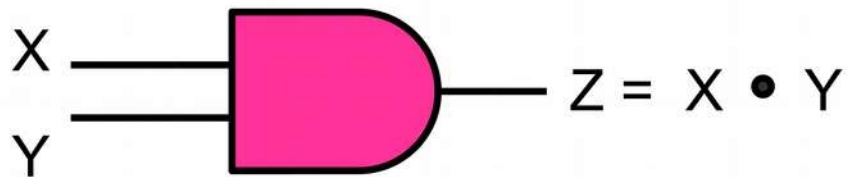


LOGICKA KOLA

Operacija logicko mnozenje

I - OPERACIJA (AND)



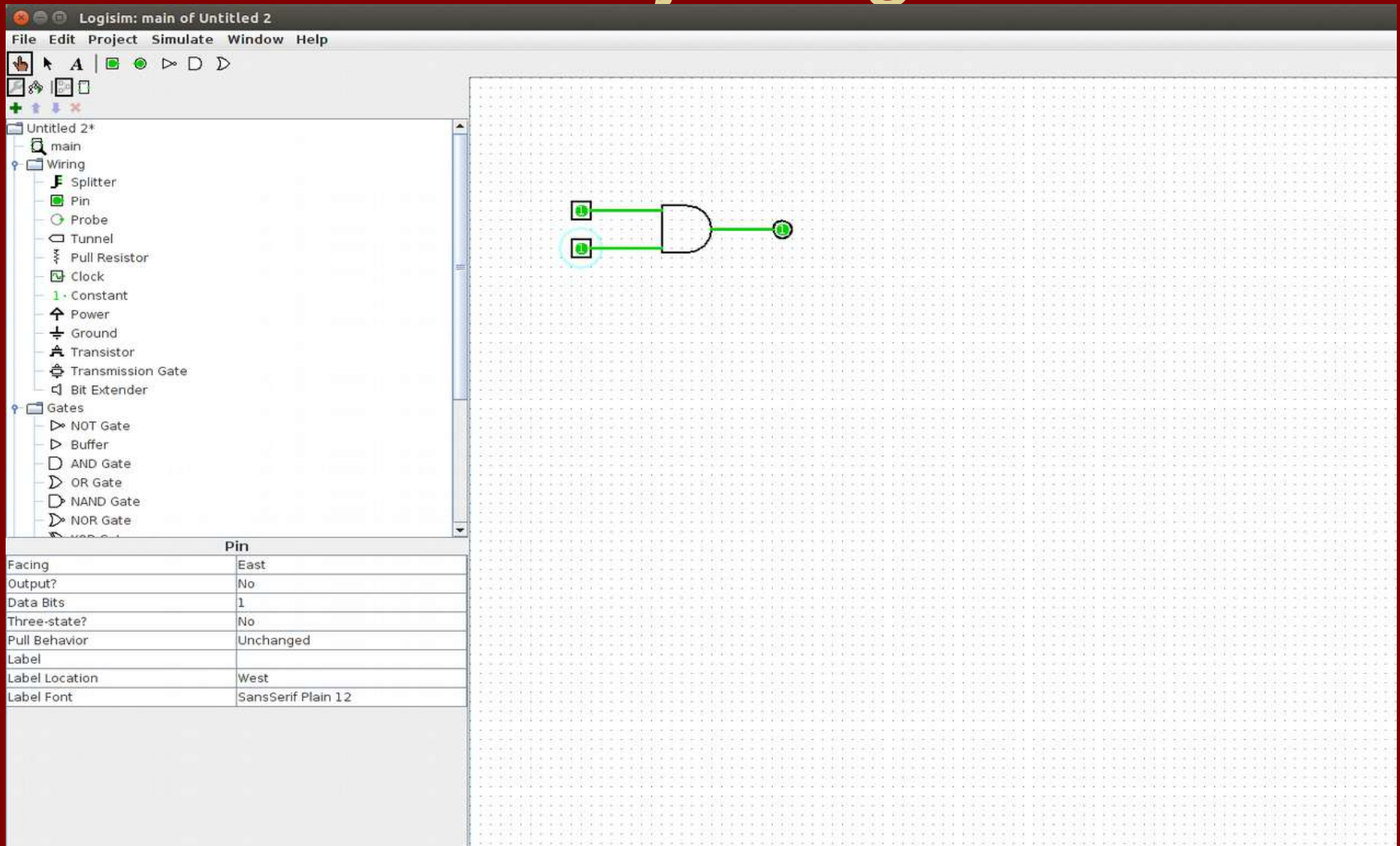
X	Y	$Z = X \cdot Y$
0	0	0
0	1	0
1	0	0
1	1	1

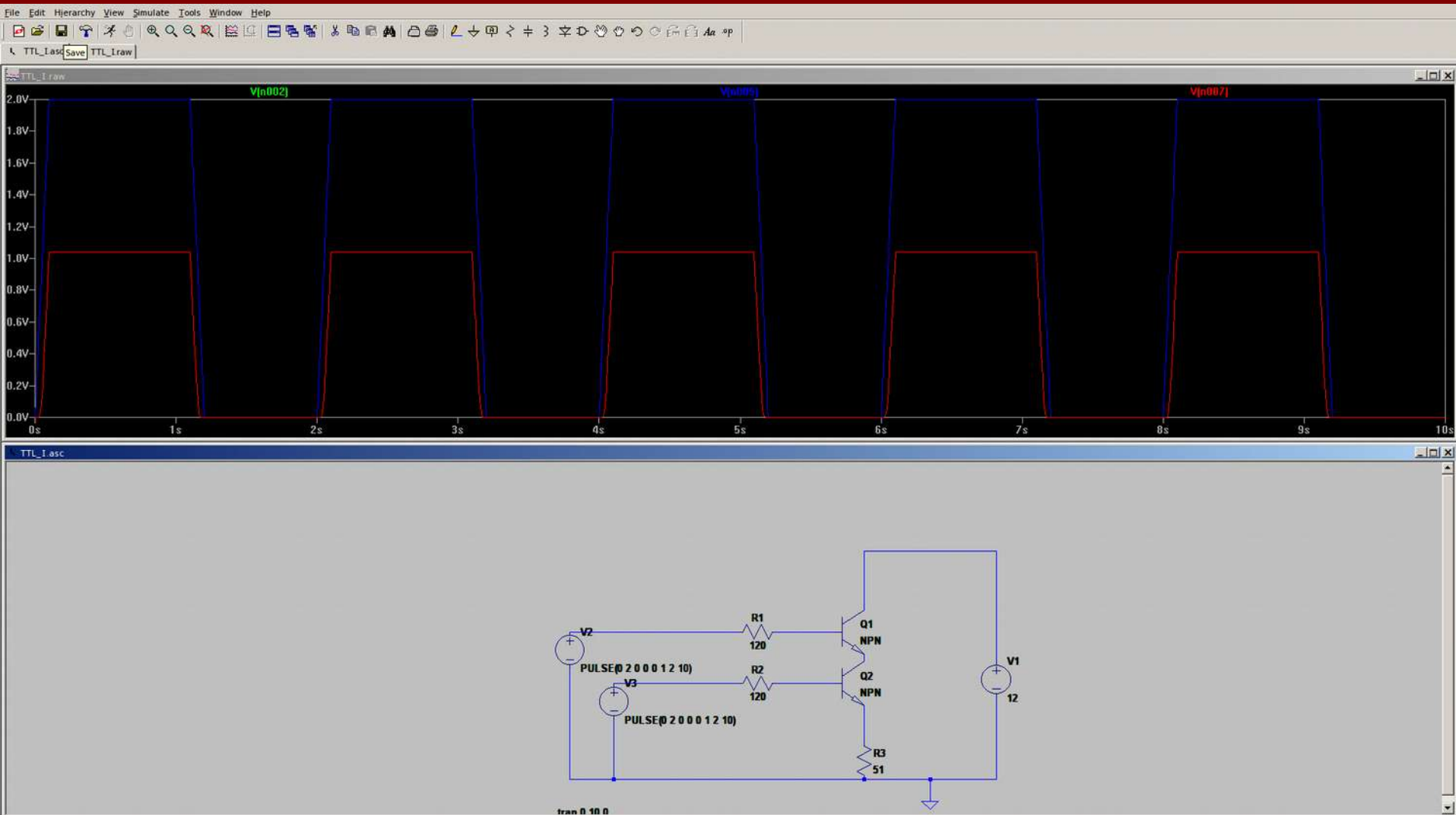
Graficki simbol i kombinaciona tabela za I operaciju

Simulacija LogiSim

```
aleksandar@aleksandar: ~/LogiSim
aleksandar@aleksandar:~$ cd LogiSim/
aleksandar@aleksandar:~/LogiSim$ ls
logisim-generic-2.7.1.jar  ORT_1_primeri.circ
aleksandar@aleksandar:~/LogiSim$ sudo java jar logisim-generic-2.7.1.jar
[sudo] password for aleksandar:
Error: Could not find or load main class jar
aleksandar@aleksandar:~/LogiSim$ sudo java -jar logisim-generic-2.7.1.jar
aleksandar@aleksandar:~/LogiSim$ sudo java -jar logisim-generic-2.7.1.jar
```

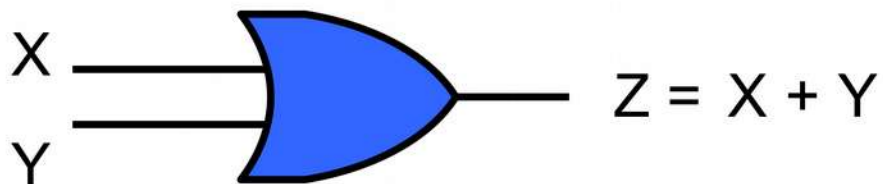
Simulacija LogiSim





Operacija logicko sabiranje

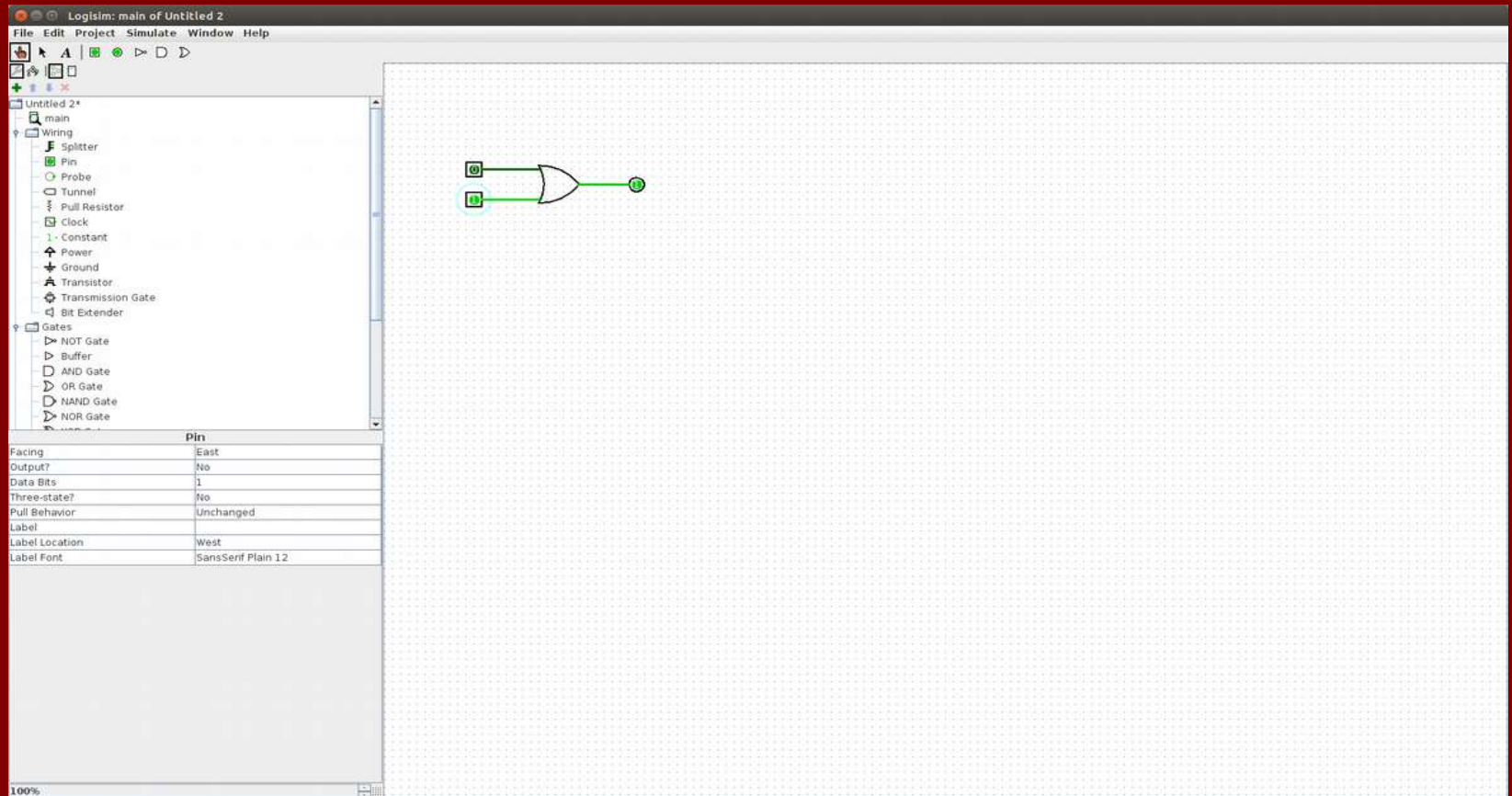
ILI - OPERACIJA (OR)

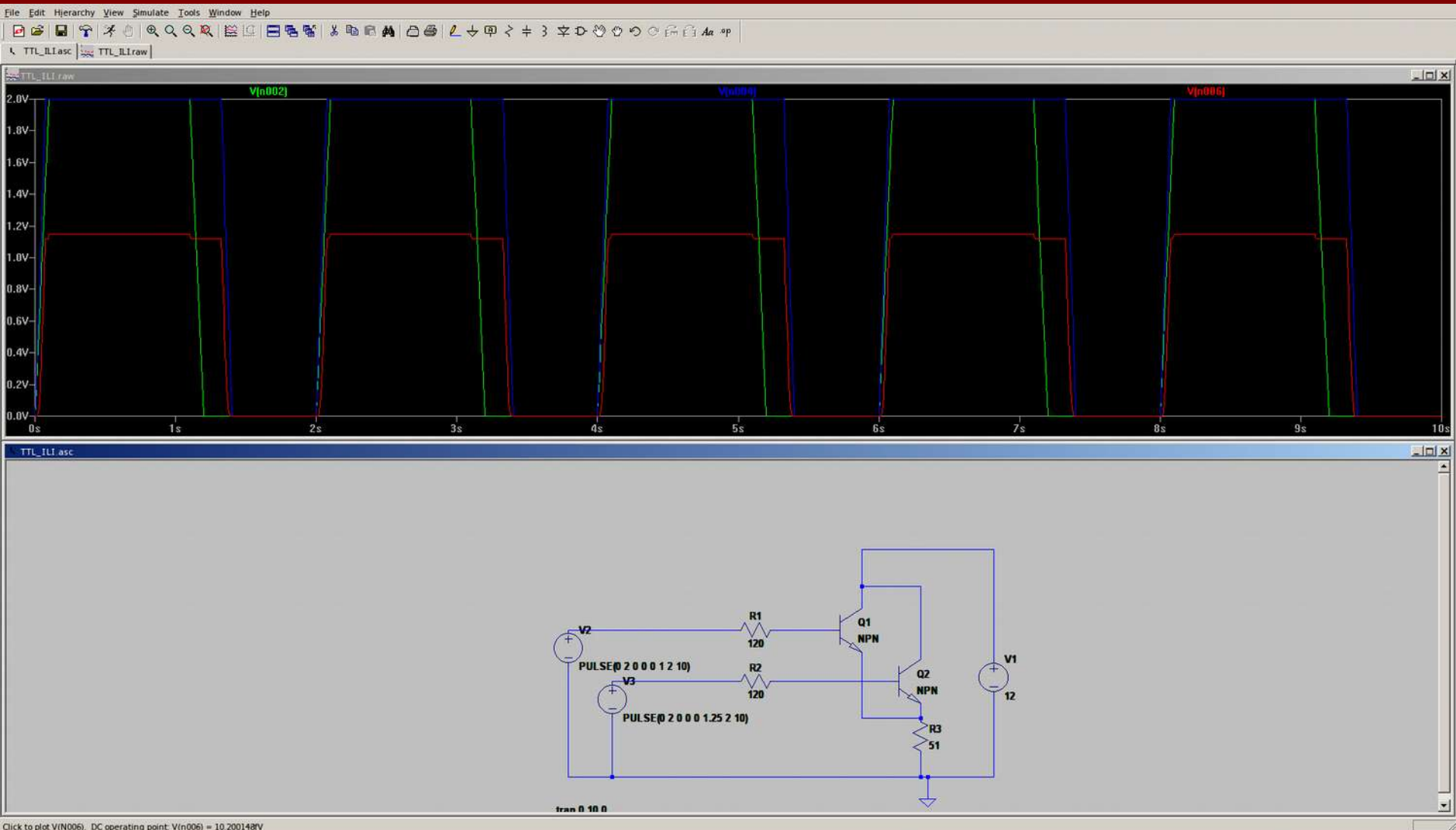


X	Y	$Z = X + Y$
0	0	0
0	1	1
1	0	1
1	1	1

Graficki simbol i kombinaciona tabela za ILI operaciju

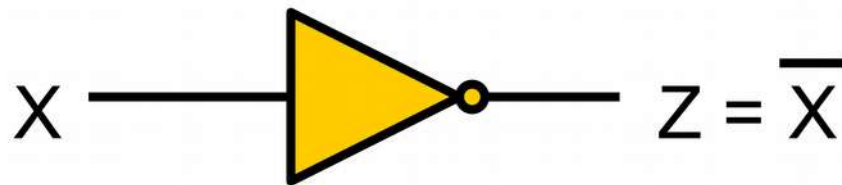
Simulacija LogiSim





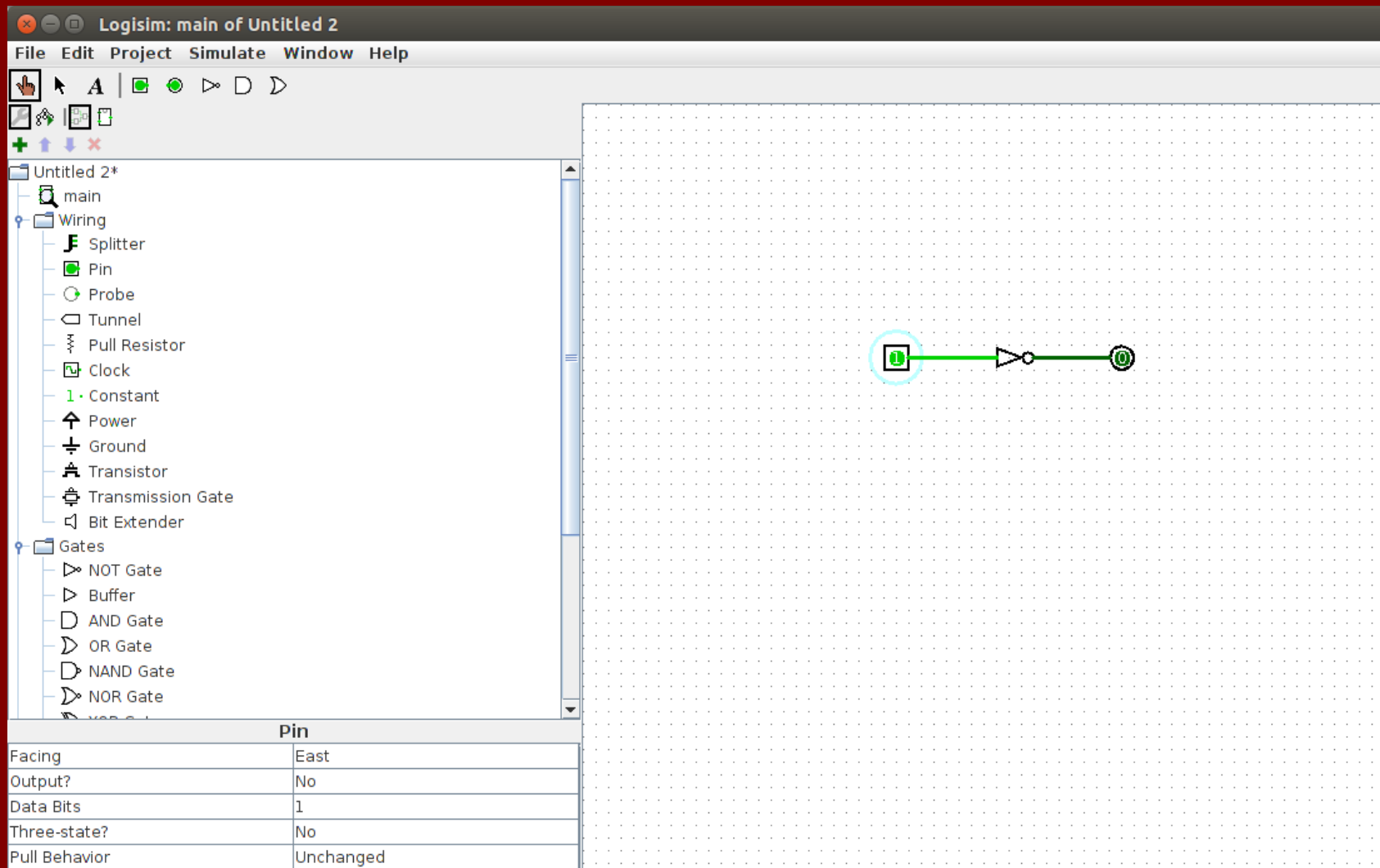
Operacija NE - komplementiranje

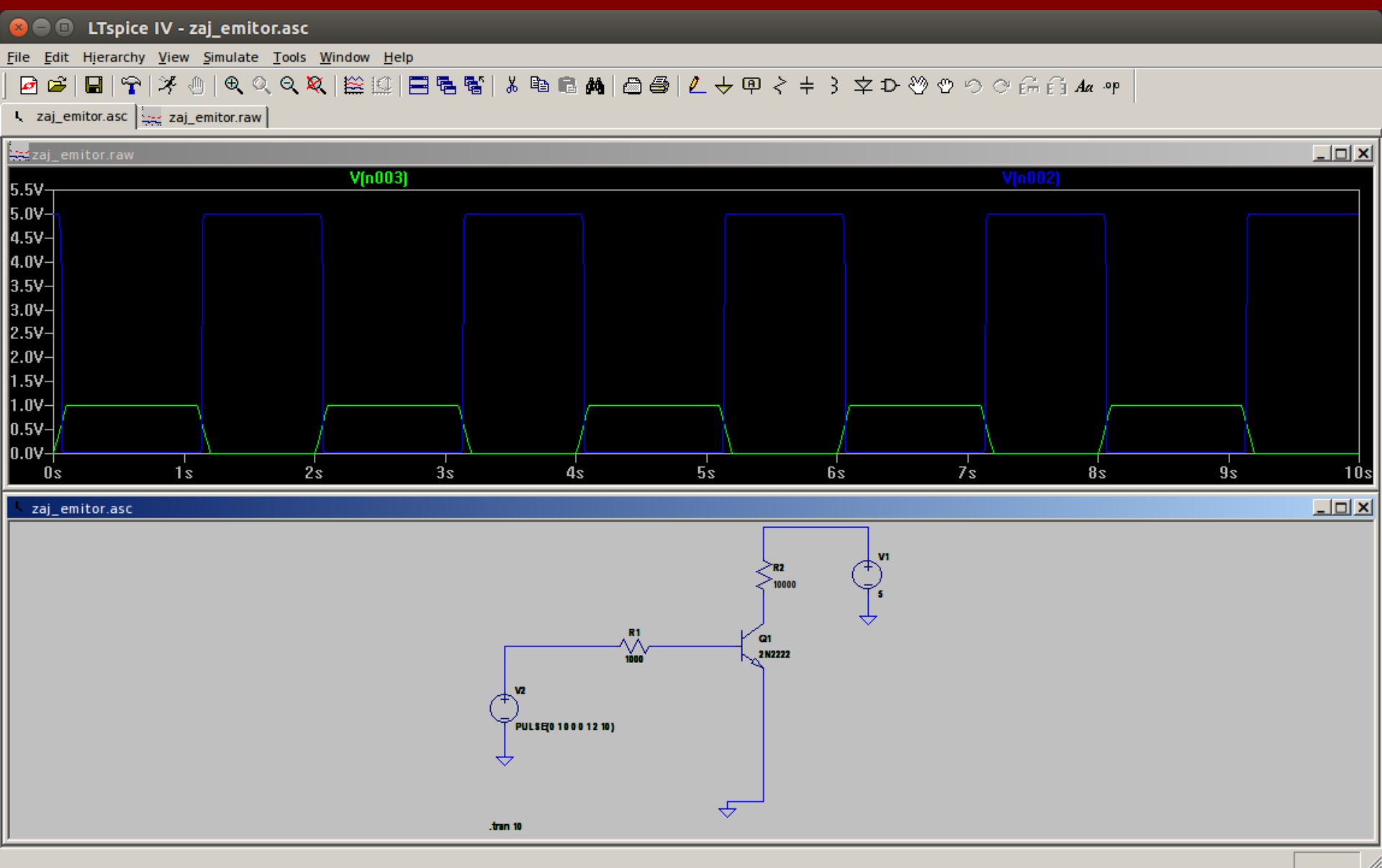
NE - OPERACIJA (NOT)

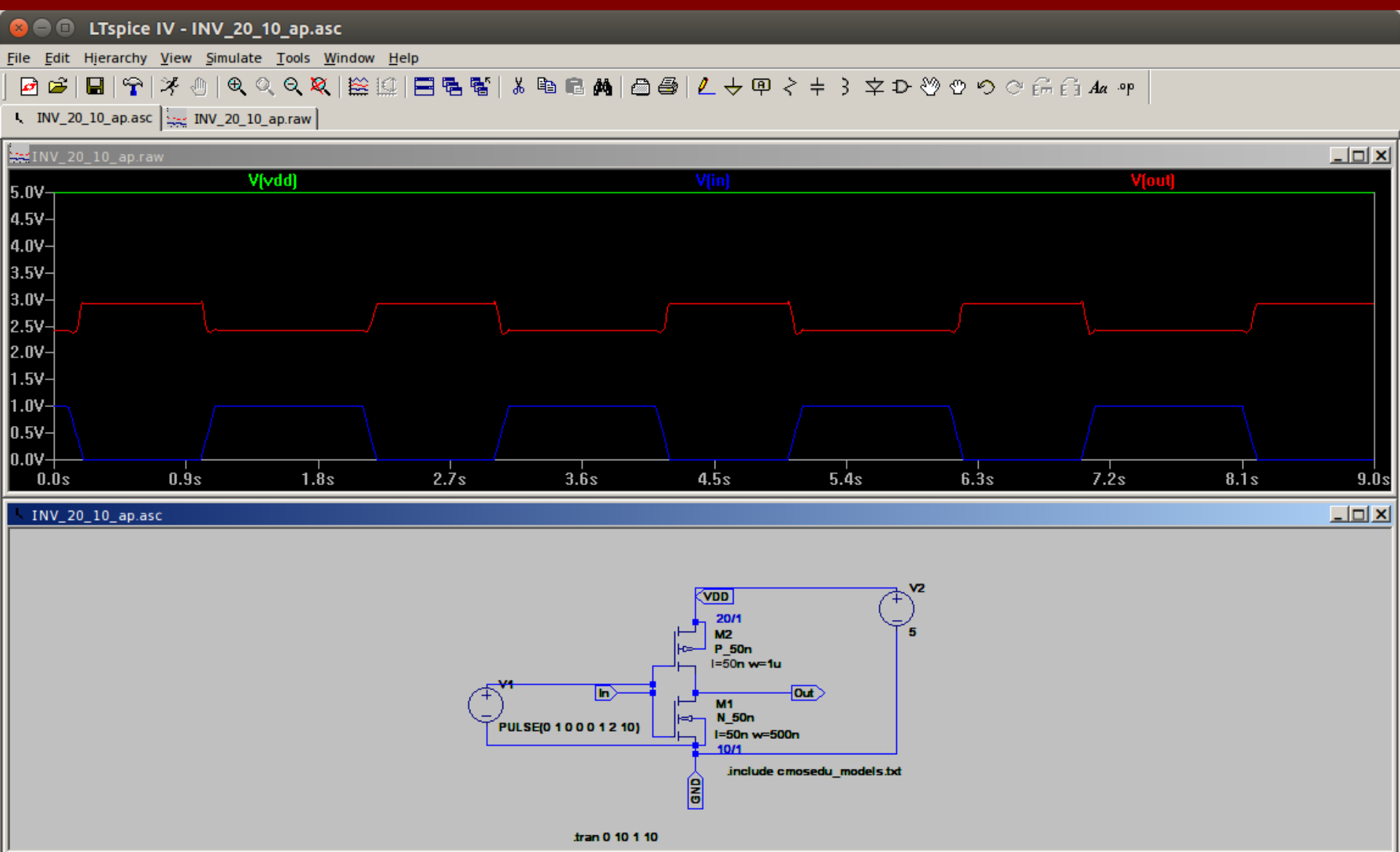


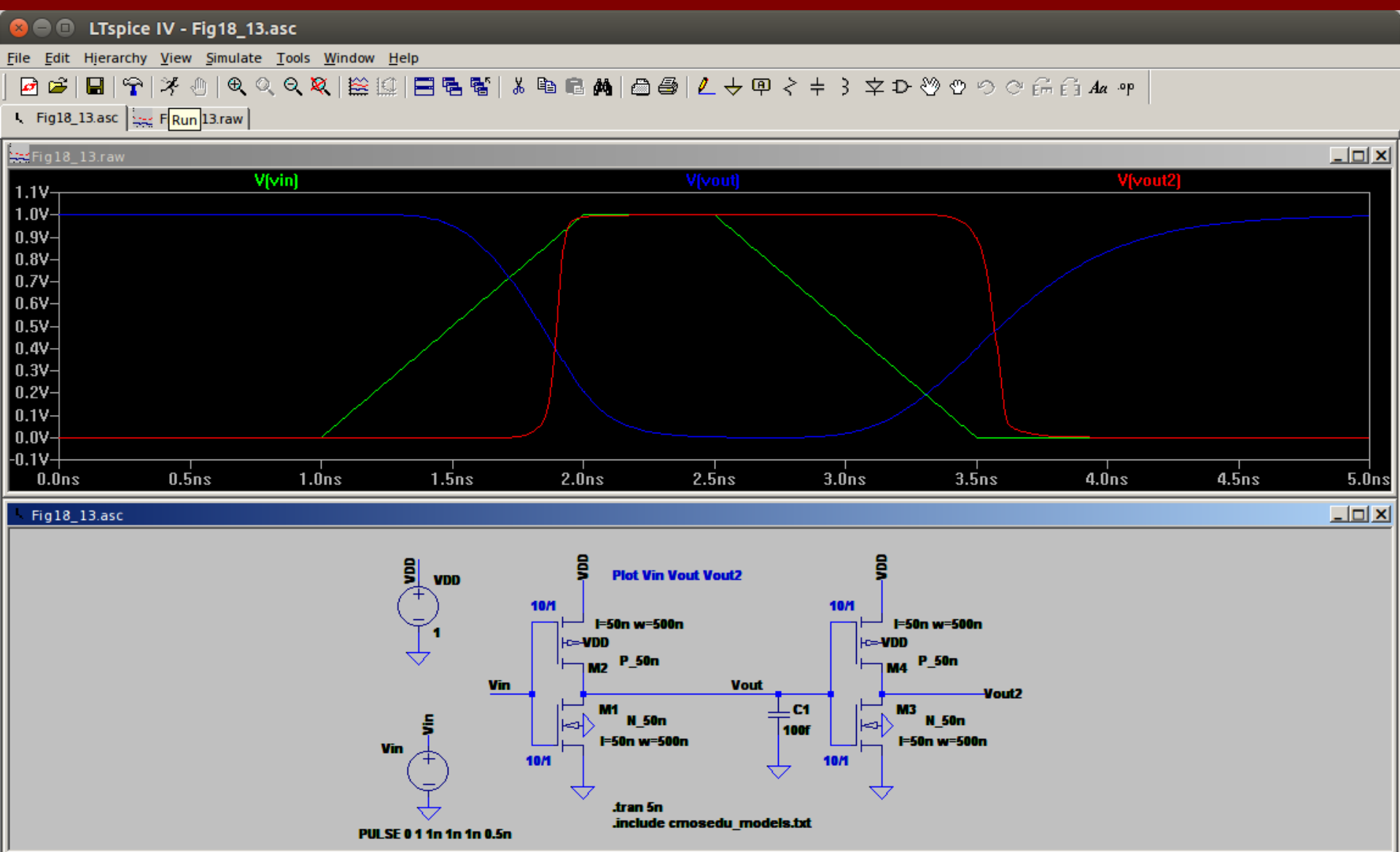
X	$Z = \overline{X}$
1	0
0	1

Graficki simbol i kombinaciona tabela za NE operaciju



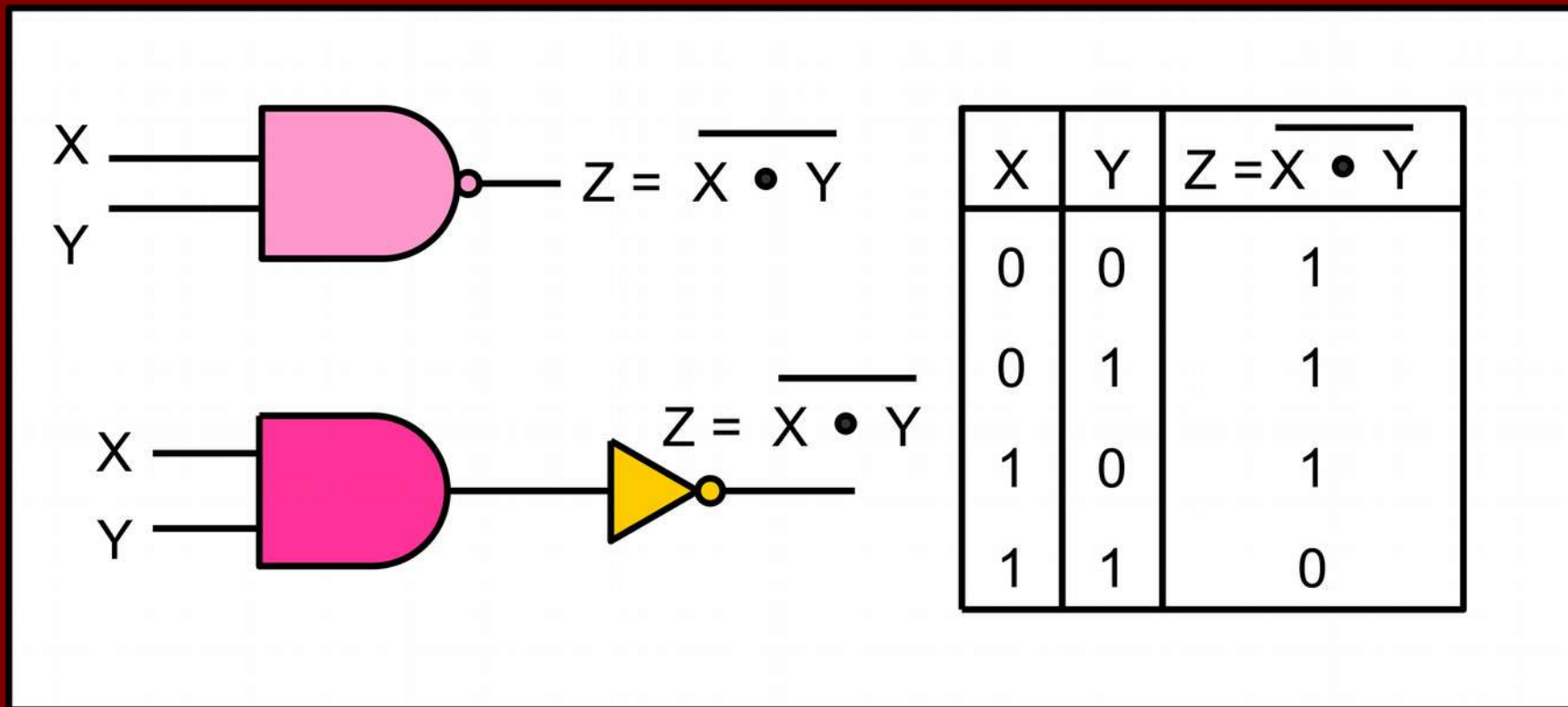




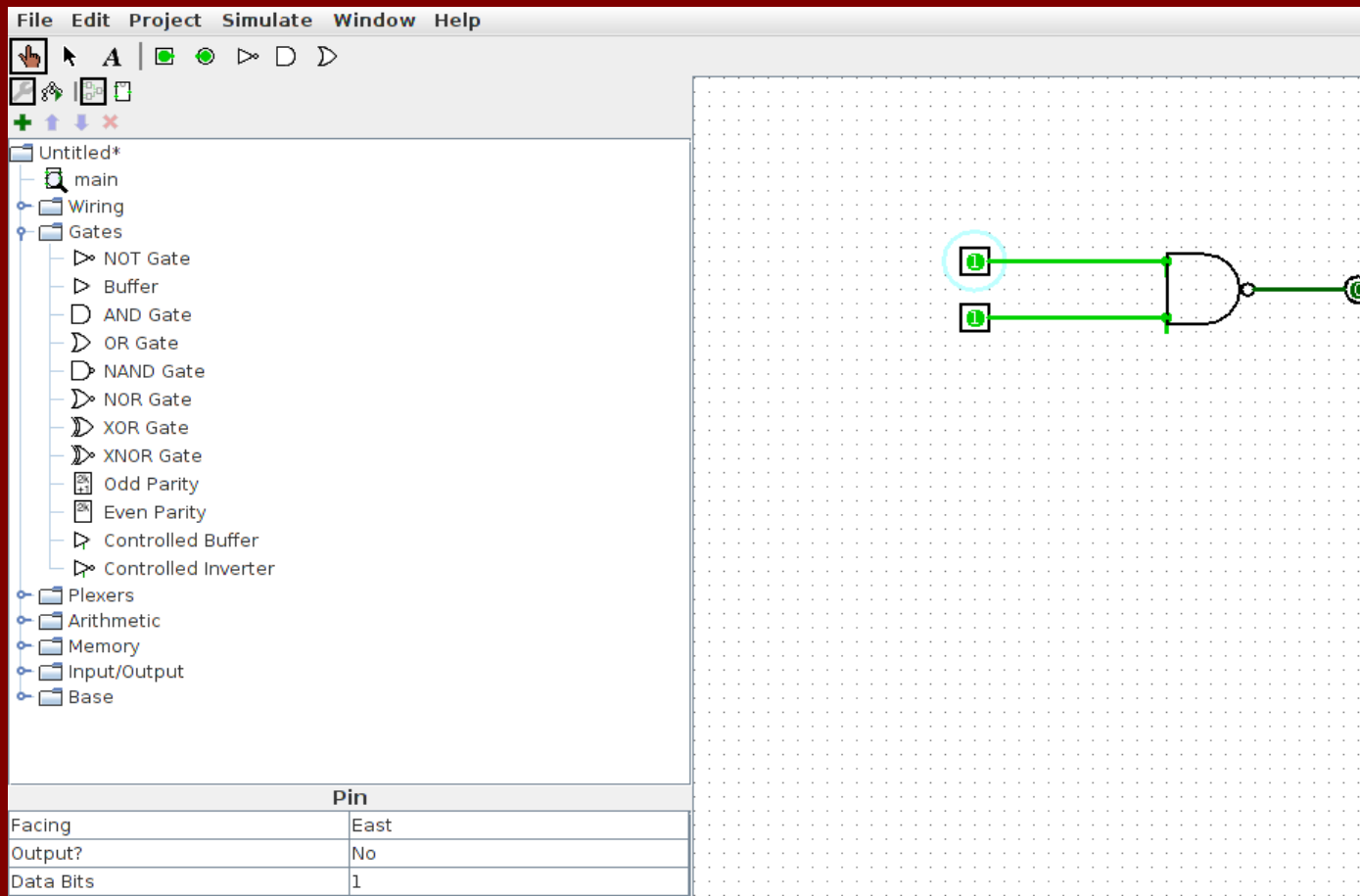


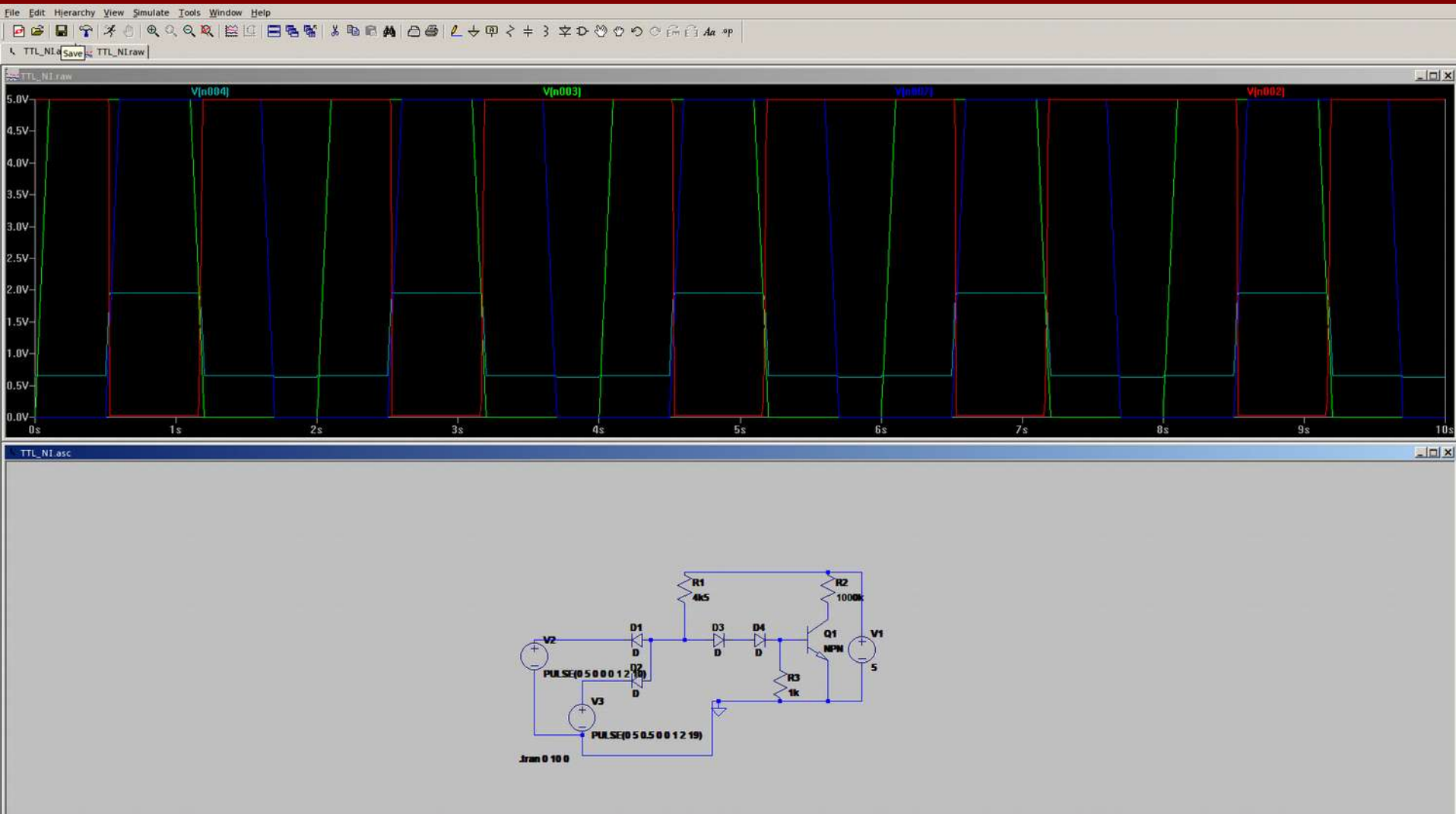
Operacija NI

Kombinacija I i NE - OPERACIJA



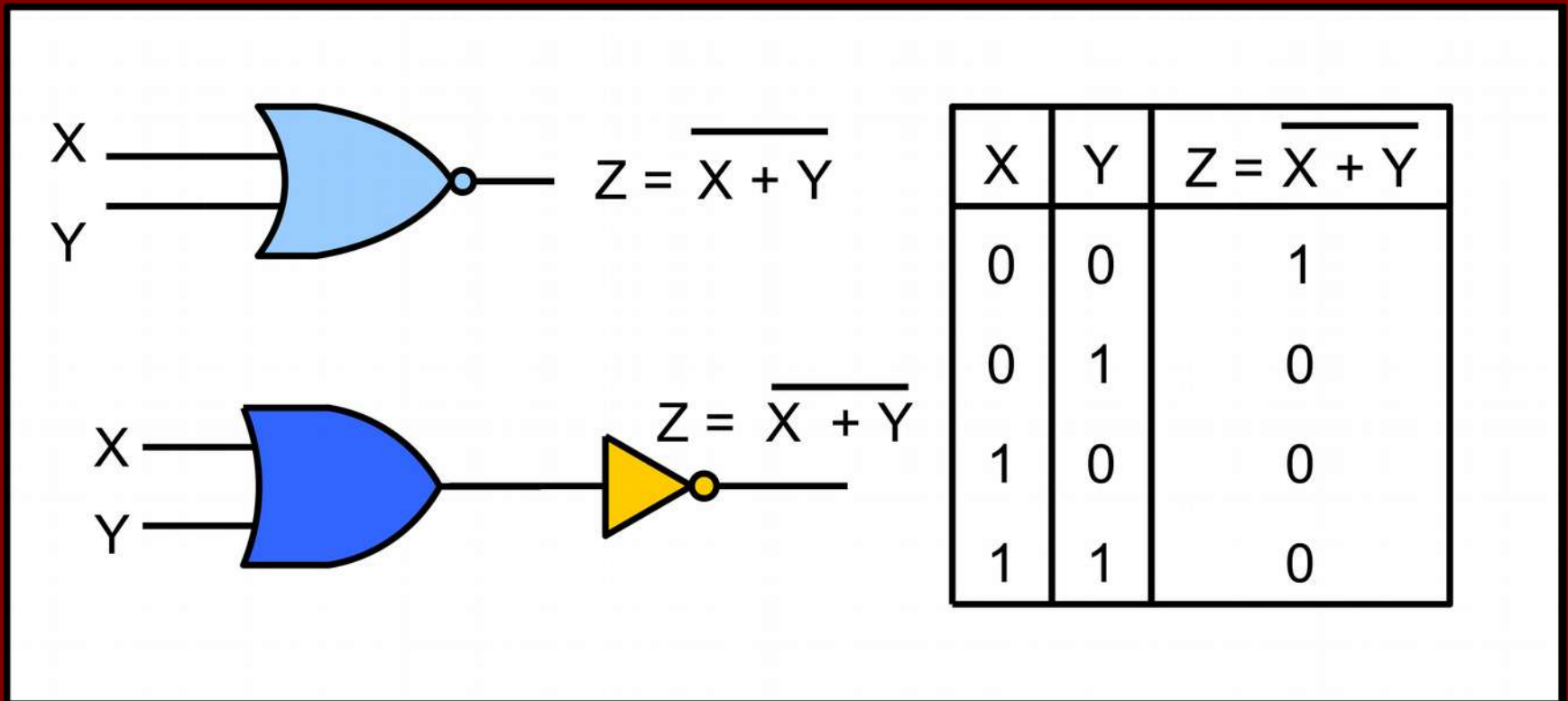
Graficki simbol i kombinaciona tabela za NI operaciju





Operacija NILI (NOR)

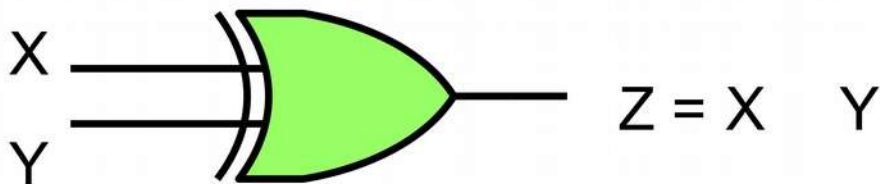
Kombinacija NILI i NE - OPERACIJA



Graficki simbol i kombinaciona tabela za NILI operaciju



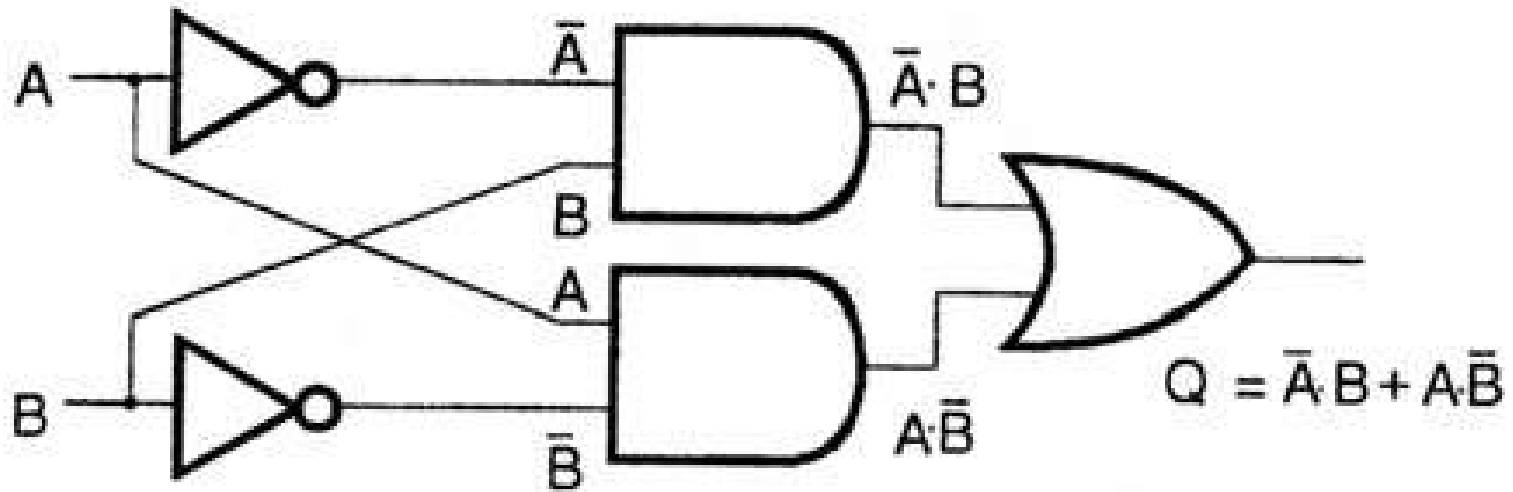
Operacija Isključivo ILI (Exclusive OR, EX-OR)



X	Y	$Z = X \oplus Y$
0	0	0
0	1	1
1	0	1
1	1	0

Graficki simbol i kombinaciona tabela za EX-OR operaciju

Logicka sema EX OR kola

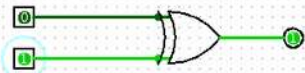




- Untitled*
- main
- Wiring
- Gates
 - NOT Gate
 - Buffer
 - AND Gate
 - OR Gate
 - NAND Gate
 - NOR Gate
 - XOR Gate
 - XNOR Gate
 - Odd Parity
 - Even Parity
 - Controlled Buffer
 - Controlled Inverter
- Plexers
- Arithmetic
- Memory
- Input/Output
- Base

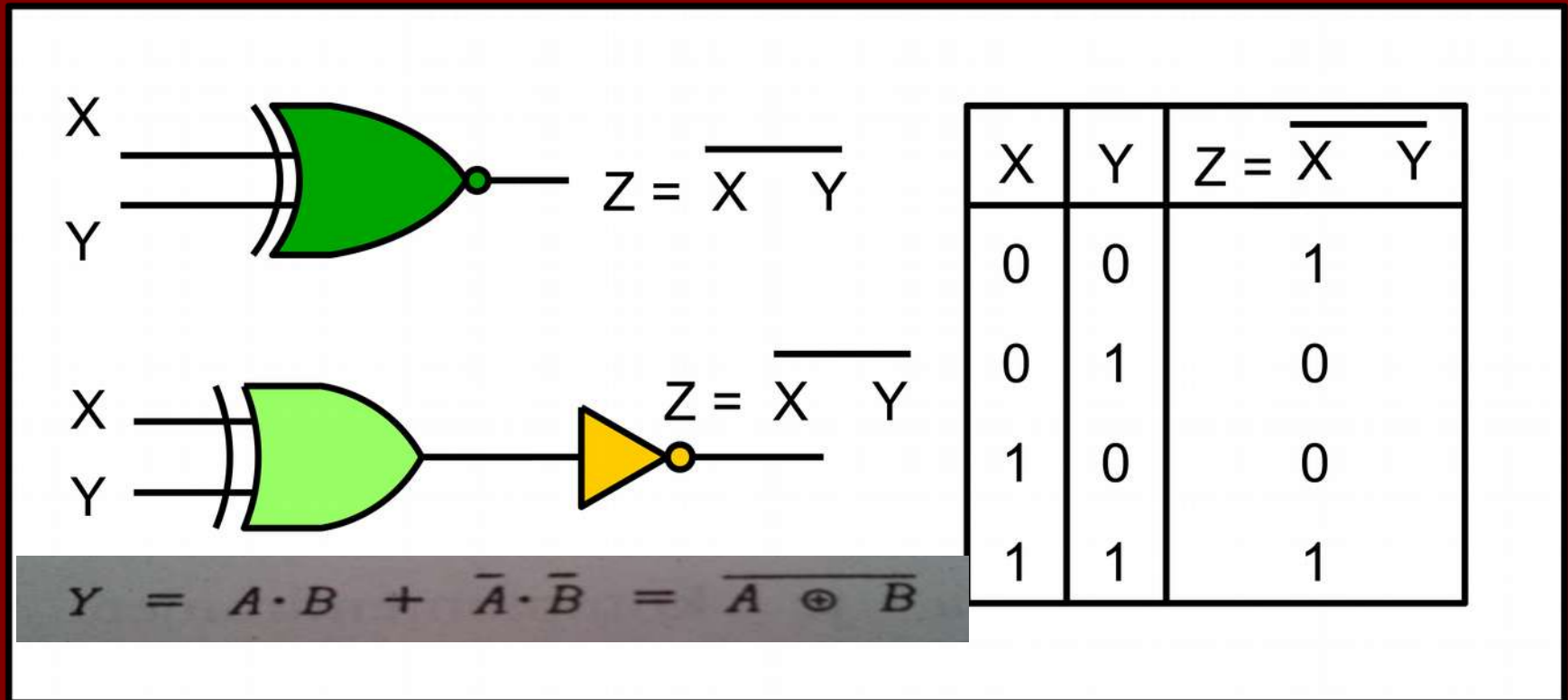
Pin	
Facing	East
Output?	No
Data Bits	1
Three-state?	No
Pull Behavior	Unchanged
Label	
Label Location	West
Label Font	SansSerif Plain 12

100%



Operacija Isključivo NILI

(Operacije koincidencije, daje kao rezultat logicku jedinicu ako su obe promenljive identicne)



Graficki simbol i kombinaciona tabela za EX-NOR operaciju

File Edit Project Simulate Window Help

Hand cursor, Arrow, Text tool, Eraser, Selection tool, Undo, Redo, Copy, Paste, Delete, Zoom in, Zoom out, Zoom reset, Zoom to fit, Zoom to width, Zoom to height, Zoom to auto, Zoom to default, Zoom to max, Zoom to min, Zoom to 100%, Zoom to 200%, Zoom to 50%, Zoom to 25%, Zoom to 125%, Zoom to 150%, Zoom to 175%, Zoom to 200%, Zoom to 225%, Zoom to 250%, Zoom to 275%, Zoom to 300%, Zoom to 325%, Zoom to 350%, Zoom to 375%, Zoom to 400%, Zoom to 425%, Zoom to 450%, Zoom to 475%, Zoom to 500%, Zoom to 525%, Zoom to 550%, Zoom to 575%, Zoom to 600%, Zoom to 625%, Zoom to 650%, Zoom to 675%, Zoom to 700%, Zoom to 725%, Zoom to 750%, Zoom to 775%, Zoom to 800%, Zoom to 825%, Zoom to 850%, Zoom to 875%, Zoom to 900%, Zoom to 925%, Zoom to 950%, Zoom to 975%, Zoom to 1000%

Untitled*

- main
- Wiring
- Gates
 - NOT Gate
 - Buffer
 - AND Gate
 - OR Gate
 - NAND Gate
 - NOR Gate
 - XOR Gate
 - XNOR Gate
 - Odd Parity
 - Even Parity
 - Controlled Buffer
 - Controlled Inverter
- Plexers
- Arithmetic
- Memory
- Input/Output
- Base

Pin

Facing	East
Output?	No
Data Bits	1
Three-state?	No
Pull Behavior	Unchanged
Label	

The diagram shows a circuit on a grid background. Two input pins, each labeled '0', are connected by green lines to the inputs of an AND gate. The output of the AND gate is connected by a green line to an output pin labeled '1'. The output pin is highlighted with a red circle.

Tehnike realizacije logičkih kola

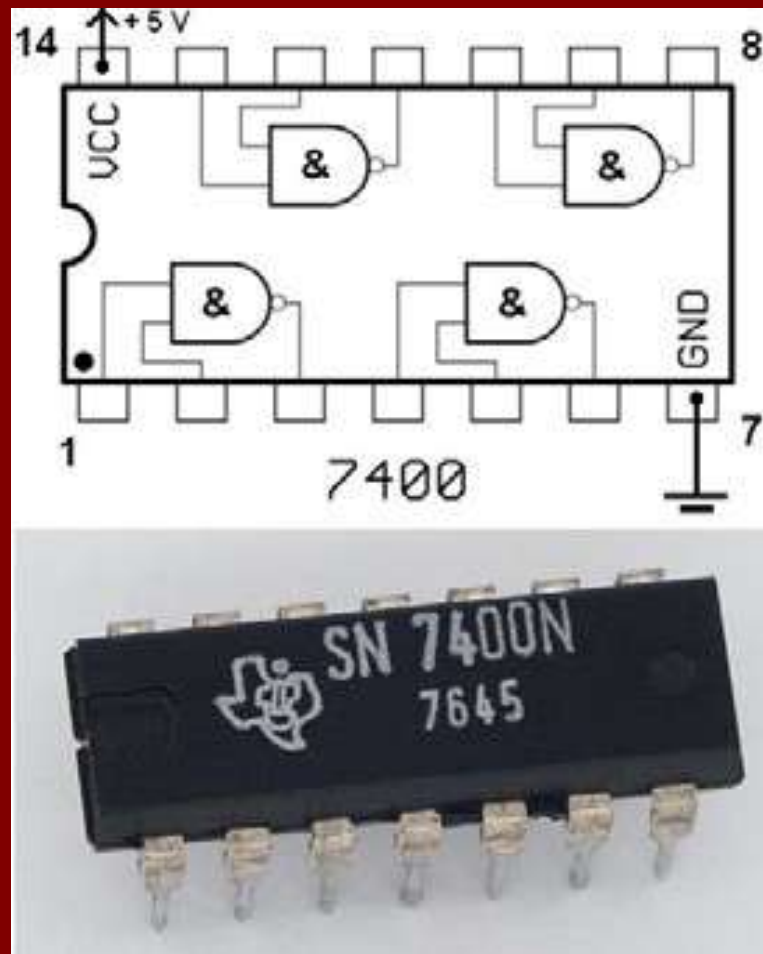
- Postoji više klasifikacija logičkih kola prema upotrebljenoj komponenti i po načinu vezivanja.
- Najčešće danas korišćene sprege logičkih kola su:
 - tranzistorsko – tranzistorska kola – TTL,
 - kola sa emitterskom spregom ECL,
 - MOS I
 - CMOS

Tehnike realizacije logičkih kola

U integrisanim kolima sastavni delovi logičkih kola su združeni, tj. integrisani u zajednički kristal poluprovodnika i ne mogu se više iz njega izdvojiti bez oštećenja.

Integrisano kolo može biti jednodelno ili višedelno ako je više razdvojivih komada kristala, na kojima su napravljeni sastavni delovi elektronskog kola, povezano u zajedničkom kućištu tako da čine celinu. Integrisana kola se izrađuju u kućištima raznih oblika i veličina.

TTL 7400



CMOS (engl. Complementary Metal – Oxide – Semiconductor)

CMOS (engl. Complementary Metal – Oxide – Semiconductor) je tehnološka generacija integralnih kola, stvorena 60-ih godina 20 veka. Za razliku od TTL tehnologije sa bipolarnim tranzistorima odlika CMOS tehnologije bila mala potrošnja struje.

Originalna CMOS tehnologija je vremenom evoluirala u mnoge podvrste. Jedna od najznačajnijih je HCMOS.

Digitalni računari

Digitalna tehnika je našla široku primenu u brojnim oblastima tehnike i tehnologije, ali je oblast *digitalnih računara* najvažnija i najzastupljenija. Mada digitalni računari utiču na značajan deo našeg života, malo je verovatno da većina ljudi zna šta računara egzaktno radi. Prosto rečeno, *računar je hardverski sistem koji izvršava aritmetičke operacije, manipulisanje podacima (obično u binarnoj formi) i donosi odluke.*

U najvećem broju slučajeva, čovek može da radi sve što i računar, ali računar to može da radi mnogo većom brzinom i sa mnogo većom tačnošću, bez obizra na činjenicu da računar sva svoja izračunavanja i operacije ostvaruje pojedinačno u vremenu.

Digitalni računari

Računar je brži i tačniji od čoveka, međutim, za razliku od većine ljudi on ima kompletan skup instrukcija koje egzaktno govore šta treba uraditi na svakom koraku svoga rada.

Ovaj skup instrukcija se naziva *program* i pripremljen je od jedne ili više osoba za svaki posao koji računar treba da obavi.

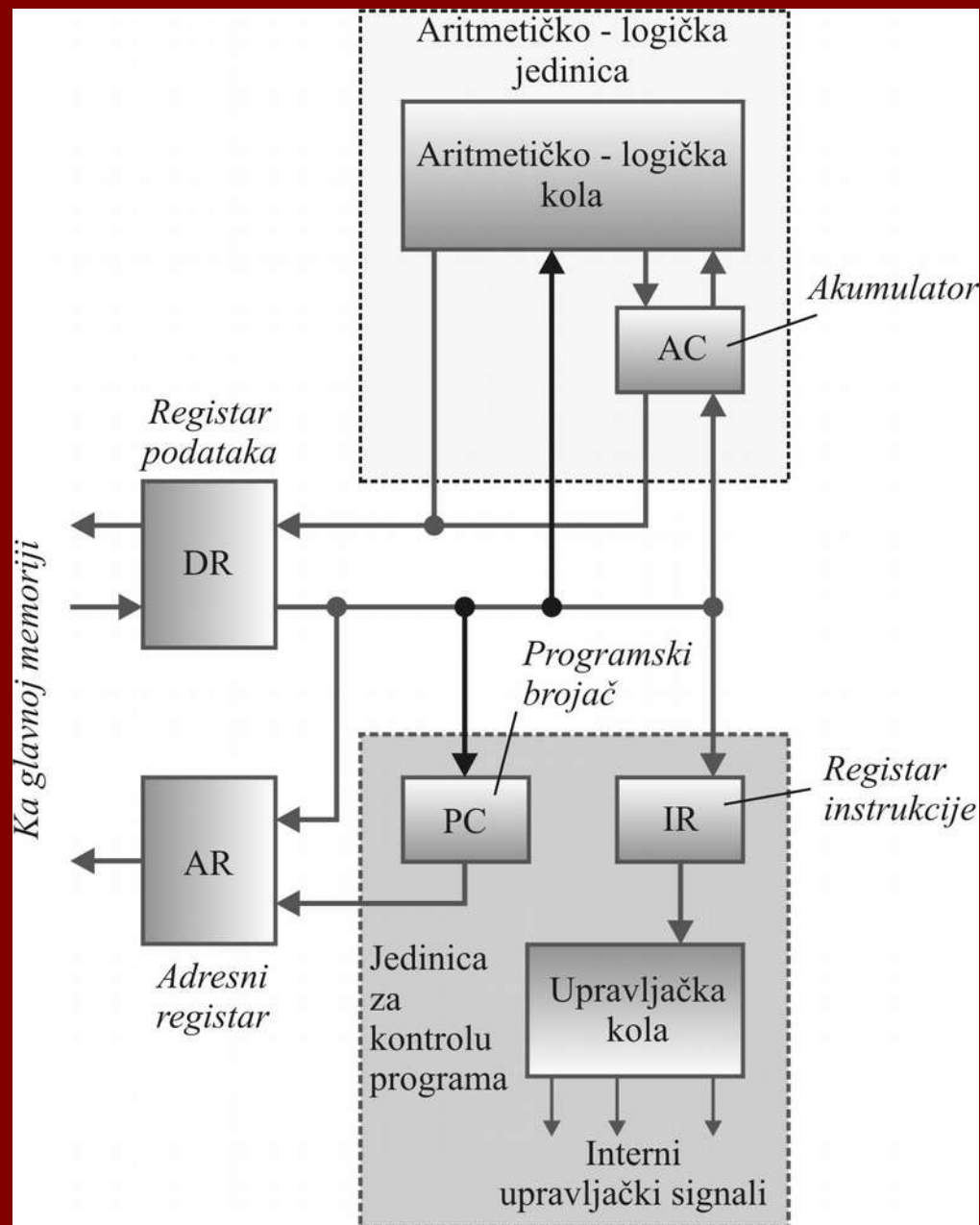
Programi se smeštaju u računarsku memoriju u binarno kodiranom obliku pri čemu svaka instrukcija ima jedinstveni kod.

Računar uzima ove kodove instrukcija iz memorije jedan po jedan i izvršava operaciju koju specificira kod.

Osnovni delovi računara

Postoji nekoliko tipova računarskih sistema, međutim, može se uočiti da se u principu svi oni sastoje iz istih funkcionalnih jedinica.

Svaka jedinica izvršava specifične funkcije, ali sve jedinice zajedno izvršavaju instrukcije koje se nalaze u programu.



Digitalni sistemi na početku razvoja zahtevali su velik broj komponenti. Veliki napredak na polju VLSI tehnologije doveo je do proizvodnje mikroračunarskih sistema kod kojih su sve neophodne komponente ugrađene u prateće komponente. Kao rezultat dalje integracije nastaju mikrokontroleri koji u sebi integrišu analogni i digitalni ulazno-izlazni podsistem, tajmere i brojače, LCD drajvere i druge logičke module.

Ove komponente su samostalno bez dodatnog hardvera sposobne da obavljaju složene upravljačke funkcije.

Mikrokontrolere delimo na 4-, 8-, 16-, i 32-bitne.

Nasuprot mikroprocesorima, mikrokontroleri su više orijentisani ka postizanju superiornijih performansi kod aplikacija namenjenih upravljanju, bezbednosti i pouzdanosti u radu. Imajući u vidu da je cena uređaja baziranih na mikrokontrolerima od izuzetne važnosti za krajnjeg korisnika, mikrokontroleri su integrisali na čipu veliki broj standardnih perifera, kontrolere raznih tipova uređaja, kao i memoriju.

Za razliku od mikroprocesora, cena mikrokontrolera je veoma niska.

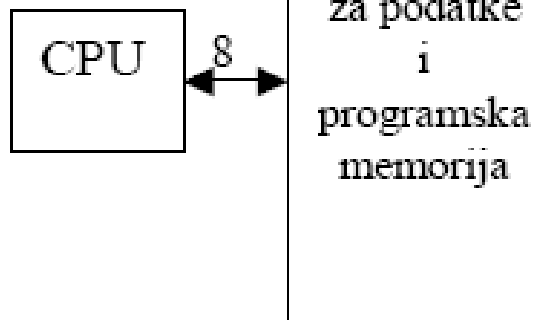
Za određene aplikacije niska cena mikrokontrolera je idealno rešenje. Taktna frekvencija na kojoj rade, veličina adresne memorije i veličina podataka sa kojima manipulišu varira od jednog mikrokontrolera do drugog, šta više i kod jednog istog proizvođača, tako da je zadatak projektanta da izabere najbolji mikrokontroler za datu aplikaciju.

Drugi važan razlog korišćenja mikrokontrolera, u odnosu na mikroprocesor, predstavlja njegova kompaktnost koja se može izraziti konstatacijom:

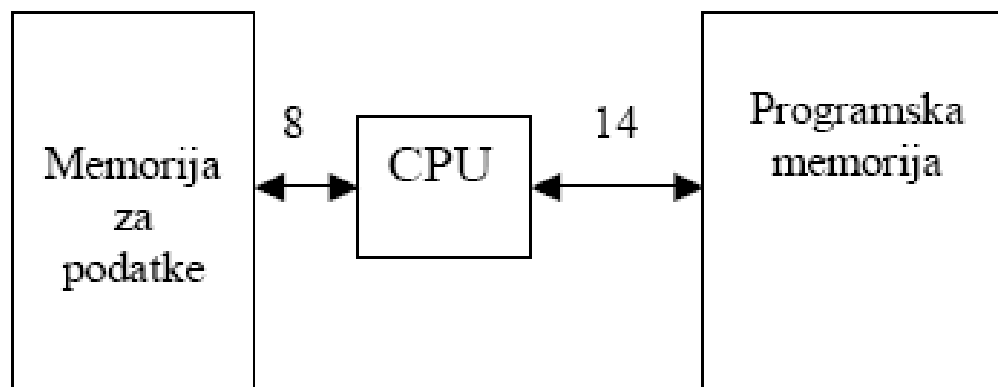
„Računar na jednom čipu.“

Danas postoji veliki broj familija mikrokontrolera kao sto su : Microchip PIC, Atmel, Texas Instruments, ...

Von Neumann

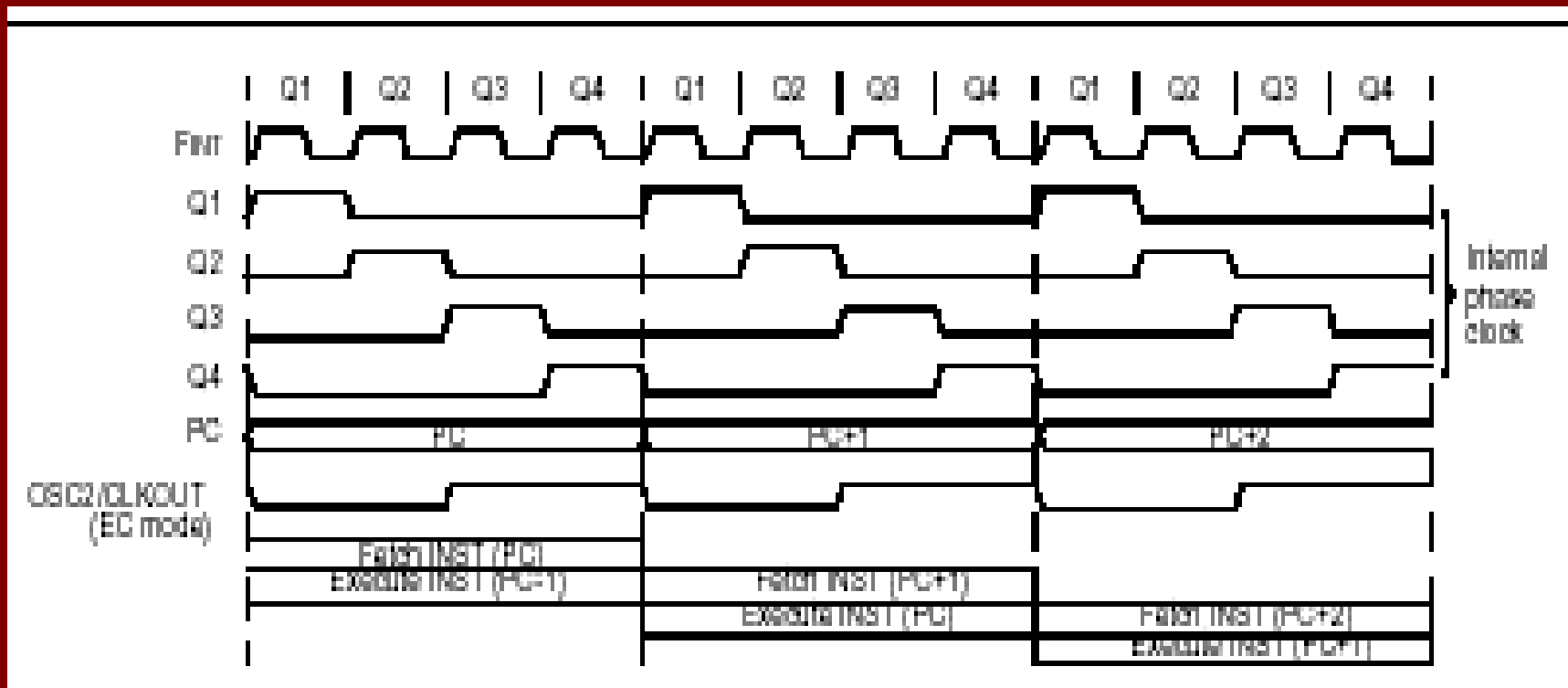


Harvard



Takt i izvršenje instrukcija, primer mikrokontrolera Microchip Pic18F452

Takt je pokretač mikrokontrolera i dobija se iz spoljnog oscilatora preko nožica OSC1 i OSC2. Takt iz oscilatora koji dolazi na nožicu OSC1, interno se deli u mikrokontroleru na četiri podjednaka takta Q1, Q2, Q3 i Q4. Ova četiri takta čine instrukcijski ciklus, a to je vreme za koje se izvrši jedna instrukcija. Na svaki Q1 iz programske memorije uzima se instrukcija i upisuje u instrukcijski registar na Q4. Između Q1 i Q4 obavlja se dešifrovanje i izvršenje instrukcije. Na slici je prikazan odnos instrukcijskog ciklusa i takta oscilatora na pinu OSC1. PC je programski brojač ili Program Counter.



Za pozivanje instrukcije koristi se jedan instrukcijski ciklus, a za dešifrovanje i izvršanje još dva. Zbog protočne obrade koja se u literaturi zove **Pipelining** svaka instrukcija se izvršava u jednom ciklusu, ukoliko to nije instrukcija koja zahteva promene u programskom brojaču, kao na primer instrukcije skoka. Tada su potrebna dva instrukcijska ciklusa.