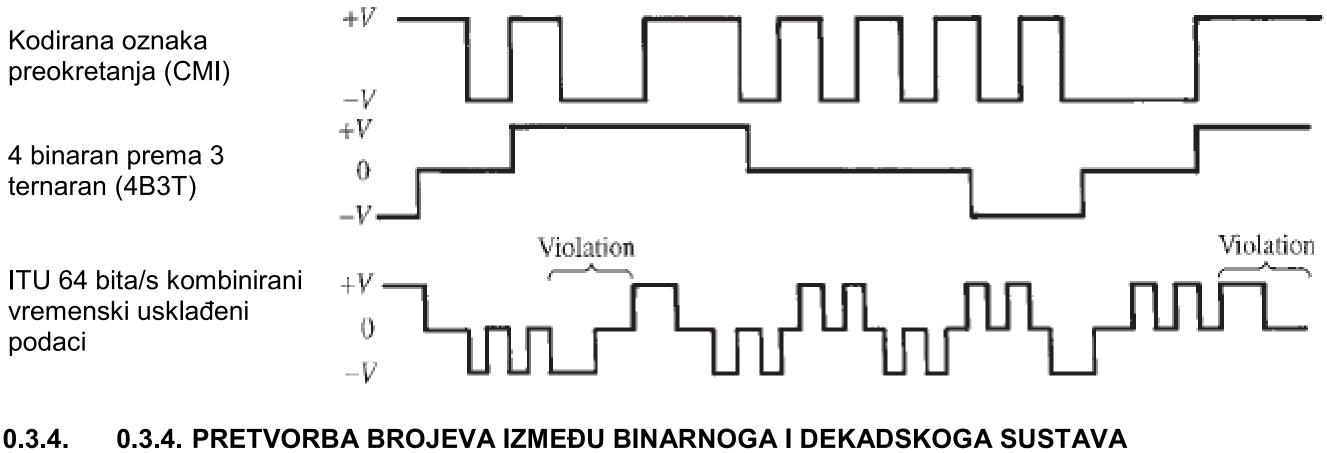
Binarna 1 kodira se impulsom pozitivne razine, a binarna 0 izostankom impulsa. Za dugotrajan niz "1" ili "0", nema prijelaza u digitalnome signalu, pa se gubi informacija o taktu. Zato se ovaj format ne može koristiti za sinkronizaciju između predajnika i prijemnika.



Binarna 1 predstavlja se pozitivnim, a binarna 0 negativnim impulsom. Na sredini svakoga intervala, signal pada na nultu razinu. Kako na sredini svakoga interval postoji prijelaz, sinkronizacija između predajnika i prijemnika lako se održava. Nedostatak je, dvostruko brže promjene signala što zahtijeva dvostruko veću propusnost.



Za vrijeme prijenosa jednoga bita, razina signala je konstantna. Koriste se dvije naponske razine, ali za razliku unipolarnoga NRZ koda, pozitivan impuls kodira binarnu 1, a negativan impuls kodira binarnu 0. I ovdje, pri dugotrajnome nizu jedinica 1, odnosno nula 0, postoji problem održavanja sinkronizacije.



0.3.4. 0.3.4. PRETVORBA BROJEVA IZMEĐU BINARNOGA I DEKADSKOGA SUSTAVA

Broj iz binarnoga brojevnoga sustava pretvara se u odgovarajući broj dekadskoga sustava tako da se svaka znamenka binarnoga broja pomnoži svojom težinom mjesta, a tako dobiveni iznosi zbroje se.

0.3.4.1. Primjer: Pretvorba binarnog broja 1011001 u dekadski.

$$1011001_2 = 1\times2^6 + 0\times25 + 1\times2^4 + 1\times2^3 + 0\times2^2 + 0\times2^1 + 1\times2^0 = 1\times64 + 1\times16 + 1\times8 + 1\times1 = 89_{10}$$

Indeksi uz brojeve pokazuju u kojemu brojevnome sustavu je broj napisan.

Pretvorba dekadskoga broja u binarni radi se tako da se dekadski broj rastavi na faktore koji su potencije broja dva. Postupak se provodi na sljedeći način: prvo se nađe najviša potencija broja 2 koja se nalazi u dekadskome broju. Zatim se traži koja sljedeća niža potencija broja dva predstavlja ostatak i tako sve dok se ne dođe do najniže potencije bez ostatka.

0.3.4.2. Primjer: Pretvorba dekadskoga broja 55 u binarni.

$$55_{10} = 32 + 16 + 8 + 4 + 2 + 1 = 1\times2^5 + 1\times2^4 + 0\times2^3 + 1\times2^2 + 1\times2^1 + 1\times2^0 = 110111_2$$

$55 - 32 = 23$

$23 - 16 = 7$

8 ne postoji

$7 - 4 = 3$

$3 - 2 = 1$

$1 - 1 = 0$

0.3.5. 0.3.5. OKTALNI BROJEVNI SUSTAV

Oktalni brojevni sustav ima osam osnovnih znamenaka: 0, 1, 2, 3, 4, 5, 6 i 7 pa mu je baza 8. Težine cjelobrojnih mjesta: $8^0 = 1$, $8^1 = 8$, $8^2 = 64$, $8^3 = 512$, …. Mjesta desno od zareza imaju težine 8^{-1}, 8^{-2}, 8^{-3} …. Najveći broj što ga se može napisati s n znamenaka je 8^n-1. Opći prikaz broja u oktalnome brojevnome sustavu:

$$A = o_m\times8^m + o_{m-1}\times8^{m-1} + \ldots + o_2\times8^2 + o_1\times8^1 + o_0\times8^0 + o_{-1}\times8^{-1} + o_{-2}\times8^{-2} + \ldots + o_{-(n-1)}\times8^{-(n-1)} + o_{-n}\times8^{-n}$$

Koeficijenti o označavaju znamenke oktalnoga brojevnoga sustava.

0.3.5.1. Primjer: Opći prikaz broja u oktalnome brojevnome sustavu

8^2	64	… težina mjesta
8^1	8	
8^0	1	
8^{-1}	0,125	
8^{-2}	0,015625	

$$234{,}56 = 2\times8^2 + 3\times8^1 + 4\times9^0 + 5\times8^{-1} + 6\times8^{-2}$$

… znamenka najnižega mjesta - najmanje težine LSD (*least significant digit*)

… ~~dekadska~~ točka, zarez (~~*decimal*~~ *point*)

… znamenka najvišega mjesta - najveće težine MSD (*most significant digit*)

0.3.6. 0.3.6. PRETVORBA BROJEVA IZMEĐU OKTALNOG I DRUGIH BROJEVNIH SUSTAVA

Pretvorba brojeva oktalnoga sustava u odgovarajući dekadski broj provodi se istim postupkom kao i binarnoga u dekadski. Razlika je samo u težinama brojevnih mjesta. Svaka znamenka oktalnoga sustava množi se svojom težinom, a dobiveni iznosi se zbroje.

0.3.6.1. Primjer: Pretvorba oktalnoga broja 237,51 u dekadski.

$$237{,}51_8 = 2\times8^2 + 3\times8^1 + 7\times8^0 + 5\times8^{-1} + 1\times8^{-2} =$$
$$= 2\times64 + 3\times8 + 7\times1 + 5\times0{,}125 + 1\times0{,}015625 = 159{,}640625_{10}$$

Pretvorba dekadskoga broja u odgovarajući oktalni može se provesti istim postupcima kao i kod pretvorbe u binarni broj. Ako je to metoda uzastopnoga dijeljenja dekadskoga broja bazom oktalnoga brojevnoga sustava, dobiveni ostaci dijeljenja označavaju znamenke oktalnoga brojevnoga sustava.

0.3.6.2. Primjer: Pretvorba dekadskoga broja 267 u oktalni.

$267 : 8 = 33$ + ostatak 3

$33 : 8 = 4$ + ostatak 1

$4 : 8 = 0$ + ostatak 4

$$267_{10} = 4\,1\,3_8$$

0.3.6.3. 1.1.6.1. Pretvorba oktalnoga broja u binarni broj - koder (8, 3)

Broj u oktalnome brojevnome sustavu pretvara se u broj binarnoga sustava tako da se svaka oktalna znamenka nadomjesti odgovarajućim binarnim brojem od 3 bita.

Tablica 1.2. Znamenke oktalnoga sustava i njihovi binarni ekvivalenti

Oktalna znamenka	0	1	2	3	4	5	6	7
Binarni ekvivalent	000	001	010	011	100	101	110	111

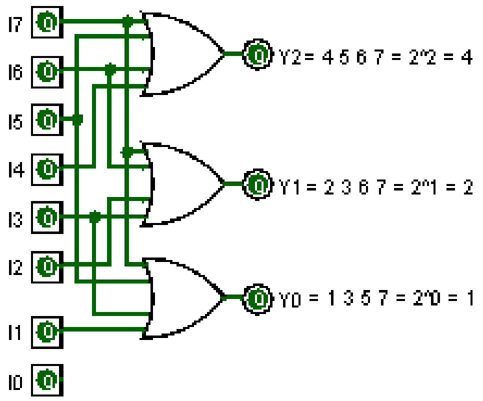
0.3.6.4. Primjer: Pretvorba oktalnoga broja 321 u binarni

$$321_8 = 011 \quad 010 \quad 001_2$$

001
010
011

I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	Y_2	Y_1	Y_0	
0	0	0	0	0	0	0	1	0	0	0	I_0
0	0	0	0	0	0	1	0	0	0	1	I_1
0	0	0	0	0	1	0	0	0	1	0	I_2
0	0	0	0	1	0	0	0	0	1	1	I_3
0	0	0	1	0	0	0	0	1	0	0	I_4
0	0	1	0	0	0	0	0	1	0	1	I_5
0	1	0	0	0	0	0	0	1	1	0	I_6
1	0	0	0	0	0	0	0	1	1	1	I_7

$$Y_0 = I_1 + I_3 + I_5 + I_7$$
$$Y_1 = I_2 + I_3 + I_6 + I_7$$
$$Y_2 = I_4 + I_5 + I_6 + I_7$$



0.3.6.5. 1.1.6.2. Pretvorba binarnoga broja u oktalni broj - dekoder (3, 8)

Broj iz binarnoga brojevnoga sustava pretvara se u oktalni tako da se binarni broj razdijeli u skupine od po 3 binarne znamenke počevši od najnižega brojevnoga mjesta. Svaka skupina binarnih znamenaka predočava se ekvivalentnom oktalnom znamenkom. Dobivene oktalne znamenke tvore oktalni broj.

0.3.6.6. Primjer: Pretvorba binarnoga broja 1011101010 u oktalni.

$$1011101010_2 = 001\ 011\ 101\ 010 = 1352_8$$
1 3 5 2

0.3.7. 0.3.7. HEKSADEKADSKI BROJEVNI SUSTAV

Baza heksadekadskoga brojevnoga sustava je 16 pa sustav ima 16 znamenaka. Za opisati znamenke od nula do devet koriste se isti znakovi kao i za dekadski brojevni sustav. Zato što za znamenke od 10_{16} do 15_{16} ne postoje posebni simboli, za iskazati te brojeve koristi se šest slova abecede: A, B, C, D, E i F.

Tablica 1.3. Znamenke heksadekadskoga brojevnoga sustava

Heksadekadska znamenka	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Dekadski broj	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Heksadekadski brojevni sustav najčešće se koristi za prikaz memorijskoga stanja procesorom upravljanih uređaja zbog jednostavne pretvorbe iz binarnoga brojevnoga sustava i obrnuto.

0.3.8. 0.3.8. PRETVORBA BROJEVA IZMEĐU HEKSADEKADSKOGA I DRUGIH BROJEVNIH SUSTAVA

Brojevi heksadekadskoga brojevnoga sustava pretvaraju se u dekadske brojeve tako da se vrijednost svake znamenke pomnoži težinom brojevnoga mjesta, a dobiveni iznosi se zbroje.

0.3.8.1. Primjer: Pretvorba između heksadekadskoga i ostalih brojevnih sustava

16^3
16^2 … težine mjesta
16^1
16^0

$$6A5C = 6 \times 16^3 + A \times 16^2 + 5 \times 16^1 + C \times 16^0 = 35134_{10} = 0110\ 1010\ 0101\ 1100_2 = 65134_8$$

… znamenka najnižega mjesta - najmanje težine LSD (*least significant digit*)
… znamenka najvišega mjesta - najveće težine MSD (*most significant digit*)

0.3.8.2. *Primjer: Pretvorba heksadekadskoga broja 2D7A u dekadski.*

$$2D7A_{16} = 2\times16^3 + D\times16^2 + 7\times16^1 + A\times16^0 = 2\times4096 + 13\times256 + 7\times16 + 10\times1 = 11642_{10}$$

Pretvorba dekadskoga broja u heksadekadski može se provesti istim postupkom kao i u ostale spomenute brojevne sustave, uzastopnim dijeljenjem dekadskoga broja sa 16, tj. bazom sustava. Ostatak dijeljenja označava heksadekadske znamenke.

0.3.8.3. *Primjer: Pretvorba dekadskoga broja 235 u heksadekadski.*

235 : 16 = 14 + ostatak 11

14 : 16 = 0 + ostatak 14

$$235_{10} = E\ B_{16}$$

Broj u heksadekadskome sustavu pretvara se u binaran broj tako da se svaka znamenka heksadekadskoga brojevnoga sustava nadomjesti odgovarajućim binarnim brojem, tj. kombinacijom od 4 bita prema tablici 1.4.

Tablica 1.4. Znamenke heksadekadskoga sustava i njihovi binarni ekvivalenti

Heksadekadska znamenka	Binarni ekvivalent	Heksadekadska znamenka	Binarni ekvivalent
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

0.3.8.4. *Primjer: Pretvorba heksadekadskoga broja B7C u binarni.*

$$B7C_{16} = 1011\ 0111\ 1100_2$$

Za pretvorbu binarnoga broja u heksadekadski potrebno je binarni broj razdijeliti u skupine od četiri binarne znamenke počevši od najnižega mjesta. Svaka skupina binarnih znamenaka predočava se ekvivalentnom heksadekadskom znamenkom prema tablici 1.4.

0.3.8.5. *Primjer: Pretvorba binarnoga broja 101101001001 u heksadekadski.*

$$101101001001_2 = 1011\ 0100\ 1001 = B49_{16}$$

0.3.8.6. *Pretvorba dekadskoga broja u broj, bilo kojega brojevnoga sustava!*

Pretvorba dekadskoga broja u broj bilo kojega brojevnoga sustava!

```
//#include <conio.h> //_getch();
```

```
#include <iostream>
using namespace std;
void main(){
int broj,duzina,baza;
char polje[18];//umjesto običnoga polja,
koristiti vector
//Problem veličine polja[] odražava se tek
//pri izlasku iz programa nakon 2. Ctrl+Z
cout<<"Prekid upisa: Ctrl+Z\n";
do{
broj=0;
cout<<"Upisi cio dekadski broj: ";
cin>>broj;
//duzina=strlen(char* broj);
cout<<"Upisi bazu sustava: ";
cin>>baza;
//cout<<" "<<broj<<endl;
if(broj==baza)cout<<"Uz broj==baza, rezultat
```

```
C:\windows\system32\cmd.exe
Upisi dekadski broj: 5
Upisi bazu sustava: 5
Uz broj==baza, rezultat je: 10
Upisi dekadski broj: 4
Upisi bazu sustava: 4
Uz broj==baza, rezultat je: 10
Upisi dekadski broj: 3
Upisi bazu sustava: 3
Uz broj==baza, rezultat je: 10
Upisi dekadski broj: 2
Upisi bazu sustava: 2
Uz broj==baza, rezultat je: 10
Upisi dekadski broj: ^Z
Upisi bazu sustava: 0
^Z
Press any key to continue . . .
```

Pretvorba dekadskoga broja u broj bilo kojega brojevnoga sustava!	

`je: ";`
`cout<<itoa(broj,polje,baza)<<endl;`
`}while(getchar()!=EOF);`
`//putchar(c);`
`}`	

0.3.9. 0.3.0. OBIČAN KOMPLEMENT

Komplementi se koriste u digitalnim uređajima za pojednostavniti operaciju oduzimanja i za rukovanje logičkim operacijama.

Pojednostavnjenjem, dobiju se jeftiniji sklopovi za provedbu operacija. Postoje dvije vrste komplementa za svaki sustav baze *r*: komplement i umanjen (*diminish*) komplement baze. Prvi se naziva *r* komplement, a drugi (*r*–1) komplement. Ako se vrijednost baze *r* zamijeni imenom, dobiju se dvije vrste za svaki brojevni sustav:

- Za binarni sustav (baza 2), komplement su: komplement dvojke (2'*s complement*) i komplement jedinice (1'*s complement*).
- Za dekadski sustav (baza 10), komplementi su: komplement desetke (10'*s complement*) i komplement devetke (9'*s complement*).
- Za oktalni sustav (baza 8), komplementi su: komplement osmice (8'*s complement*) i komplement sedmice (7'*s complement*).
- Za heksadekadski sustav (baza 16), komplementi su: komplement šesnaestice (16'*s complement*) i komplement petnaestice (15'*s complement*).

Običan komplement binarnoga broja (*komplement jedinice*) dobije se zamjenom svih jedinica nulama i obratno. Iznos broja što se dobije na ovaj način računa se prema izrazu:

$$\overline{N}_2 = 2^n - 1 - N_2$$

Običan komplement nije pogodan za izražavanje prirodnih brojeva, jer nije jednoznačno određena vrijednost nule, naime običan komplement od 0000 iznosi 1111, ali služi za jednostavno računanje punoga komplementa, jer vrijedi:

$$N_2' = \overline{N}_2 + 1$$

Ako se $\overline{N}_2$ u ovoj jednadžbi zamijeni prethodnom jednadžbom, dobije se:

$$N_2' = 2^n - 1 + 1 - N_2 \Rightarrow N_2' = 2^n - N_2$$

ili za prethodan primjer: 1111 + 1 = 10000.

Komplement baze nekoga broja u nekome brojevnome sustavu dobije se tako da se znamenke zamijene razlikama do broja baze (2, 8, 10, 16) pa im se pribroji 1, tj. *komplementu umanjene baze* doda se 1.

Komplement devetke dekadskoga broja: 546700 je 999999 – 546700 = 453299.

Komplement desetke dekadskoga broja: 546700 je 453299 + 1 = 453300.

Komplement devetke dekadskoga broja: 012398 je 999999 – 012398 = 987601.

Komplement desetke dekadskoga broja: 012398 je 012398 + 1 = 987699.

Komplement dvojke nekoga binarnoga broja dobije se tako da se jedinice zamijene nulama, a nule jedinicama te se pribroji 1, tj. *komplementu jedinice* doda se 1.

0.3.9.1.   Primjer: Prikaz broja –41 pomoću bita za predznak i komplementa dvojke

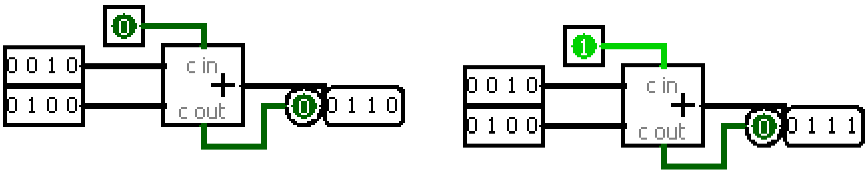
37_{10} = 1 0 1 0 0 1_2 komplement dvojke od 100101 je 011010 = 011011

-37_{10} = 1 1 0 1 0 0 1_2

vrijednost

bit za predznak

Prikaz komplementa dvojke simulacijskim krugom LogiSim dan je u nastavku.



0.3.10.   0.3.10. KOMPLEMENTI BINARNIH BROJEVA

Budući da je baza sustava (*radix*) za binaran broj 10_2, smanjena baza je 1_2. Komplement smanjene baze nađe se oduzimanjem binarnoga broja od 1_2. To se obično zove *komplement jedinice*. Na primjer, komplement jedinice za 0 nađe se kao 1–0 = 1, a komplement jedinice za 1 nađe se kao 1–1 = 0. U biti, komplement jedinice dobiva se jednostavnim obrtanjem (ili "zrcaljenjem") svakoga bita u binarnome broju pa je za 100101_2 komplement jedinice jednak 011010_2. Komplement baze (ili "komplement dvojke") binarnoga broja nađe se tako da se prvo izračuna komplement jedinice, a zatim se broj 1 pribroji tome broju. Komplement jedinice za 101101_2 je 010010_2, a komplementa dvojke za 010010_2+1_2, ili 010011_2.

0.3.10.1.   Komplementi s predznakom

U krugovima koji koriste binarnu matematiku, dizajner krug može se odlučiti za korištenje komplementa za negativne brojeve i odrediti najznačajniji bit kao bit predznaka. Ako je tako, ostali bitovi su vrijednost broja. Ako je najznačajniji bit 1, on označava negativne brojeve, a ako je najznačajniji bit 0, tada je broj pozitivan, a vrijednost broja određuje se uzimanjem komplementa preostalih bitova. Ali ako je najznačajniji bit 1, onda je broj negativan, a vrijednost broja određuje se uzimanjem komplementa jedinice broja. Tako je: 0111 = +7, a 1000 = –7 (komplement jedinice za 1000 je 0111). Sljedeća tablica može pomoći u razjašnjavanju ovoga koncepta:

Tablica 1.5. Komplement jedinice negativnih brojeva

dekadski	pozitivni	negativni
0	0000	1111
1	0001	1110
2	0010	1101
3	0011	1100
4	0100	1011
5	0101	1010
6	0110	1001
7	0111	1000

Binarnim brojem s 4 bita, može se predstaviti bilo koji dekadski broj od –7 do +7. Ali, uočite da poput sustava predznak-i-vrijednost, postoje dvije vrijednosti za 0, jedna pozitivna i jedna negativna. To zahtijeva dodatne sklopove za testiranje obje vrijednosti za 0 nakon operacije poput oduzimanja. Kako bi se pojednostavnilo oblikovanje sklopa, možemo se odlučiti za korištenje *komplementa dvojke* negativnih brojeva te odrediti najznačajniji bit kao bit predznaka. U tome slučaju, ostali bitovi predstavljaju vrijednost broja.

Pri korištenju komplementa dvojke, ako je najznačajniji bit 0, onda je broj pozitivan i vrijednost broja određuje se iz preostalih bitova. Ako je najznačajniji bit 1, onda je broj negativan, a vrijednost broja određuje se uzimanje komplementa svakoga od bitova, a zatim se pribraja 1 (izrada komplementa dvojke). Tako je: 0111 = 7, a 1001 = –7 (komplement jedinice od 1001 je 0110, a komplement dvojke od 0110 je 0110+1 = 0111). Sljedeća tablica može pomoći u razjašnjavanju ovoga koncepta:

Tablica 1.6. Komplement dvojke negativnih brojeva

dekadski	pozitivni	negativni
0	0000	10000
1	0001	1111
2	0010	1110
3	0011	1101
4	0100	1100
5	0101	1011
6	0110	1010
7	0111	1001

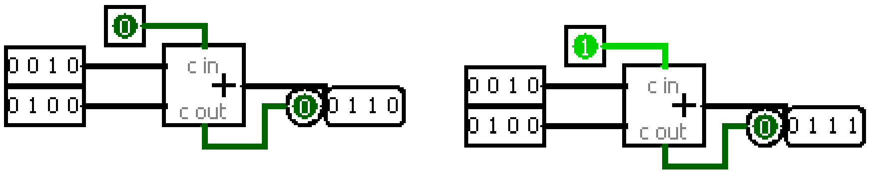
Komplement dvojke uklanja hirovitost označavanja nule na dva načina. Gornja tablica pokazuje da je nula ili 0000 ili 10000; a budući da se pretpostavlja rad s 4 bita, "1" se inicijalno odbacuje, ostavljajući 4 nule (0000) bez obzira je li nula pozitivan ili negativan broj.

0.3.10.2. Izračun komplementa dvojke

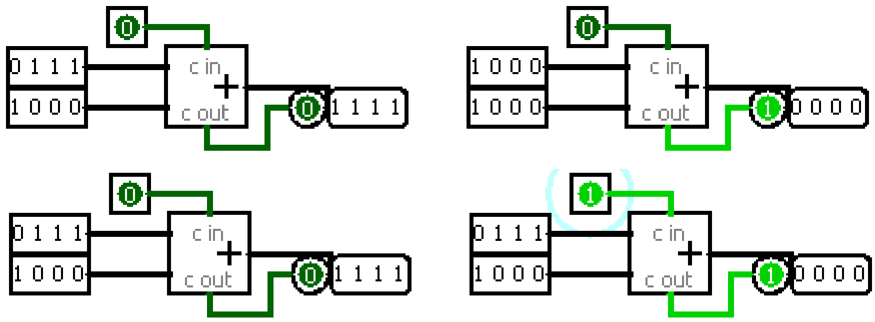
Komplement dvojke - korijen (*radix*) računa se pomoću komplementa jedinice broja i nakon toga dodavanjem jedinice. Za strojeve, to je najučinkovitiji način izračuna. Postoji način koji je puno lakši za ljude kada koriste izračun komplementa dvojke nekoga broja. Započne se najmanje značajnim bitom (bit sasvim desno), a zatim se čita broj s desna u lijevo. Potraži se prva "1", a zatim preokrene svaki bit lijevo od te "1".

Zbrajalo je napravljeno tako da prvi broj što se zbraja ulazi na ulaz lijevo gore, a drugi broj ulazi na ulaz lijevo dolje.

Utjecaj i ponašanje bita posudbe (*Carry in*)



Utjecaj i ponašanje bita pretek-prijenos (*Carry out*)



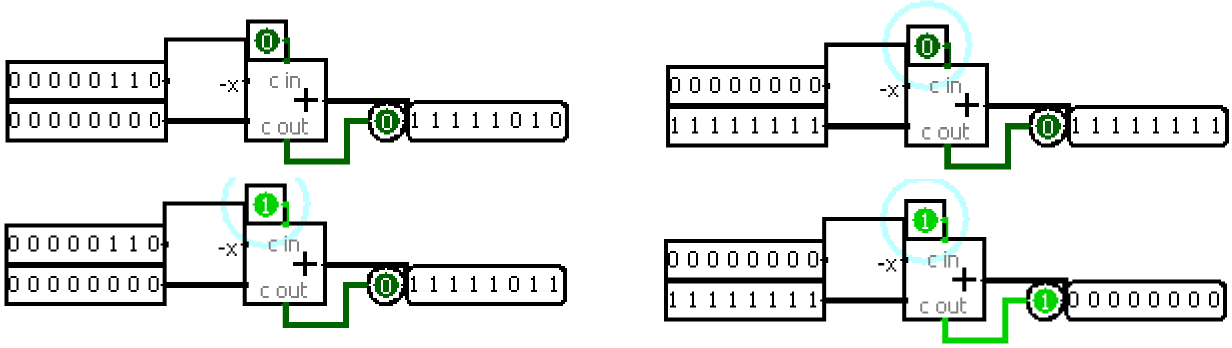
Ulaz "*bita za posudbu*" (***carry in bit***) je na vrhu (on se zbraja s bitom postavljenim sasvim desno na izlazu). Primjena je u *punome komplementu* ili *komplementu dvojke*, a koristi se za označavanje ***negativnih*** brojeva. *Pun komplement* izračuna se tako da se najprije izračuna *običan komplement* te mu se doda jedinica.

komplement dvojke broja 6:

```
00000110   +6
11111001   običan komplement od +6
     + 1   dodaj 1
11111010   -6 u komplementu dvojke
```

pun komplement od 0 iznosi:

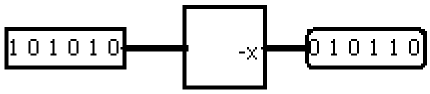
```
 00000000   0
 11111111   običan komplement od 0
      + 1   dodaj 1
100000000   0 u komplementu dvojke
```



U navedenome primjeru *punoga komplementa* od 0, jedinica predstavlja deveti bit, te se stoga u sustavu od 8 bitova odbacuje pa rezultat opet predstavlja nulu. *Komplement dvojke* omogućuje jednoznačno određivanje nule i zbog toga je izabran za predstavljanje cijelih brojeva, a ne običan komplement (komplement jedinice).

U LogiSimu, bit, što se kao rezultat zbrajanja prenosi i naznačuje da je potrebno osigurati još jedno binarno mjesto - npr. u registru (***carry out bit***), izlazi na dnu. Zbroj dvaju brojeva pojavljuje se u čvoru na desnoj strani.

Kao primjer, komplement dvojke broja 101010 oblikuje se počevši od najmanje značajnoga bita (0 na desnoj strani) i obrađuju se bitovi prema lijevoj strani, u potrazi za prvom 1, koja je u ovome primjeru na drugome mjestu s desne strane.

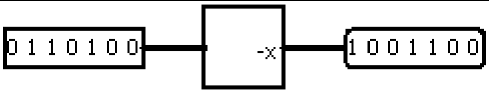
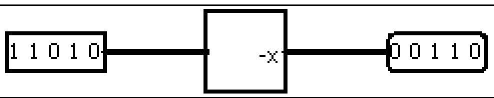
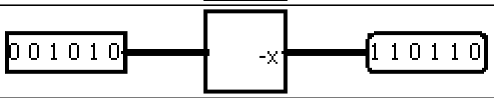
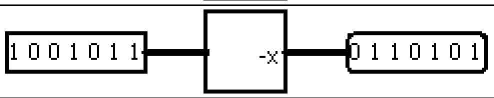
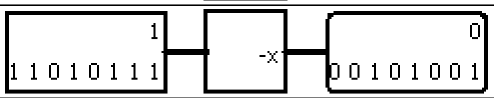


Slika:

Svaki bit u lijevome stupcu obrće se i ukupan vektor završava kao 010110.

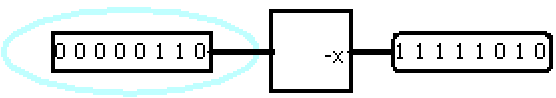
Evo još nekoliko primjera:

Tablica 1.7. Komplementi dvojke

Broj	LogiSim prikaz	Komplement dvojke
0110100		1001100
11010		00110
001010		110110
1001011		0110101
111010111		000101001

0.3.10.3. *Primjer: Računanje komplementa dvojke ulaznoga broja*

Logisim uključuje uređaj koji računa komplement dvojke ulaznoga broja. Zove se "negator" i nalazi se u mapi "Aritmetika". On ima oznaku "-x" na svome istočnome dijelu.



0.3.11. 0.3.11. PREGLED KLJUČNIH POJMOVA

binarna znamenka (*binary digit-bit*)

- znamenka binarnoga brojevnoga sustava

binarna znamenka najnižega brojevnoga mjesta LSB (*least significant bit*)

- binarna znamenka najmanje vrijednosti brojevnoga mjesta, krajnja desna znamenka binarnoga broja

binarna znamenka najvišega brojevnoga mjesta MSB (*most significant bit*)

- binarna znamenka najveće vrijednosti brojevnoga mjesta, krajnja lijeva znamenka binarnoga broja

binarni brojevni sustav (*binary number system*)

- brojevni sustav s dvije znamenke: 0 i 1

binarni zarez (*binary point*)

- zarez koji razdvaja cjelobrojna od razlomljenih mjesta u binarnom brojevnom sustavu

brojno mjesto

- položaj znamenke u bilo kojemu broju

dekadski brojevni sustav (*decimal number system*)

- brojevni sustav s deset znamenaka

digitalna elektronika (*digital electronics*)

- područje elektronike u kojemu signali mogu imati dva iznosa kojima se pridružuju znamenke 0 i 1 što omogućuje prikaz podataka u brojčanome obliku binarnoga brojevnoga sustava

komplement dvojke (2*'s-complement*)

- ili komplement do baze, dobije se tako da se komplementu jedinice doda jedinica

heksadekadski brojevni sustav (*hexadecimal number system*)

- brojevni sustav sa 16 znamenaka

oktalni brojevni sustav (*octal number system*)

- brojevni sustav s 8 znamenaka

komplement jedinice (1*'s-complement*)

- ili komplement do najvećega broja, u binarnome brojevnome sustavu dobije se međusobnom zamjenom nula i jedinica.

težina brojevnoga mjesta

- vrijednost brojevnoga mjesta, može se prikazati kao potencija osnovice (baze) brojevnoga sustava

znamenka najnižega brojevnoga mjesta (*least significant digit*)

- znamenka najmanje vrijednosti (težine) brojevnoga mjesta, krajnja desna znamenka broja bilo kojega brojevnoga sustava

znamenka najnižega brojevnoga mjesta LSD (*least significant digit*)

- znamenka najmanje vrijednosti (težine) brojevnoga mjesta, krajnja desna znamenka broja bilo kojega brojevnoga sustava

znamenka najvišega brojevnoga mjesta MSD (*most significant digit*)

- znamenka najviše vrijednosti (težine) brojevnoga mjesta, krajnja lijeva znamenka broja bilo kojega brojevnoga sustava

Tablica 1.8. Pregled brojevnih sustava

Brojevni sustav	Dekadski	Binarni	Oktalni	Heksadekadski
Znamenke	0	0	0	0

Brojevni sustav	Dekadski	Binarni	Oktalni	Heksadekadski
	1	1	1	1
	2		2	2
	3		3	3
	4		4	4
	5		5	5
	6		6	6
	7		7	7
	8			8
	9			9
				A
				B
				C
				D
				E
				F
Baza sustava	10	2	8	16
	$10^0 = 1$	$2^0 = 1$	$8^0 = 1$	$16^0 = 1$
	$10^1 = 10$	$2^1 = 2$	$8^1 = 8$	$16^1 = 16$
	$10^2 = 100$	$2^2 = 4$	$8^2 = 64$	$16^2 = 256$
	$10^3 = 1000$	$2^3 = 8$	$8^3 = 512$	$16^3 = 4096$
Prvih 16 brojeva	0	0	0	0
Težine mjesta	1	1	1	1
	2	10	2	2
	3	11	3	3
	4	100	4	4
	5	101	5	5
	6	110	6	6
	7	111	7	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
	16	10000	20	10

Tablica 1.9. Prikaz relativnih brojeva

Dekadski broj	Prikaz pomoću binarnoga broja	Prikaz pomoću komplementa jedinice	Prikaz pomoću komplementa dvojke
	$P2^3 2^2 2^1 2^0$	$P2^3 2^2 2^1 2^0$	$P2^3 2^2 2^1 2^0$
15	0 1 1 1 1	0 1 1 1 1	0 1 1 1 1
14	0 1 1 1 0	0 1 1 1 0	0 1 1 1 0
13	0 1 1 0 1	0 1 1 0 1	0 1 1 0 1
12	0 1 1 0 0	0 1 1 0 0	0 1 1 0 0
11	0 1 0 1 1	0 1 0 1 1	0 1 0 1 1
10	0 1 0 1 0	0 1 0 1 0	0 1 0 1 0
9	0 1 0 0 1	0 1 0 0 1	0 1 0 0 1
8	0 1 0 0 0	0 1 0 0 0	0 1 0 0 0
7	0 0 1 1 1	0 0 1 1 1	0 0 1 1 1
6	0 0 1 1 0	0 0 1 1 0	0 0 1 1 0
5	0 0 1 0 1	0 0 1 0 1	0 0 1 0 1
4	0 0 1 0 0	0 0 1 0 0	0 0 1 0 0
3	0 0 0 1 1	0 0 0 1 1	0 0 0 1 1
2	0 0 0 1 0	0 0 0 1 0	0 0 0 1 0
1	0 0 0 0 1	0 0 0 0 1	0 0 0 0 1
0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
–1	1 0 0 0 1	1 1 1 1 0	1 1 1 1 1
–2	1 0 0 1 0	1 1 1 0 1	1 1 1 1 0
–3	1 0 0 1 1	1 1 1 0 0	1 1 1 0 1
–4	1 0 1 0 0	1 1 0 1 1	1 1 1 0 0

Dekadski broj	Prikaz pomoću binarnoga broja	Prikaz pomoću komplementa jedinice	Prikaz pomoću komplementa dvojke
	$P2^32^22^12^0$	$P2^32^22^12^0$	$P2^32^22^12^0$
–5	1 0 1 0 1	1 1 0 1 0	1 1 0 1 1
–6	1 0 1 1 0	1 1 0 0 1	1 1 0 1 0
–7	1 0 1 1 1	1 1 0 0 0	1 1 0 0 1
–8	1 1 0 0 0	1 0 1 1 1	1 1 0 0 0
–9	1 1 0 0 1	1 0 1 1 0	1 0 1 1 1
–10	1 1 0 1 0	1 0 1 0 1	1 0 1 1 0
–11	1 1 0 1 1	1 0 1 0 0	1 0 1 0 1
–12	1 1 1 0 0	1 0 0 1 1	1 0 1 0 0
–13	1 1 1 0 1	1 0 0 1 0	1 0 0 1 1
–14	1 1 1 1 0	1 0 0 0 1	1 0 0 1 0
–15	1 1 1 1 1	1 0 0 0 0	1 0 0 0 1

0.3.12. 0.3.12. PITANJA I ZADACI ZA PONAVLJANJE

1. Objasnite razlike između dekadskoga i binarnoga brojevnoga sustava.

2. Koji se najveći broj (izražen dekadsko) može napisati s 8 znamenaka binarnoga sustava?

3. Koliko je binarnih znamenaka potrebno za prikaz dekadskoga broja 47?

4. Prikažite binarni signal 11010 u serijskome i paralelnome obliku.

5. Pretvorite binarni broj 110110 u dekadski.

6. Pretvorite dekadski broj 53 u binarni.

7. Navedite značajke oktalnoga brojevnoga sustava.

8. Koji se najveći broj (izražen dekadski) može napisati s četiri znamenke oktalnoga sustava?

9. Pretvorite oktalni broj 465 u dekadski.

10. Pretvorite dekadski broj 372 u oktalni.

11. Pretvorite oktalni broj 524 u binarni.

12. Pretvorite binarni broj 11001110 u oktalni.

13. Navedite značajke heksadekadskoga brojevnoga sustava,

14. Koji se najveći broj (izražen dekadski) može napisati s tri znamenke heksadekadskoga sustava?

15. Koje su težine mjesta znamenaka najnižega i najvišega brojevnoga mjesta heksadekadskoga broja A3F,5D?

16. Pretvorite heksadekadski broj 12BF u dekadski.

17. Pretvorite dekadski broj 3127 u heksadekadski.

18. Pretvorite heksadekadski broj 2C4E u binarni.

19. Pretvorite binarni broj 100101000111111 u heksadekadski.

20. Pretvorite heksadekadski broj D3A u oktalni.

21. Pretvorite oktalni broj 5437 u heksadekadski.

22. Prikažite broj –25 pomoću binarnoga broja, komplementa jedinice i komplementa dvojke.

0.4. 0.4. Logisim_.doc

0.5. 0.5. Flip Flop

Flip Flop.doc (kao priručnik)
Digitalna elektronika. D-bistabil, LFSR. XOR vrata i paritet.

2 D bistabila.circ
D-bistabil (vidi: Flip Flop.doc), LFSR
XOR vrata i paritet. XOR vrata.circ

0.6. 0.6. Univerzalan posmični Registar za 4 bita

Univerzalan posmični Registar za 4 bita.doc (kratak priručnik: MOODLE.pdf)
Arhitektura kodera pomoću LFSR.doc (napraviti dio vježbe)

LFSR (vidi: Flip Flop.doc)