

Aritmetičko-logička jedinica (ALU)

Funkcije aritmetičko-logičke jedinice

- Funkcijska jedinica u računalu koja se bavi procesiranjem podataka
- višefunkcijski digitalni sklop
 - izvodi operacije nad operandima
 - računa efektivne adrese operanada
- Izvodi osnovne aritmetičke i logičke operacije
- Aritmetičke operacije:
 - Zbrajanje
 - Oduzimanje
 - Množenje
 - Dijeljenje
- Logičke operacije:
 - I, ILI, NE, ISKLJUČIVO ILI
- Posmak

Poluzbrajalo (eng. Half-Adder)

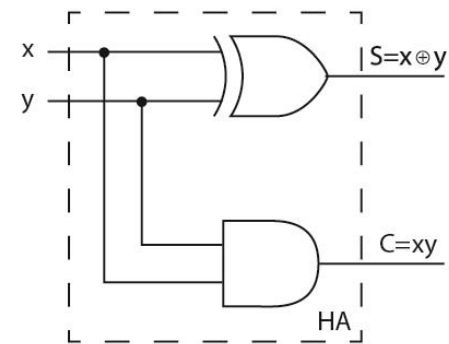
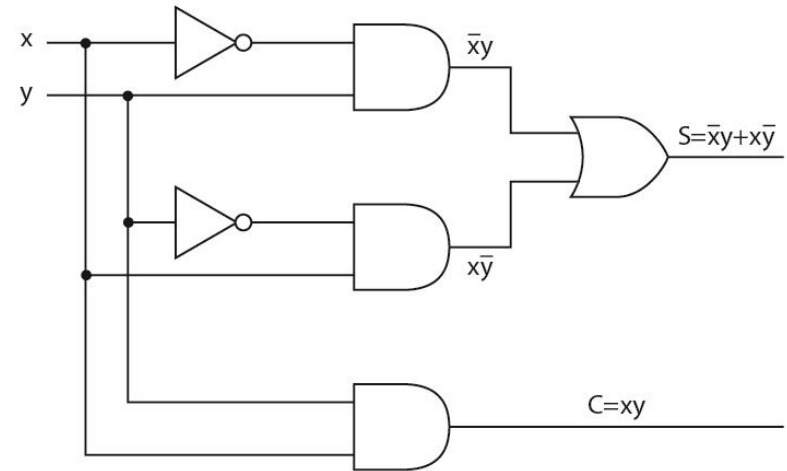
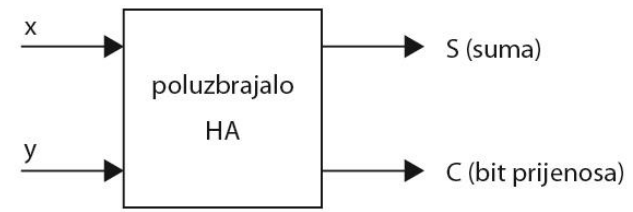
Zbrajanje dvaju binarnih znamenaka:

PRAVILO:

$$\begin{aligned} 0 + 0 &= 0 \\ 0 + 1 &= 1 \\ 1 + 0 &= 1 \\ 1 + 1 &= \mathbf{1} 0 \end{aligned}$$

X	Y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$S = \bar{x}y + x\bar{y} = x \oplus y$$
$$C = xy$$



Zbrajanje višebitnih binarnih brojeva

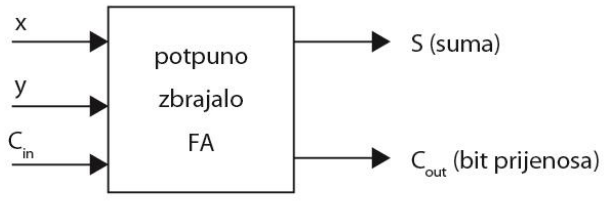
Npr:

$$\begin{array}{r} 1011 \\ +0010 \\ \hline 1101 \\ \mathbf{1} \end{array}$$

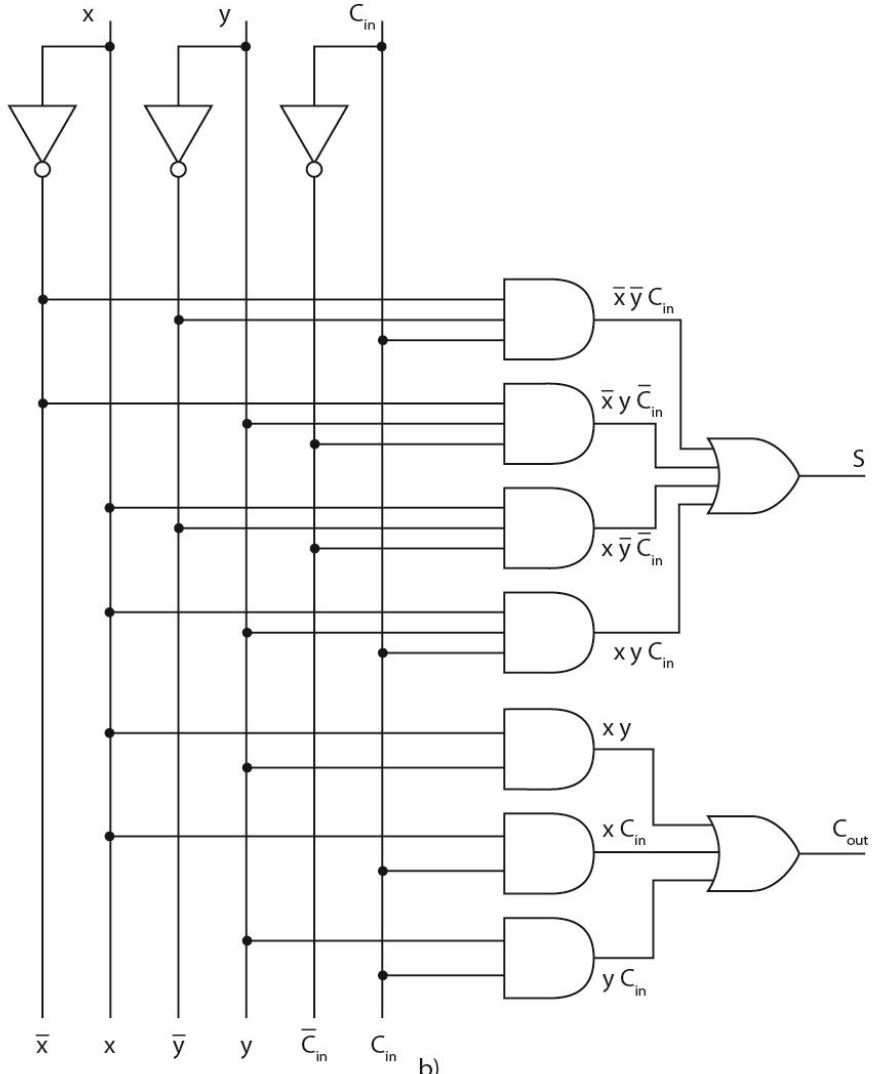
- Na svakoj bitovnoj poziciji može se pojaviti **prijenos („1 dalje“)**
- Prijenos treba uzeti u obzir na sljedećoj bitovnoj poziciji
- Dakle, u općem slučaju, treba nam zbrajanje triju binarnih znamenaka:
 - bit prvog podatka
 - bit drugog podatka
 - prijenos sa prethodne bitovne pozicije

Potpuno zbrajalo (eng. Full-Adder)

X	Y	Cin	Cout	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



a)



b)

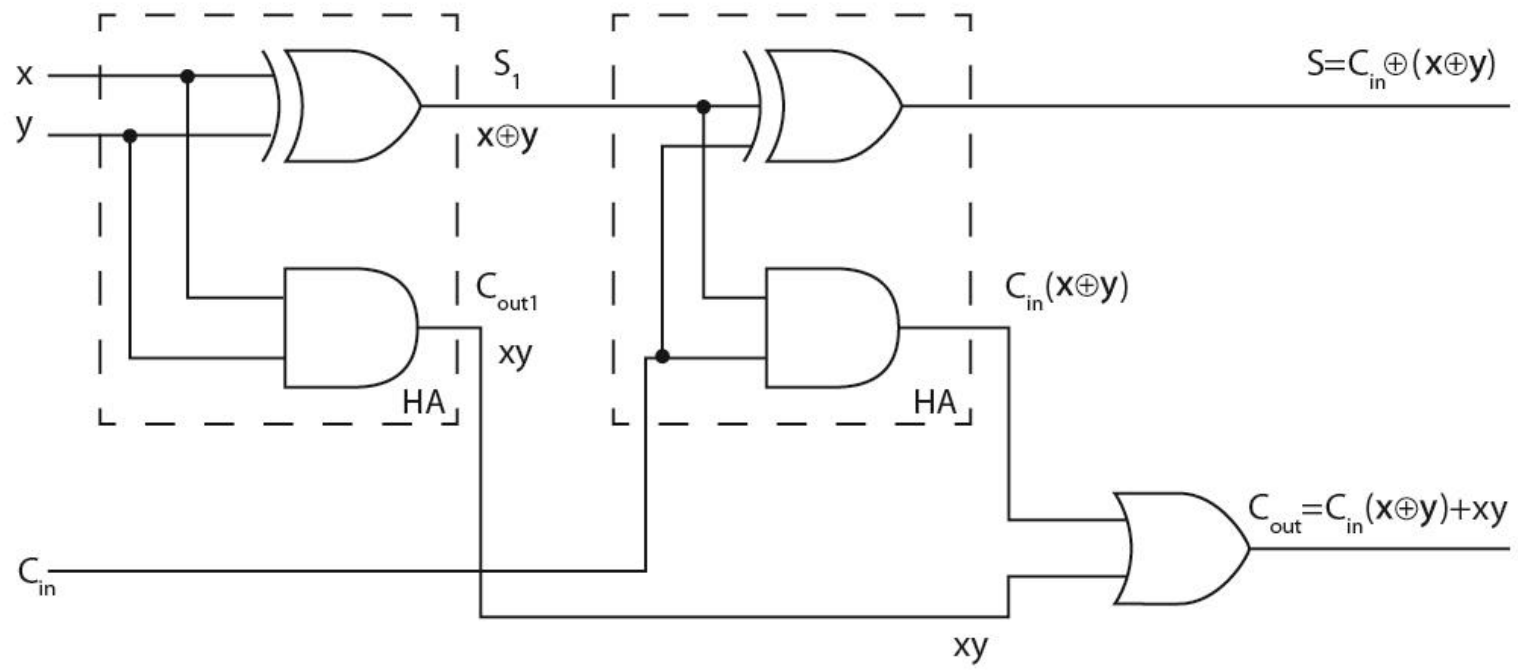
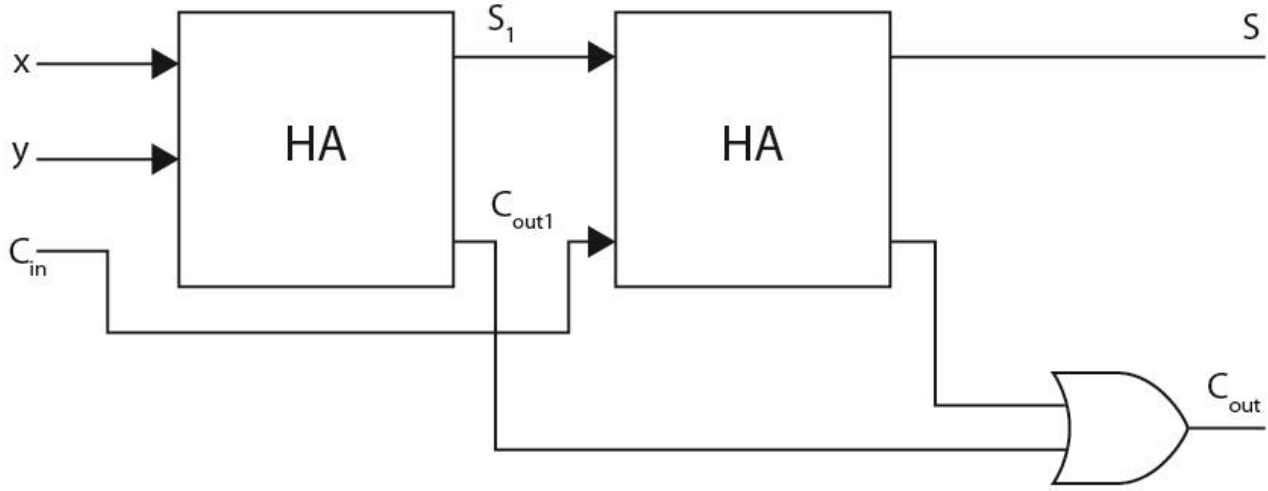
$$S = \bar{x}\bar{y}c_{in} + \bar{x}y\bar{c}_{in} + x\bar{y}\bar{c}_{in} + xyc_{in}$$

$$C_{out} = \bar{x}yc_{in} + x\bar{y}c_{in} + xyc_{in} + xyc$$

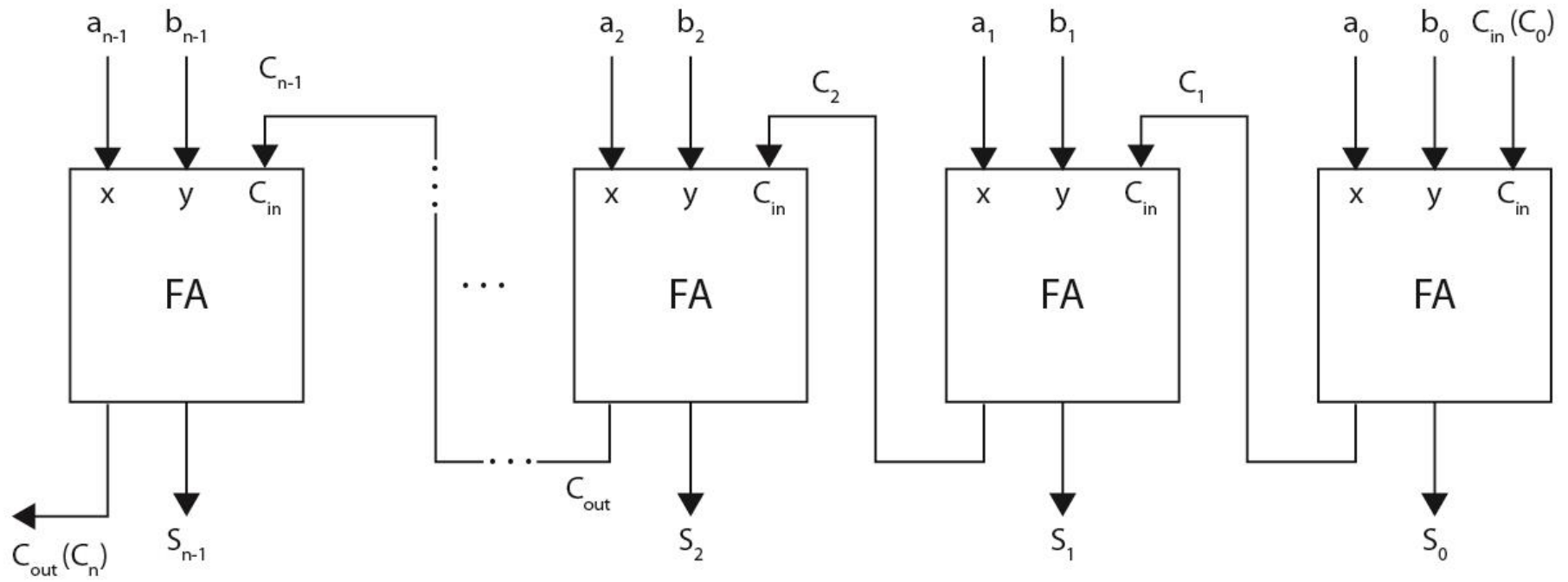
⇓

$$C_{out} = xy + xC_{in} + yC_{in}$$

Potpuno zbrajalo – implementacija s dva poluzbrajala



Paralelno zbrajalo



Poluoduzimalo (eng. Half-Subtractor)

Oduzimanje dvaju binarnih brojeva:

PRAVILO:

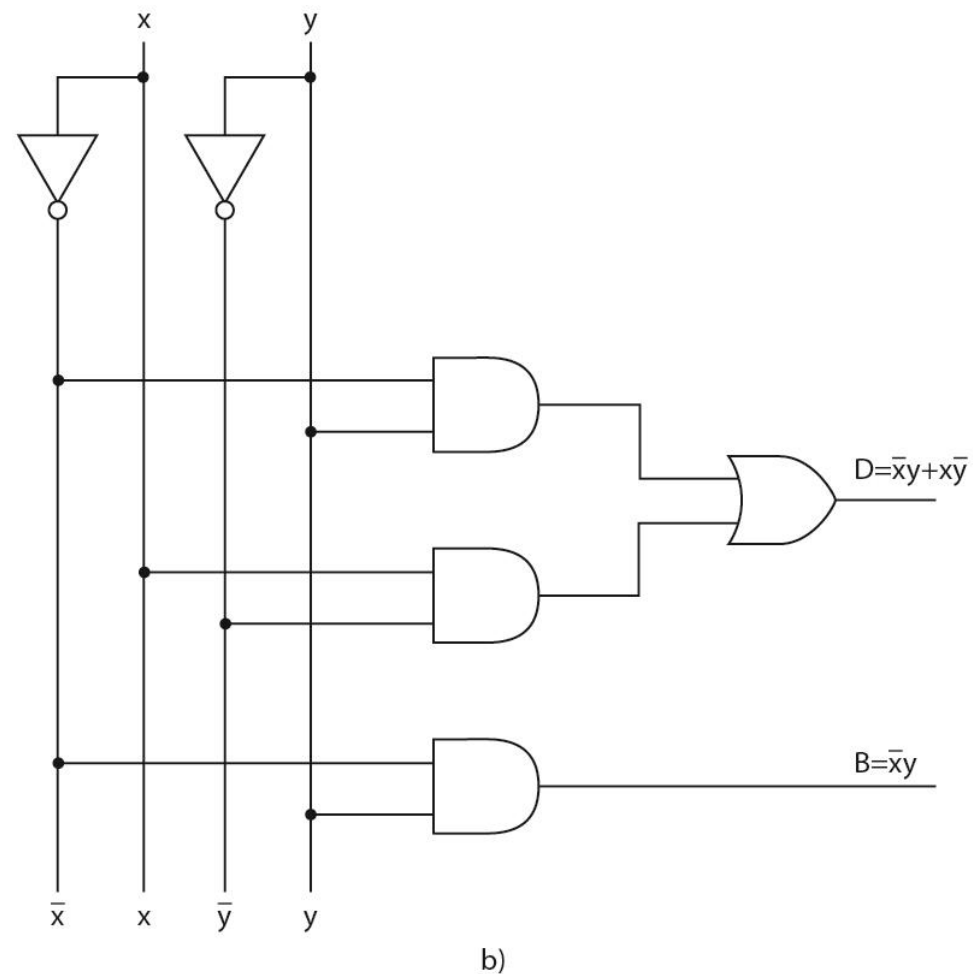
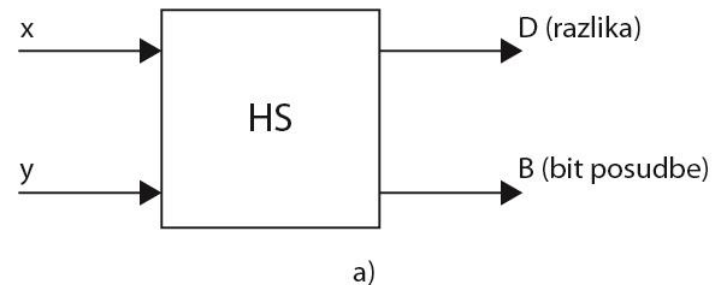
0	-	0	=	0
1	-	0	=	1
1	-	1	=	0
0	-	1	=	1 uz posudbu

1 s značajnije bitovne pozicije

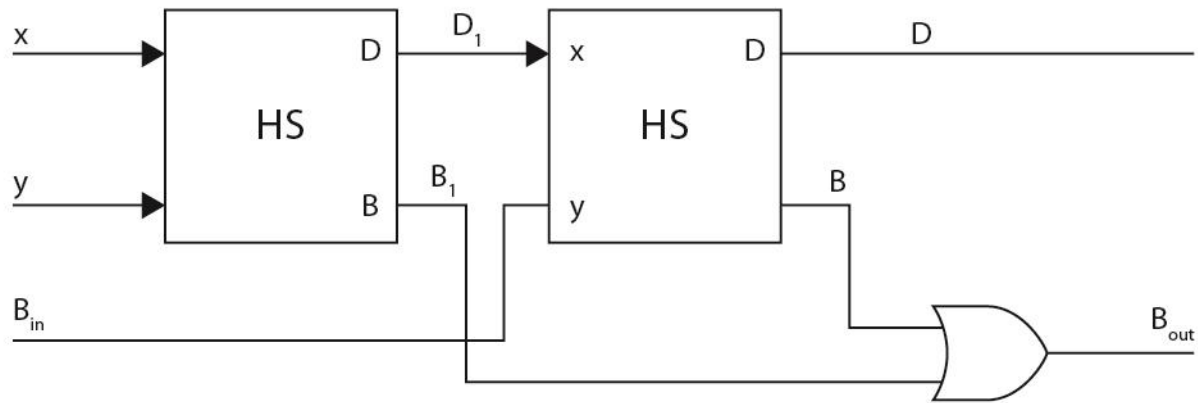
X	Y	D	B
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

$$D = \bar{x}y + x\bar{y} = x \oplus y$$

$$B = \bar{x}y$$



Potpuno oduzimalo (eng. Full-Subtractor)



Potpuno oduzimalo (eng. Full-Subtractor)

Češće se, ipak, umjesto izvedbe zasebnog sklopa, oduzimanje ostvaruje kao zbrajanje s dvojnim komplementom suptrahenda

- Npr. registri: $A = 3$, $B = 2$

$$A : 00000011 \quad B : 00000010 \quad \overline{B}_2 = 11111110$$

$$A + \overline{B}_2 : \begin{array}{|c|} \hline 00000011 \\ \hline \end{array}$$

$$+ \begin{array}{|c|} \hline 11111110 \\ \hline \end{array}$$

$$1 \begin{array}{|c|} \hline 00000001 \\ \hline \end{array}$$

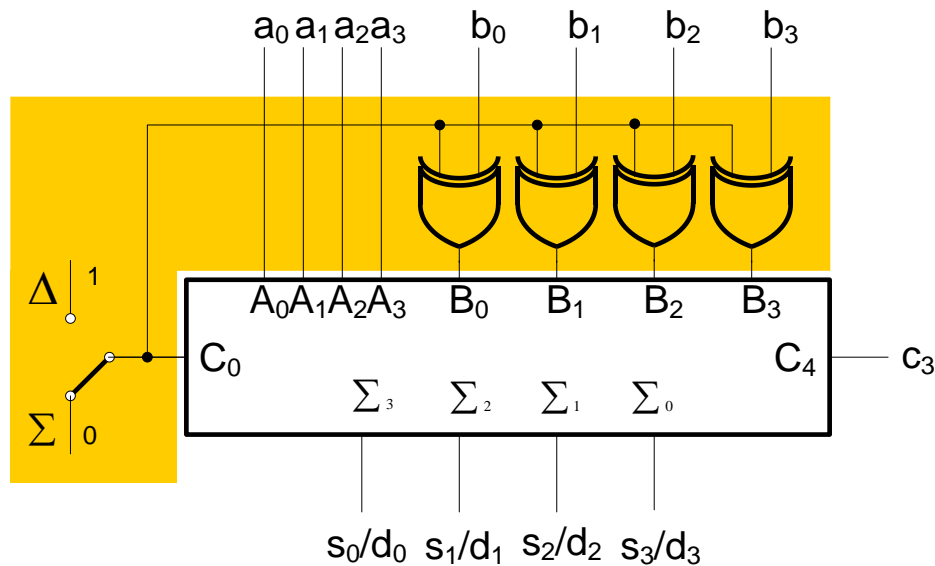
← traženi rezultat

↑
ne stane u registar – preljev!

složenost operacije:
2 x zbrajanje
1 x inverzija bitova

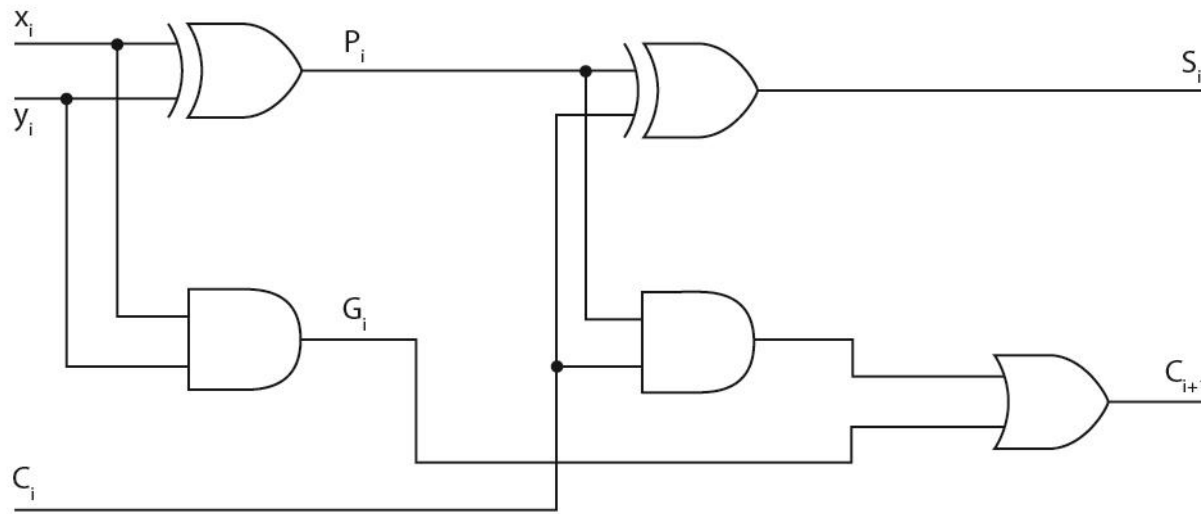
Potpuno oduzimalo (eng. Full-Subtractor)

- potpuno odbijalo *s dvojnim komplementom*
~ binarno zbrajalo + *dodatno sklopovlje*
- Sklop „Isključivo ILI” = upravljivi invertor
($b_i \text{ XOR } 0 = b_i$; $b_i \text{ XOR } 1 = \overline{b_i}$)



$$c_0 = \begin{cases} 0 & \Sigma & a_3 a_2 a_1 a_0 + b_3 b_2 b_1 b_0 = s_3 s_2 s_1 s_0 \\ 1 & \Delta & a_3 a_2 a_1 a_0 - b_3 b_2 b_1 b_0 = d_3 d_2 d_1 d_0 \end{cases}$$

Paralelno zbrajalo s predviđanjem bita prijenosa (engl. Carry lookahead)



$P_i = x_i \oplus y_i$ signal širenja (propagacije) bita prijenosa

$G_i = x_i y_i$ signal generiranja bita prijenosa

$S_i = P_i \oplus C_i$ suma

$C_{i+1} = G_i + (P_i C_i)$ izlazni signal bita prijenosa koji se prenosi stupnju $i+1$

Sklop za predviđanje bita prijenosa

$$C_{i+1} = G_i + (P_i C_i)$$

$$C_{i+2} = G_{i+1} + (P_{i+1} C_{i+1})$$

$$C_{i+2} = G_{i+1} + (P_{i+1} (G_i + (P_i C_i)))$$

$$C_{i+2} = G_{i+1} + P_{i+1} G_i + P_{i+1} P_i C_i$$

$$C_{i+3} = G_{i+2} + (P_{i+2} C_{i+2})$$

$$C_{i+3} = G_{i+2} + P_{i+2} (G_{i+1} + P_{i+1} G_i + P_{i+1} P_i C_i)$$

$$C_{i+3} = G_{i+2} + P_{i+2} G_{i+1} + P_{i+2} P_{i+1} G_i + P_{i+2} P_{i+1} P_i C_i$$

Sklop za predviđanje bita prijenosa – implementacija

$$C_1 = G_0 + (P_0 C_0)$$

$$C_2 = G_1 + (P_1 C_1)$$

$$C_2 = G_1 + (P_1(G_0 + (P_0 C_0)))$$

$$C_2 = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$C_3 = G_2 + (P_2 C_2)$$

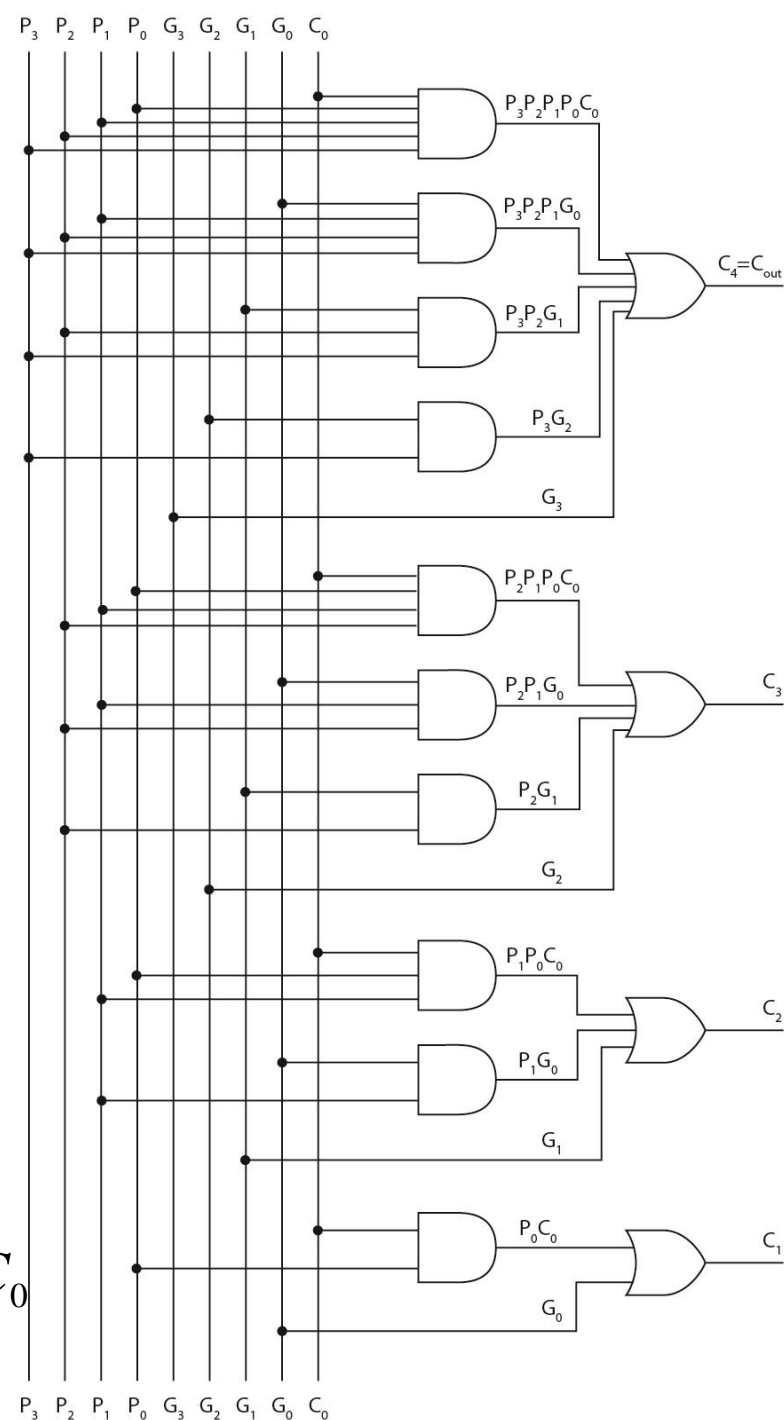
$$C_3 = G_2 + P_2(G_1 + P_1 G_0 + P_1 P_0 C_0)$$

$$C_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0$$

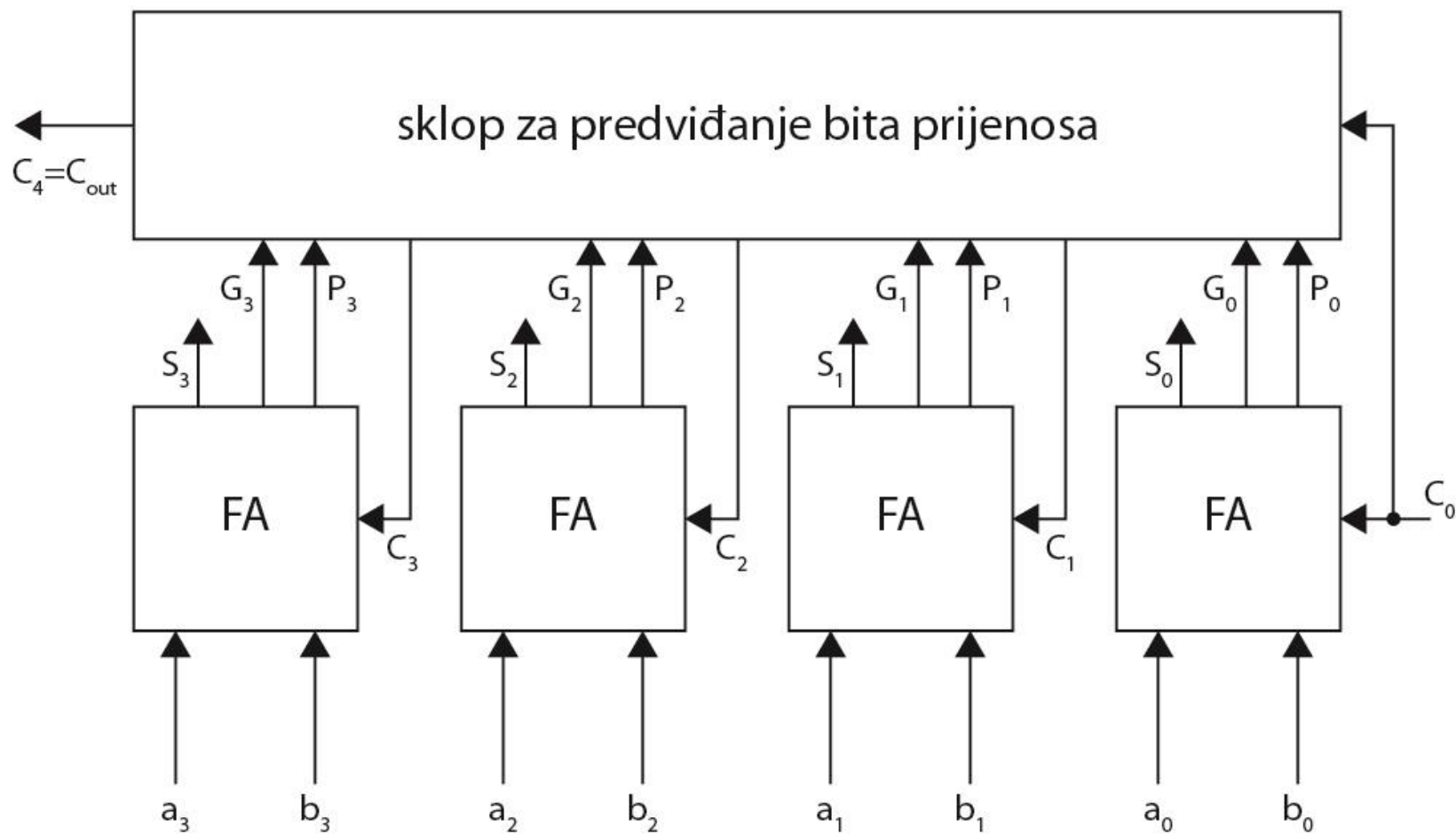
$$C_4 = G_3 + (P_3 C_3)$$

$$C_4 = G_3 + P_3(G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0)$$

$$C_4 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0$$

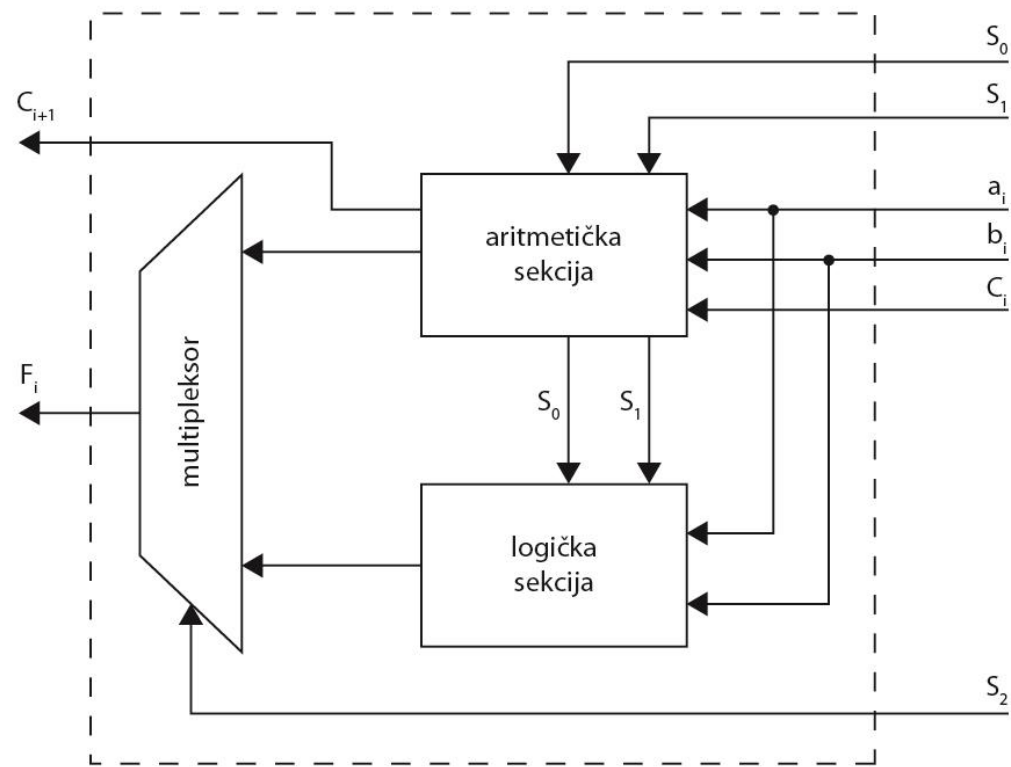


Izvedba četverobitnog paralelnog zbrajala sa sklopom za predviđanje bita prijenosa

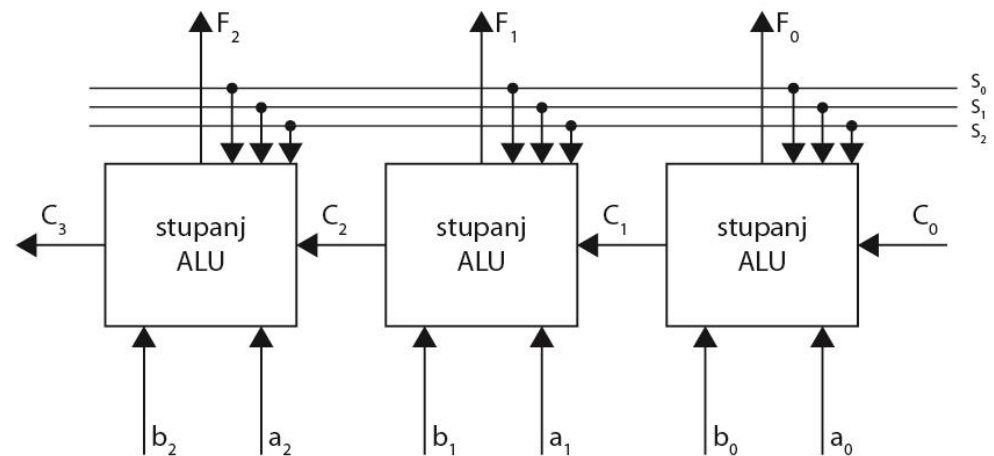


Analiza slučaja: Izgraditi jedan stupanj ALU

- Aritmetička sekcija
- Logička sekcija
- Odabir operacije:
signali S_2, S_1, S_0
 - S_2 – aritmetička ili logička operacija
 - S_1, S_0 – jedna od četiriju aritmetičkih ili četiriju logičkih operacija



a)



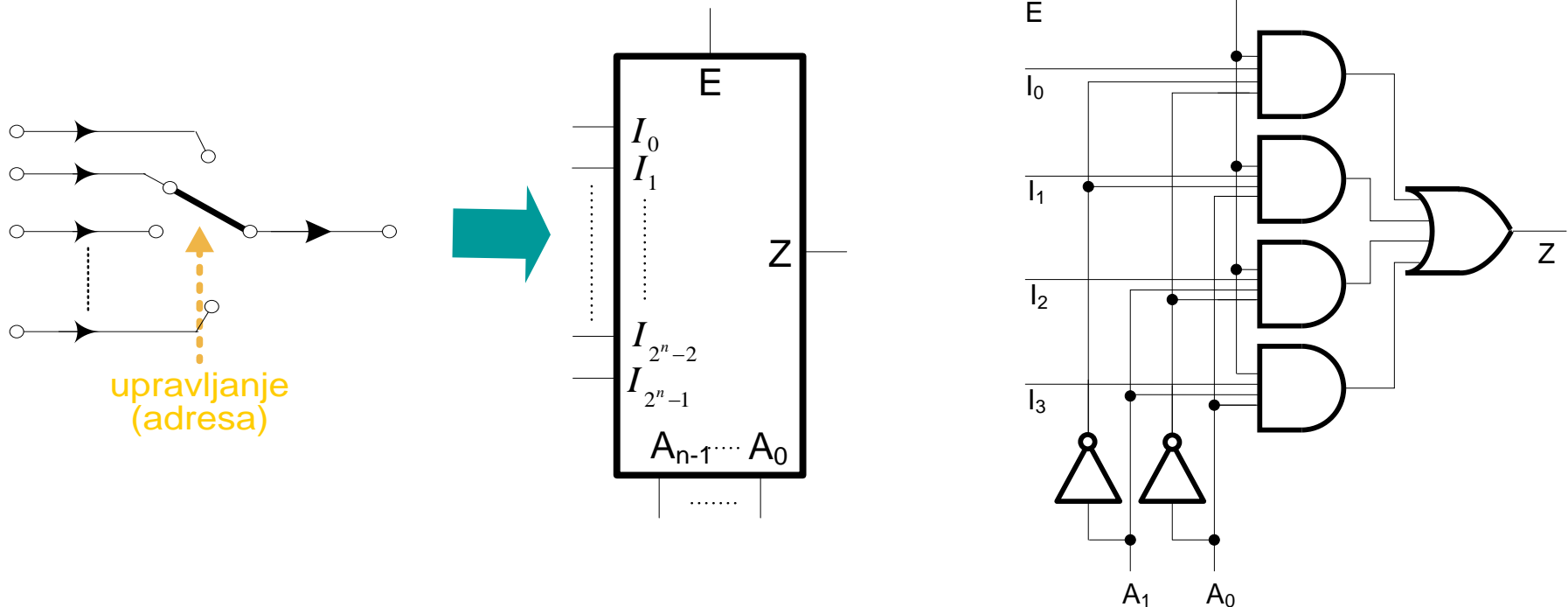
b)

- **multipleksor:**

- odabir podatka koji će se proslijediti na izlaz
~ "multipleksiranje"
- funkcija *upravljane preklopke* (engl. rotary switch)

- **Slika:**

- Lijevo – funkcija
- Sredina – simbol
- Desno – realizacija



Aritmetička sekcija

Izlazi potpunog zbrajala mogu se opisati sljedećim Booleovim jednadžbama:

$$F_i = C_{in} \oplus (x_i \oplus y_i)$$

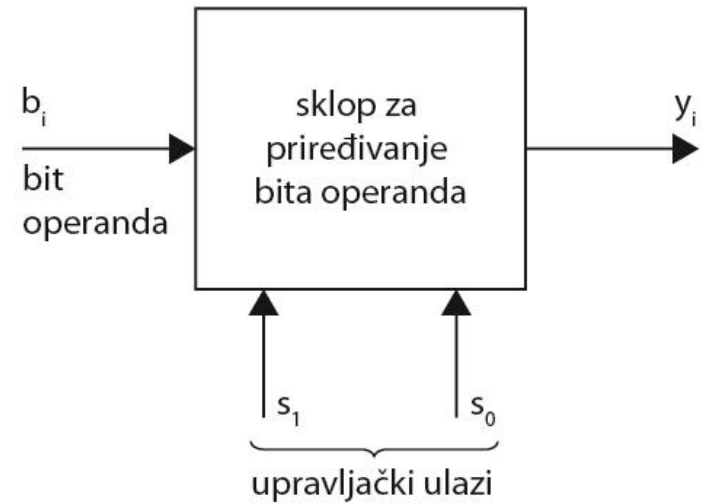
$$C_{i+1} = C_{in} (x_i \oplus y_i) + x_i y_i$$

Želimo proširiti funkcionalnost potpunog zbrajala prilagodbom ulaza y_i

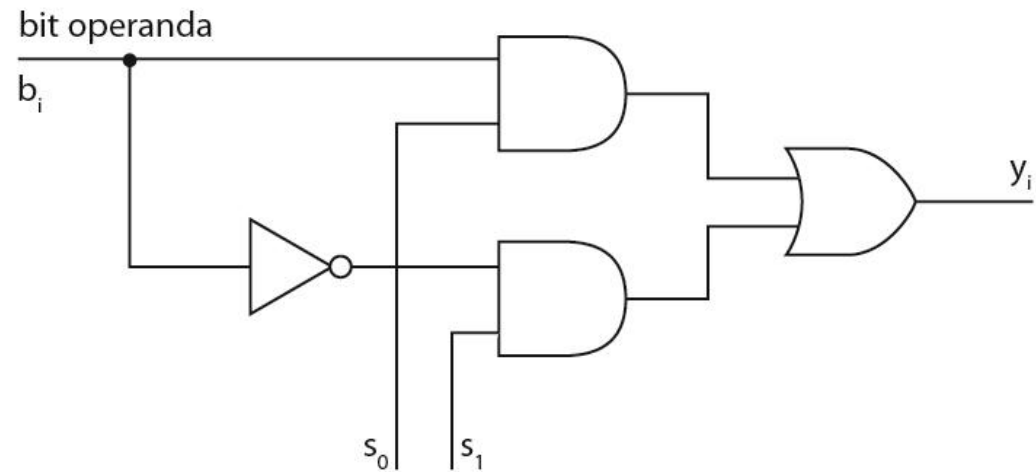
X	Y	C_{in}	Izlaz F
X	Y	0	Zbrajanje: $F=X+Y$
X	Y	1	Zbrajanje s bitom prijenosa: $F=X+Y+1$
X	Jedinični komplement: \bar{Y}	0	$F=X+\bar{Y}$
X	Jedinični komplement: \bar{Y}	1	Oduzimanje: $F=X+\bar{Y}+1$
X	0	0	Prijenos: $F=X$
X	0	1	Inkrementiranje: $F=X+1$
X	Sve jedinice: 111...111	0	Dekrementiranje: $F=X-1$
X	Sve jedinice: 111...111	1	Prijenos: $F=X$

Sklop za priređivanje operanda y_i

S_1	S_0	Ulaz: bit operanda	Izlaz iz sklopa za priređivanje
0	0	b_i	0
0	1	b_i	b_i
1	0	b_i	$\overline{b_i}$
1	1	b_i	1

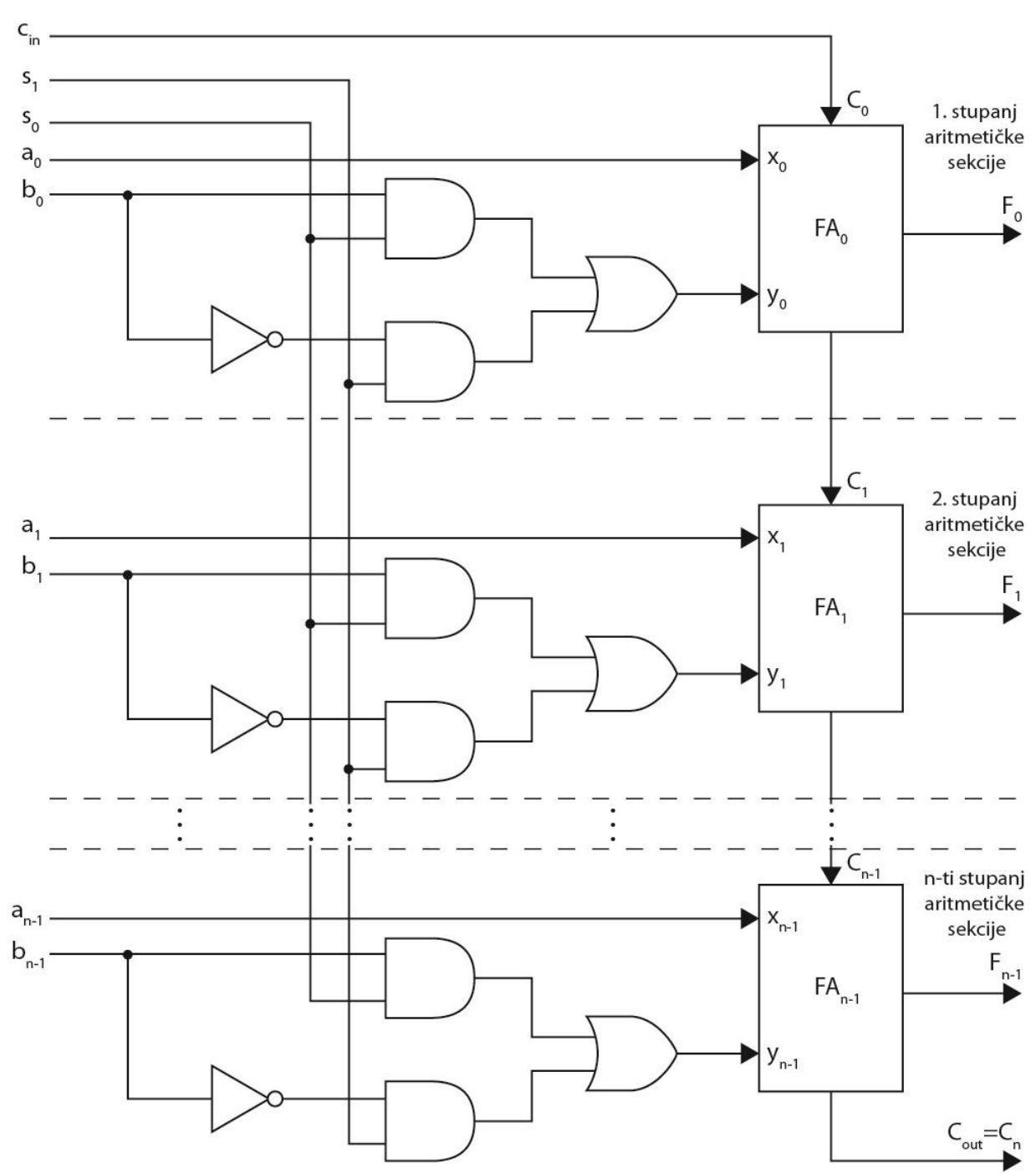


a)



b)

Aritmetička sekcija ALU



Logička sekcija ALU

Jednostavna logička sekcija – npr. podržava četiri osnovne logičke operacije:
I, ILI, Isključivo ILI, NE

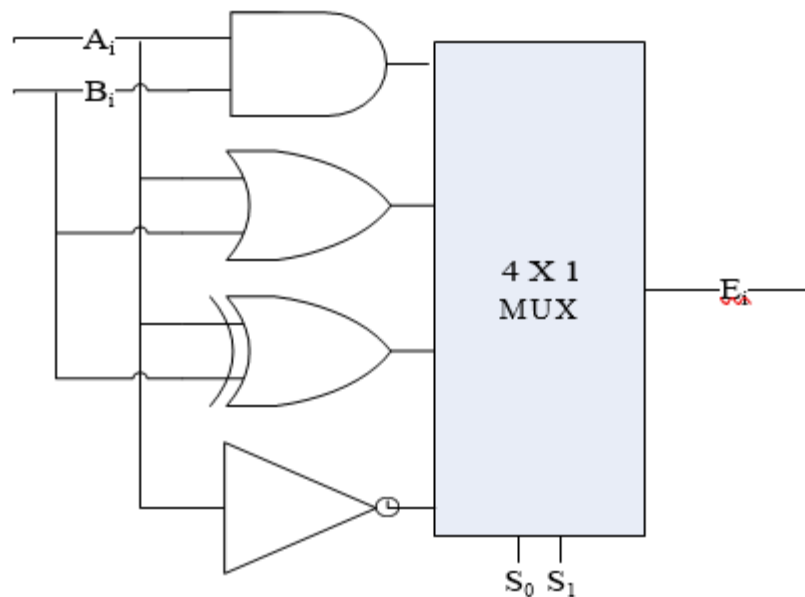


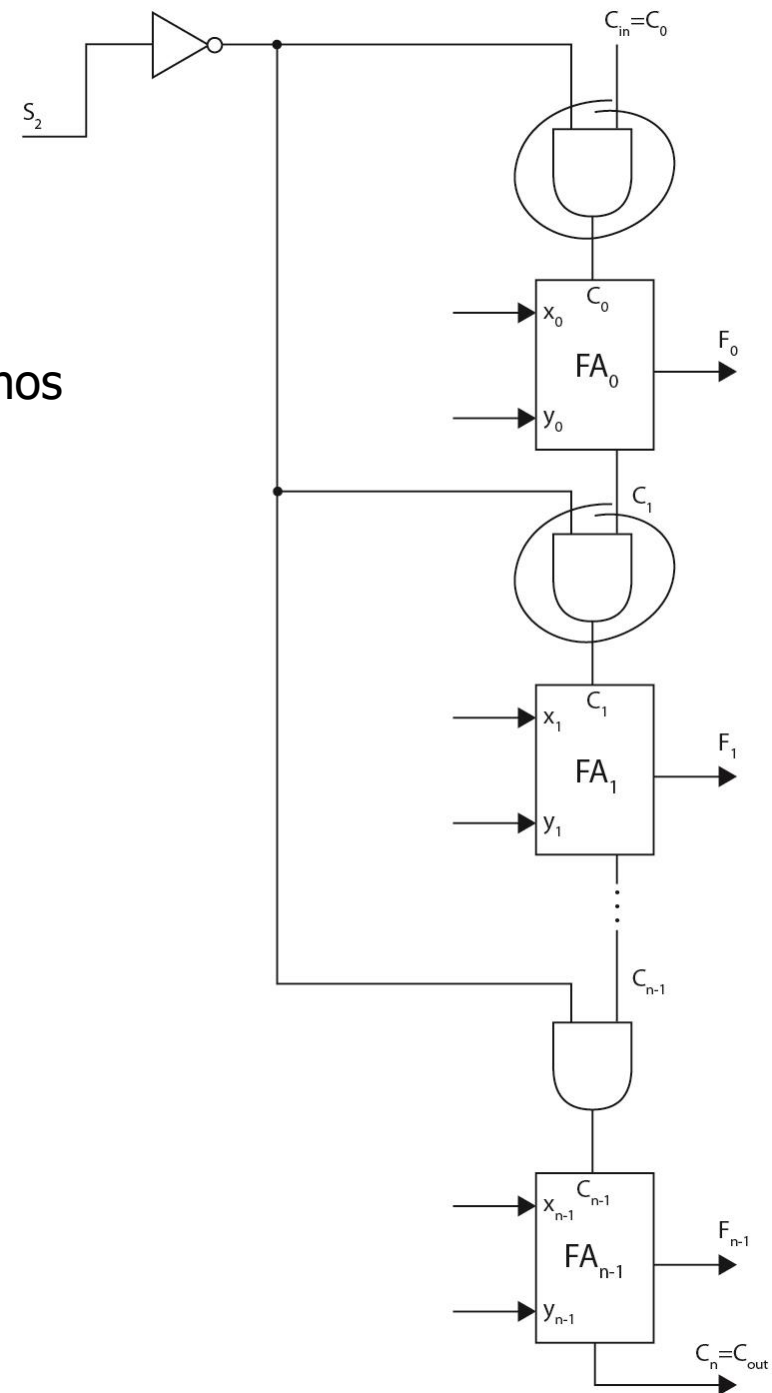
Fig: Block diagram of Logic Unit

Logička sekcija ALU

Za logičke operacije, dodatno, isključuje se prijenos

Kombinacija upravljačkih bitova:

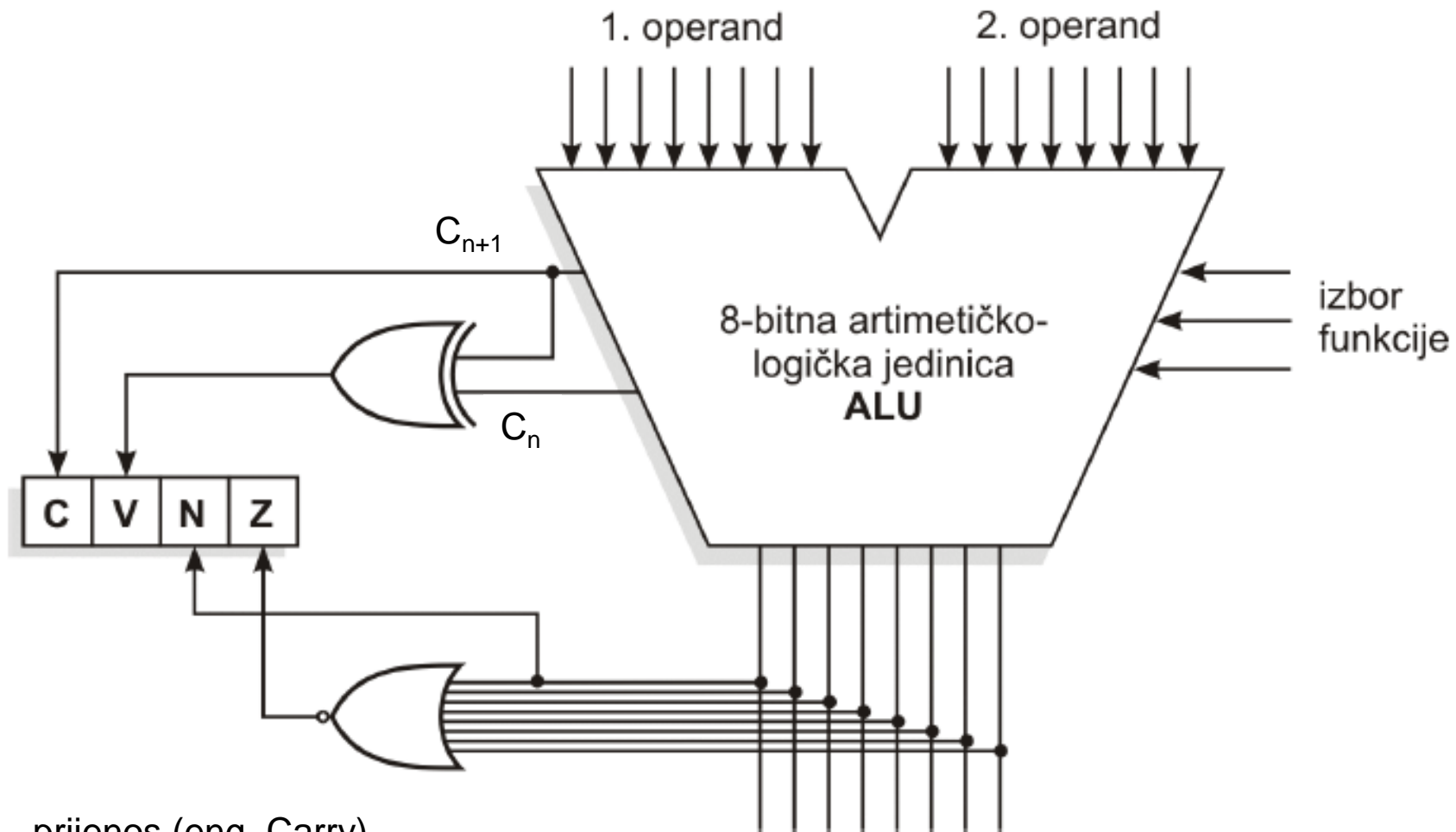
- $S_2=1, S_1=X, S_0=X$
- S_2 preko invertora isključuje prijenos



Pristup oblikovanju ALU jedinice

1. Prvo se oblikuje aritmetička sekcija **nezavisno** od logičke sekcije
 2. Određuju se logičke operacije koje se mogu izvesti sa sklopovima iz aritmetičke sekcije
 3. Modificiraju se aritmetički sklopovi da bi se dobile željene logičke operacije
- Prednosti ovakvog pristupa:
 - Manje sklopovlja za istu funkcionalnost
 - Manje zauzeće na površini čipa
 - Detaljnije: S. Ribarić, Građa računala, str. 260-263

Veza ALU i statusnog registra



C – prijenos (eng. Carry)

V – preljev (eng. oVerflow)

N – negativan broj (eng. Negative)

Z – broj je nula (eng. Zero)

Množenje

$$\begin{array}{r} \underline{1100 * 1101} \\ 1100 \\ 0000 \\ 1100 \\ \underline{1100} \\ 10011100 \end{array}$$

$$12_{(10)} * 13_{(10)}$$

1. korak: najmanje značajni bit množitelja je 1, prepisuje se množenik
2. korak: drugi bit množitelja je 0, upisuje se 0000 s posmakom ulijevo
3. korak: treći bit množitelja je 1, prepisuje se množenik s posmakom ulijevo
4. korak: četvrti bit množitelja je 1, prepisuje se množenik s posmakom ulijevo
5. korak : zbrajamo parcijalne produkte (rezultat $156_{(10)}$)

Množenje brojeva predodčenih u notaciji dvojnog komplementa:

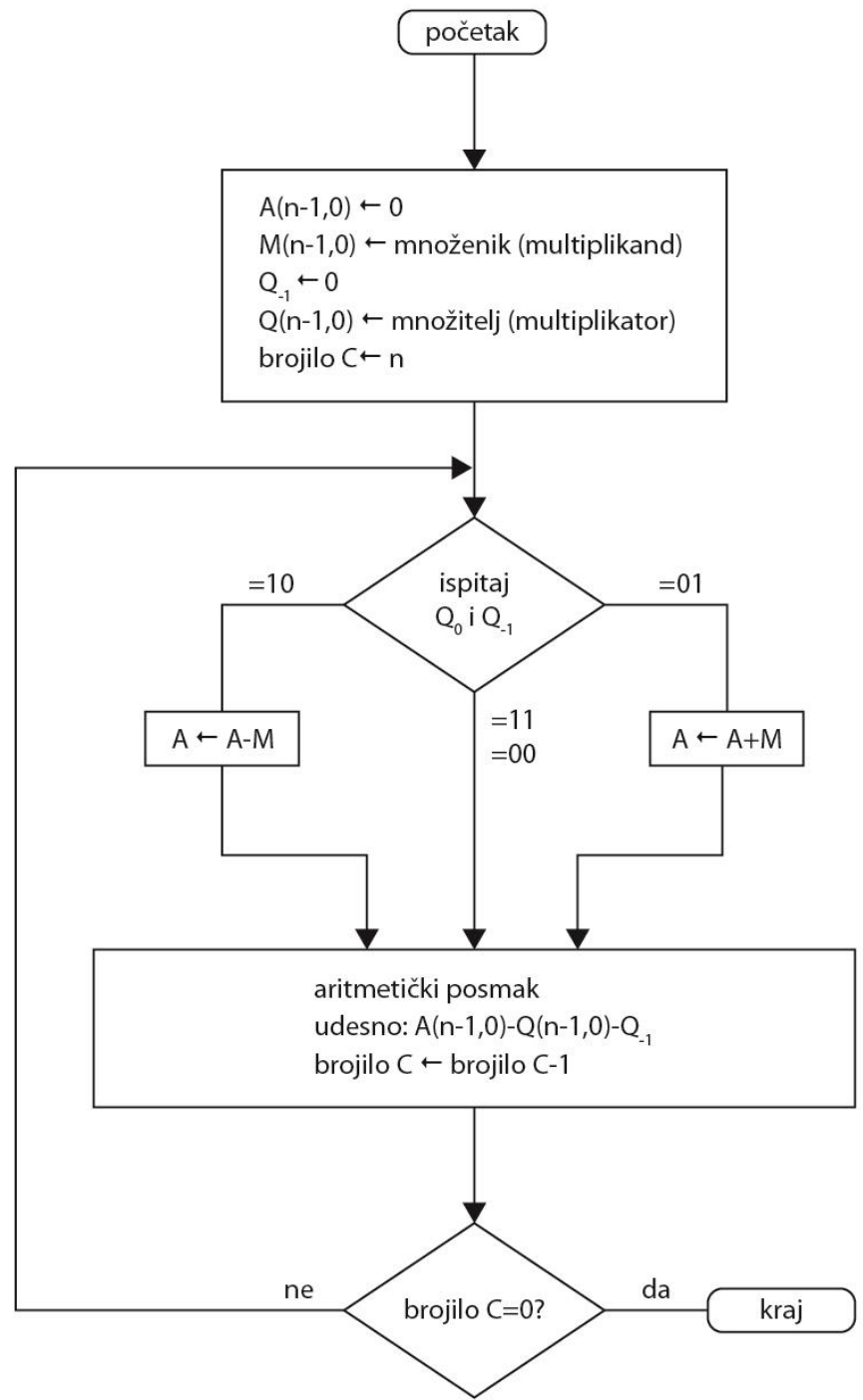
Zahtijeva se korekcijski faktor nakon operacije množenja i to ako:

- je množitelj Y negativan (korekcijski faktor je dvojni komplement množenika X pomnožen s 2^n)
- je množenik X negativan (korekcijski faktor je dvojni komplement množitelja Y pomnožen s 2^n)
- Su i množenik X i množitelj Y negativni (korekcijski faktori su tada $2^n X$ i $2^n Y$)

Boothov algoritam

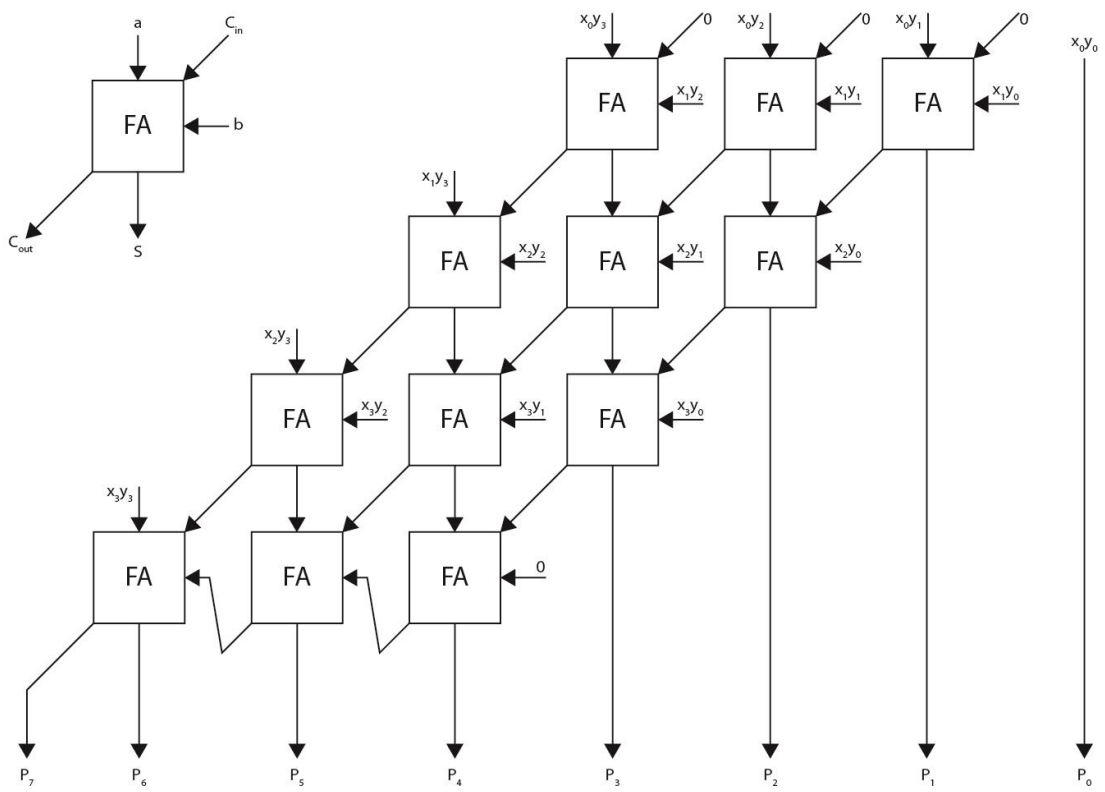
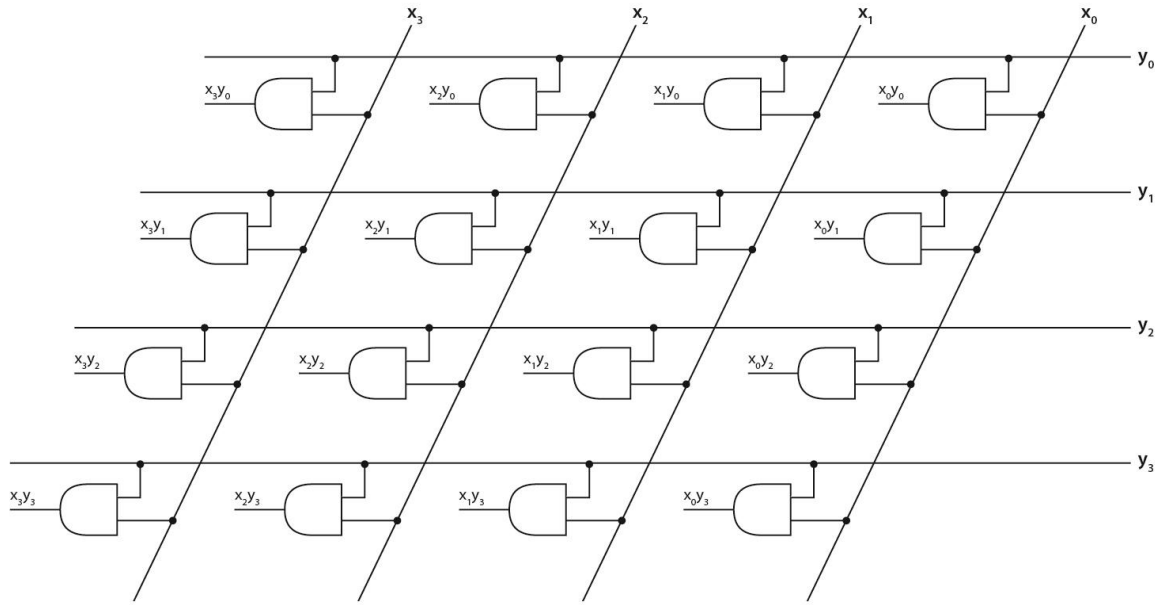
Svojstva:

- algoritam vrijedi za dva pozitivna broja, za slučaj kad je jedan od brojeva negativan i kad su oba broja negativna
- Ubrzava postupak množenja (u odnosu na klasičan pristup)
- Ne zahtijeva korekciju
- Podesan je za sklopovlje



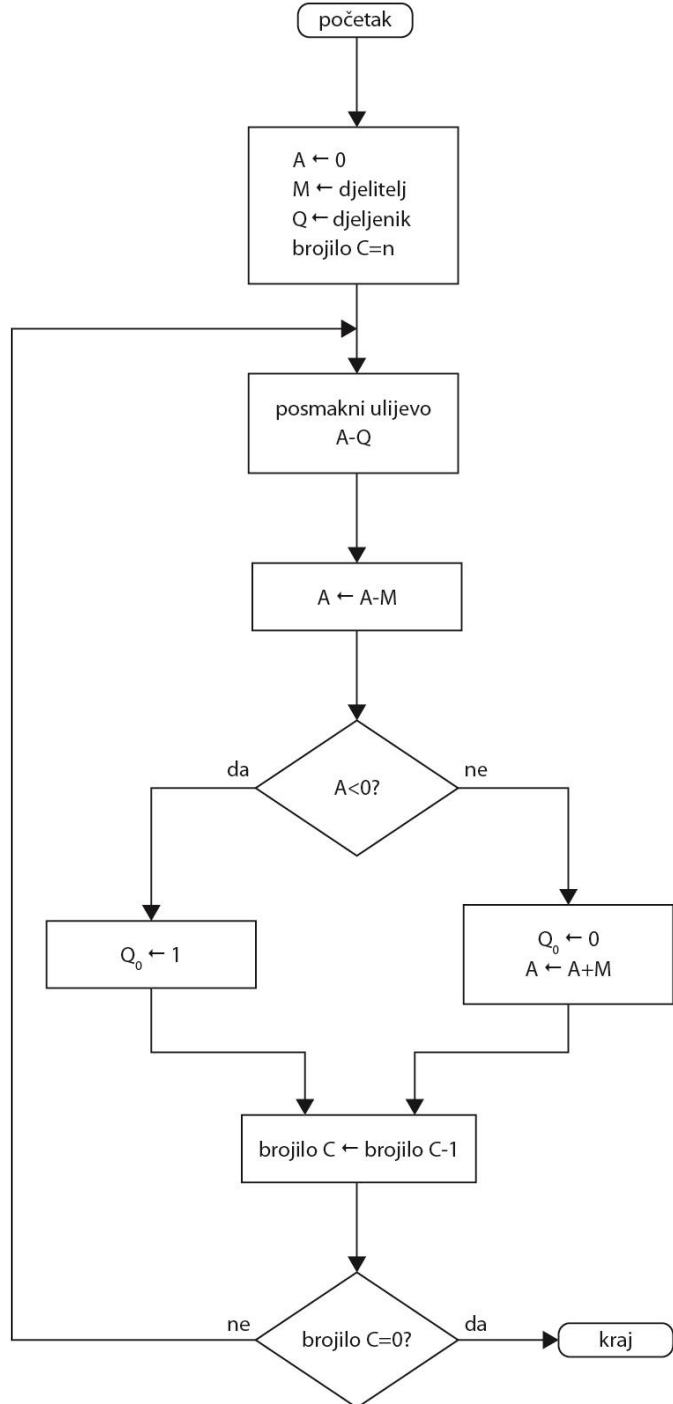
Posebne izvedbe množila

- Množila sastavljena od polja potpunih zbrajala
- Množila temelja na tablici preslikavanja
- Množila s brzim posmačnim sklopovima
- Množila sastavljena od zbrajala s očuvanjem bita prijenosa koja su povezana u kaskadu



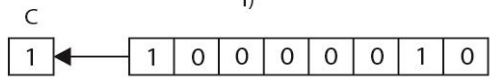
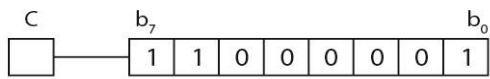
Dijeljenje

$$\begin{array}{r} \underline{1111101 : 101 = 11001} \\ - \quad \underline{101} \\ \quad 0101 \\ - \quad \underline{101} \\ \quad \quad 0001 \\ \quad \quad \quad 00010 \\ \quad \quad \quad \quad 000101 \\ - \quad \quad \quad \quad \underline{101} \\ \quad \quad \quad \quad \quad 000 \end{array}$$

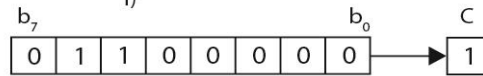
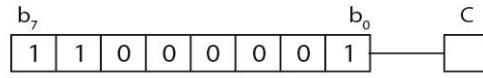


Posmak

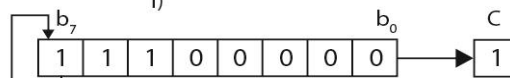
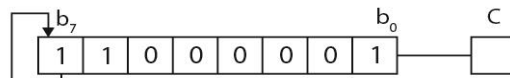
zastavica



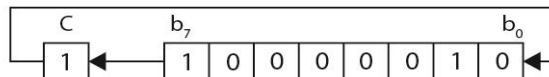
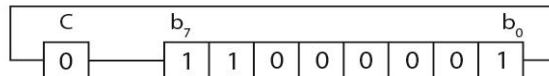
a)



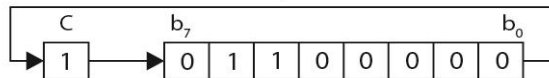
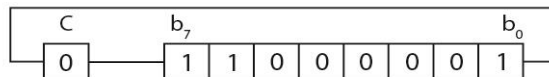
b)



c)



d)



e)

- Logički posmak ulijevo

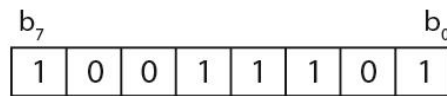
- Logički posmak udesno

- Aritmetički posmak udesno

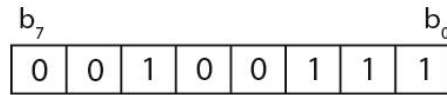
- Kružni posmak ulijevo

- Kružni posmak udesno

Posmak



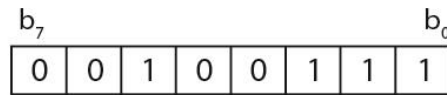
a)



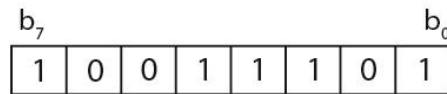
b)



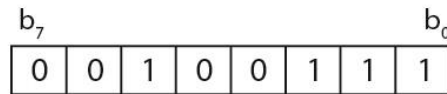
c)



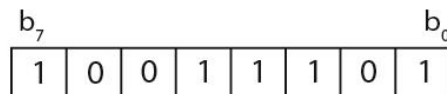
d)



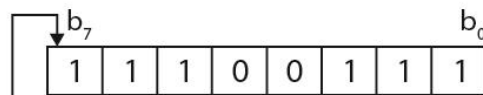
e)



f)



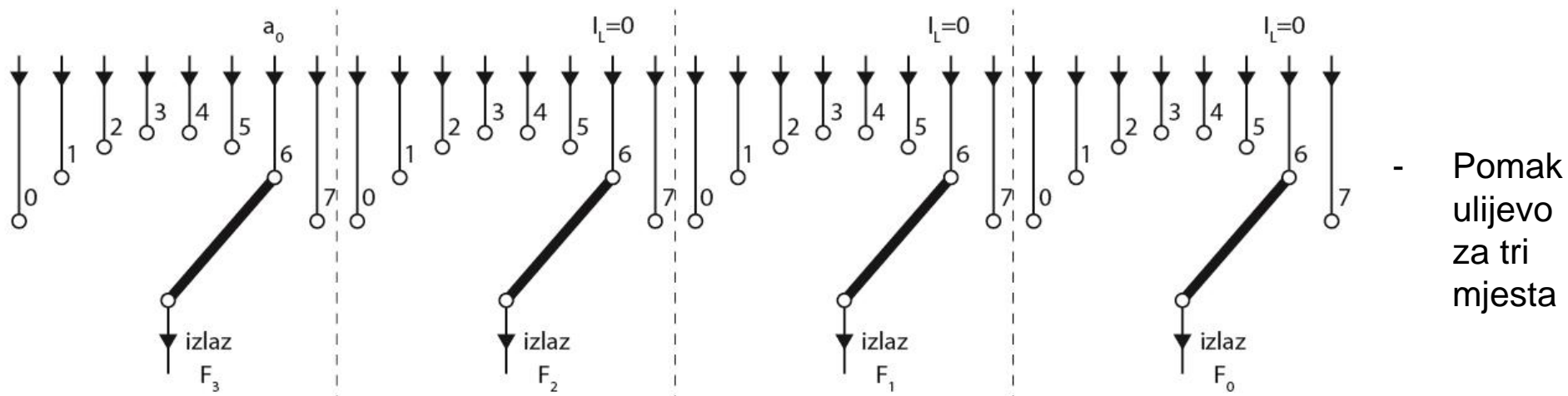
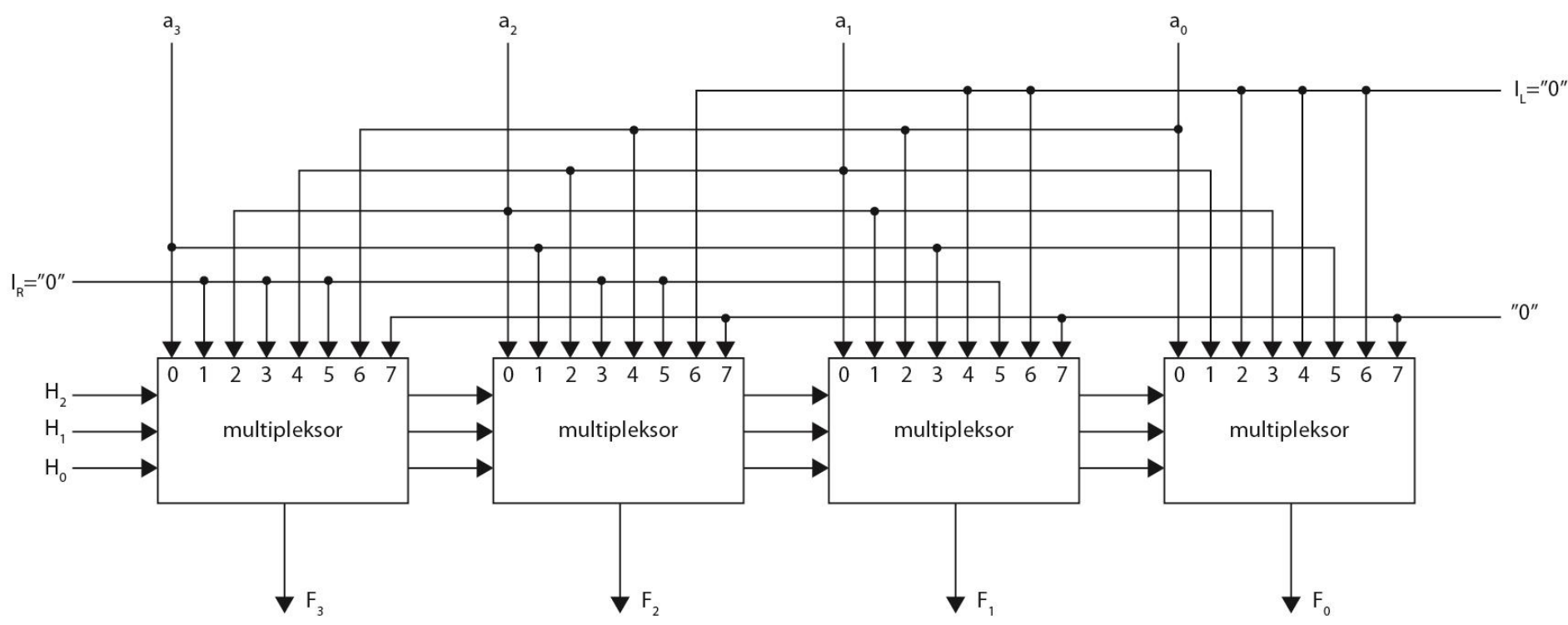
g)



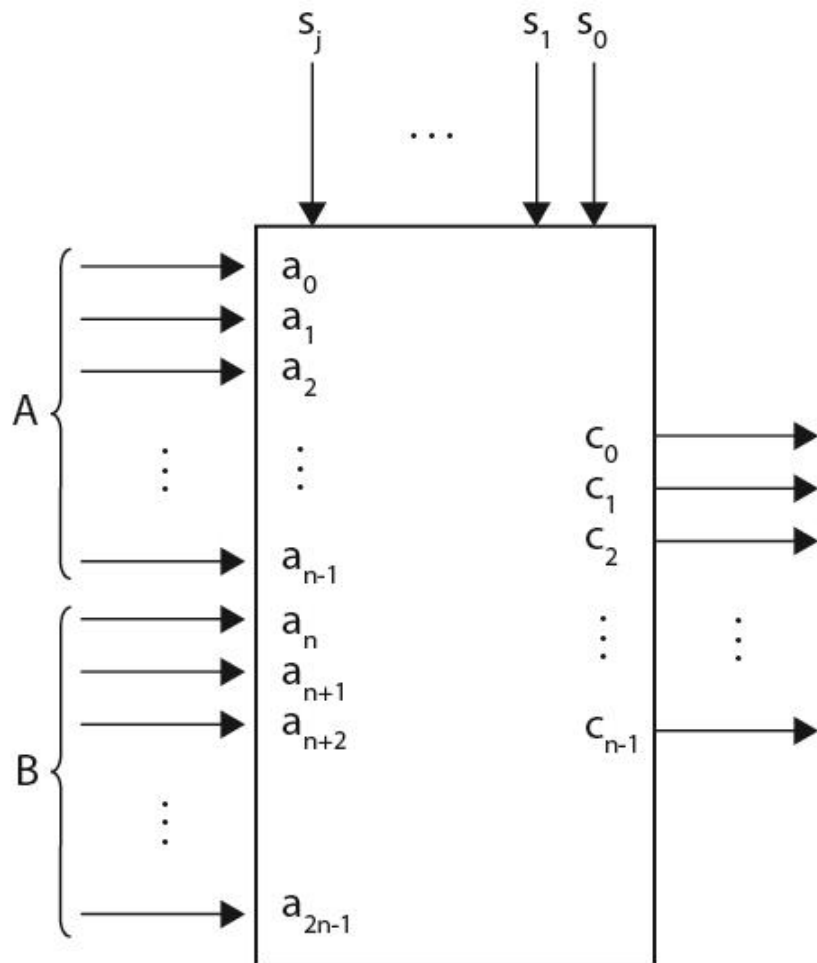
h)

- Nepredznačni broj
- Vrijednost: 157
- Logički posmak udesno za 2 mjesta
- Vrijednost: 39
- Predznak-apsolutna vrijednost
- Vrijednost: -29
- Logički posmak udesno za 2 mjesta
- Vrijednost: +39
- Dvojni komplement
- Vrijednost: -99
- Logički posmak udesno za 2 mjesta
- Vrijednost: +39
- Dvojni komplement
- Vrijednost: -99
- Aritmetički posmak udesno za 2 mjesta
- Vrijednost: -25

Izvedba posmačnog sklopa multipleksorima



Bačvasti posmačni sklop



$S=0,$

$$c_{n-1}c_{n-2}\dots c_1c_0 = a_{n-1}a_{n-2}\dots a_1a_0$$

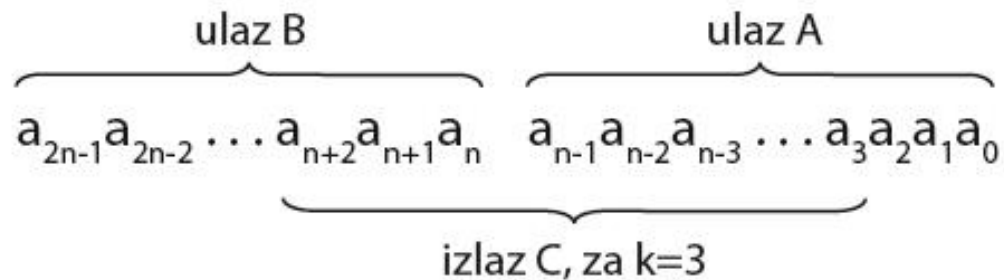
$S=k, 0 < k \leq n$

$$c_{n-1}c_{n-2}\dots c_1c_0 = a_{n-1+k}a_{n-2+k}\dots a_{k+1}c_{k+0}$$

- Obavlja posmak u kratkom vremenu neovisno o iznosu posmaka
- Ulazi: dva n-bitna podatka, vrijednosti podataka određuju tip posmaka
- Ulaz: j-bitni upravljački operand određuje vrijednost posmaka, tj. definira broj bitova ulaza B koji će se pojaviti na izlazu ulančan s dijelom bitova iz ulaza A
- Izlaz: jedan n-bitni operand

Bačvasti posmačni sklop – princip rada

- Npr. $S=3$,



- $A=B$, kružni posmak udesno za 3 binarna mjesta
kružni posmak ulijevo za $n-3$ binarna mjesta

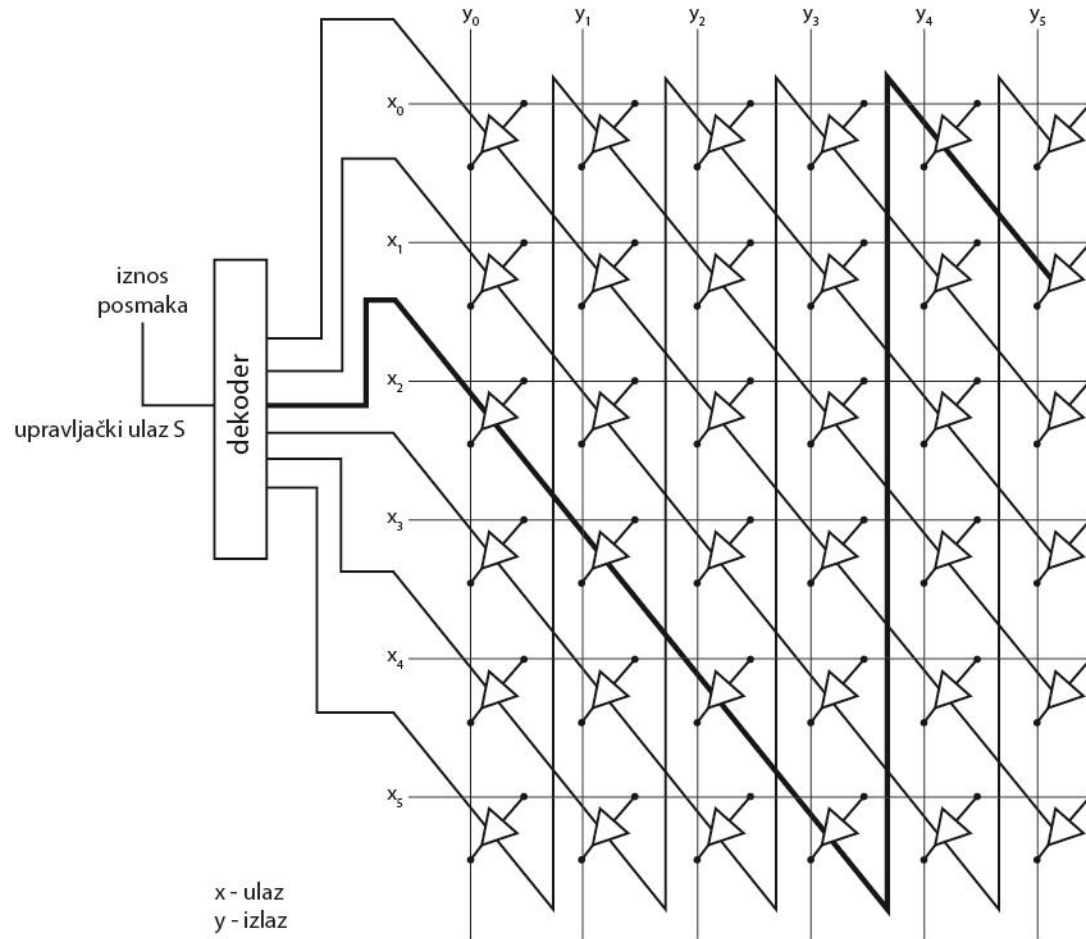
- $B=0$, logički posmak desno za 3 binarna mjesta

- $$c_{n-1} c_{n-2} \dots c_1 c_0 = 000 a_{n-1+k} a_{n-2+k} \dots a_4$$

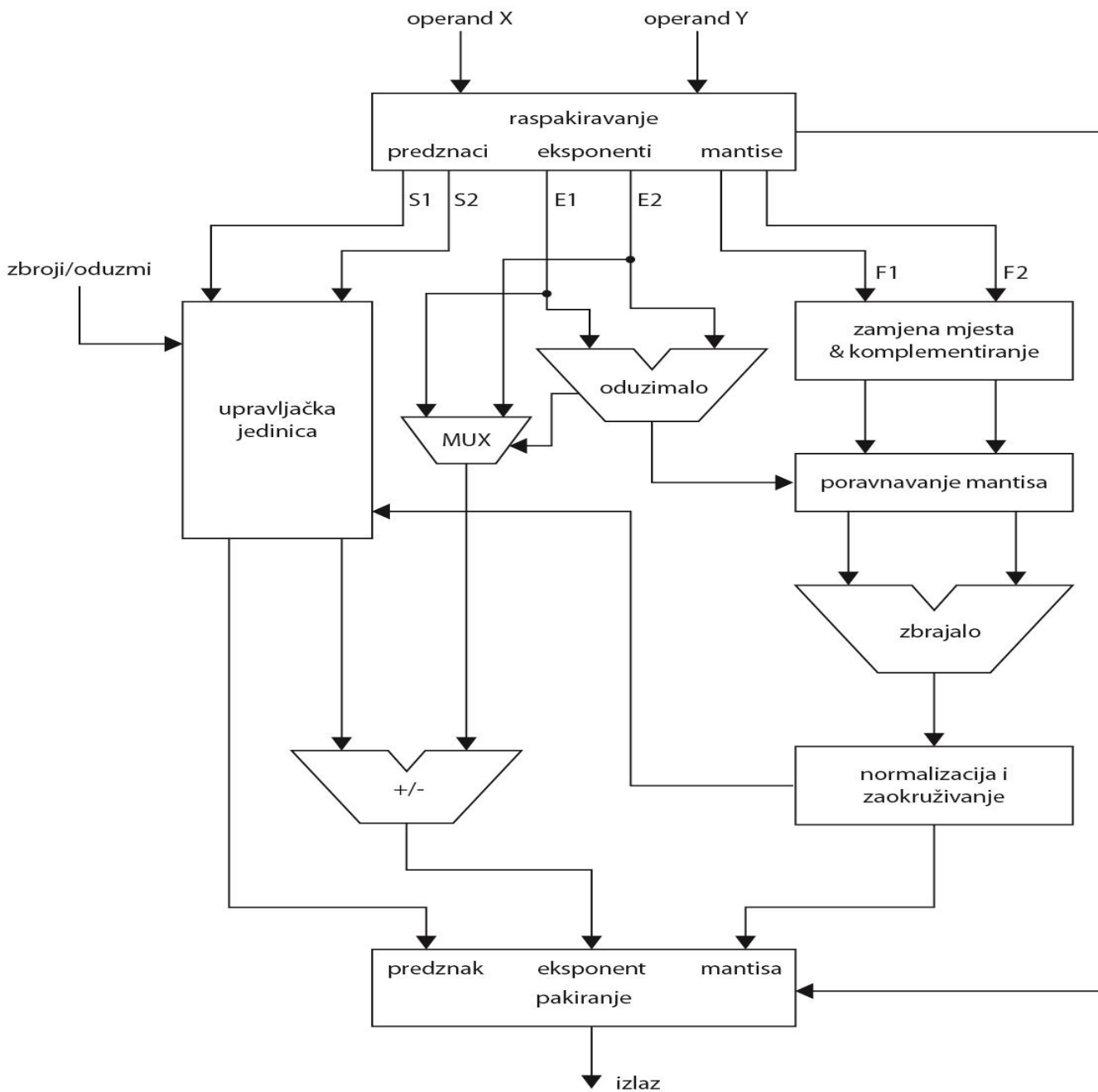
- $A=0$, logički posmak ulijevo za $n-3$ binarna mjesta

- $$c_{n-1} c_{n-2} \dots c_1 c_0 = a_{n+2} a_{n+1} a_n 00 \dots 0$$

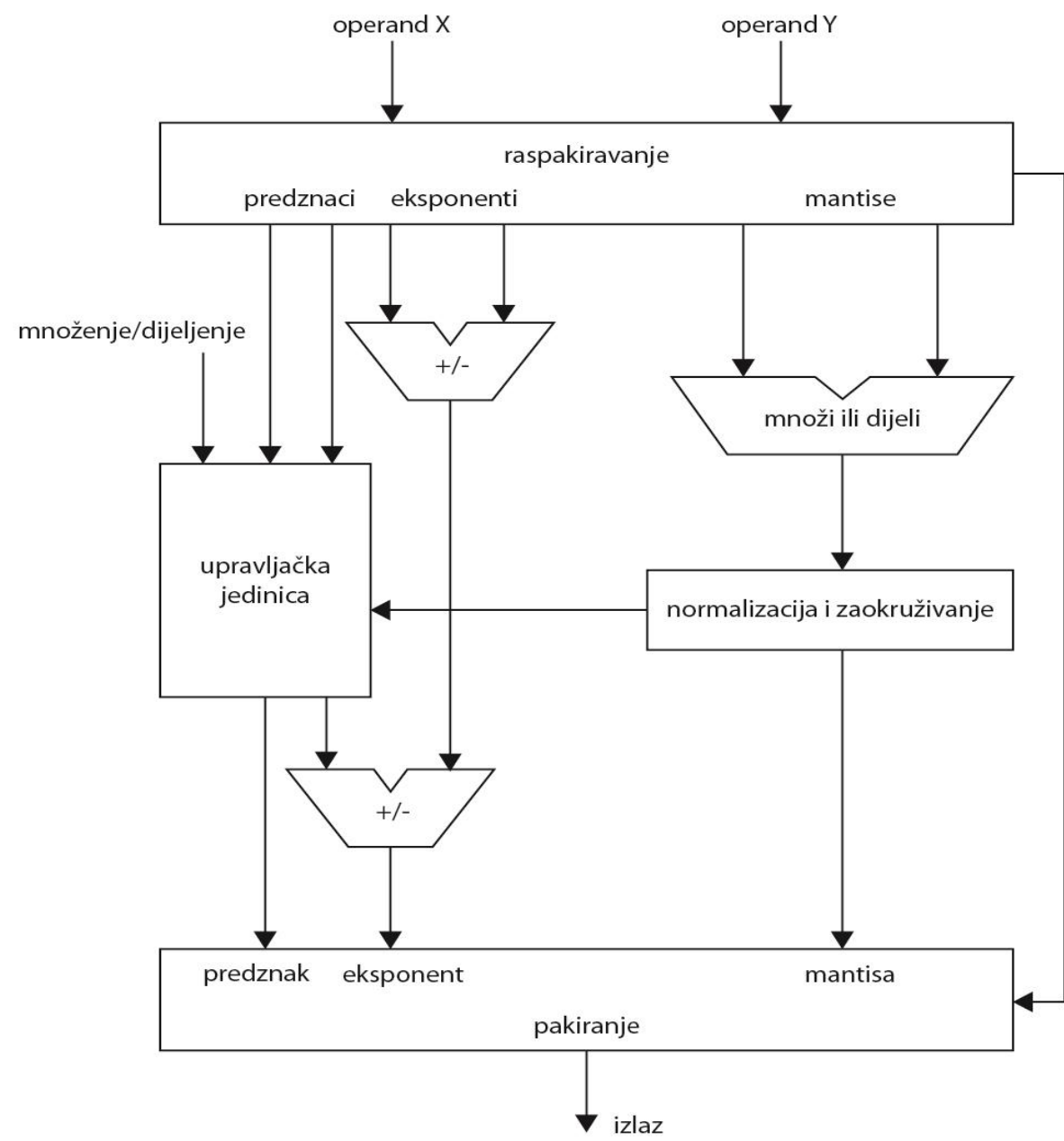
Bačvasti posmačni sklop - implementacija



Aritmetičke operacije s brojevima s pomičnim zarezom – zbrajanje i oduzimanje



Aritmetičke operacije s brojevima s pomičnim zarezom – množenje i dijeljenje



Jednostavna višefunkcijska ALU

