

Izbor funkcije

$X_3 X_2 X_1 X_0 \Rightarrow$ Funkcija

0000 $\Rightarrow A + B$

0001 $\Rightarrow A - B$

0010 $\Rightarrow -A$

0011 $\Rightarrow -B$

0100 $\Rightarrow A++$

0101 $\Rightarrow A--$

0110 $\Rightarrow A \text{ AND } B$

0111 $\Rightarrow A \text{ OR } B$

1000 $\Rightarrow A \text{ XOR } B$

1001 $\Rightarrow \text{NOT } A$

1010 $\Rightarrow \text{NOT } B$

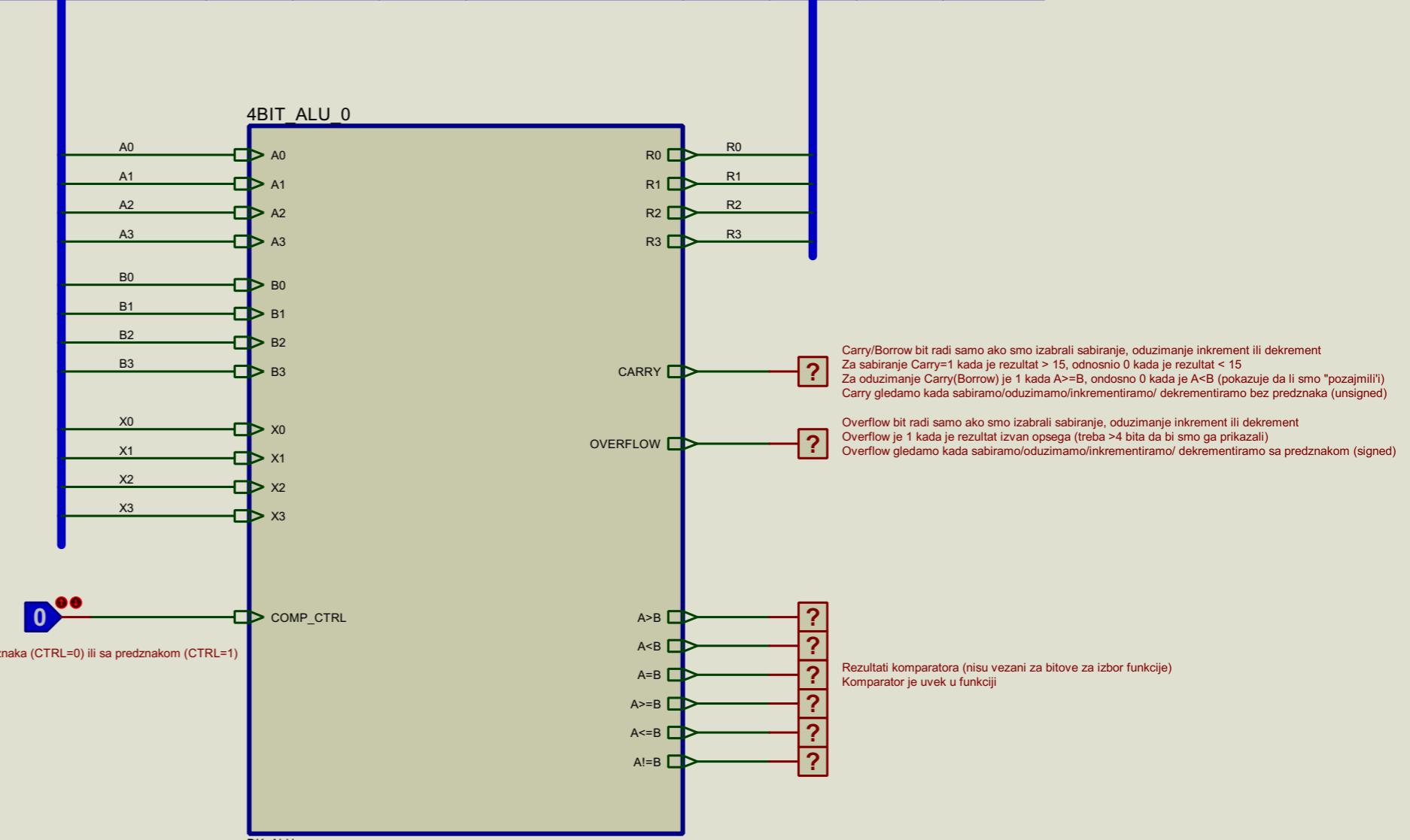
1011 $\Rightarrow A \text{ NAND } B$

1100 $\Rightarrow A \text{ NOR } B$

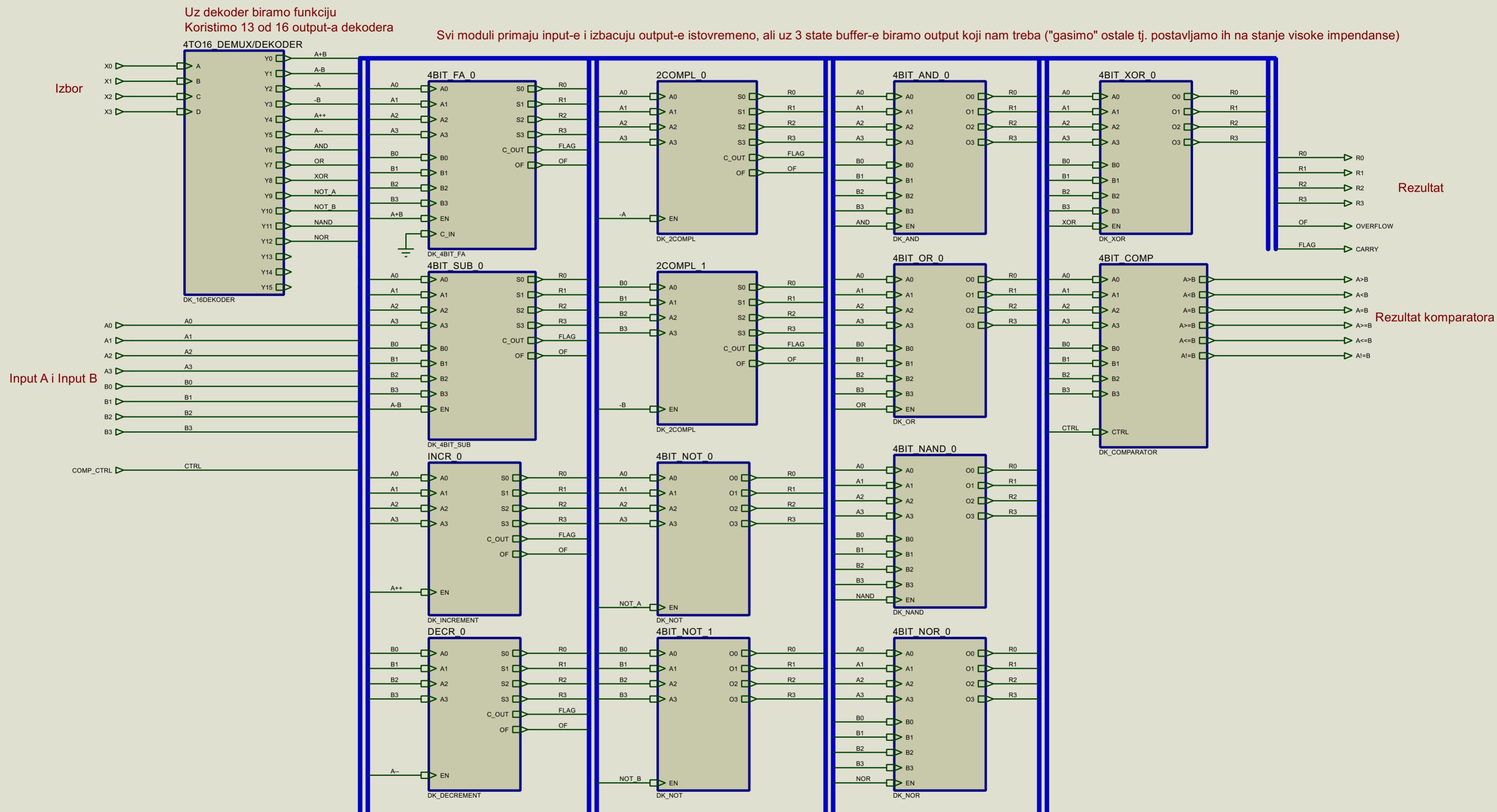
1101 $\Rightarrow \text{NONE}$

1110 $\Rightarrow \text{NONE}$

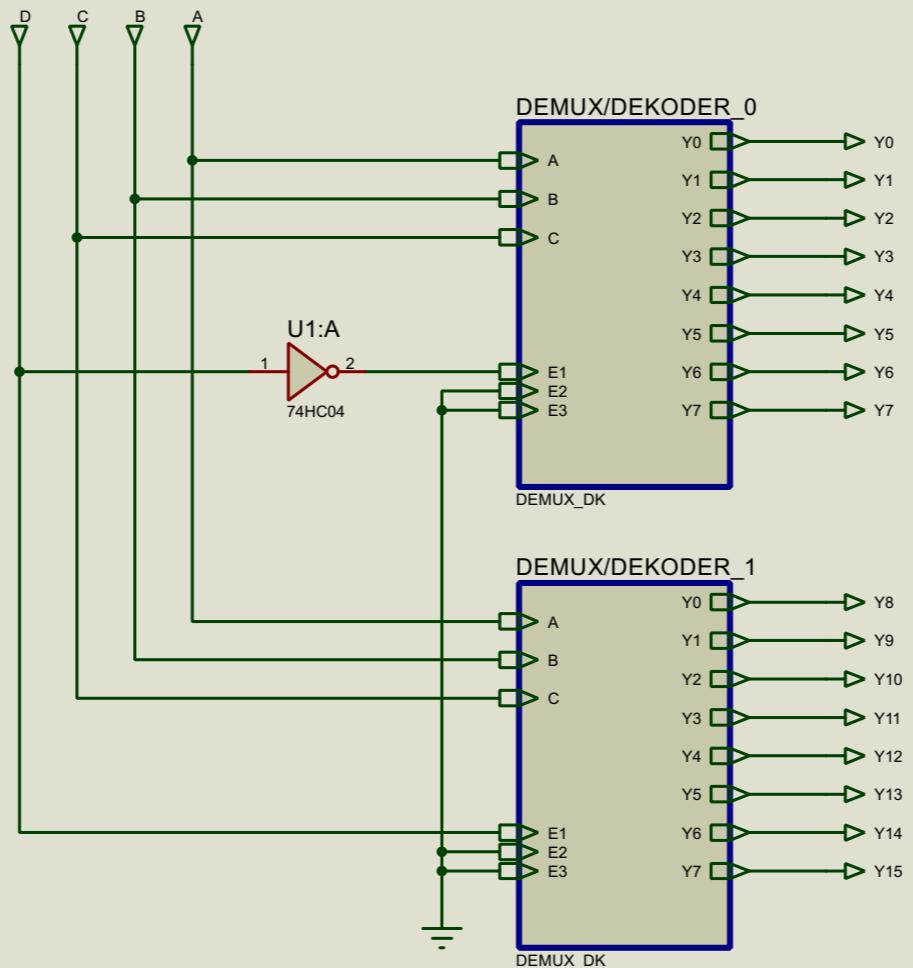
1111 $\Rightarrow \text{NONE}$



4-bit ALU



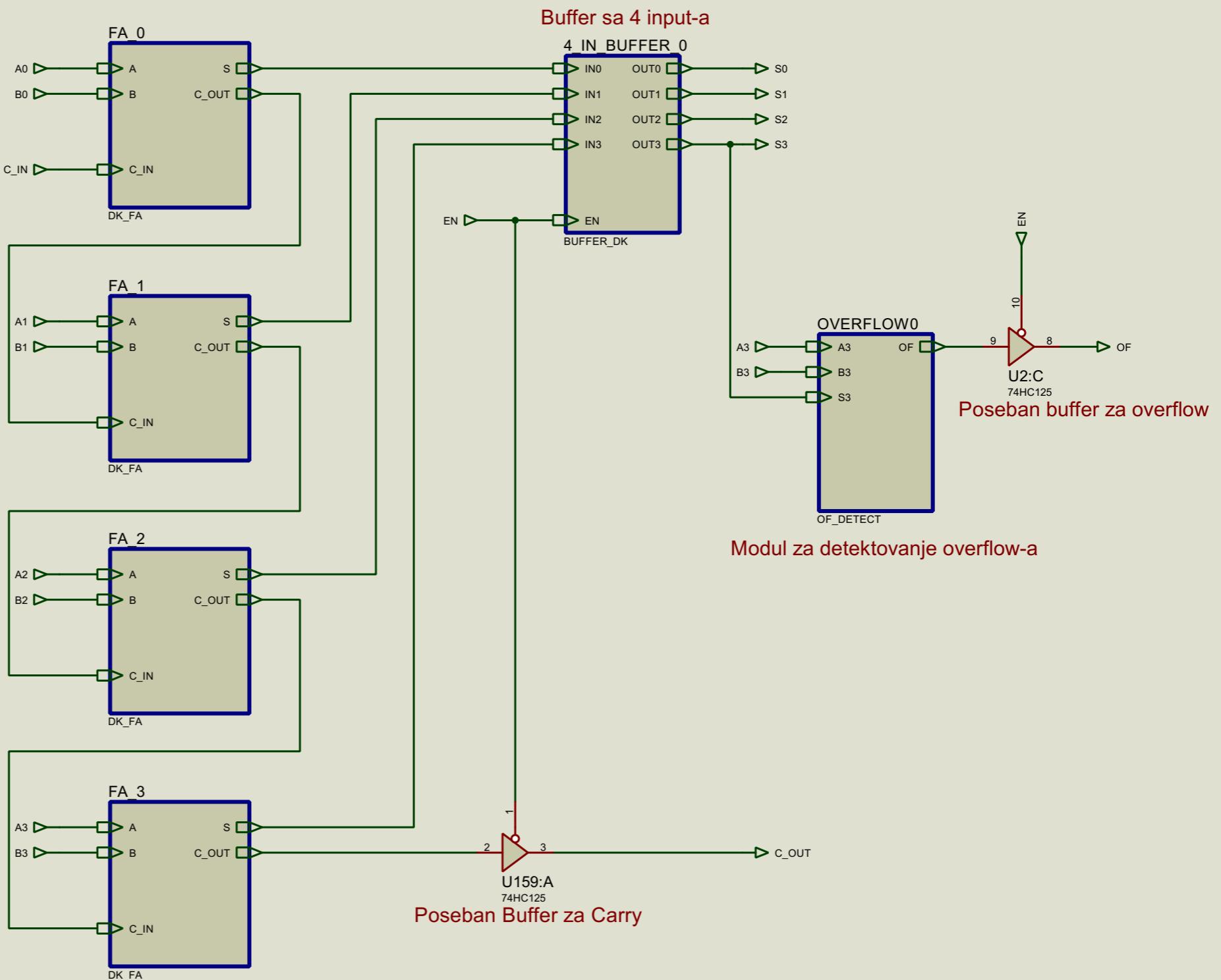
16-bit MUX/DECODER



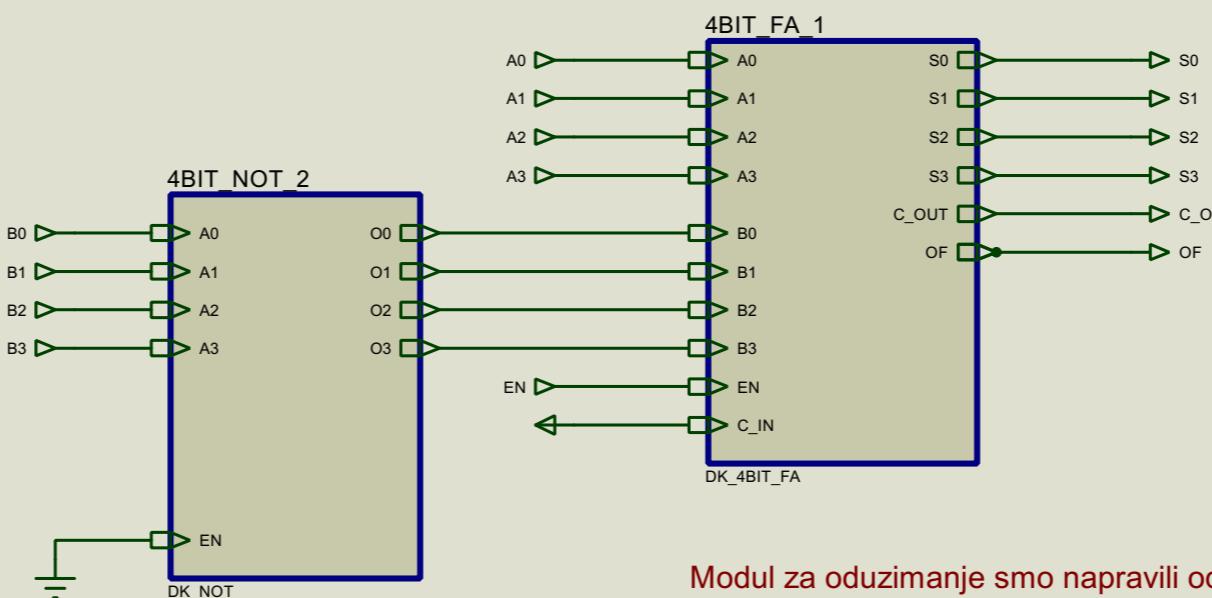
A	B	Cin	s	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

4 1-bitna modula za sabiranje

4-bit modul za sabiranje sa carry i overflow



4-bit modul za oduzimanje sa borrow (c_out) i overflow



Modul za oduzimanje smo napravili od modula za sabiranje

Prvi ulaz ostaje nepromenjen

Drugi ulaz prebacimo u 1. komplement (4-bitni NOT) i uz Cin iz 1. u 2. komplement

Imamo $A + (-B)$, koje je jednako sa $A - B$

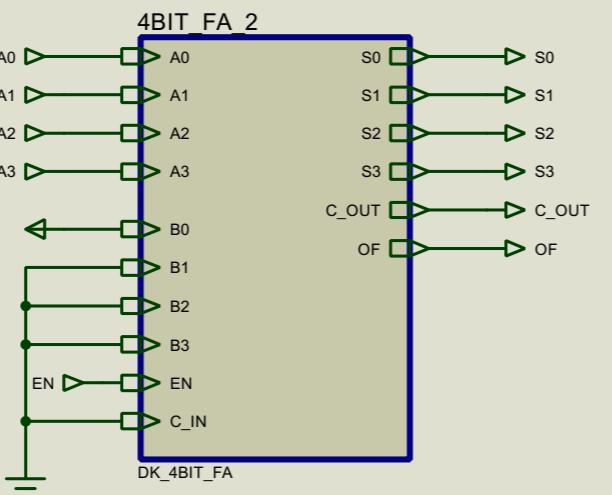
Koristimo OF iz modula za sabiranje

Borrow (c_out) je 1 kada je $A \geq B$ tj. 0 kada je $A < B$

A	B	S	Cout
0	0	0	1
0	1	0	0
1	0	1	1
1	1	0	1

A3	A2	A1	A0	R3	R2	R1	R0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

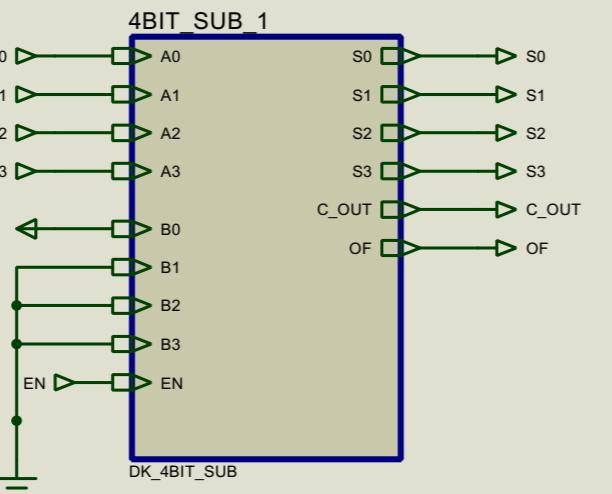
4-bit inkrement sa carry i overflow



Koristimo modul za sabiranje u koga ubacujemo broj koji inkrementiramo u A input i u B-input-u LSB postavljajmo na 1 a ostale na 0

A3	A2	A1	A0	R3	R2	R1	R0
0	0	0	0	1	1	1	1
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	1
0	0	1	1	0	0	1	0
0	1	0	0	0	0	1	1
0	1	0	1	0	1	0	0
0	1	1	0	0	1	0	1
0	1	1	1	0	1	1	0
1	0	0	0	0	1	1	1
1	0	0	1	1	0	0	0
1	0	1	0	1	0	0	1
1	0	1	1	1	0	1	0
1	1	0	0	1	0	1	1
1	1	0	1	1	1	0	0
1	1	1	0	1	1	0	1
1	1	1	1	1	1	1	0

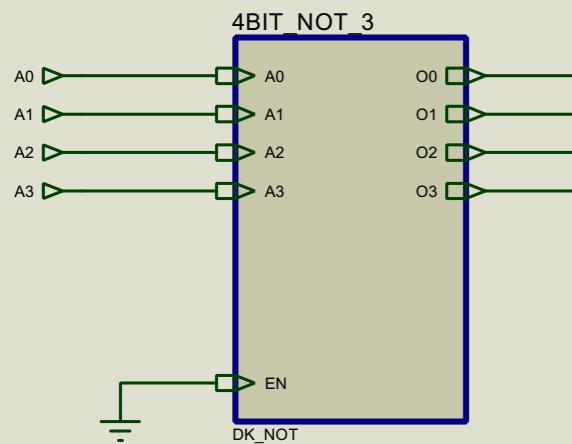
4-bit dekrement sa borrow i overflow



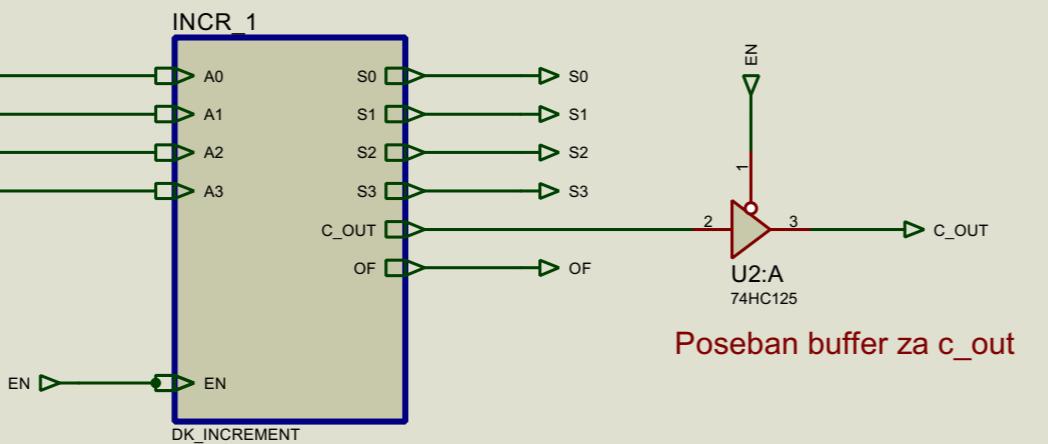
Koristimo modul za oduzimanje u koga ubacujemo broj koji dekrementiramo u A-input i u B-input-u LSB postavljajmo na 1 a ostale na 0

A3	A2	A1	A0	R3	R2	R1	R0
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1
1	0	0	0	1	0	0	0
1	0	0	1	0	1	1	1
1	0	1	0	0	1	1	0
1	0	1	1	0	0	1	1
1	1	0	0	0	1	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	1

4-bit drugi komplement sa carry i overflow



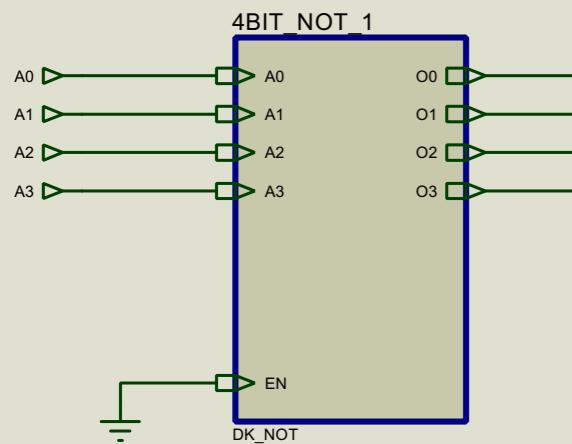
Prvo negiramo ulaz (1. komplement)



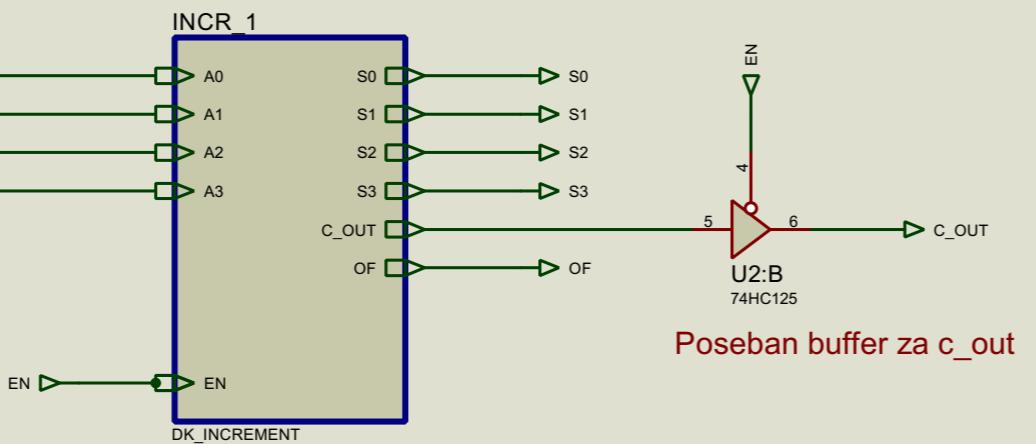
Pa inkrementiramo 1. komplement i dobijamo 2. komplement

A3	A2	A1	A0	R3	R2	R1	R0
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
1	0	0	0	1	0	0	0
1	0	0	1	0	1	1	1
1	0	1	0	0	1	1	0
1	0	1	1	0	0	1	1
1	1	0	0	0	1	0	0
1	1	0	1	0	0	1	1
1	1	1	0	1	0	1	0
1	1	1	1	0	0	0	1
1	1	1	1	0	0	0	1

4-bit drugi komplement sa carry i overflow



Prvo negiramo ulaz (1. komplement)

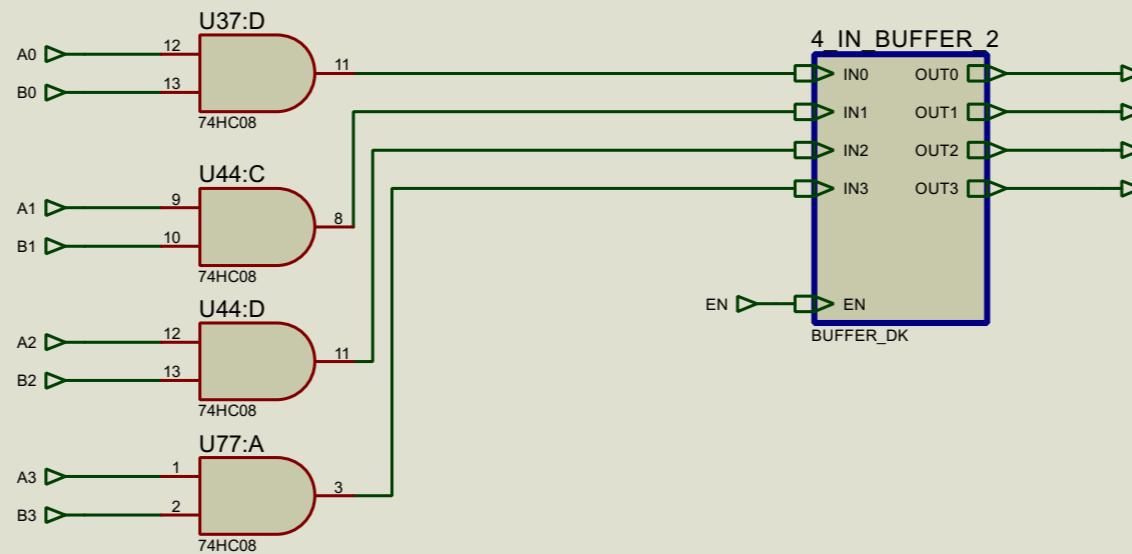


Pa inkrementiramo 1. komplement i dobijamo 2. komplement

Za svaki bit posebno

A	B	R
0	0	0
0	1	0
1	0	0
1	1	1

4-bit AND

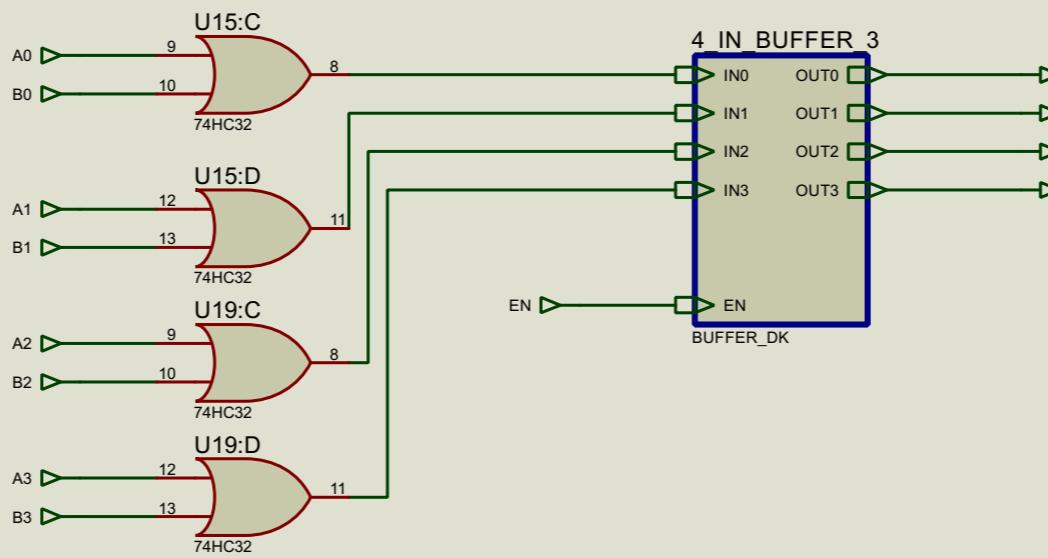


Radimo AND svakog bita A i B posebno

Za svaki bit posebno

A	B	R
0	0	0
0	1	1
1	0	1
1	1	1

4-bit OR

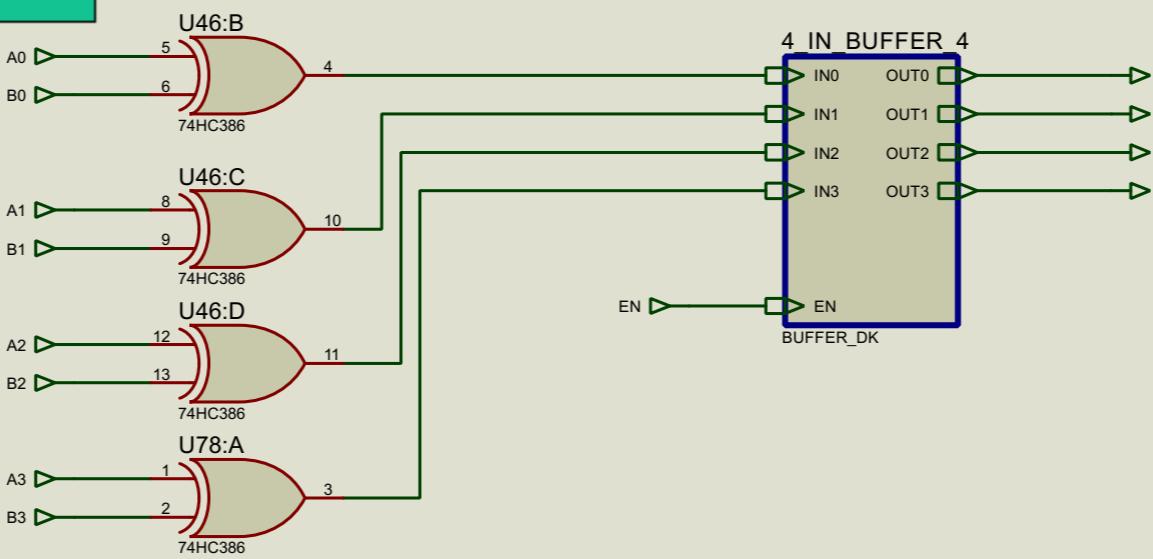


Radimo OR svakog bita A i B posebno

Za svaki bit posebno

A	B	R
0	0	0
0	1	1
1	0	1
1	1	0

4-bit XOR

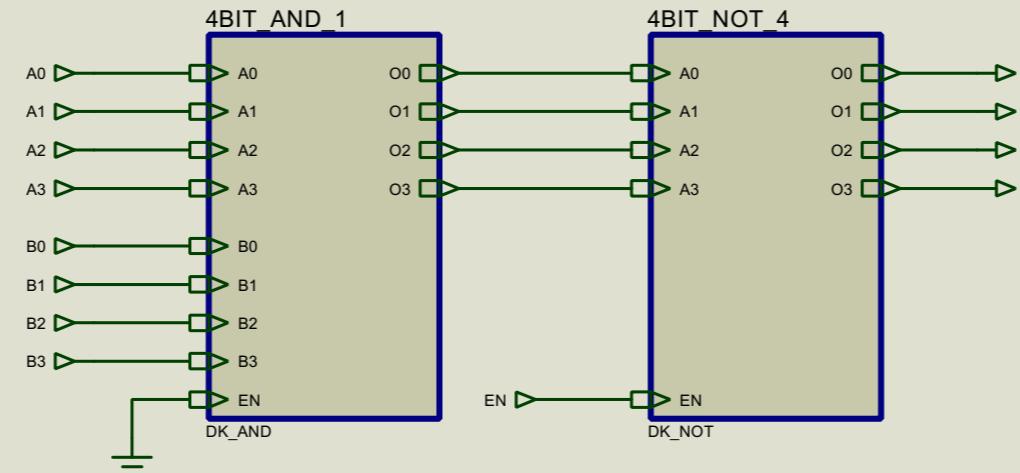


Radimo XOR svakog bita A i B posebno

Za svaki bit posebno

A	B	R
0	0	1
0	1	1
1	0	1
1	1	0

4-bit NAND

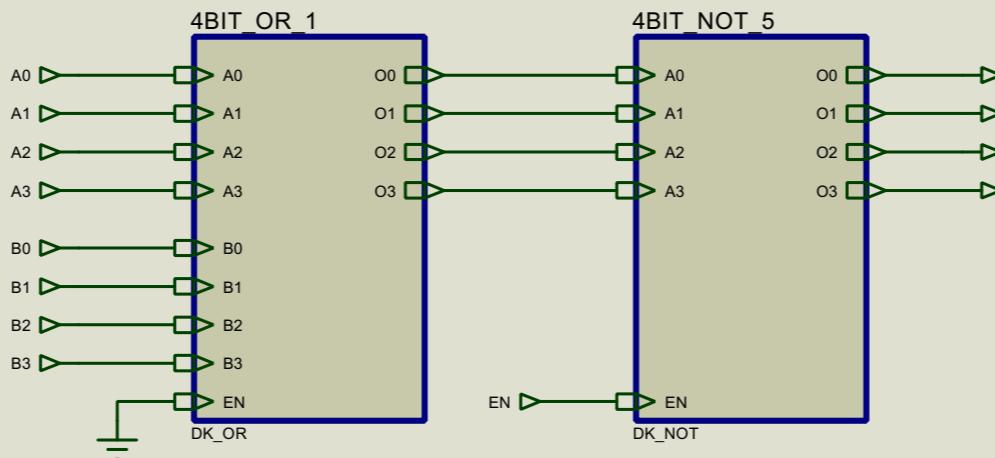


Negiramo izlaz AND modula

Za svaki bit posebno

A	B	R
0	0	1
0	1	0
1	0	0
1	1	0

4-bit NOR

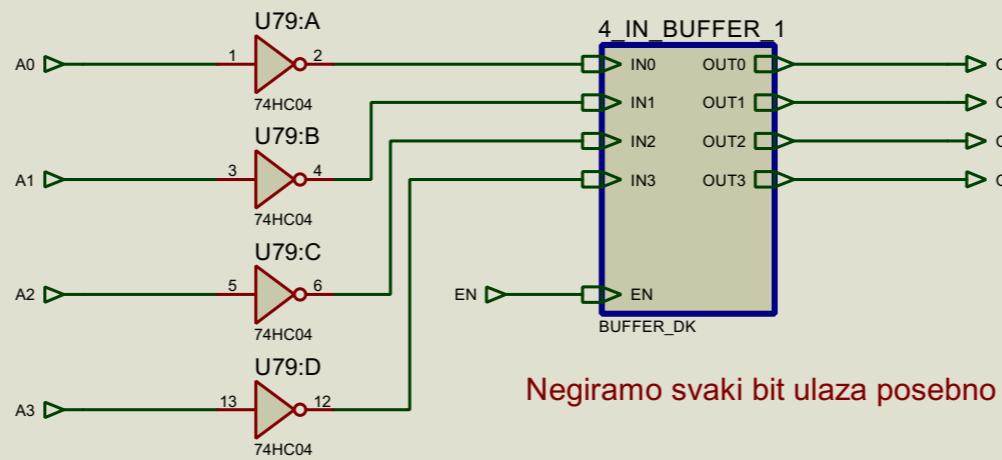


Negiramo izlaz OR modula

Za svaki bit posebno

A	R
0	1
1	0

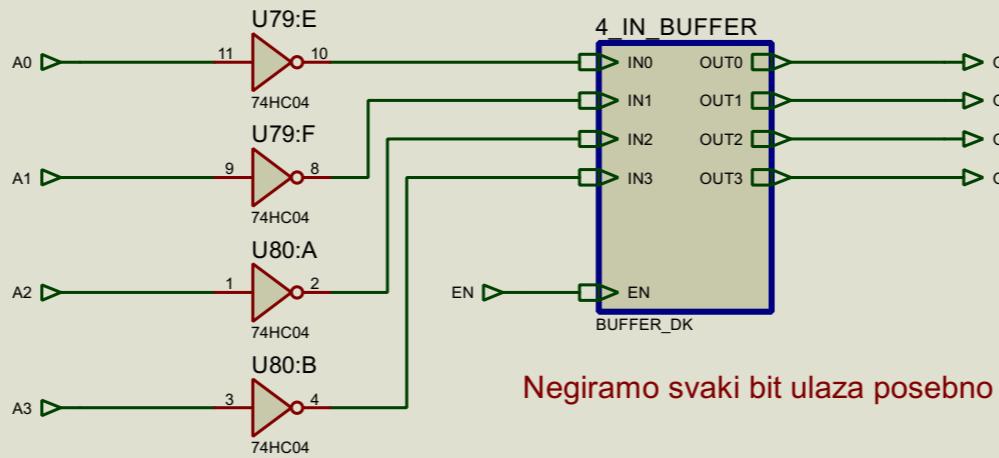
4-bit NOT



Za svaki bit posebno

A	R
0	1
1	0

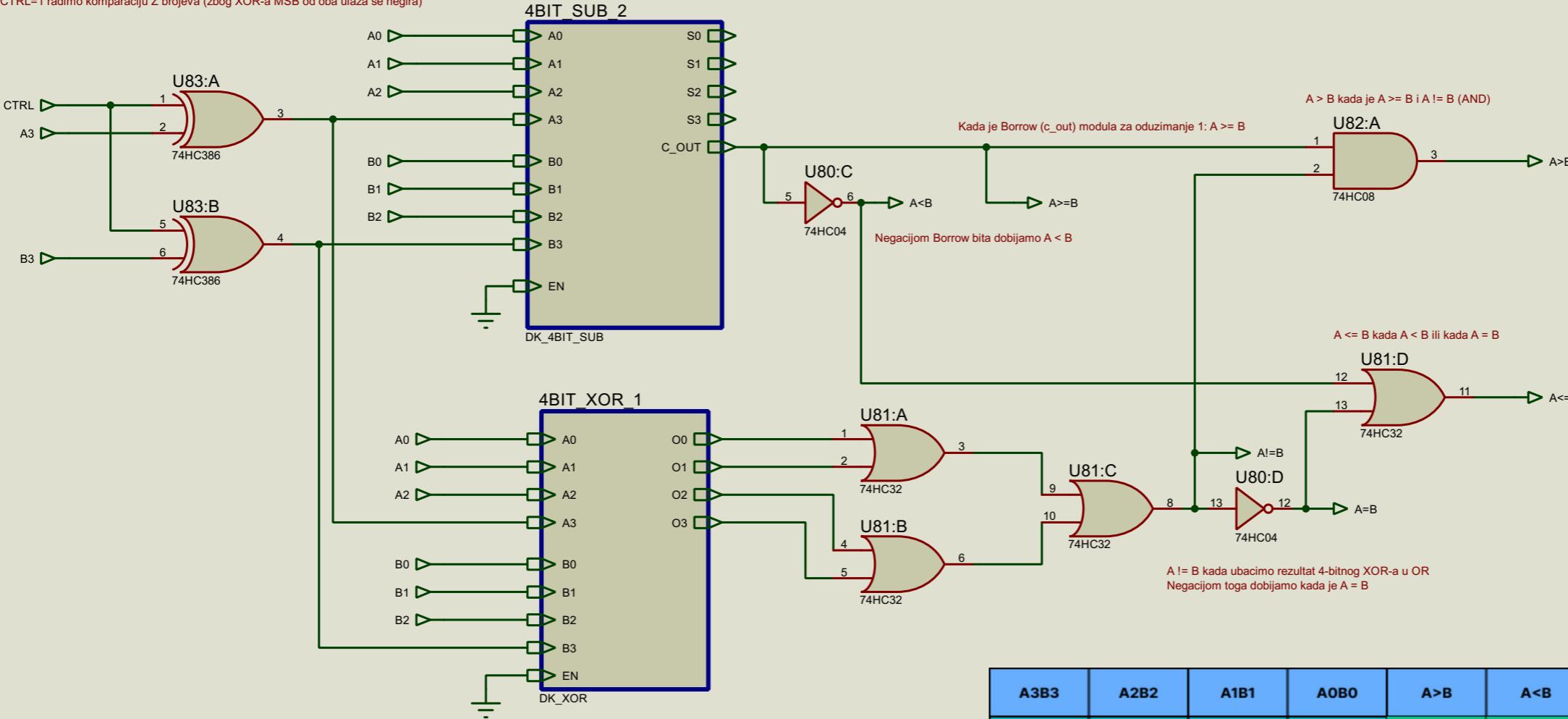
4-bit NOT



4-bit komparator za N i Z brojeve

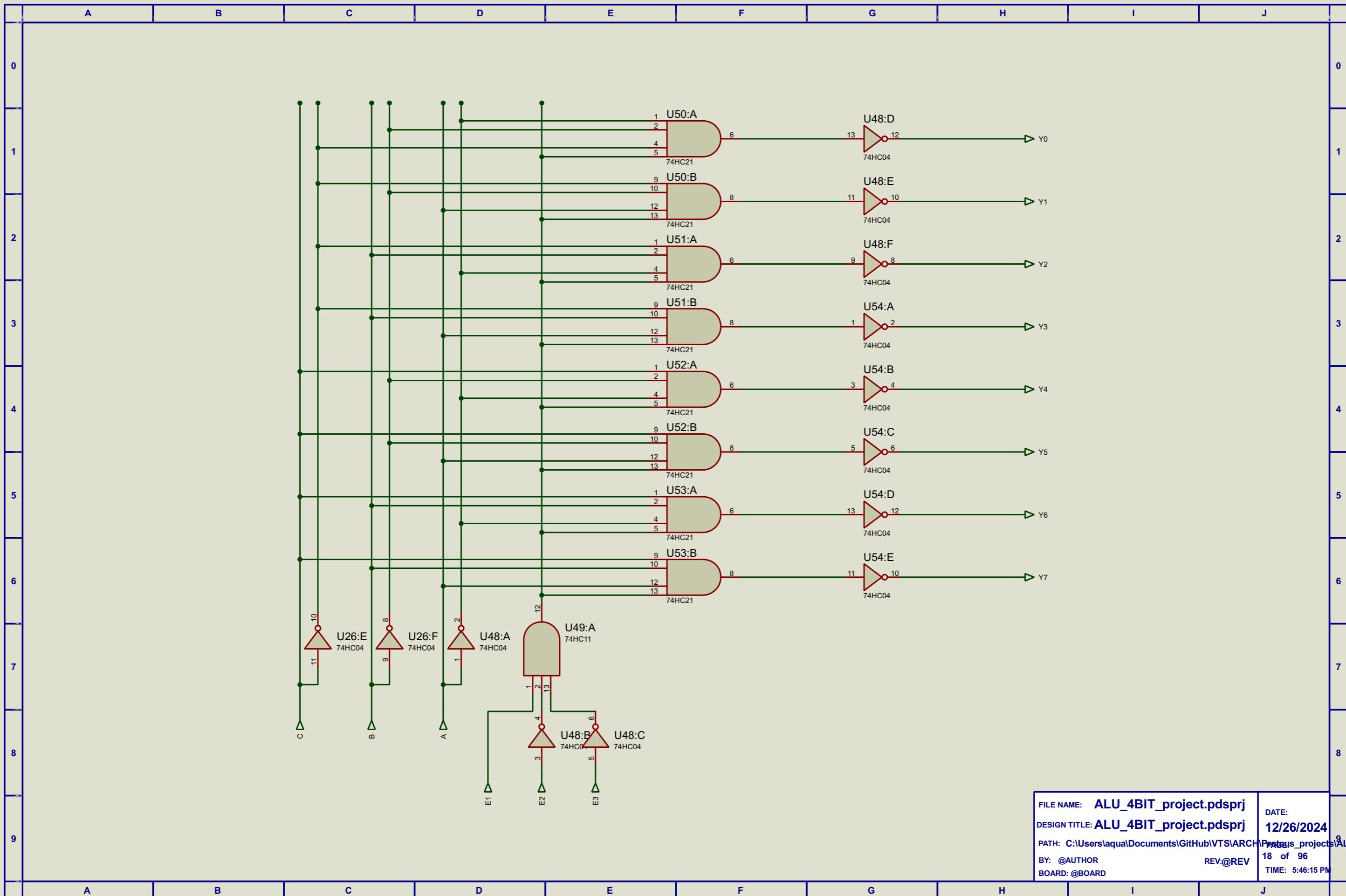
Za komparator koristimo modul za sabiranje i XOR modul

Kada je CTRL=0 radimo komparaciju N brojeva (ulazi ostaju nepromjenjeni)
Kada je CTRL=1 radimo komparaciju Z brojeva (zbog XOR-a MSB od oba ulaza se negira)

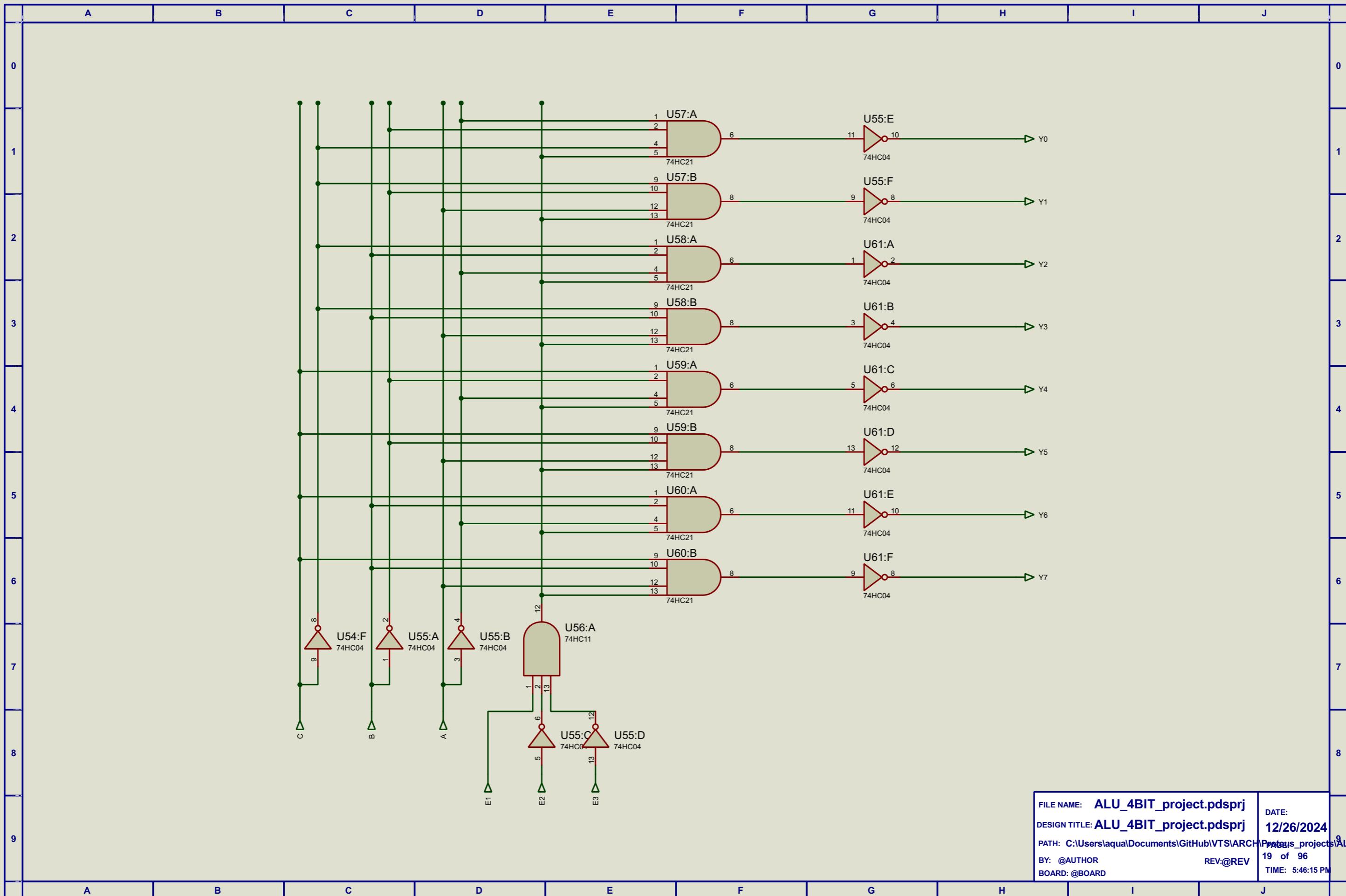


A	B	A>B	A<B	A=B
0	0	0	0	1
0	1	0	1	0
1	0	1	0	0
1	1	0	0	1

A3B3	A2B2	A1B1	A0B0	A>B	A<B	A=B	A>=B	A<=B	A!=B
A3>B3	X	X	X	1	0	0	1	0	1
A3<B3	X	X	X	0	1	0	0	1	1
A3=B3	A2>B2	X	X	1	0	0	1	0	1
A3=B3	A2<B2	X	X	0	1	0	0	1	1
A3=B3	A2=B2	A1>B1	X	1	0	0	1	0	1
A3=B3	A2=B2	A1<B1	X	0	1	0	0	1	1
A3=B3	A2=B2	A1=B1	A0>B0	1	0	0	1	0	1
A3=B3	A2=B2	A1=B1	A0<B0	0	1	0	0	1	1
A3=B3	A2=B2	A1=B1	A0=B0	0	0	1	1	1	0

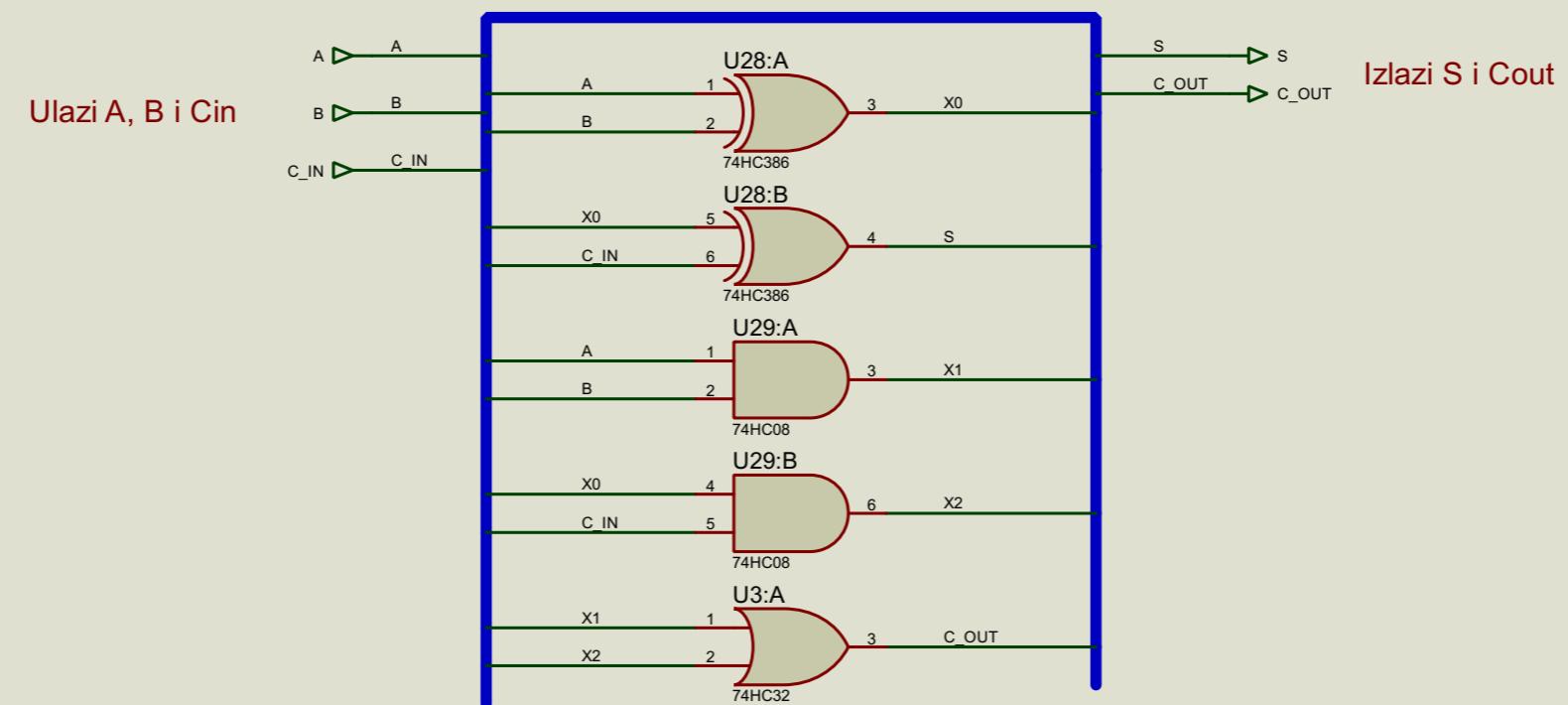


FILE NAME: ALU_4BIT_project.pdsprj
 DESIGN TITLE: ALU_4BIT_project.pdsprj
 PATH: C:\Users\lqual\Documents\GitHub\VTS\ARCHI\Projects\ALU_4BIT_project.pdsprj
 BY: @AUTHOR
 BOARD: @BOARD
 DATE: 12/26/2024
 REV:@REV
 18 of 96
 TIME: 5:46:15 PM



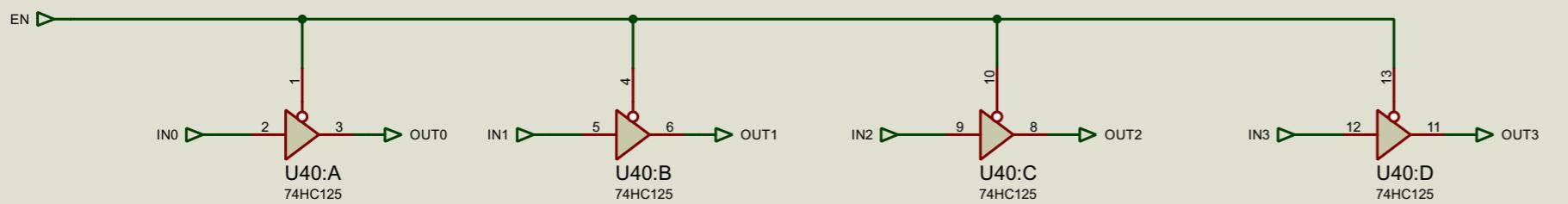
FILE NAME: ALU_4BIT_project.pdsprj
 DESIGN TITLE: ALU_4BIT_project.pdsprj
 PATH: C:\Users\lqual\Documents\GitHub\VTS\ARCHI\Projects\ALU_4BIT_project.pdsprj
 BY: @AUTHOR
 BOARD: @BOARD
 DATE: 12/26/2024
 REV:@REV
 19 of 96
 TIME: 5:46:15 PM

1-bit FA sa carry



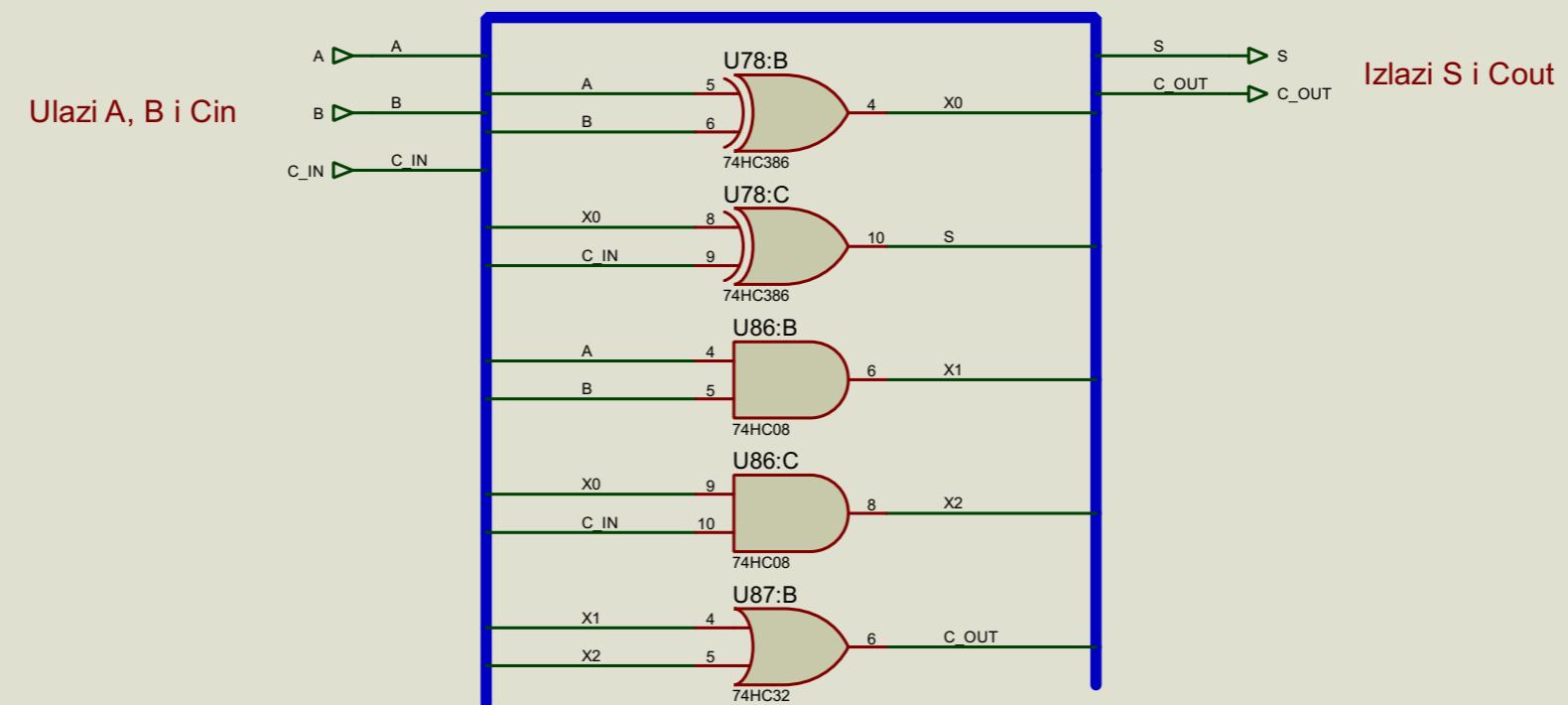
IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)

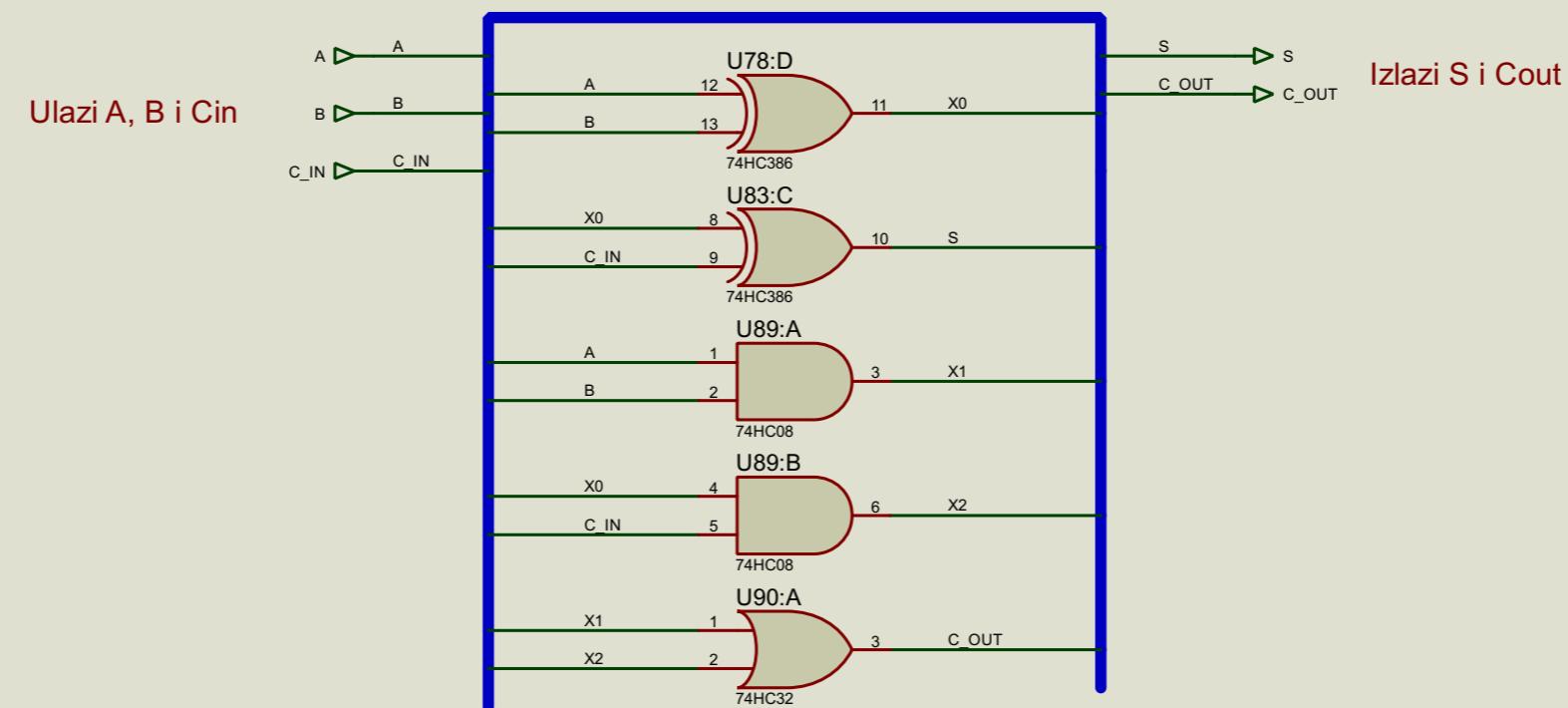


Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

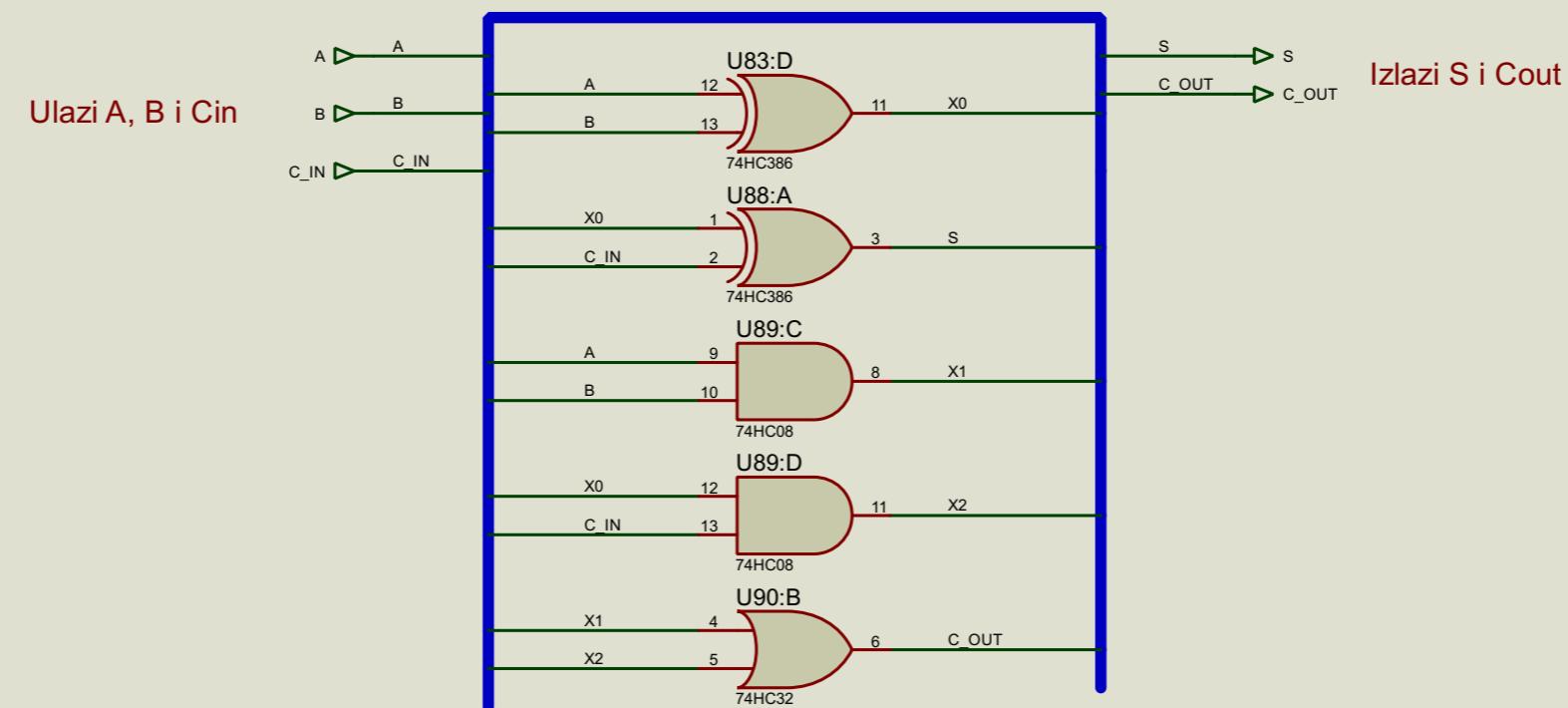
1-bit FA sa carry



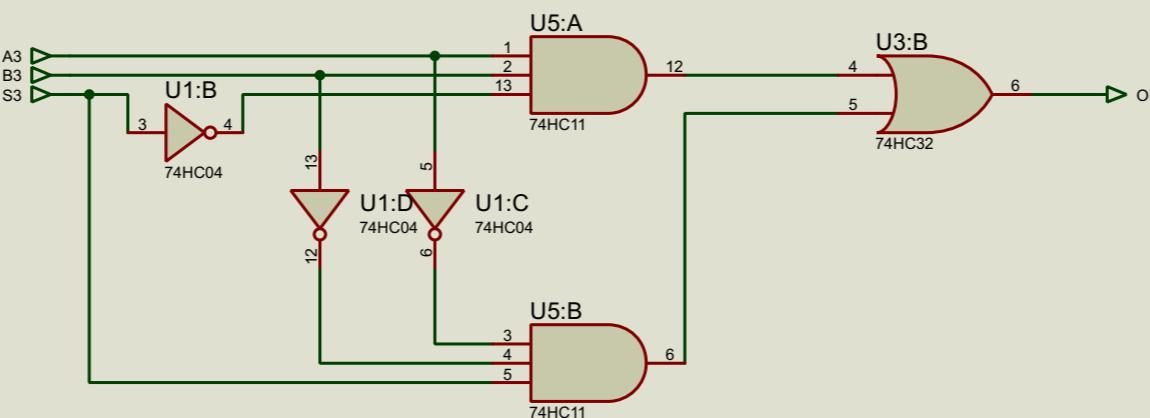
1-bit FA sa carry



1-bit FA sa carry



OVERFLOW CHECK

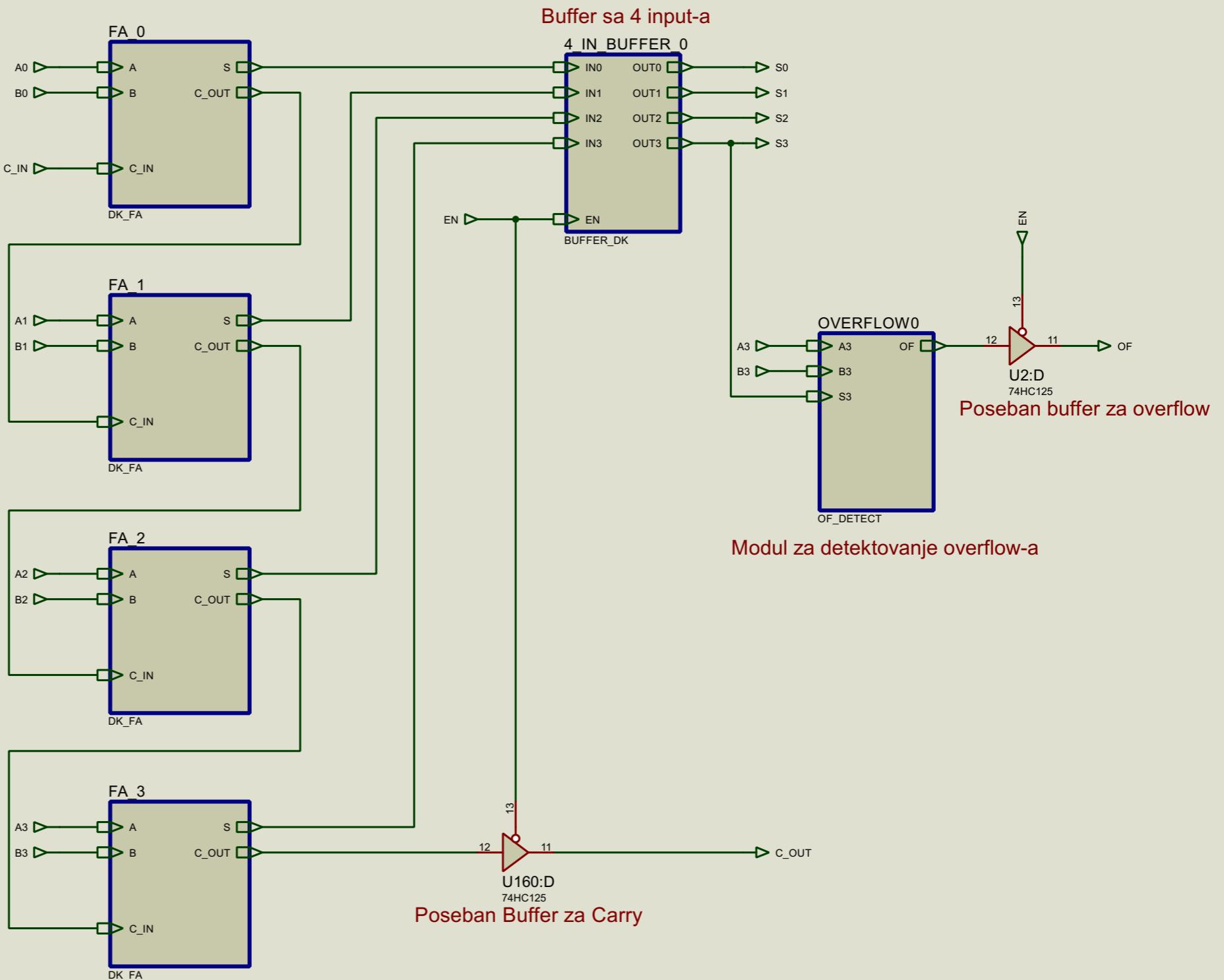


Proveravamo MSB-ove ulaza i rezultata
Ako znak rezultata nije isti kao znak input-a OF=1

A3	B3	S3	OF
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

A	B	Cin	s	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

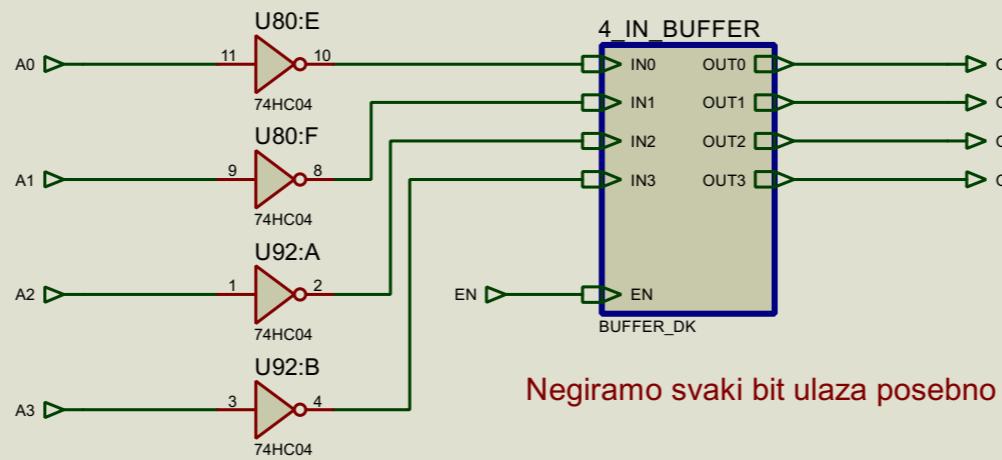
4-bit modul za sabiranje sa carry i overflow



Za svaki bit posebno

A	R
0	1
1	0

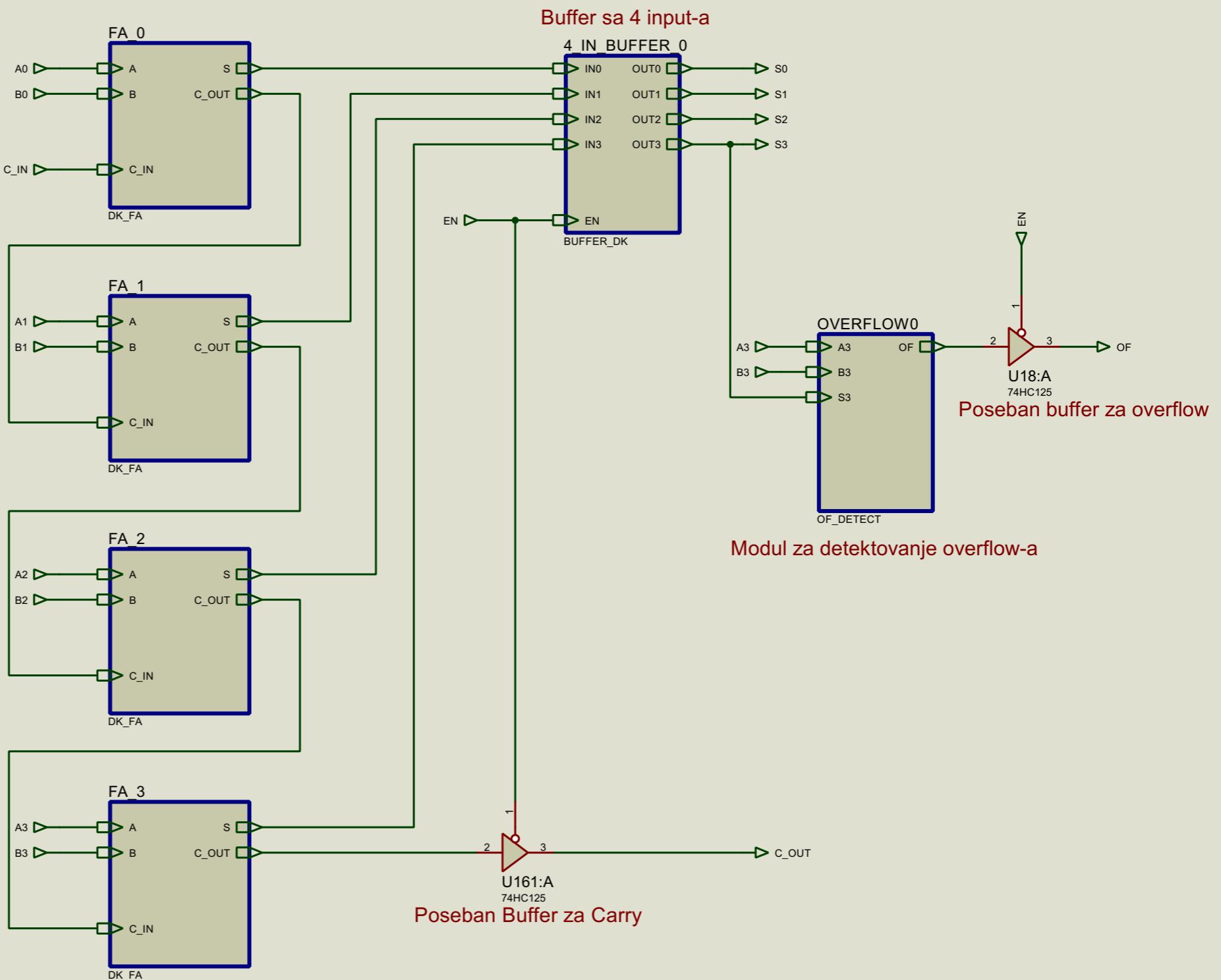
4-bit NOT



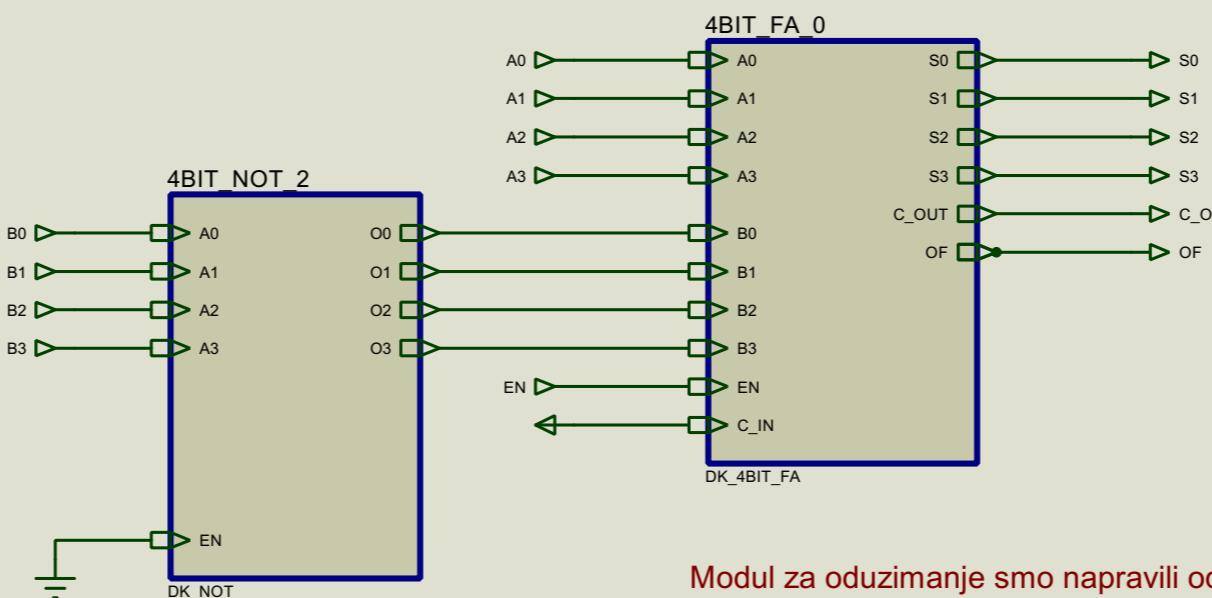
A	B	Cin	s	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

4 1-bitna modula za sabiranje

4-bit modul za sabiranje sa carry i overflow



4-bit modul za oduzimanje sa borrow (c_out) i overflow



A	B	S	Cout
0	0	0	1
0	1	0	0
1	0	1	1
1	1	0	1

Modul za oduzimanje smo napravili od modula za sabiranje

Prvi ulaz ostaje nepromenjen

Drugi ulaz prebacimo u 1. komplement (4-bitni NOT) i uz Cin iz 1. u 2. komplement

Imamo $A + (-B)$, koje je jednako sa $A - B$

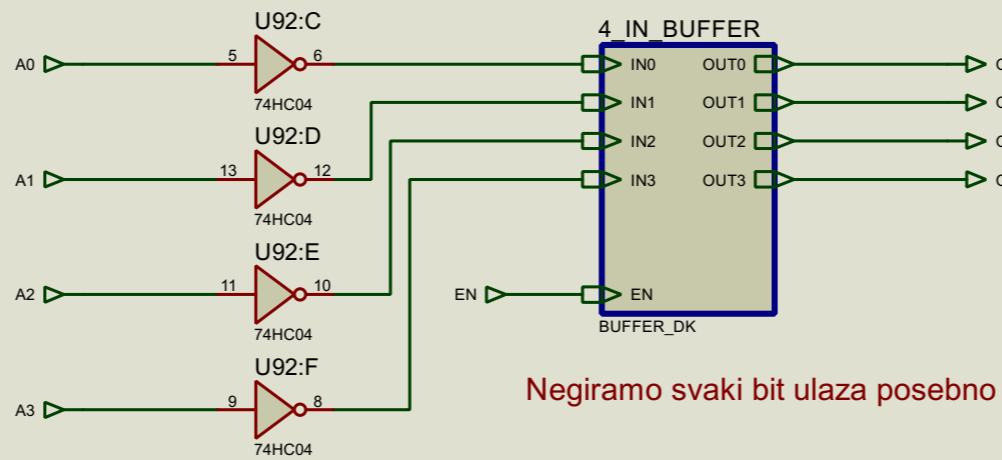
Koristimo OF iz modula za sabiranje

Borrow (c_out) je 1 kada je $A \geq B$ tj. 0 kada je $A < B$

Za svaki bit posebno

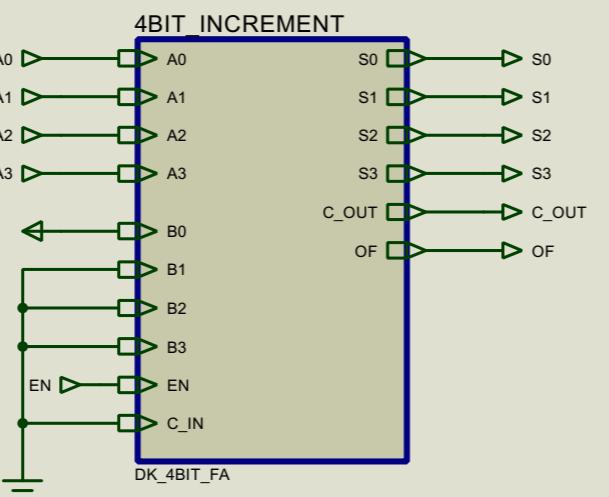
A	R
0	1
1	0

4-bit NOT



A3	A2	A1	A0	R3	R2	R1	R0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

4-bit inkrement sa carry i overflow

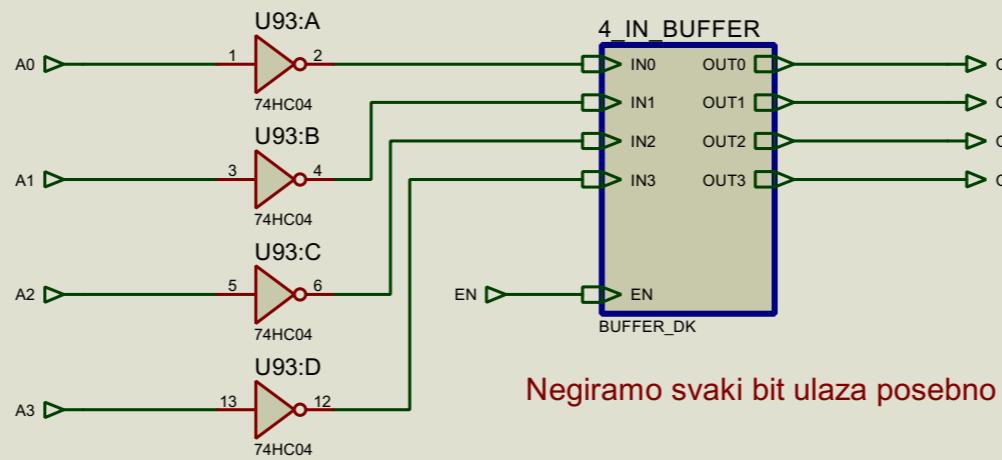


Koristimo modul za sabiranje u koga ubacujemo broj koji inkrementiramo u A input i u B-input-u LSB postavljajmo na 1 a ostale na 0

Za svaki bit posebno

A	R
0	1
1	0

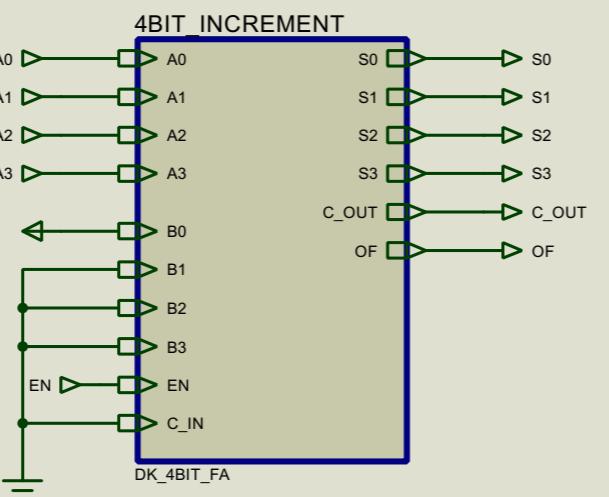
4-bit NOT



Negiramo svaki bit ulaza posebno

A3	A2	A1	A0	R3	R2	R1	R0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

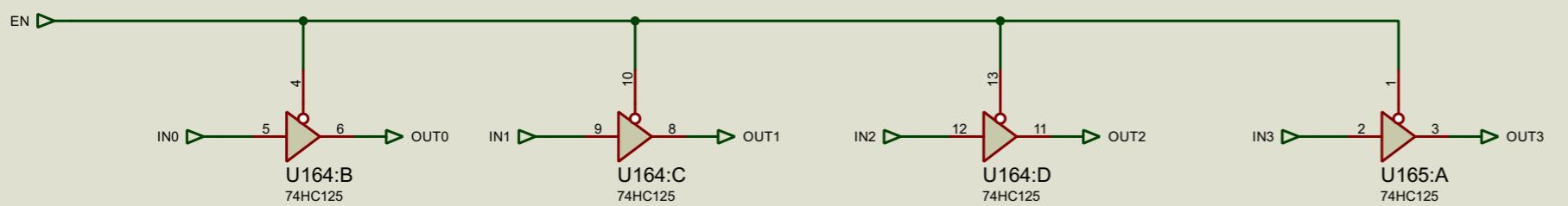
4-bit inkrement sa carry i overflow



Koristimo modul za sabiranje u koga ubacujemo broj koji inkrementiramo u A input i u B-input-u LSB postavljajmo na 1 a ostale na 0

IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

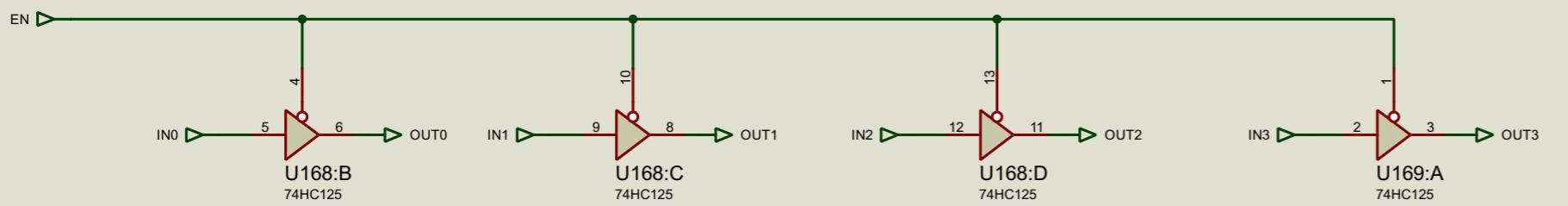
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

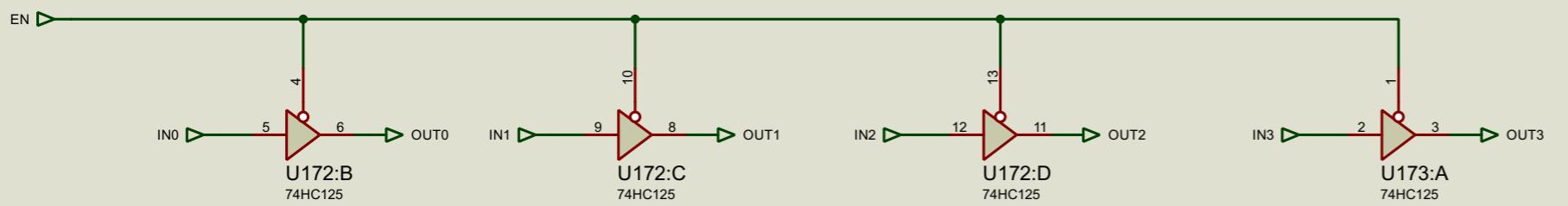
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)

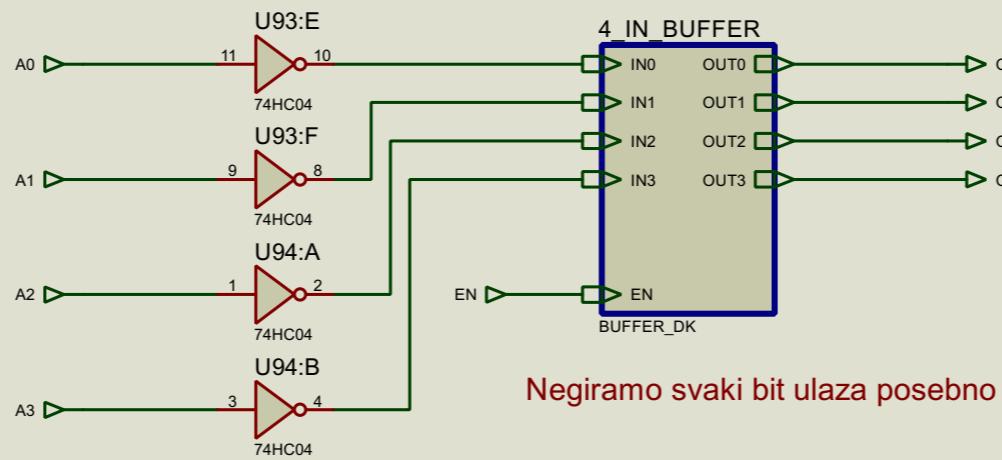


Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

Za svaki bit posebno

A	R
0	1
1	0

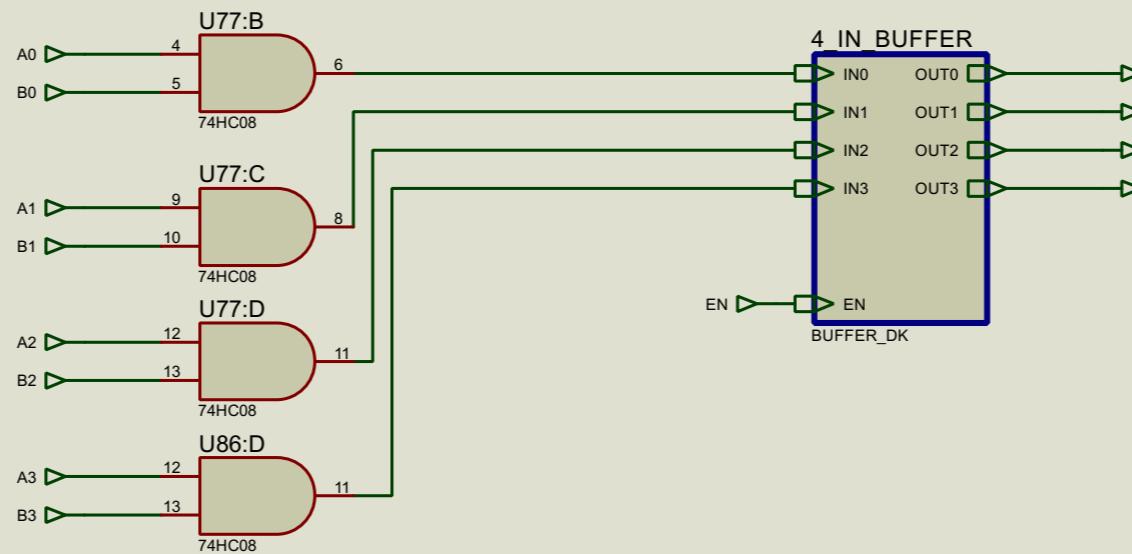
4-bit NOT



Za svaki bit posebno

A	B	R
0	0	0
0	1	0
1	0	0
1	1	1

4-bit AND

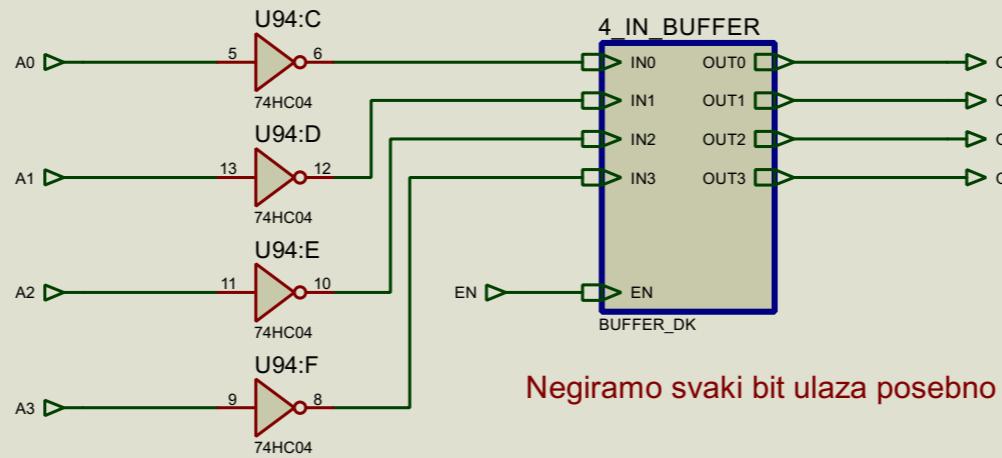


Radimo AND svakog bita A i B posebno

Za svaki bit posebno

A	R
0	1
1	0

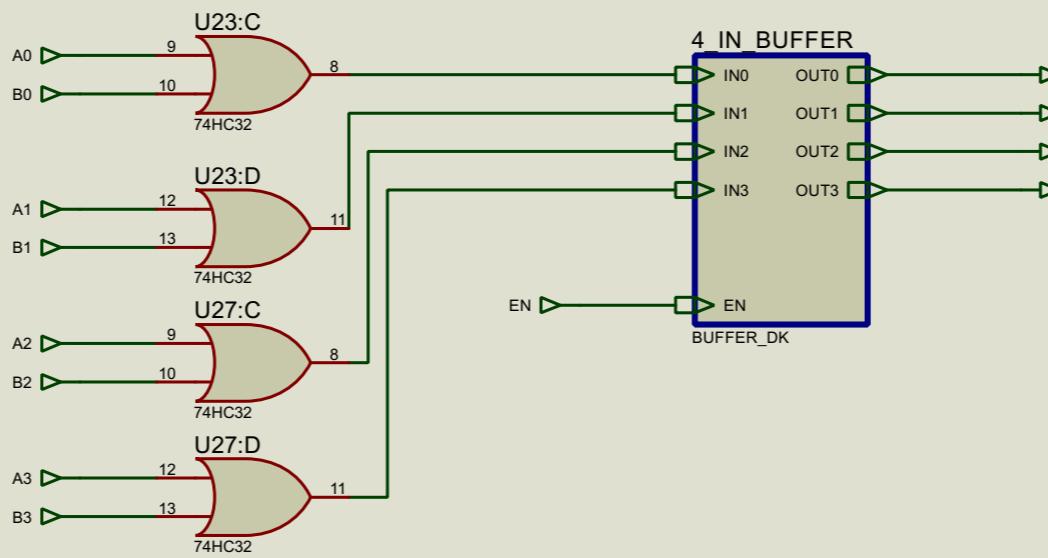
4-bit NOT



Za svaki bit posebno

A	B	R
0	0	0
0	1	1
1	0	1
1	1	1

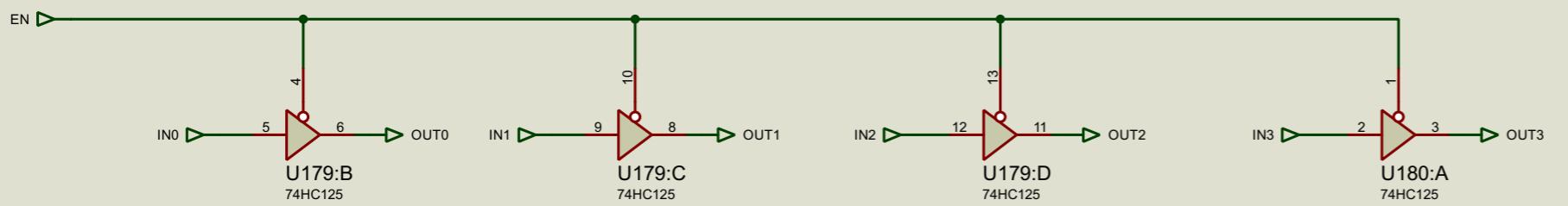
4-bit OR



Radimo OR svakog bita A i B posebno

IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

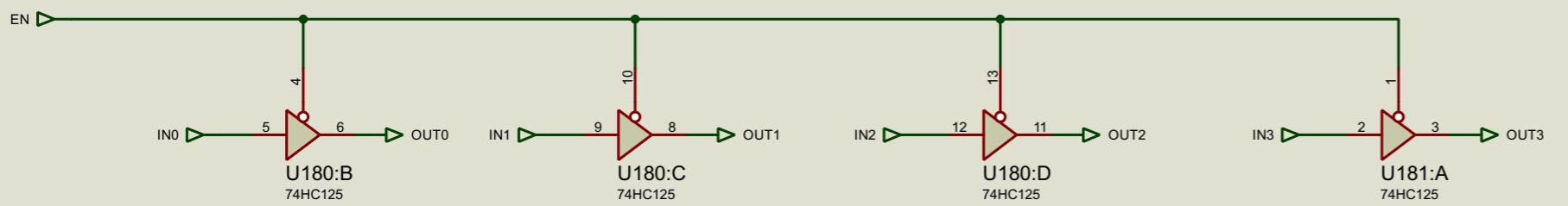
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

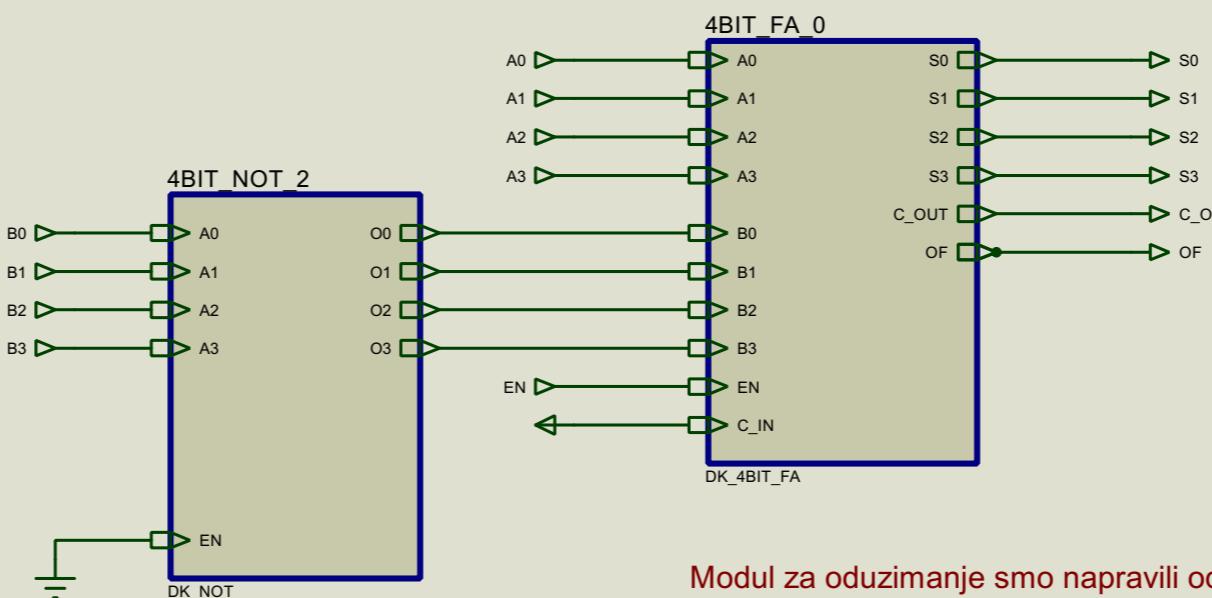
IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	1
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

4-bit modul za oduzimanje sa borrow (c_out) i overflow



A	B	S	Cout
0	0	0	1
0	1	0	0
1	0	1	1
1	1	0	1

Modul za oduzimanje smo napravili od modula za sabiranje

Prvi ulaz ostaje nepromenjen

Drugi ulaz prebacimo u 1. komplement (4-bitni NOT) i uz Cin iz 1. u 2. komplement

Imamo $A + (-B)$, koje je jednako sa $A - B$

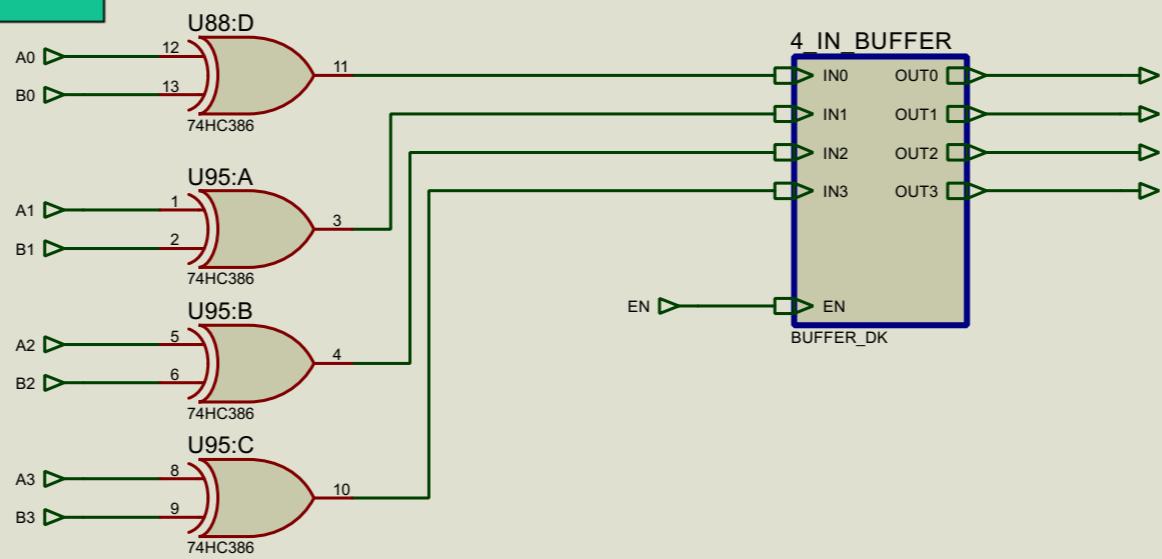
Koristimo OF iz modula za sabiranje

Borrow (c_out) je 1 kada je $A \geq B$ tj. 0 kada je $A < B$

Za svaki bit posebno

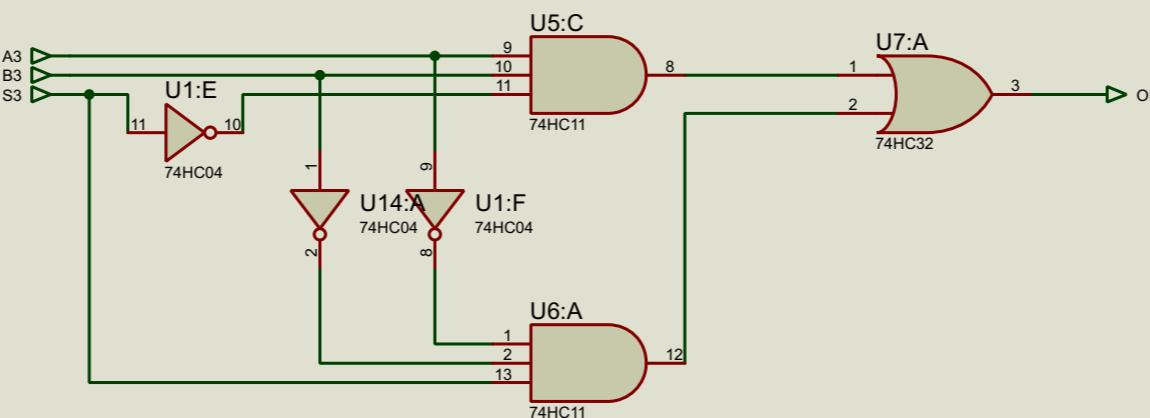
A	B	R
0	0	0
0	1	1
1	0	1
1	1	0

4-bit XOR



Radimo XOR svakog bita A i B posebno

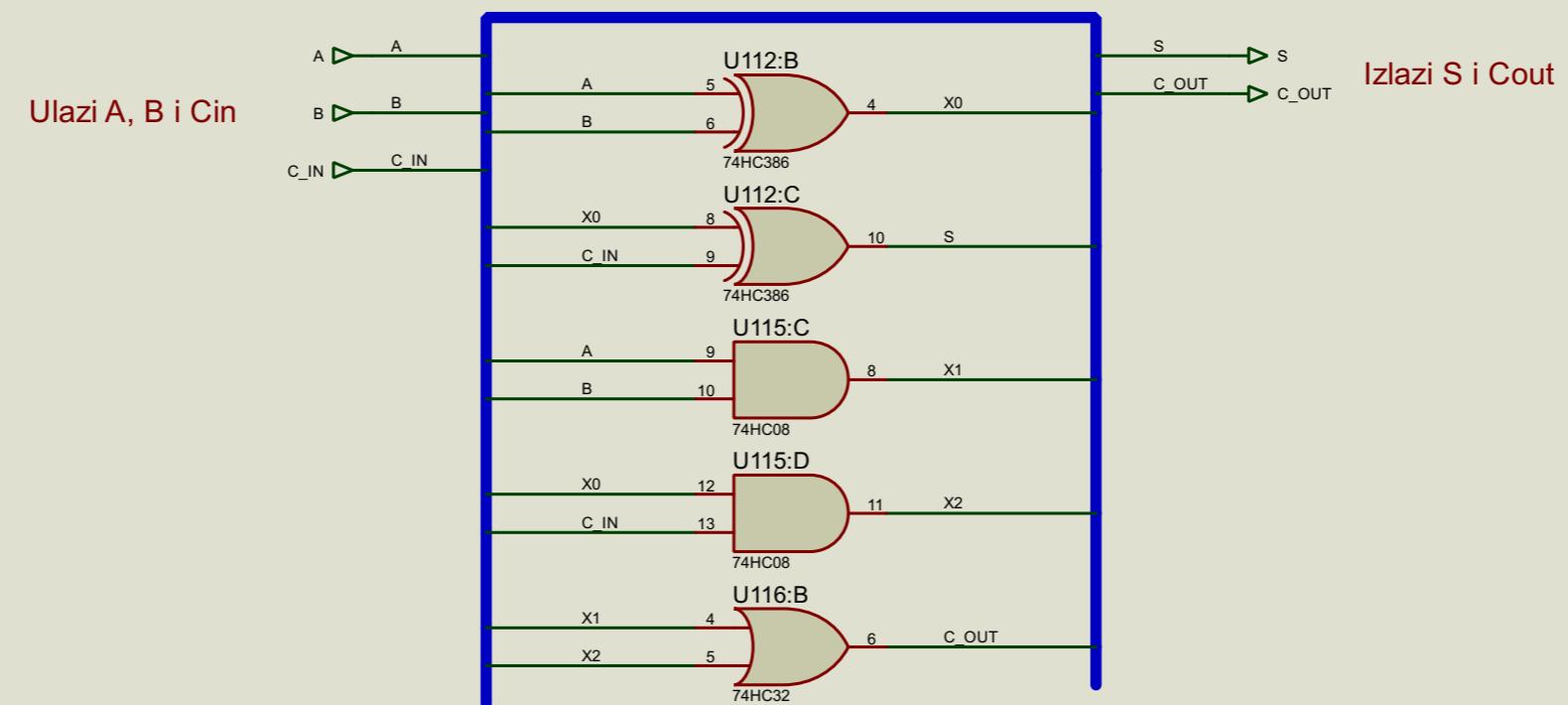
OVERFLOW CHECK



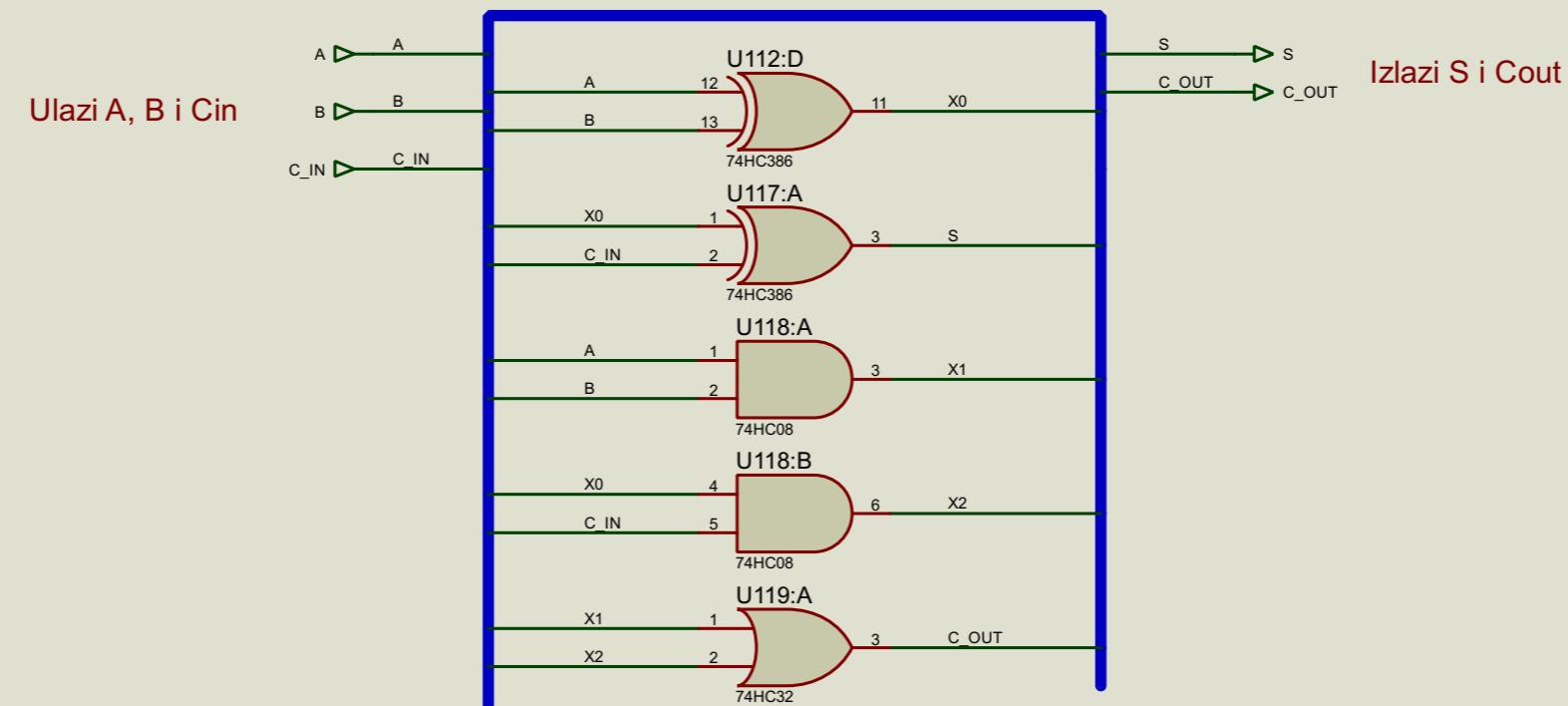
Proveravamo MSB-ove ulaza i rezultata
Ako znak rezultata nije isti kao znak input-a OF=1

A3	B3	S3	OF
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

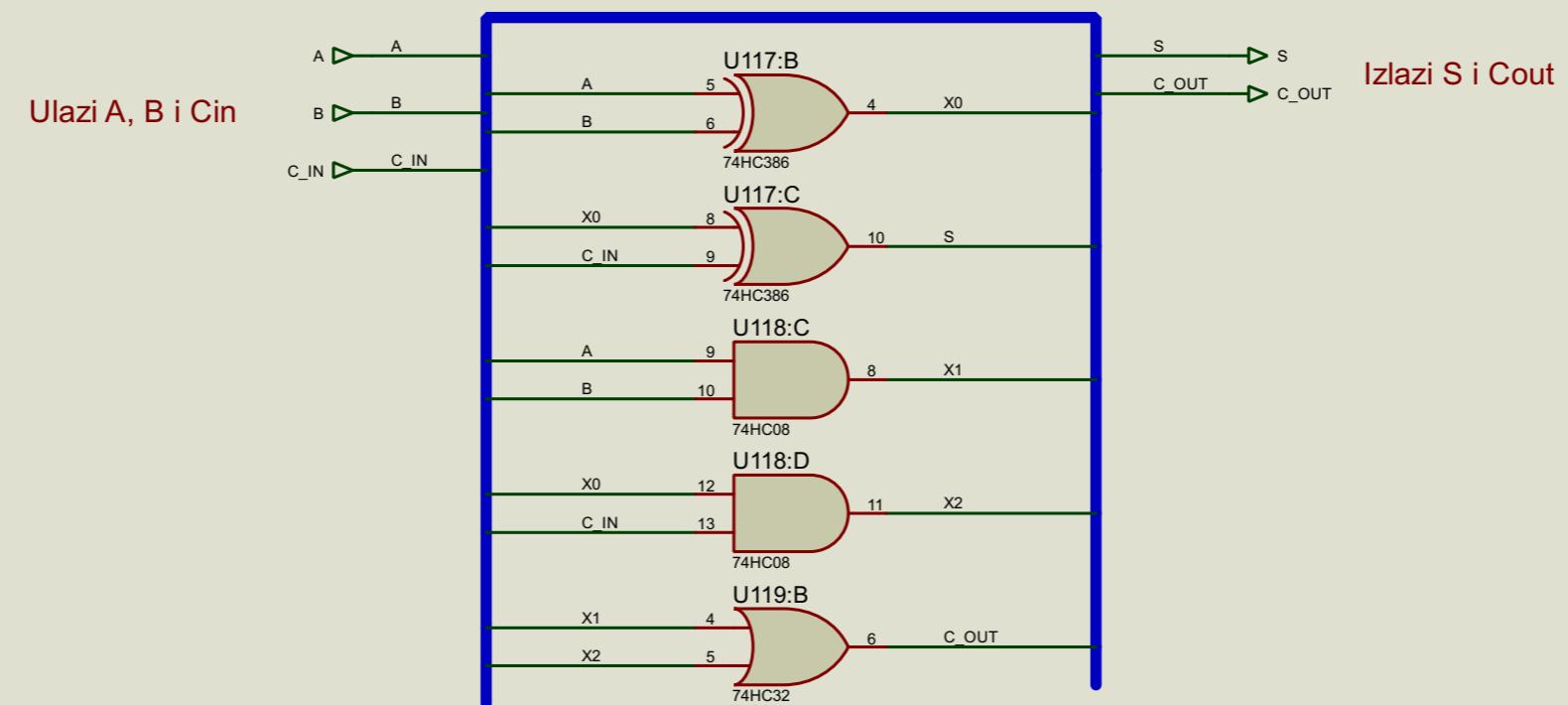
1-bit FA sa carry



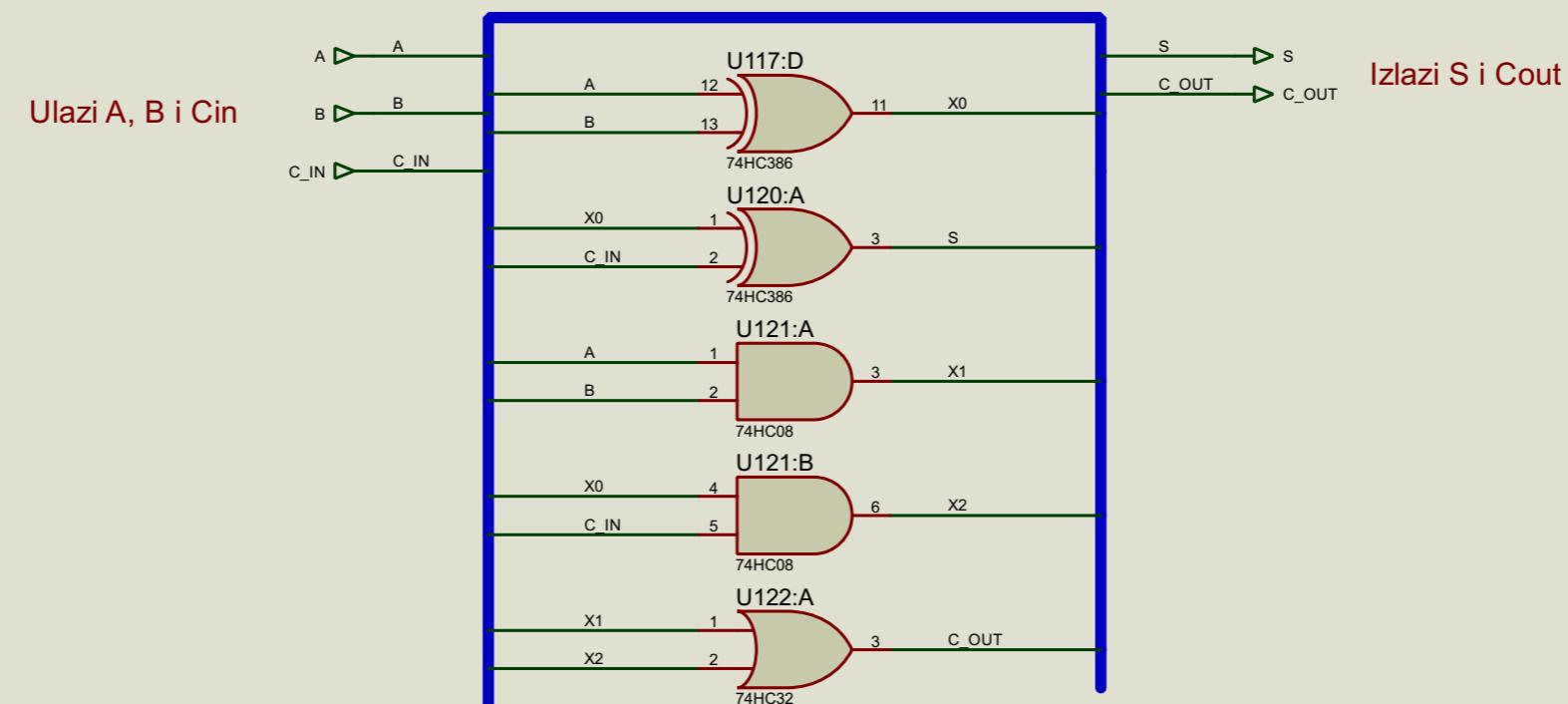
1-bit FA sa carry



1-bit FA sa carry

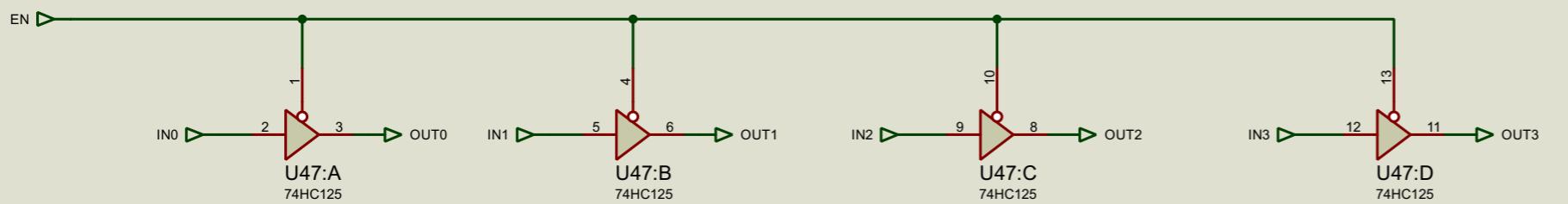


1-bit FA sa carry



IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	1	0	0	1	1	1	0
1	1	1	1	0	1	1	1	1

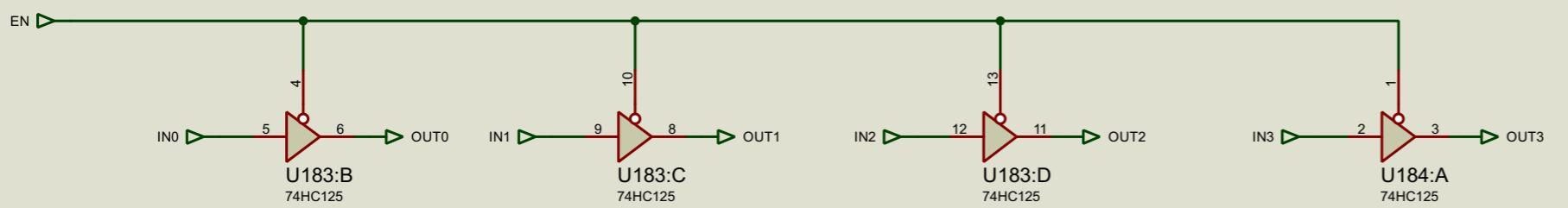
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

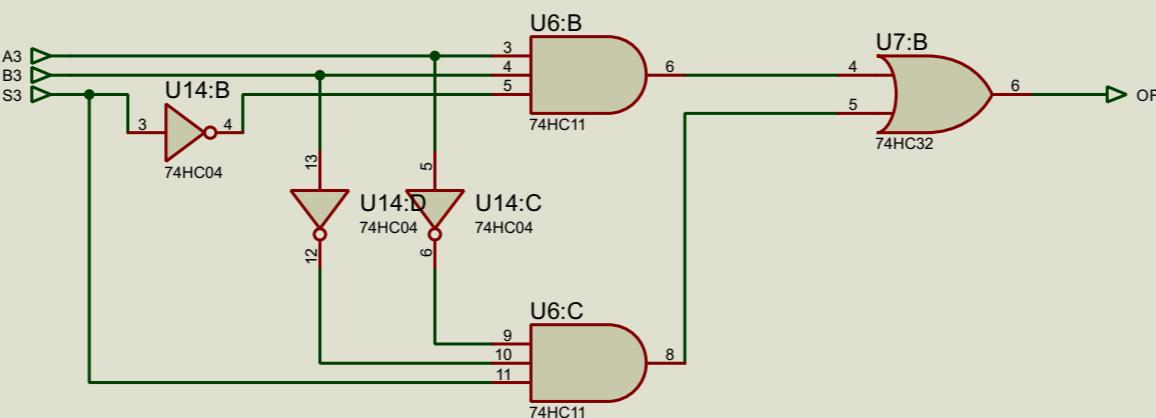
IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

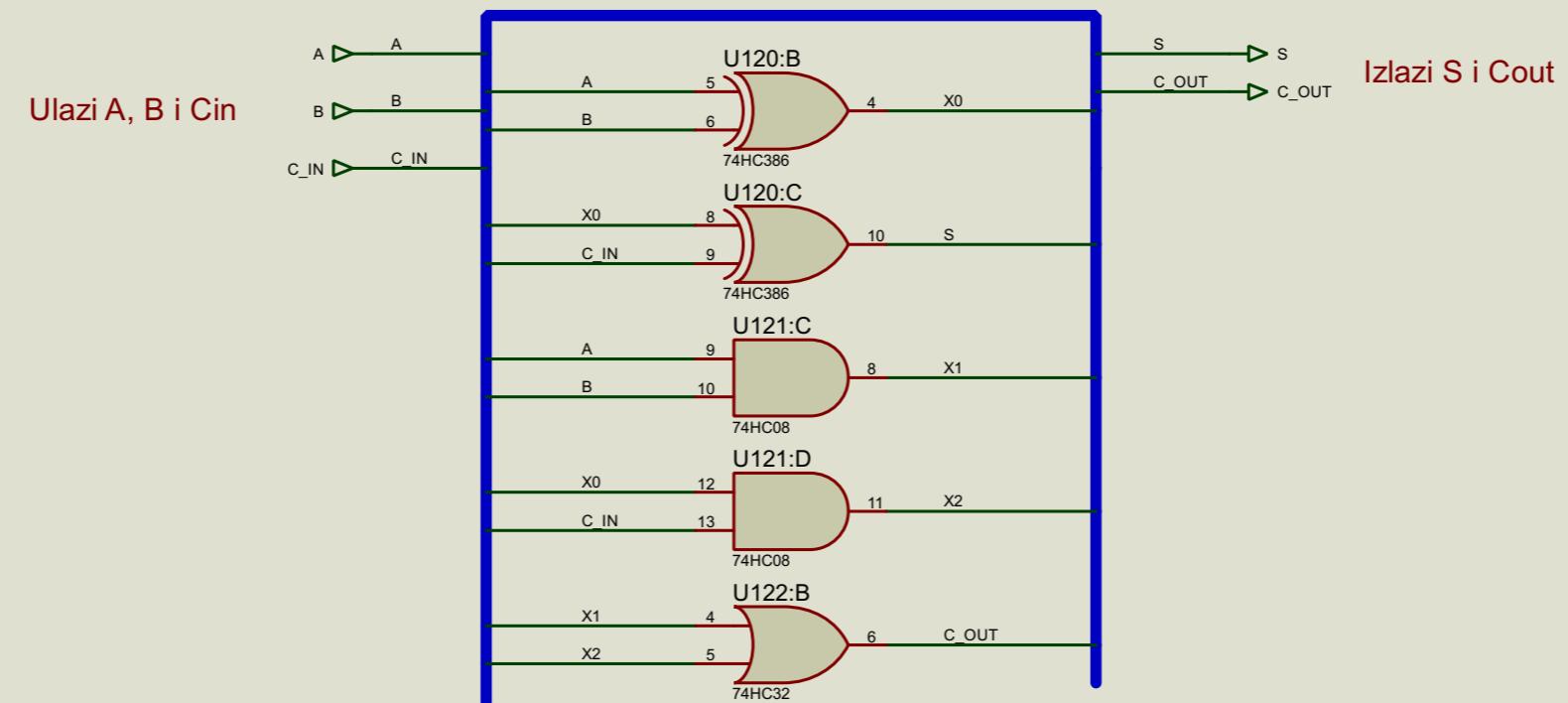
OVERFLOW CHECK



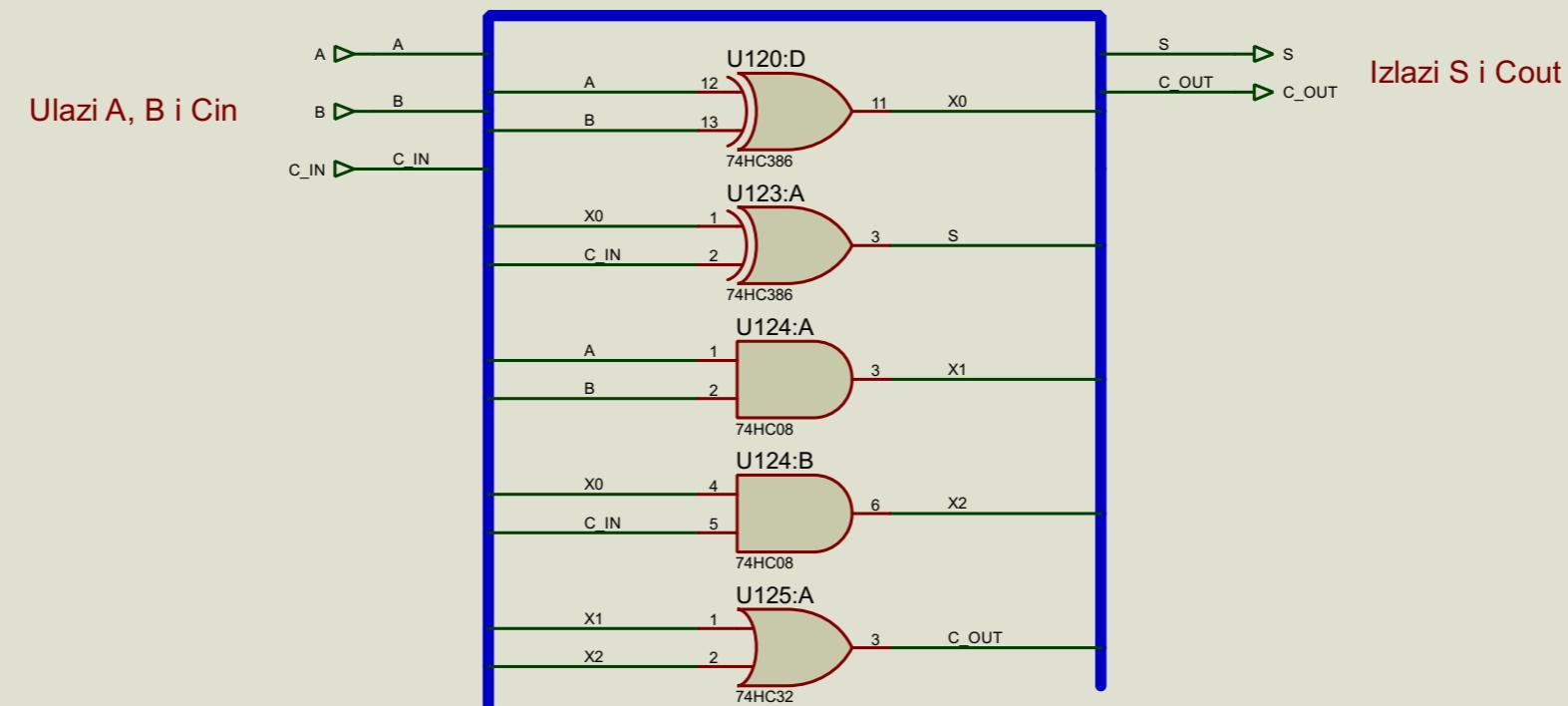
Proveravamo MSB-ove ulaza i rezultata
Ako znak rezultata nije isti kao znak input-a OF=1

A3	B3	S3	OF
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

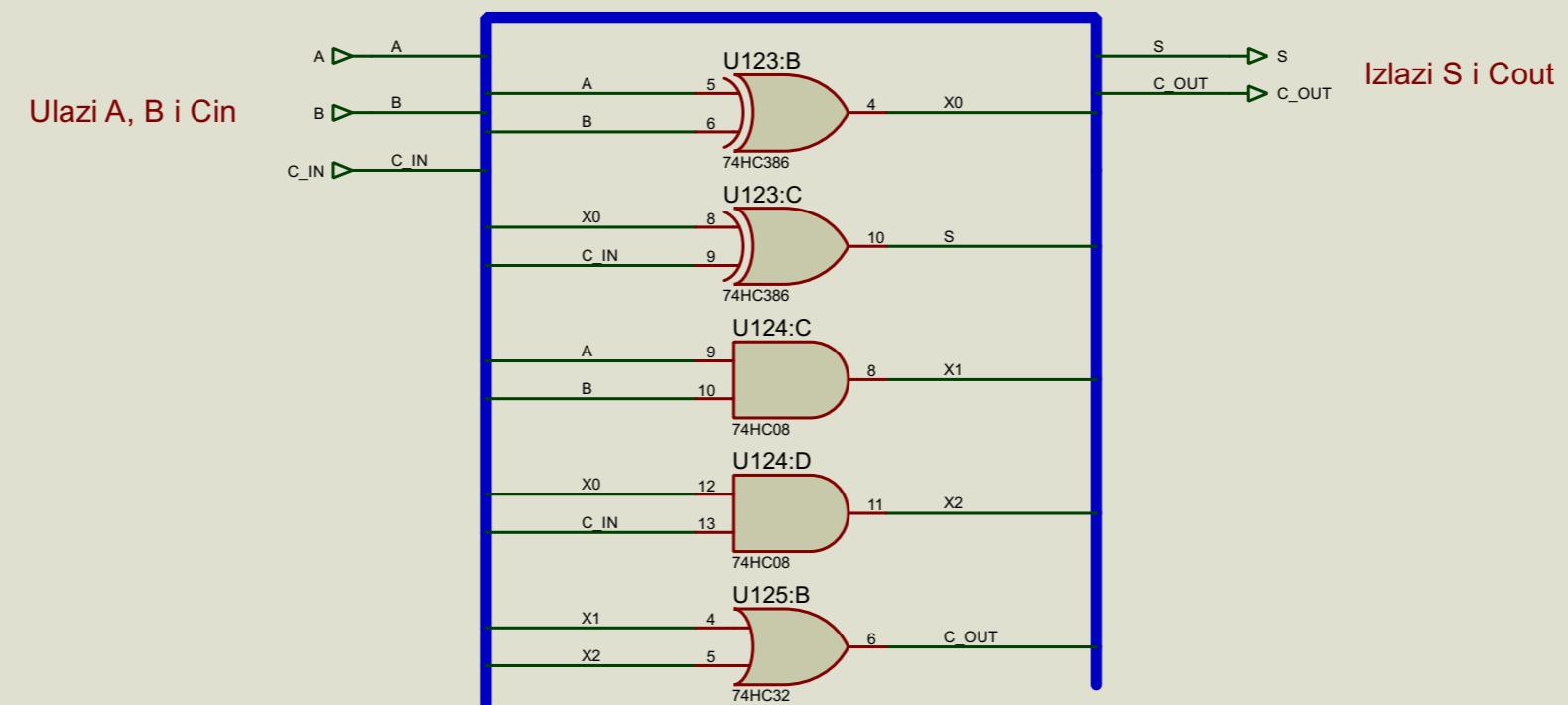
1-bit FA sa carry



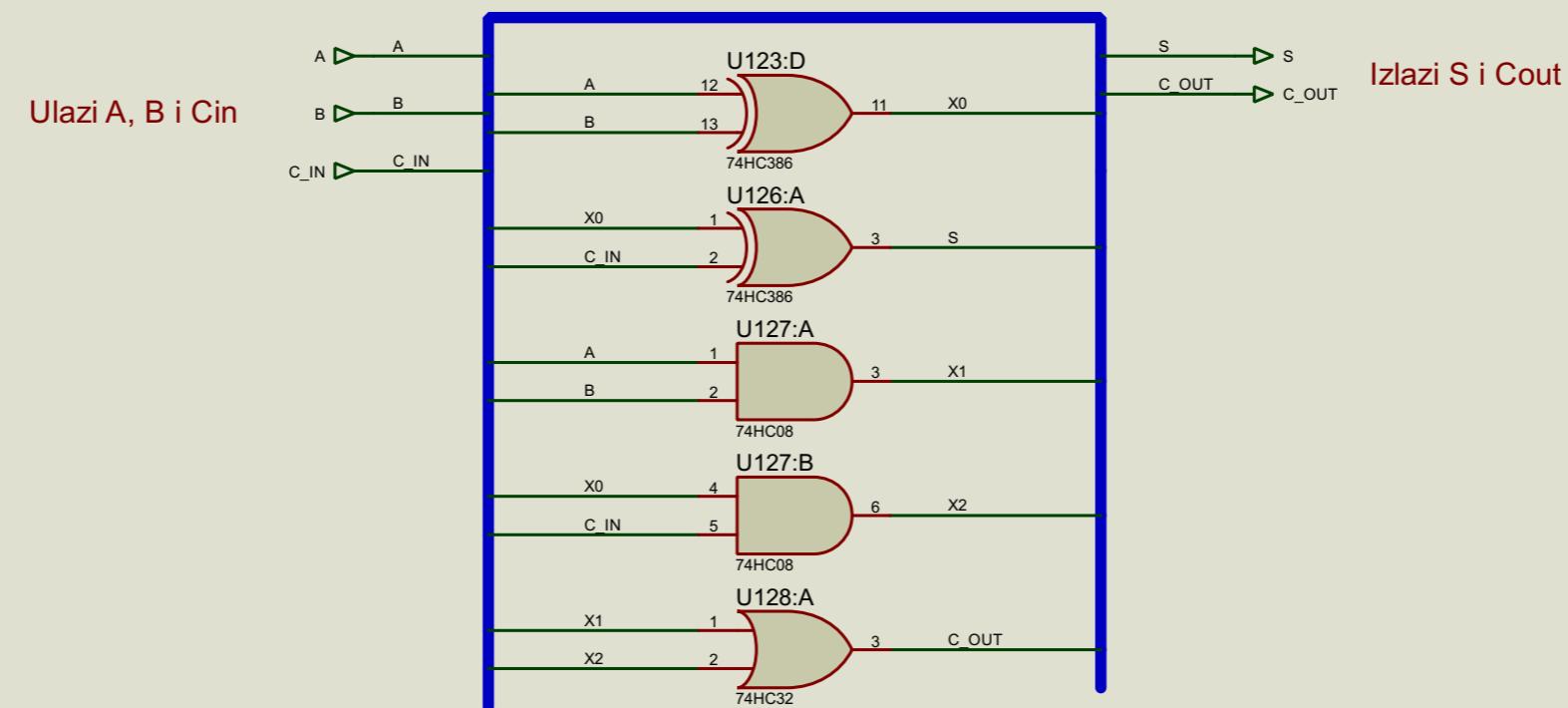
1-bit FA sa carry



1-bit FA sa carry

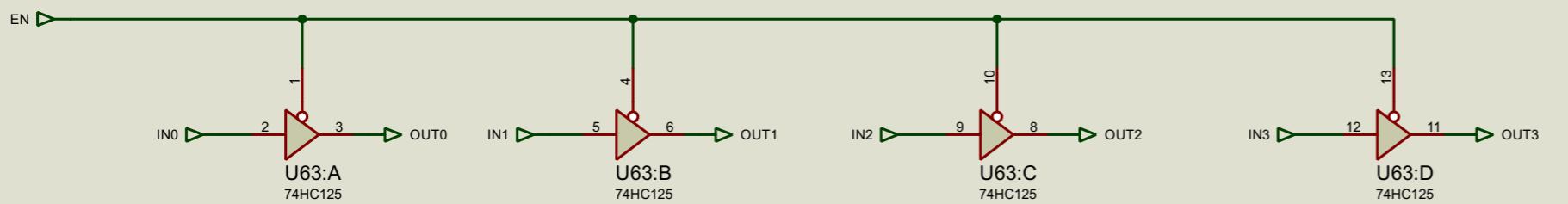


1-bit FA sa carry



IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	1	0	0	1	1	1	0
1	1	1	1	0	1	1	1	1

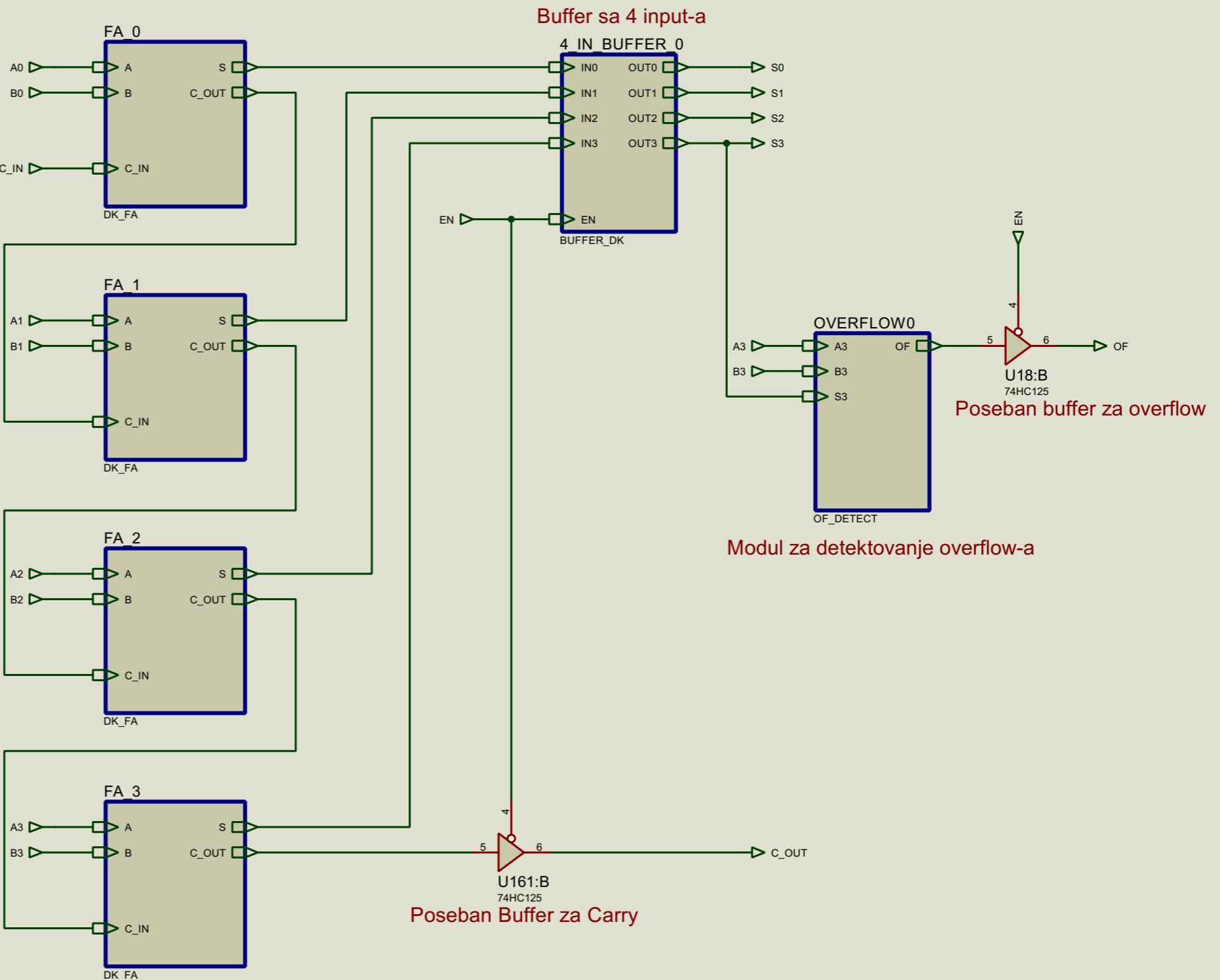
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

A	B	Cin	s	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

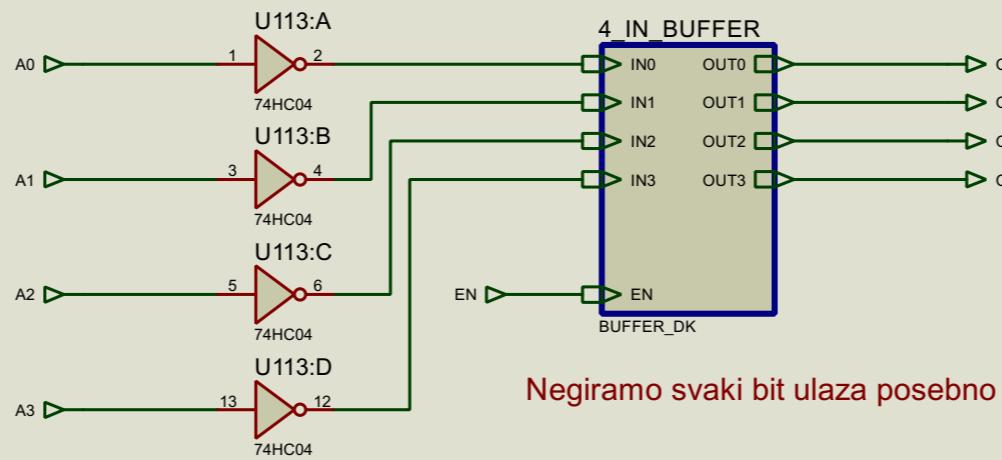
4-bit modul za sabiranje sa carry i overflow



Za svaki bit posebno

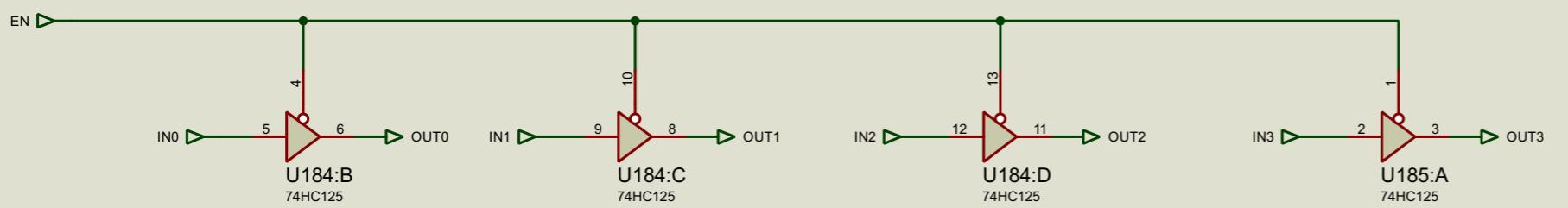
A	R
0	1
1	0

4-bit NOT



IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)

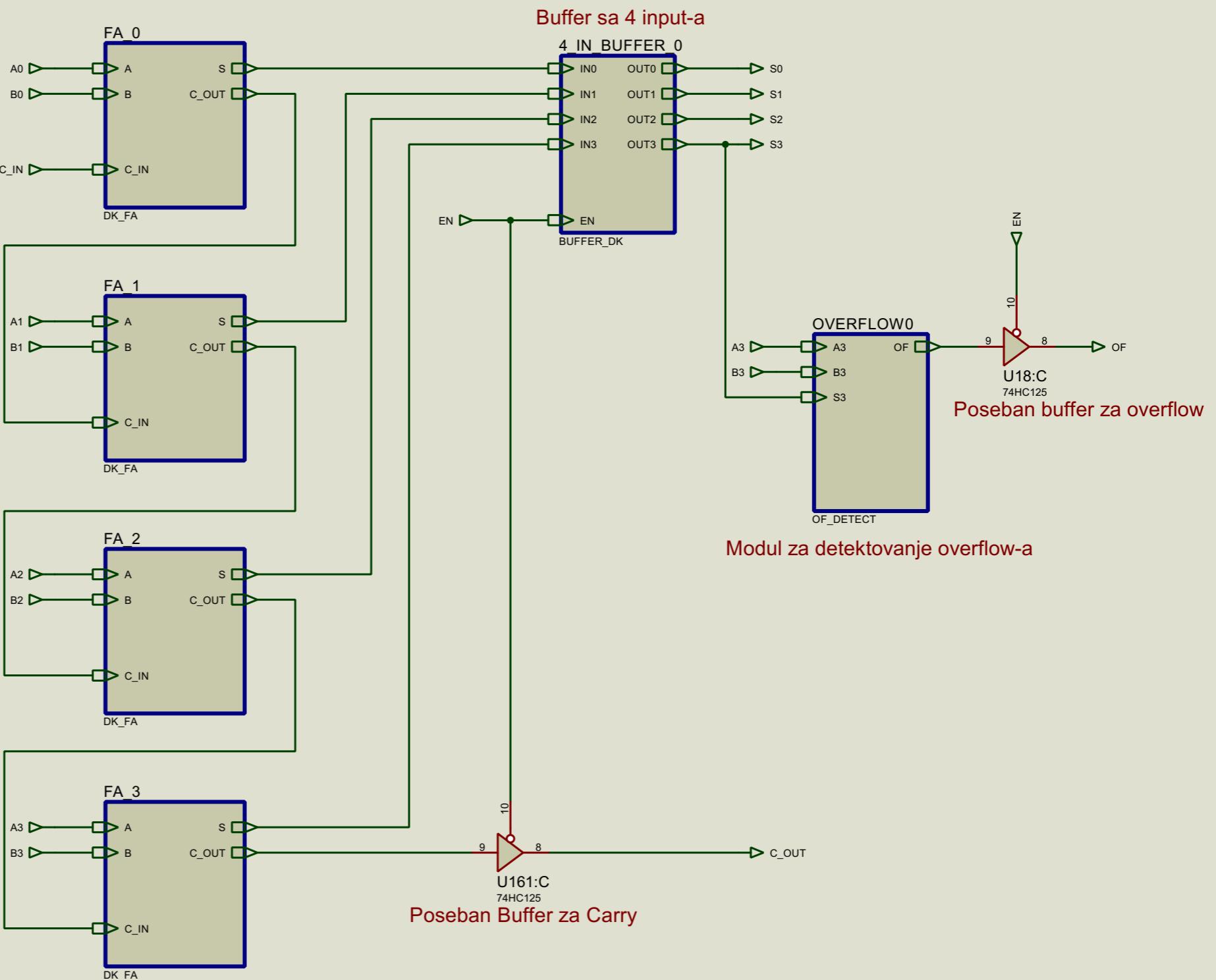


Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

A	B	Cin	s	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

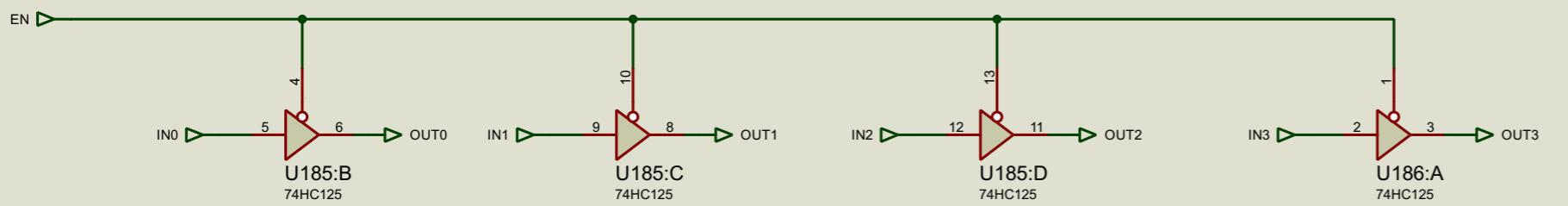
4 1-bitna modula za sabiranje

4-bit modul za sabiranje sa carry i overflow



IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

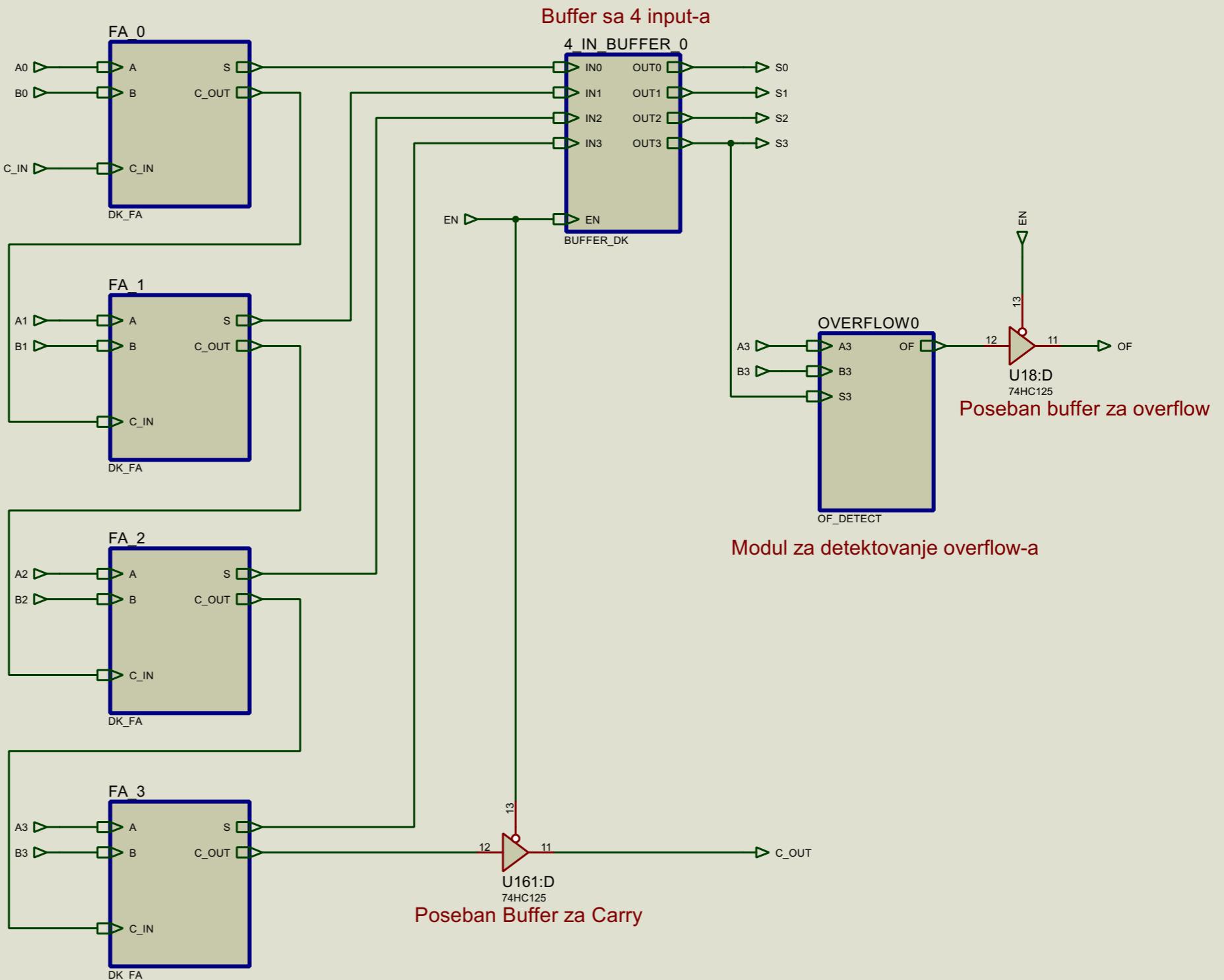
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

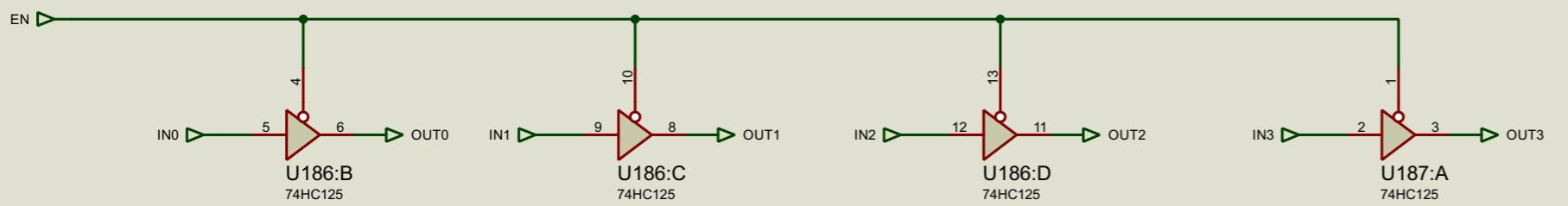
A	B	Cin	s	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

4-bit modul za sabiranje sa carry i overflow



IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

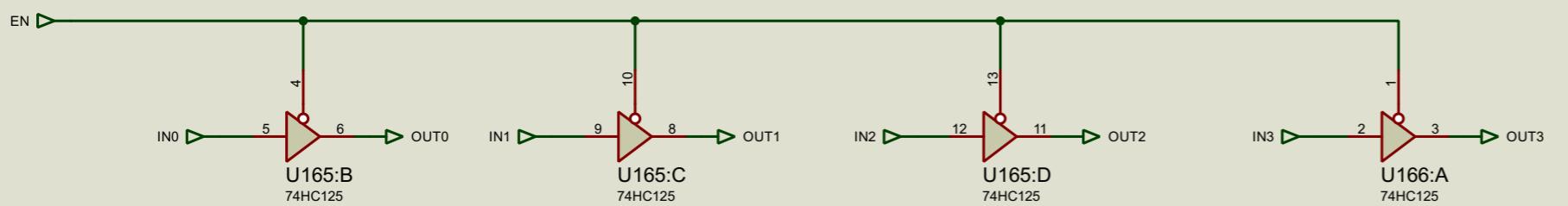
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

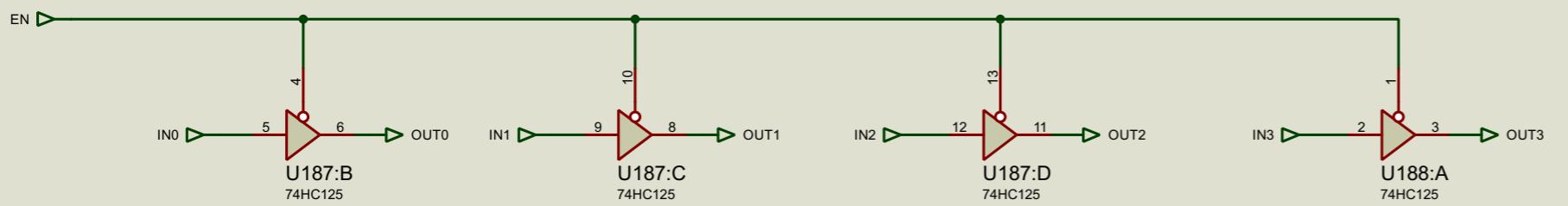
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

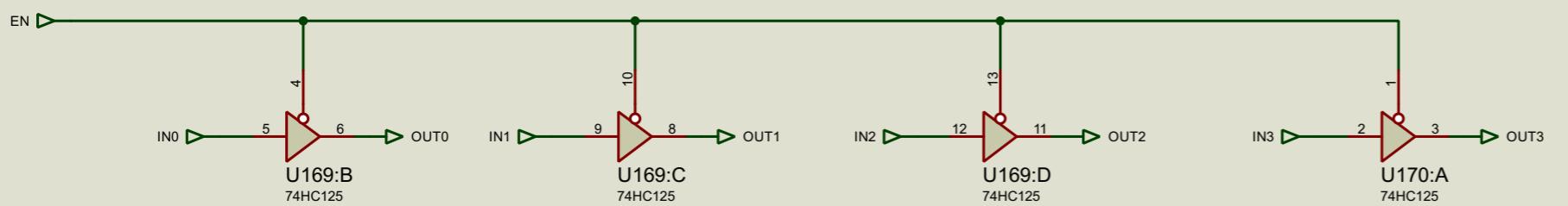
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)

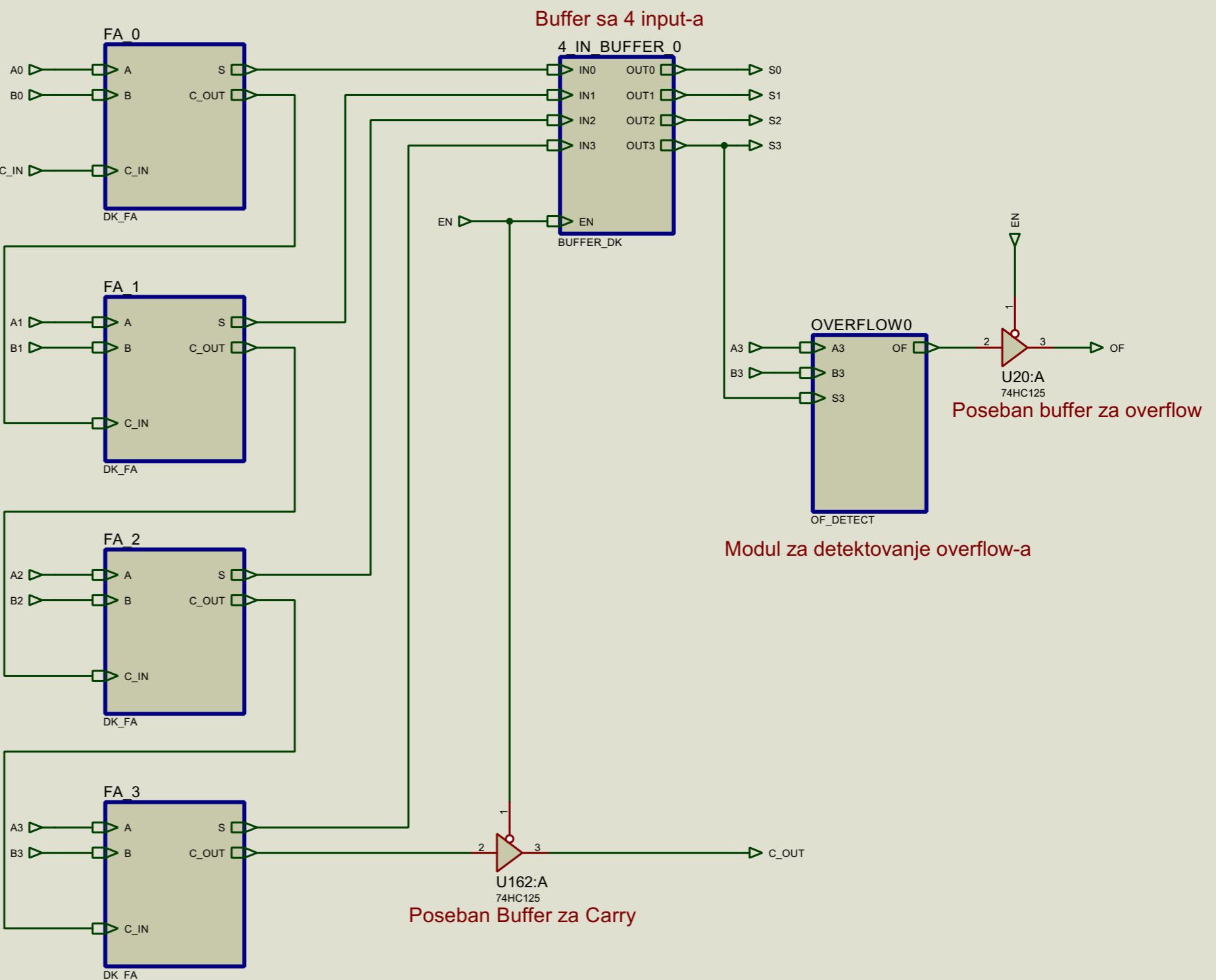


Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

A	B	Cin	s	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

4 1-bitna modula za sabiranje

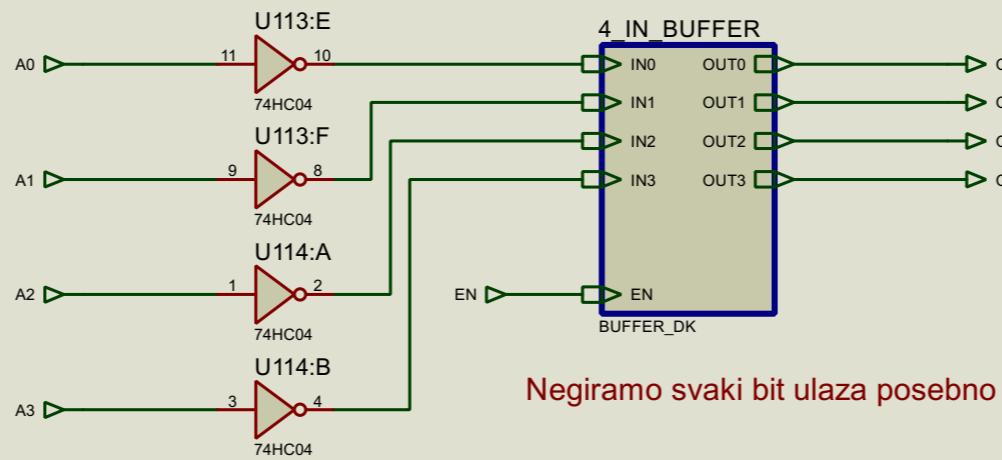
4-bit modul za sabiranje sa carry i overflow



Za svaki bit posebno

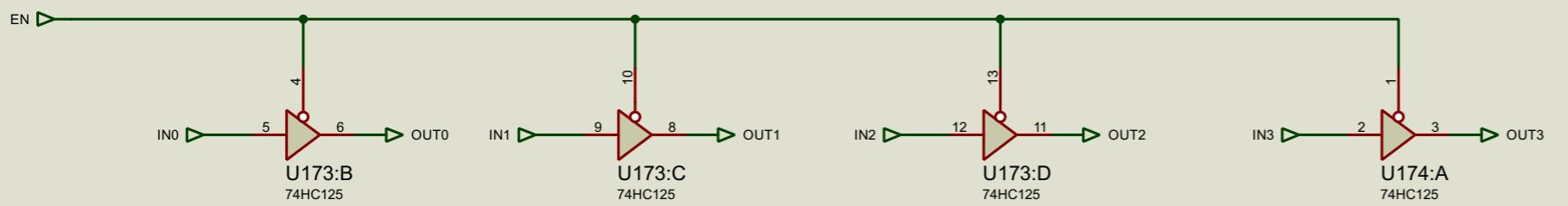
A	R
0	1
1	0

4-bit NOT



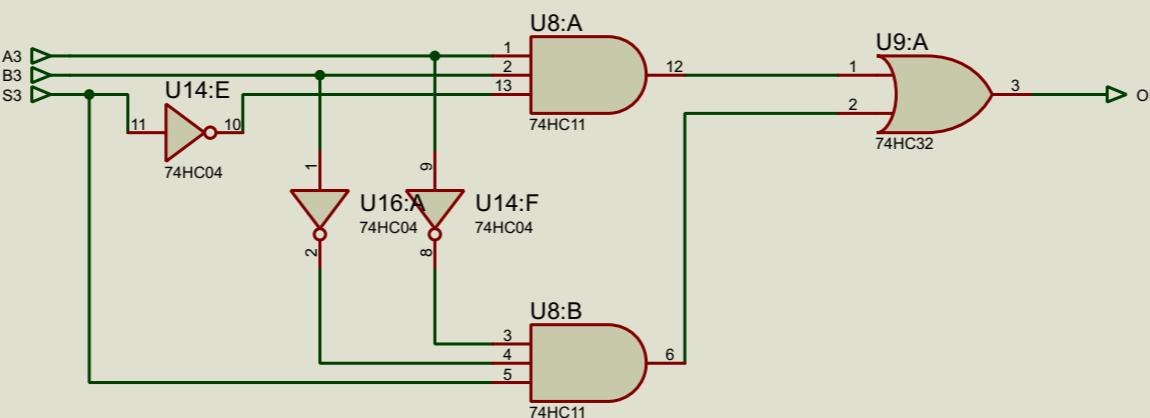
IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

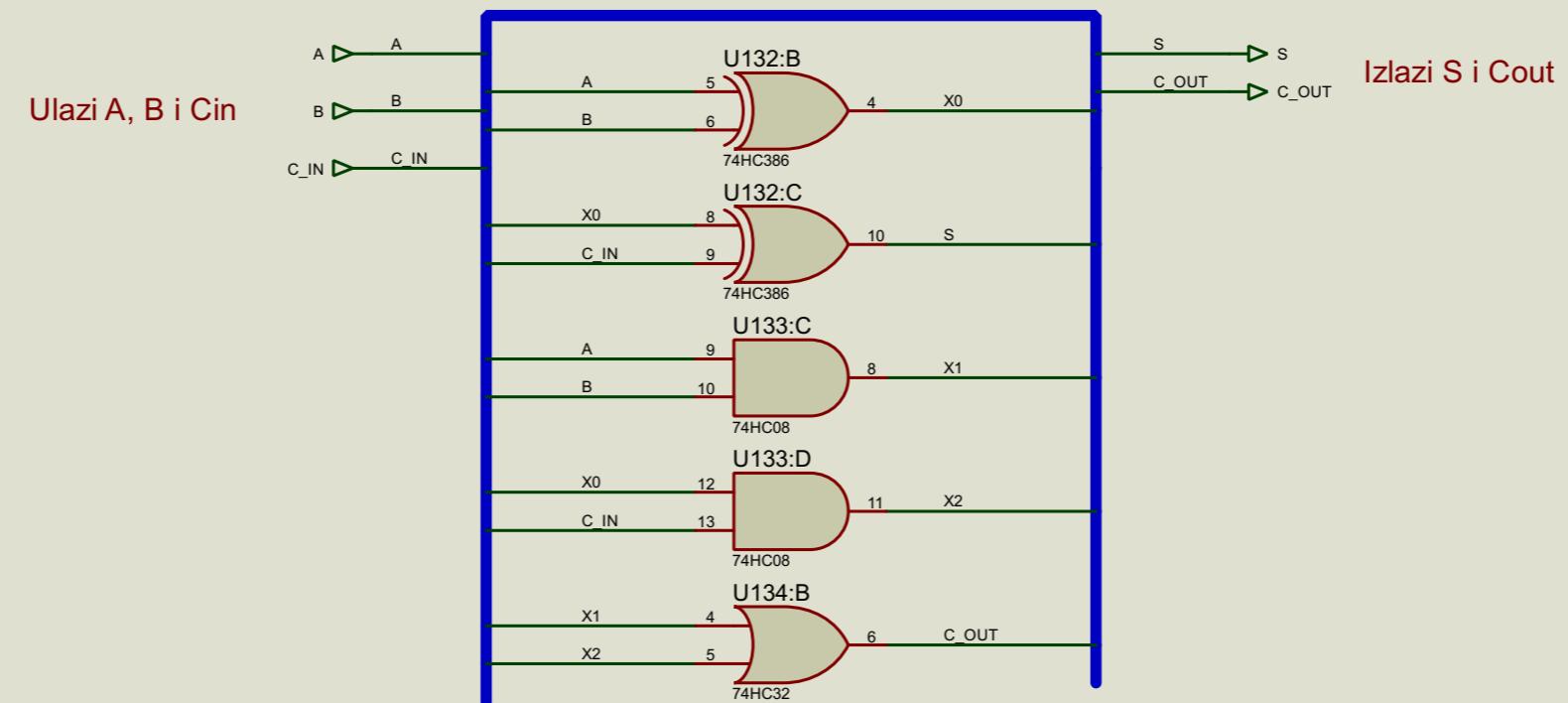
OVERFLOW CHECK



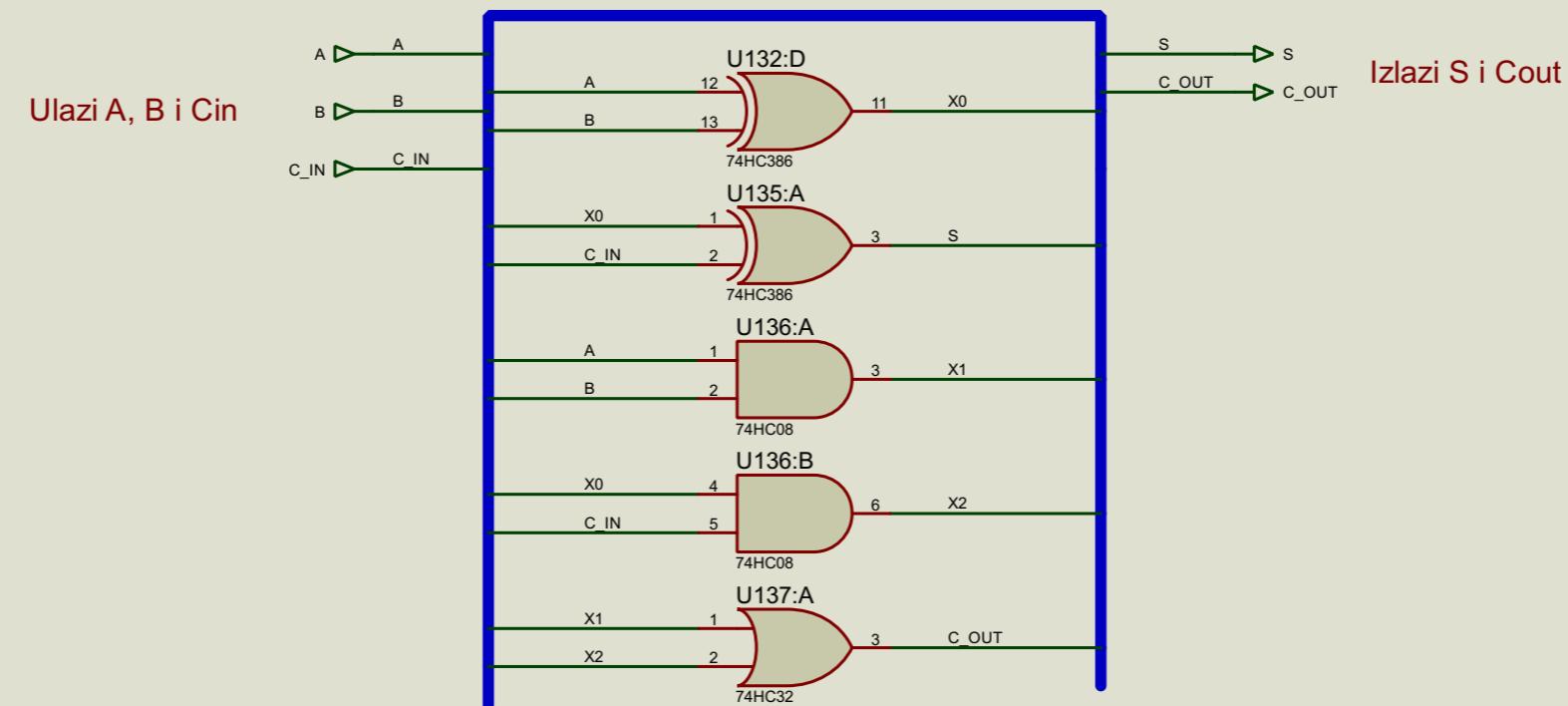
Proveravamo MSB-ove ulaza i rezultata
Ako znak rezultata nije isti kao znak input-a OF=1

A3	B3	S3	OF
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

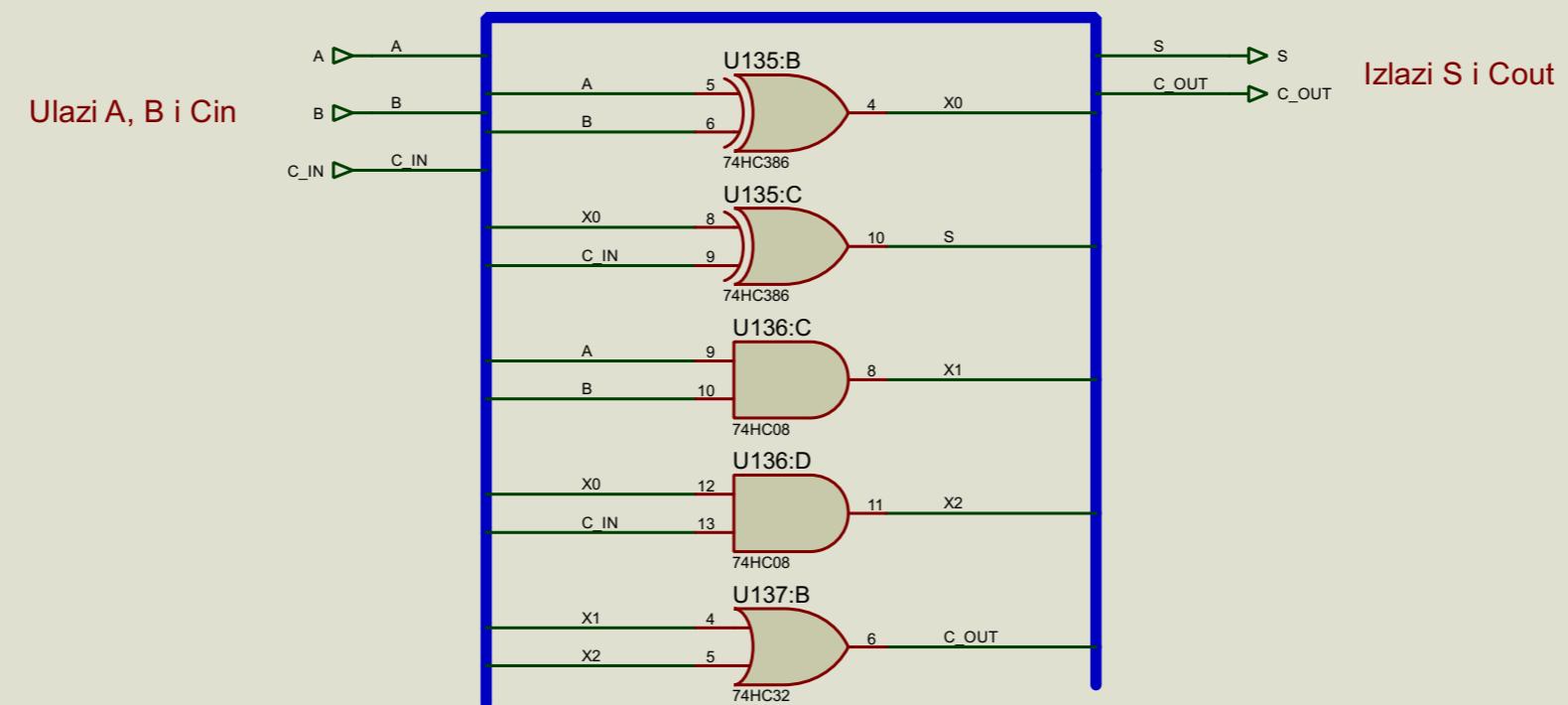
1-bit FA sa carry



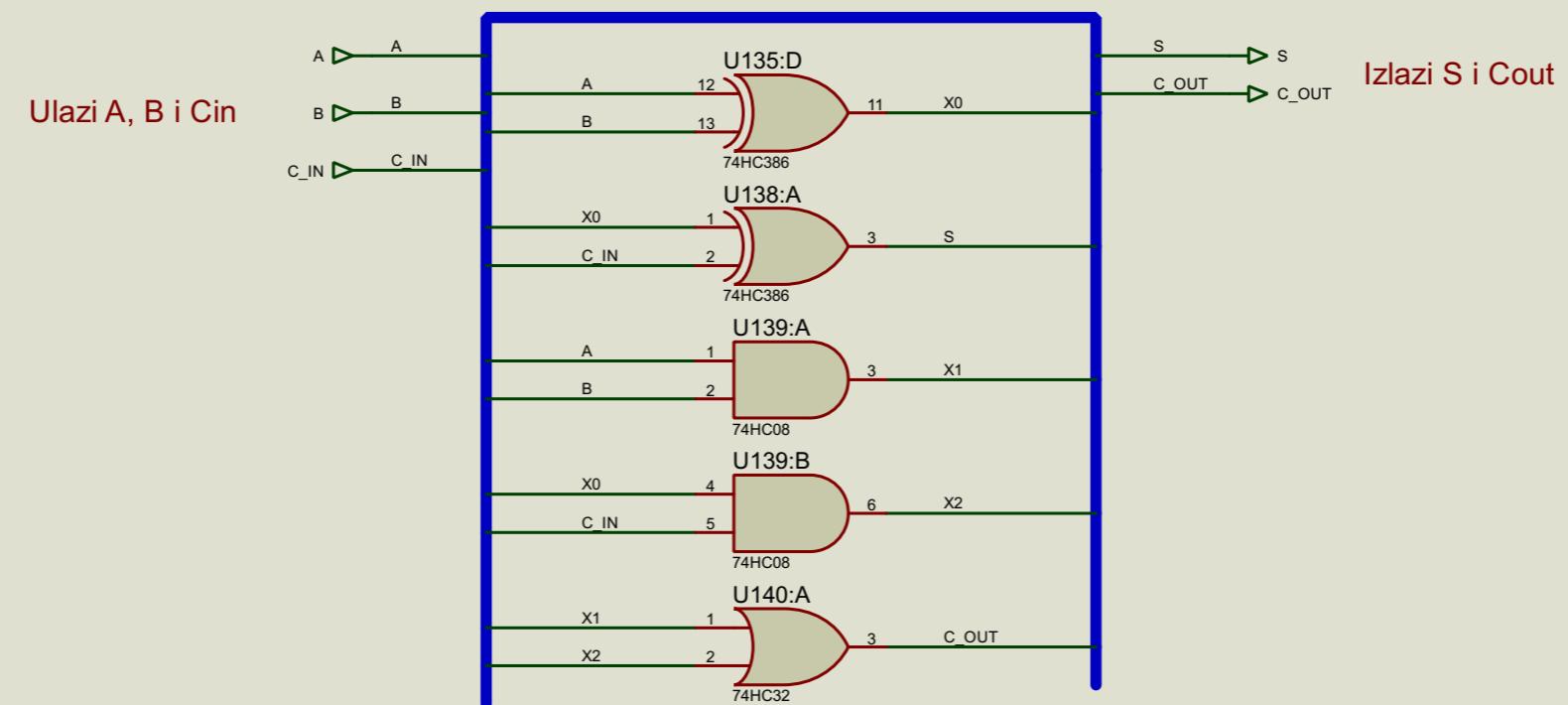
1-bit FA sa carry



1-bit FA sa carry

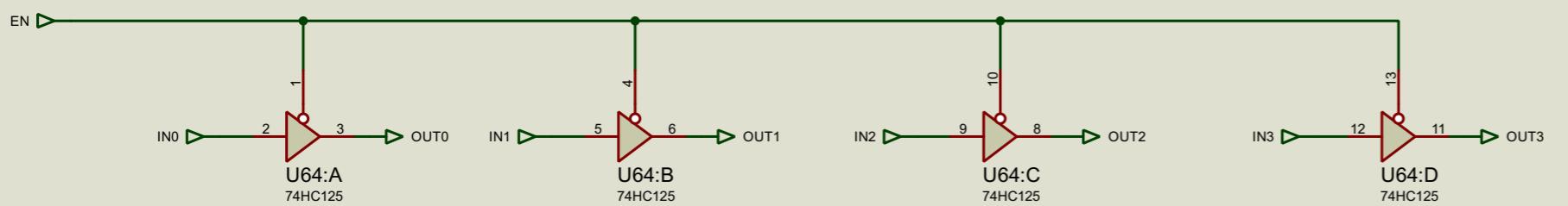


1-bit FA sa carry



IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

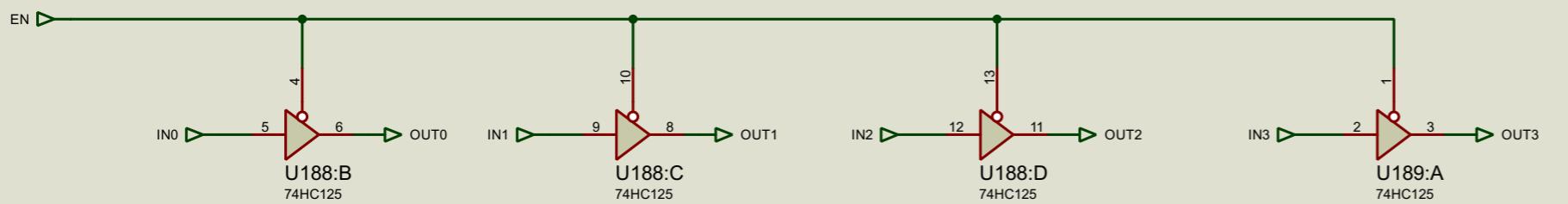
4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

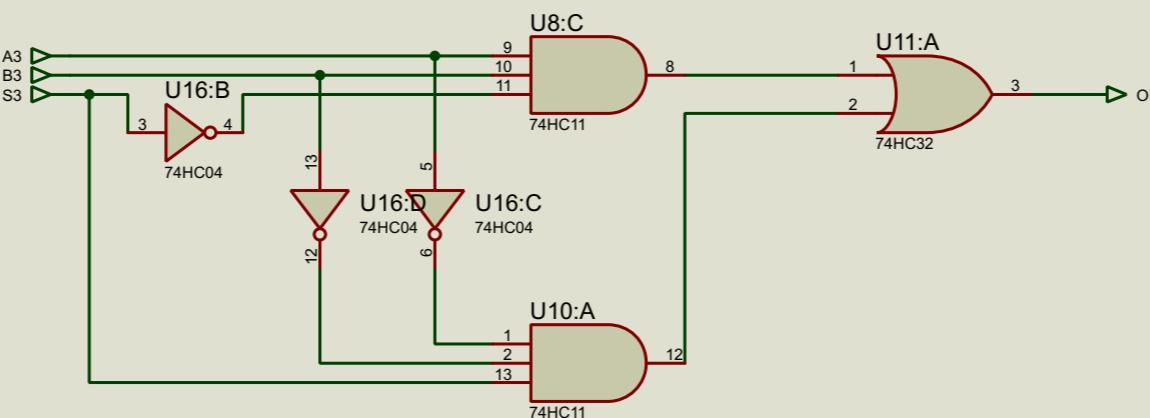
IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	1	0	0	1	1	1	0
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

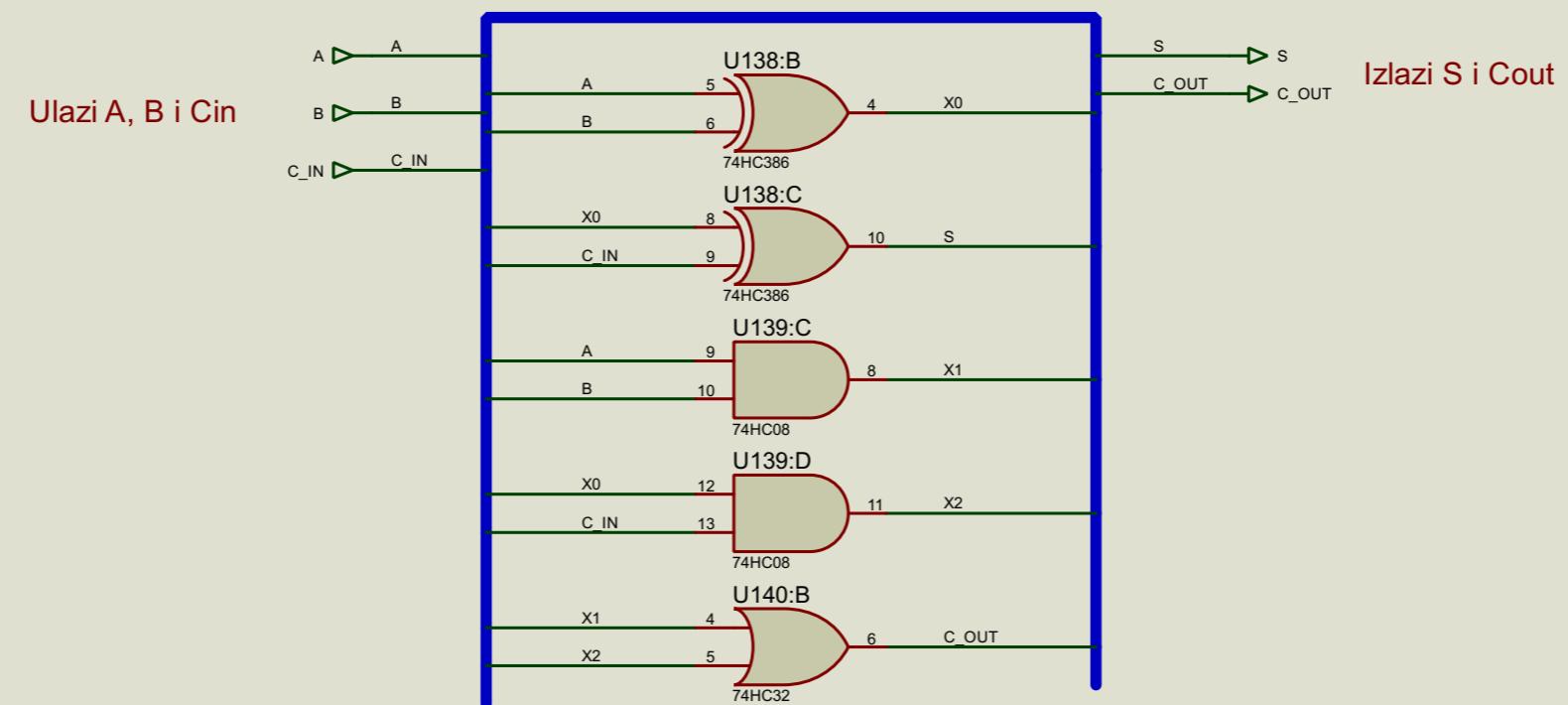
OVERFLOW CHECK



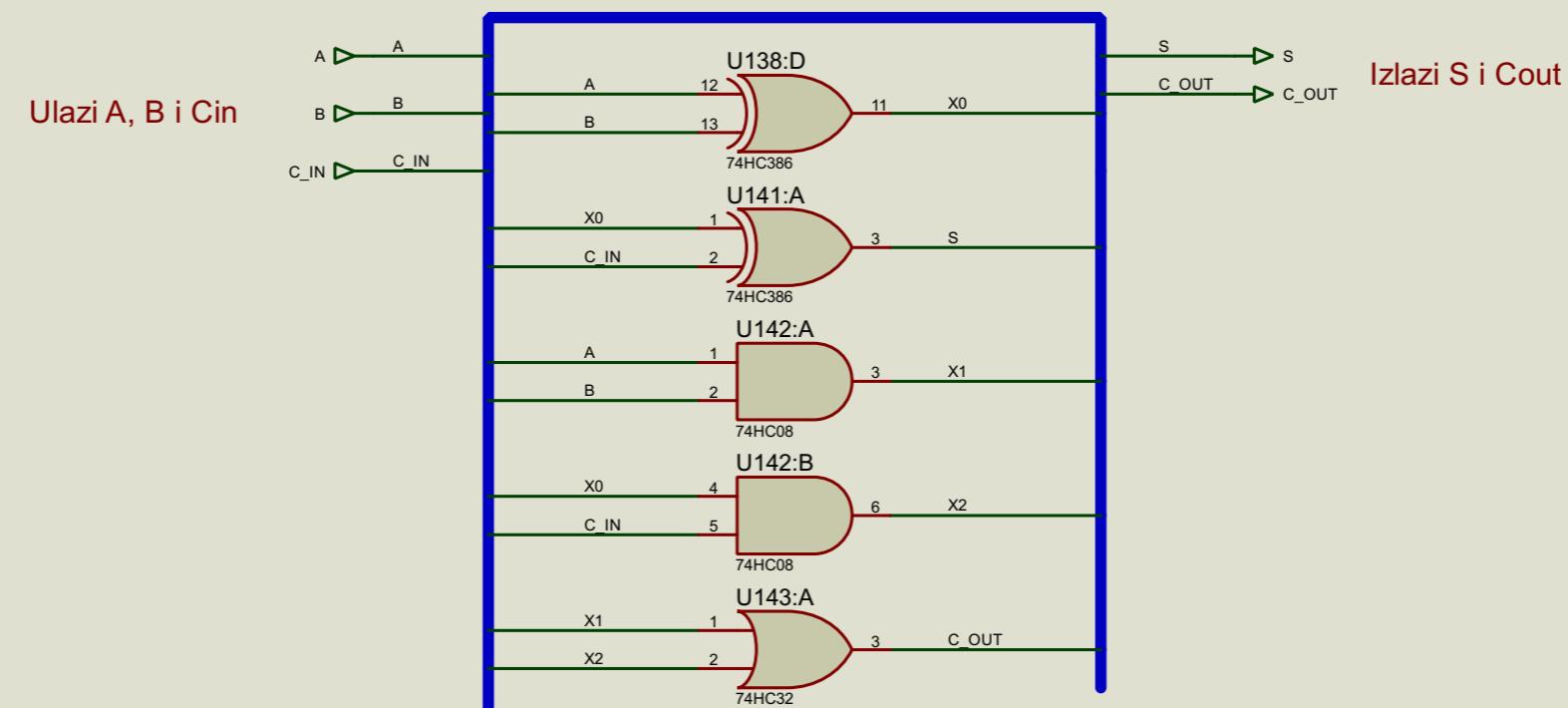
Proveravamo MSB-ove ulaza i rezultata
Ako znak rezultata nije isti kao znak input-a OF=1

A3	B3	S3	OF
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

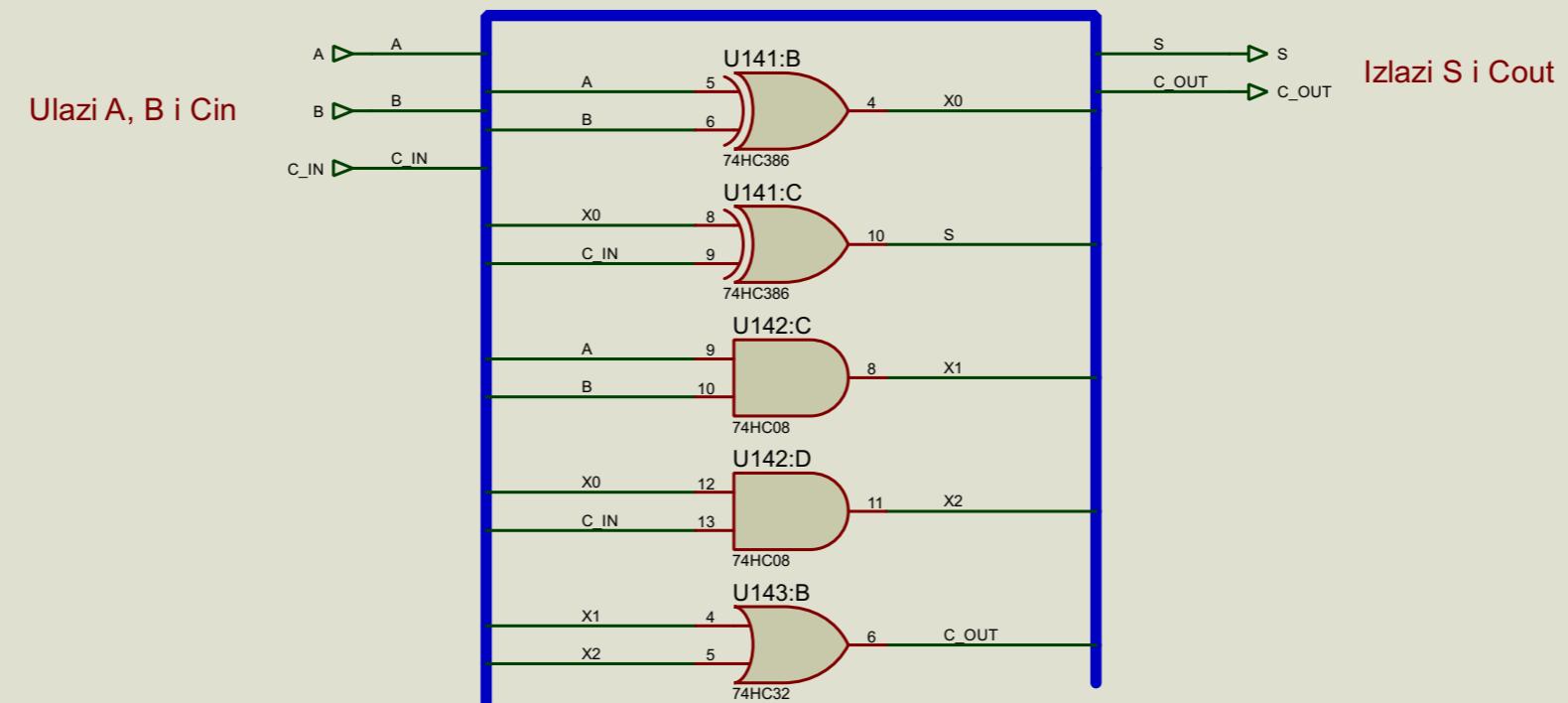
1-bit FA sa carry



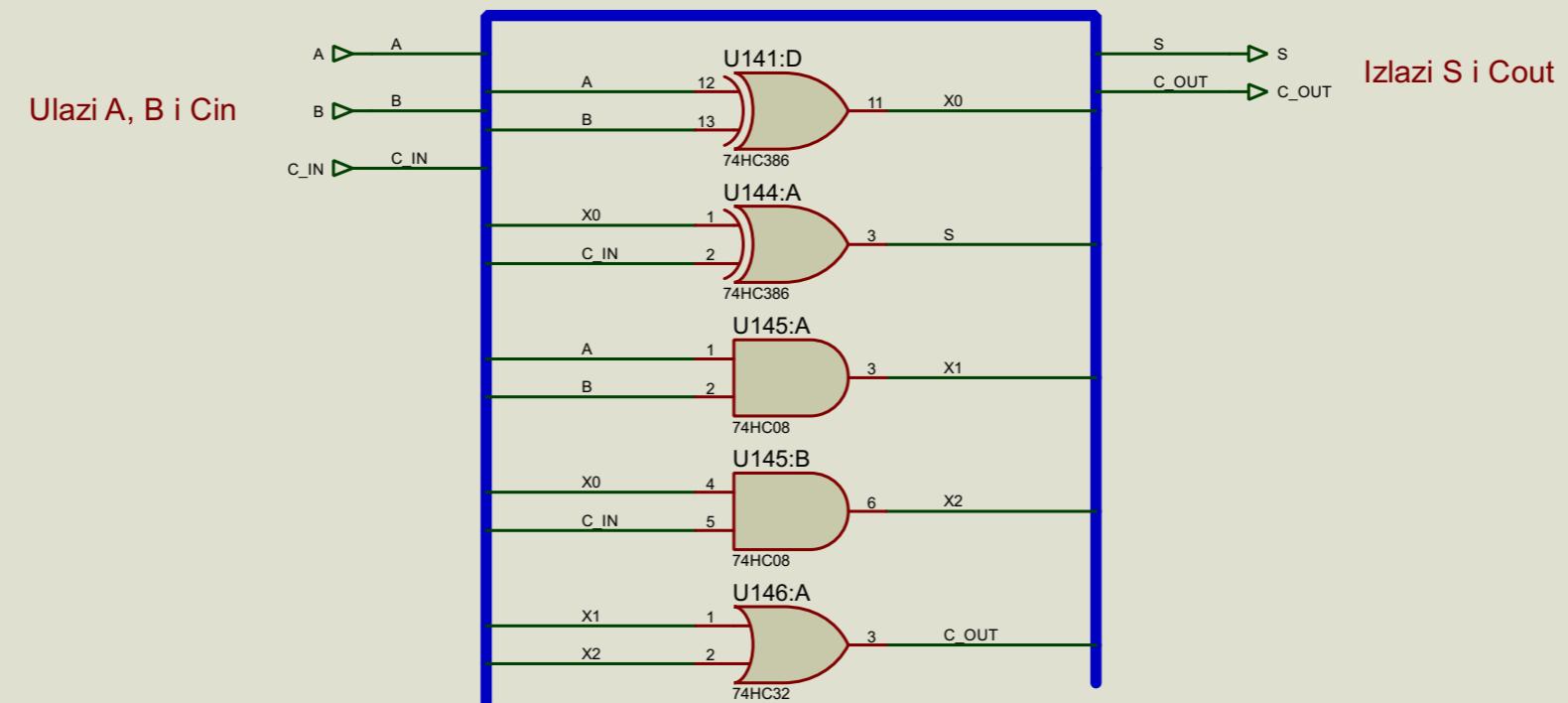
1-bit FA sa carry



1-bit FA sa carry

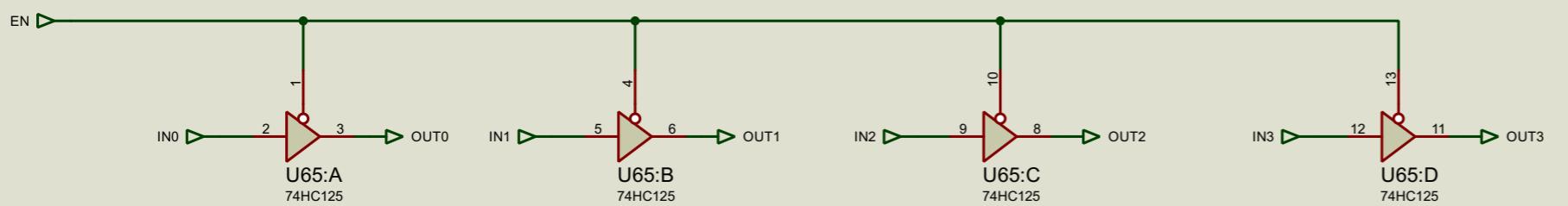


1-bit FA sa carry



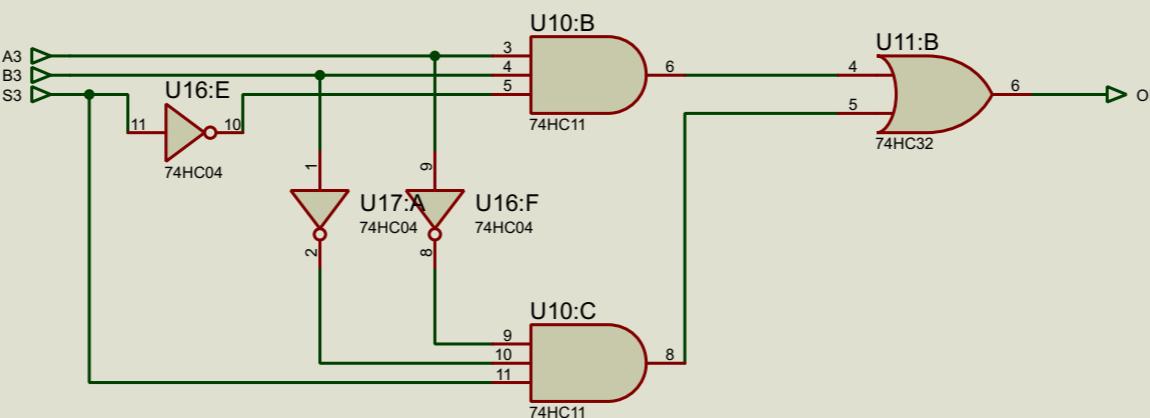
IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

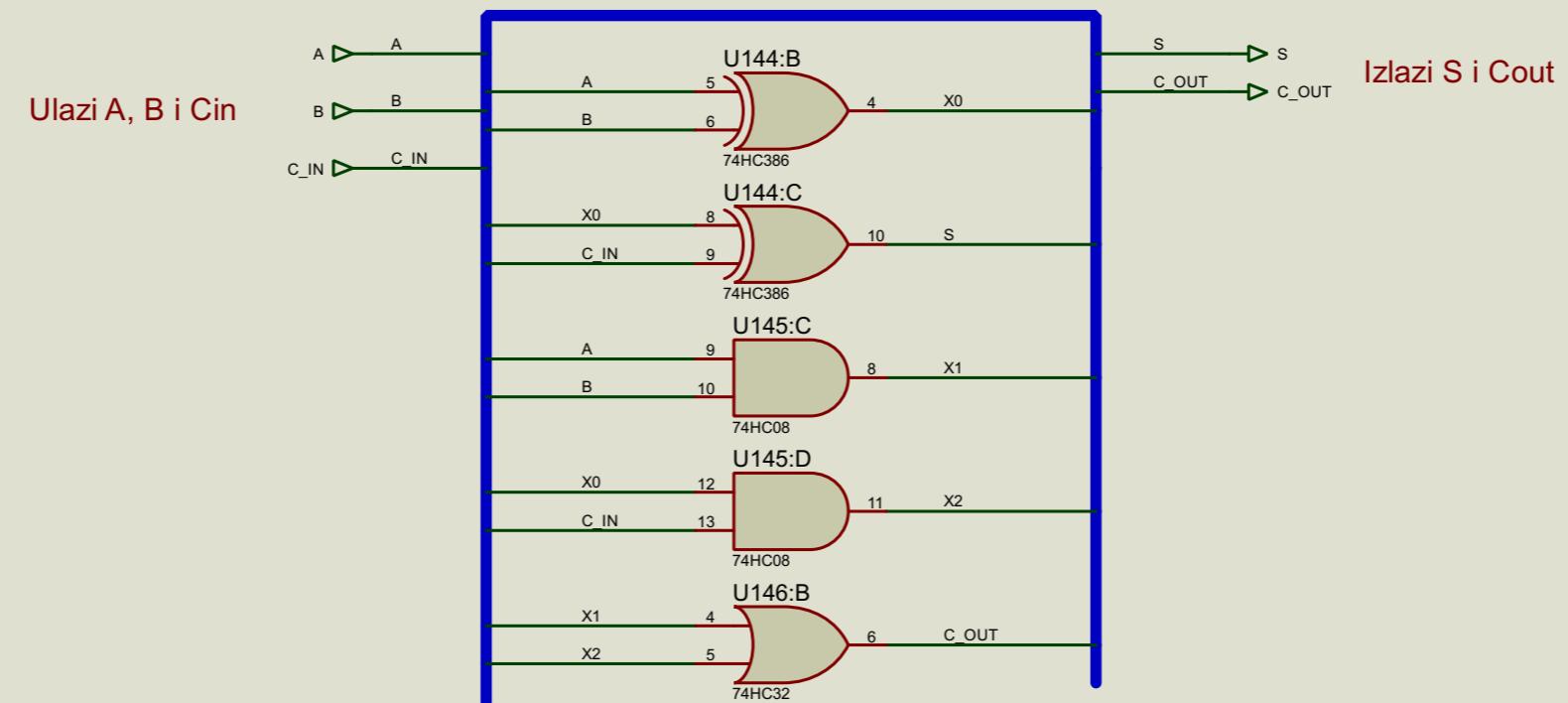
OVERFLOW CHECK



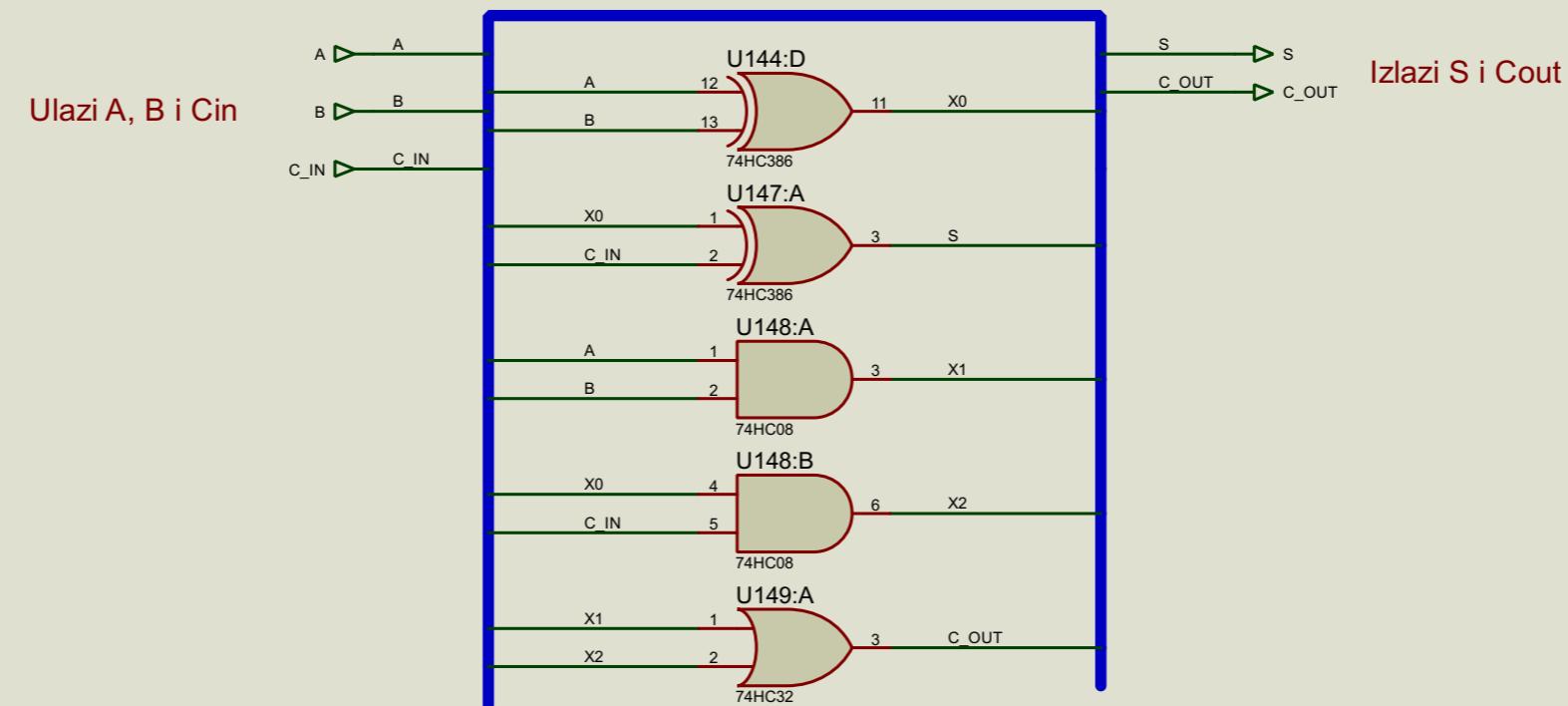
Proveravamo MSB-ove ulaza i rezultata
Ako znak rezultata nije isti kao znak input-a OF=1

A3	B3	S3	OF
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

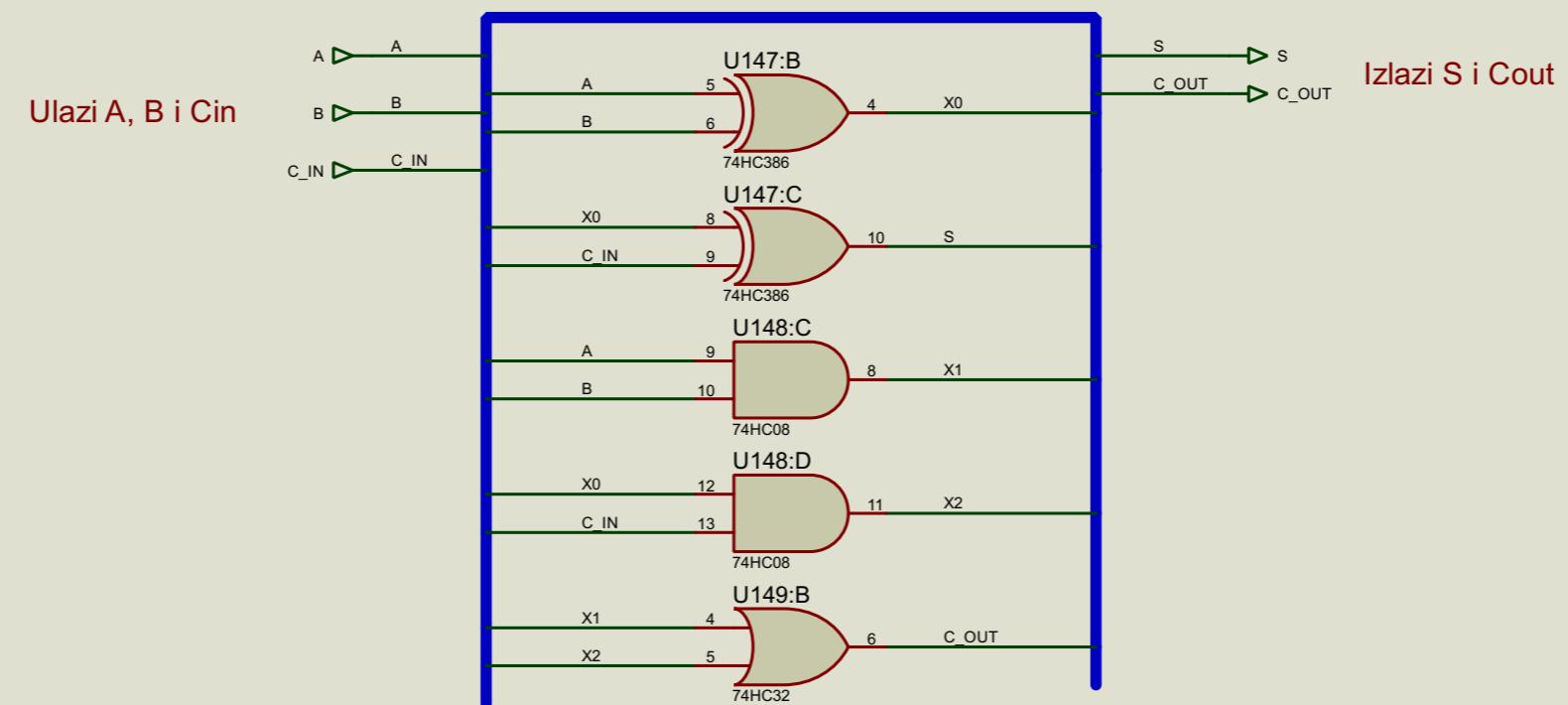
1-bit FA sa carry



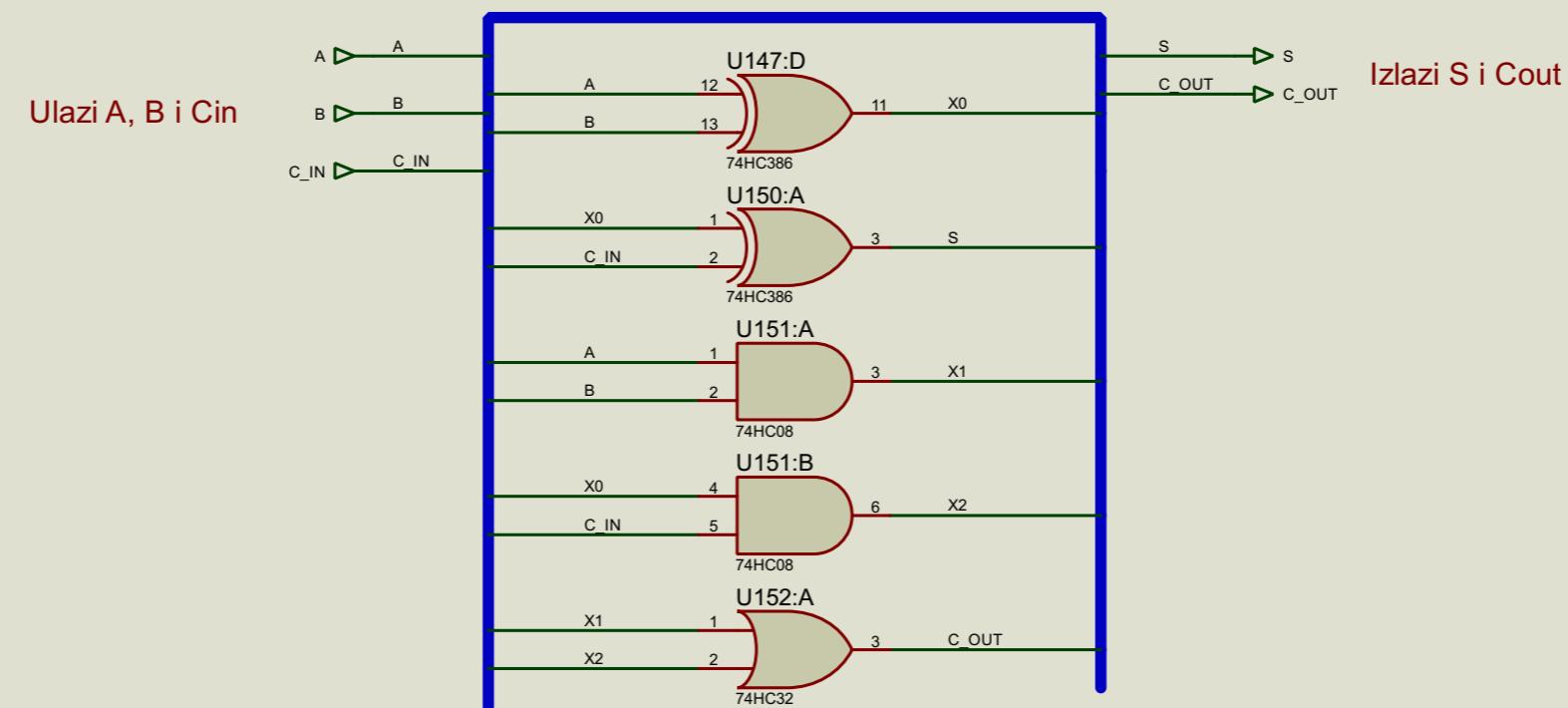
1-bit FA sa carry



1-bit FA sa carry

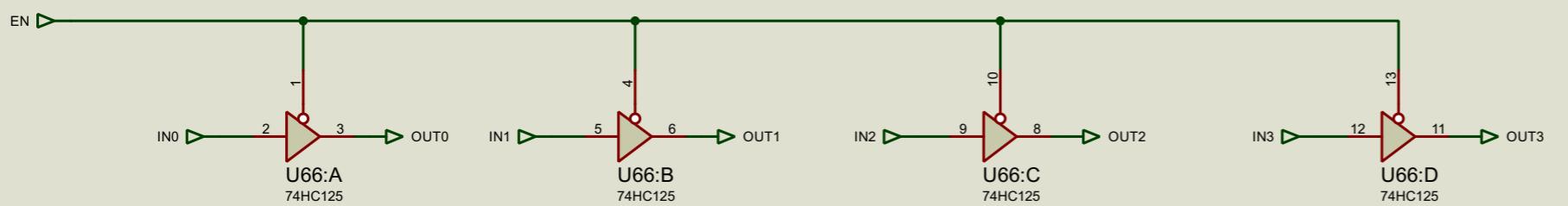


1-bit FA sa carry



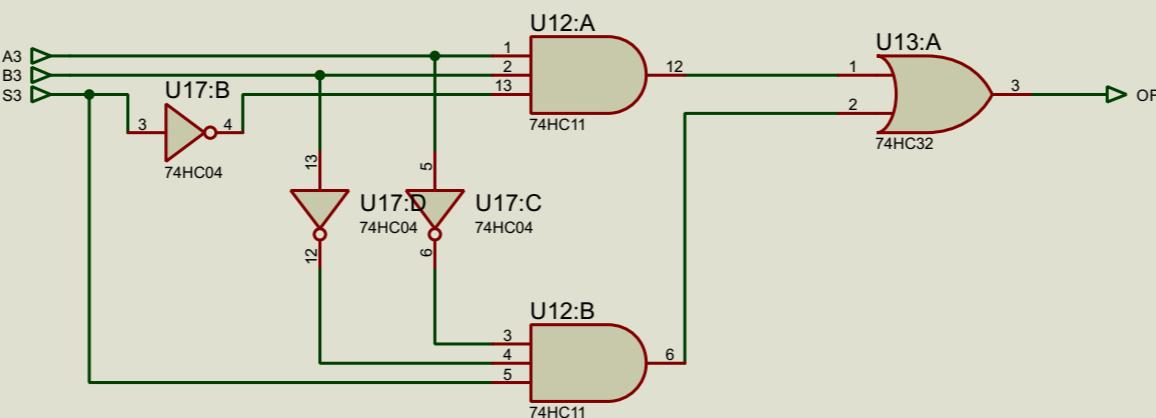
IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	0	1	0	0	1	1	0
1	1	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

OVERFLOW CHECK

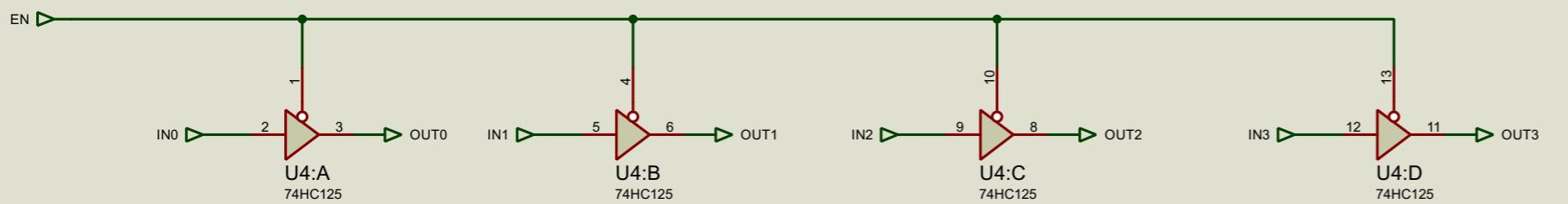


Proveravamo MSB-ove ulaza i rezultata
Ako znak rezultata nije isti kao znak input-a OF=1

A3	B3	S3	OF
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

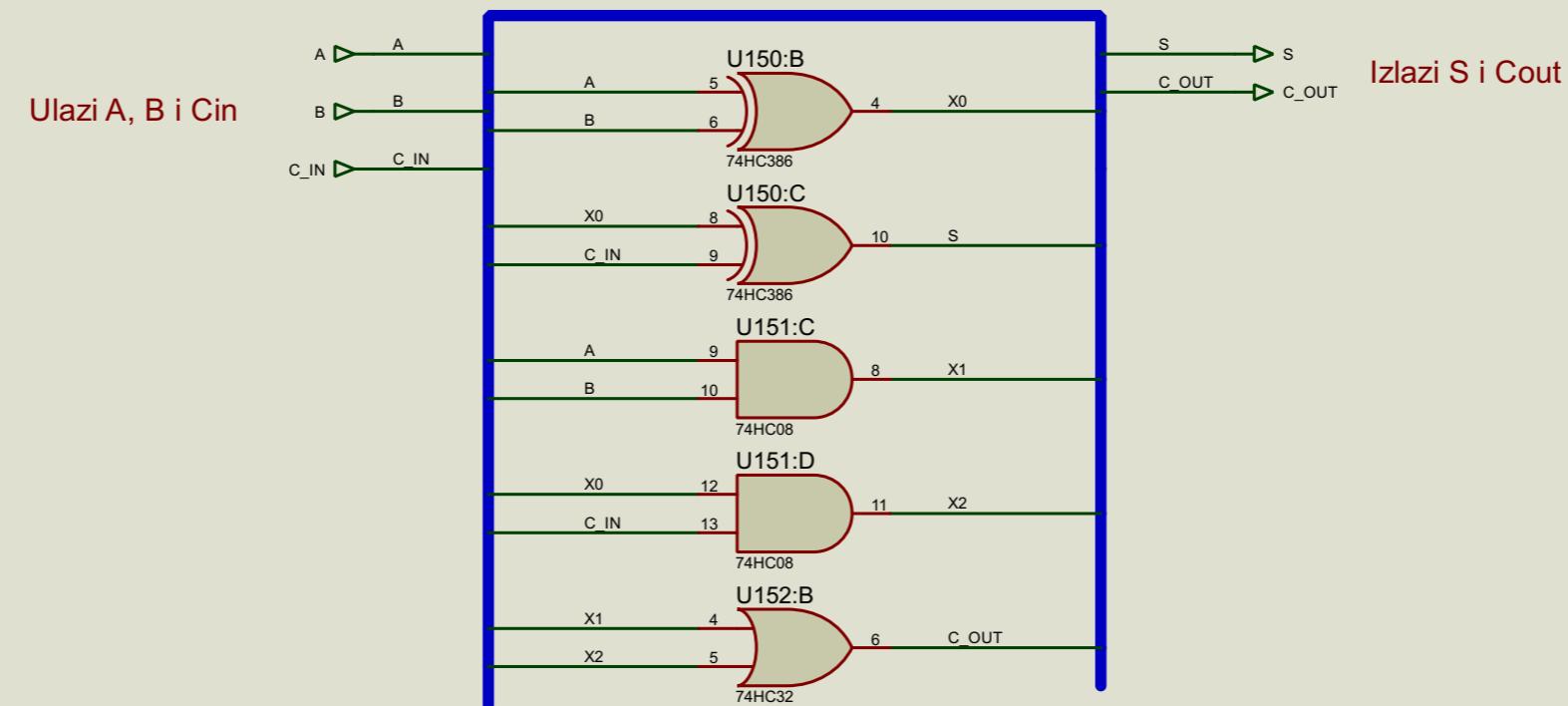
IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	1	0	0	1	1	1	0
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)

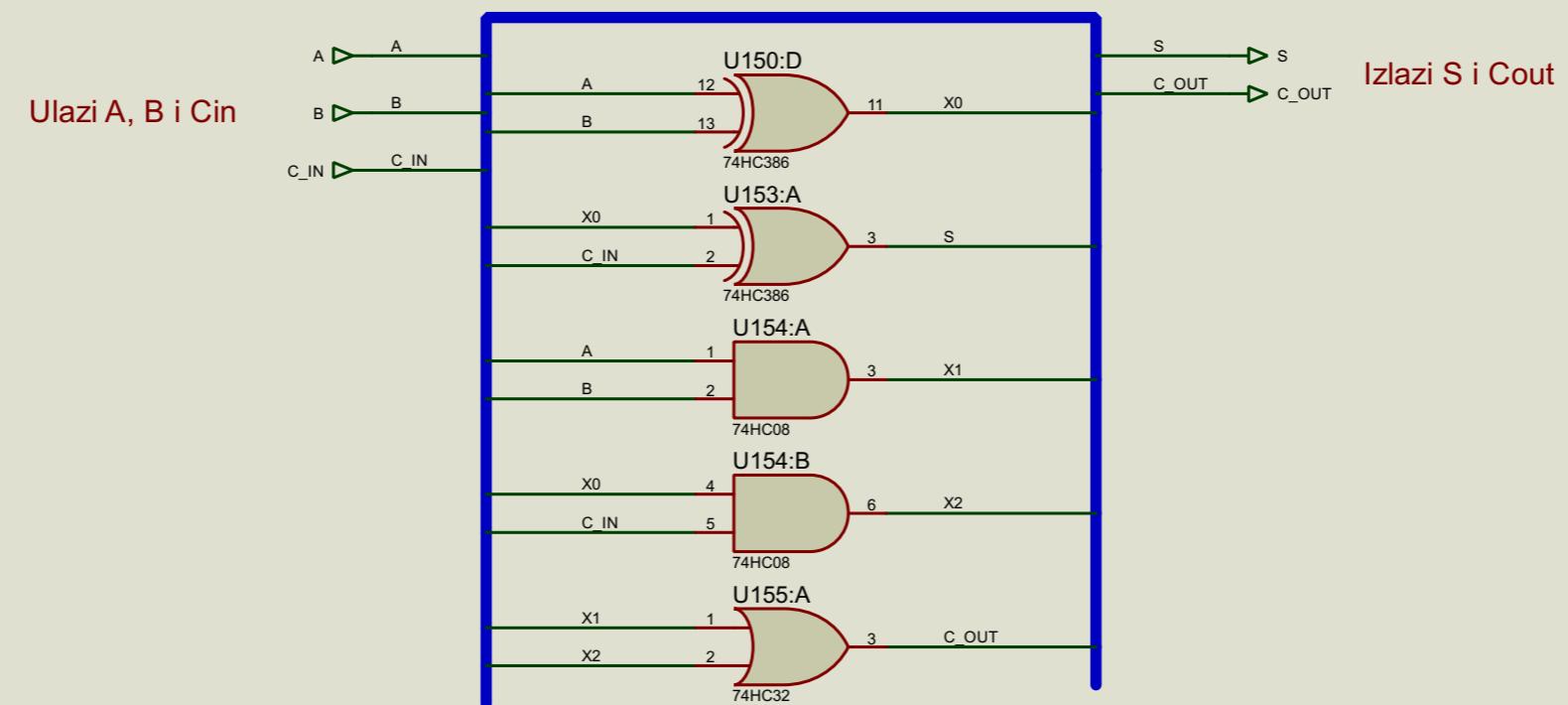


Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1

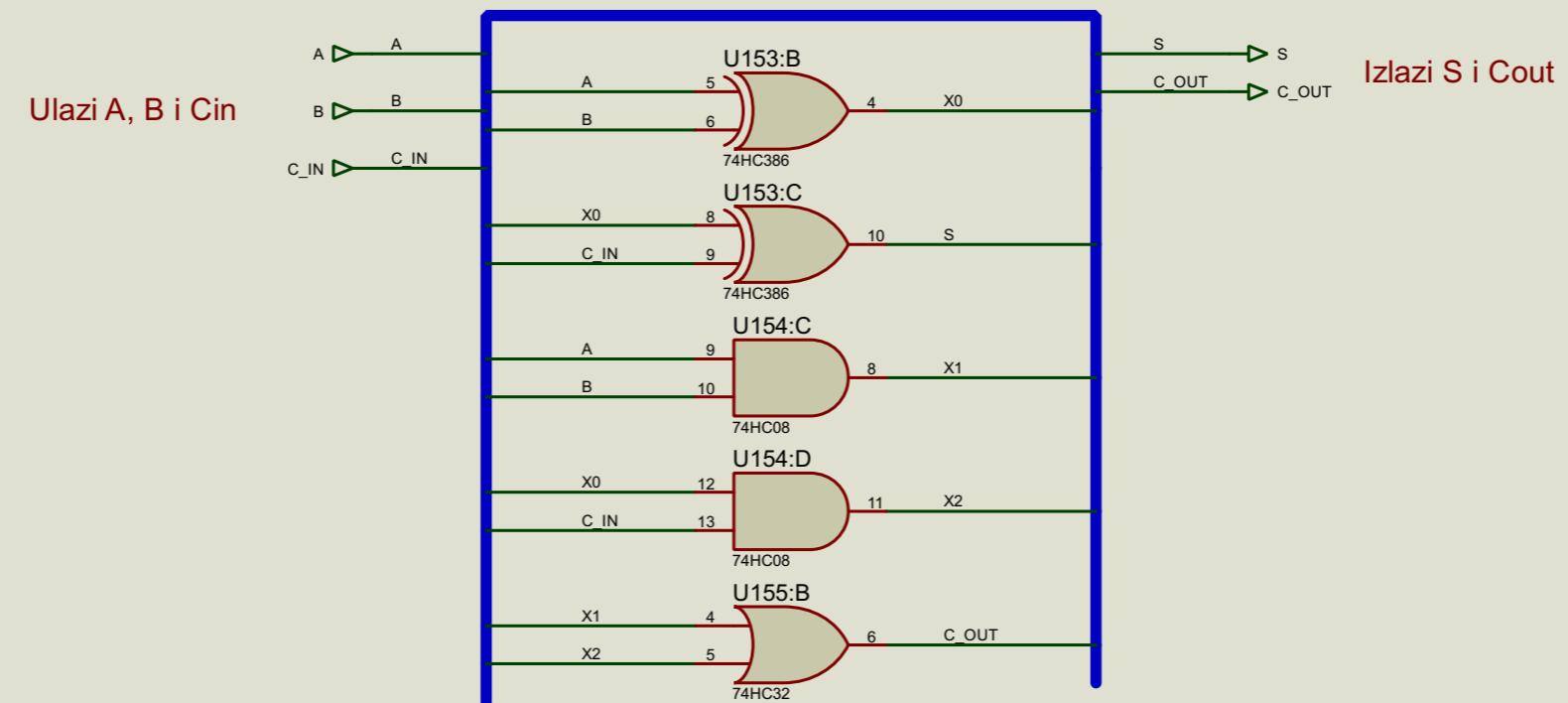
1-bit FA sa carry



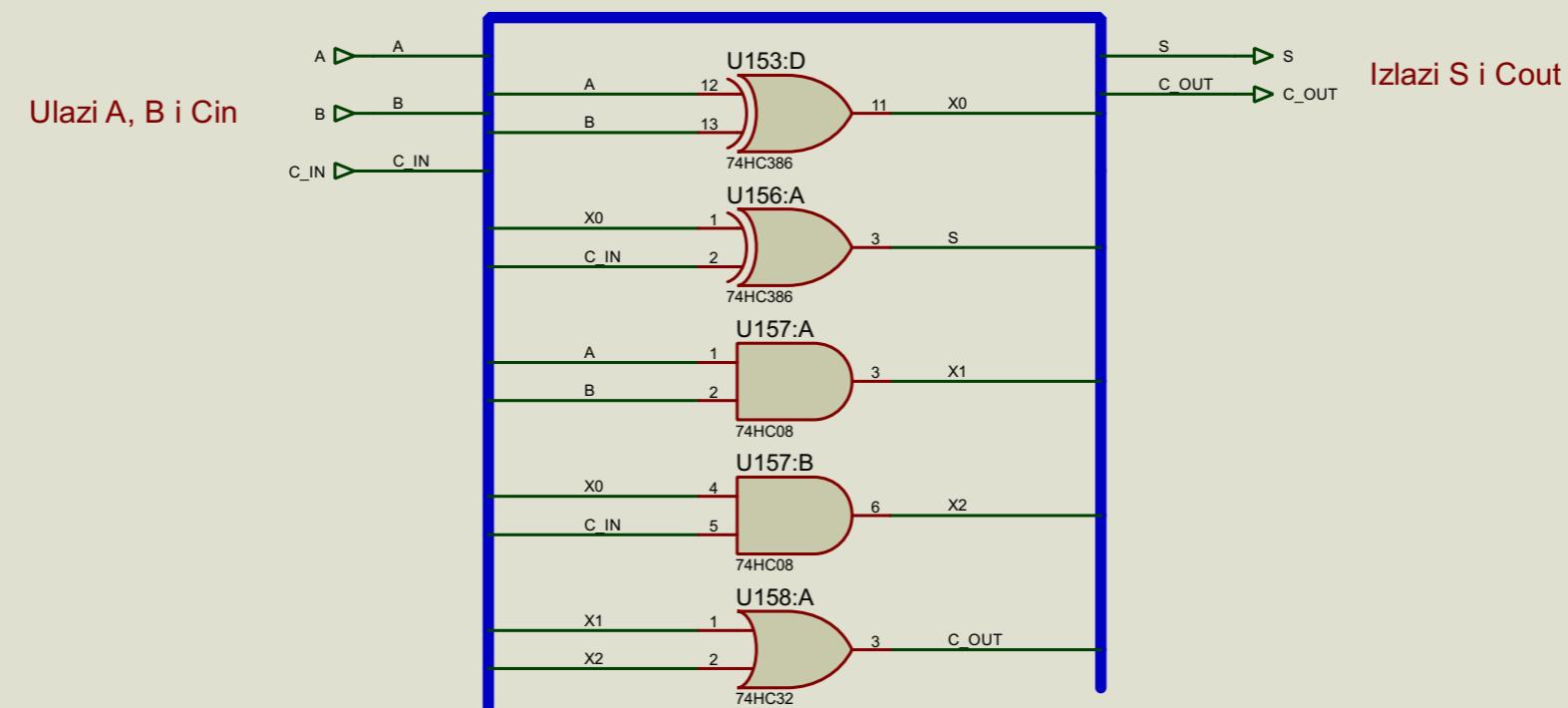
1-bit FA sa carry



1-bit FA sa carry

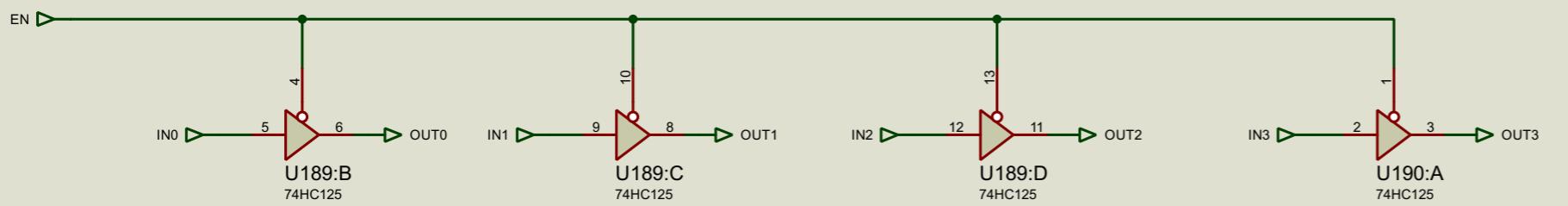


1-bit FA sa carry



IN3	IN2	IN1	IN0	EN	OUT3	OUT2	OUT1	OUT0
X	X	X	X	1	hi-Z	hi-Z	hi-Z	hi-Z
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	0	1	1
1	0	0	0	0	0	1	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	0	1	0	1	0
1	0	1	1	0	1	0	1	1
1	1	0	0	1	0	1	1	0
1	1	1	0	0	1	1	1	0
1	1	1	1	0	1	1	1	1

4 bit tri-state buffer sa negativnom logikom (EN=0-enabled EN=1-disabled)



Uzima 4-bitni ulaz i izbacuje ga nepromjenjenog ako je EN=0 tj. u stanju visoke impedanse ako je EN=1