Simulacija rada logičkih sklopova

Kovačić, Stipe

Undergraduate thesis / Završni rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Organization and Informatics / Sveučilište u Zagrebu, Fakultet organizacije i informatike**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:211:213135

Rights / Prava: Attribution 3.0 Unported/Imenovanje 3.0

Download date / Datum preuzimanja: **2024-04-20**

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN

Repository / Repozitorij:

Faculty of Organization and Informatics - Digital Repository

zir.nsk.hr

DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Stipe Kovačić

SIMULACIJA RADA LOGIČKIH SKLOPOVA

ZAVRŠNI RAD

Varaždin, 2020.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Stipe Kovačić

Matični broj: 45130/16-R

Studij: Informacijski sustavi

SIMULACIJA RADA LOGIČKIH SKLOPOVA

ZAVRŠNI RAD

Mentor:

Luka Milić, mag. ing. comp.

Varaždin, rujan 2020.

Sadržaj

1. Uvod 1
2. Digitalna logika 2
2.1 Povijest nastanka logičkih sklopova 3
2.2 Osnovna logička vrata 4
2.2.1 Logička I-vrata 5
2.2.2 Logička ILI-vrata 7
2.2.3 Logička NE-vrata 8
2.3 Izvedeni logički sklopovi 9
2.3.1 Logički sklopovi NI i NILI 10
2.3.2 Logički sklop isključivo-ILI 12
2.3.3 Logički sklop isključivo-NILI 13
3. Osnovni kombinacijski sklopi 14
3.1 Dekoder 2/4 14
3.2 Koder 4/2 15
3.3 Poluzbrajalo 16
3.4 Zbrajalo 17
3.5 Demultipleksor 1/4 19
3.6 Multipleksor 4/1 20
4. Dodatni logički sklopi 21
4.1 Generator signala takta 21
4.2 Digitalni dekoder 21
5. Osnovni sekvencijski sklopi 22
5.1 SR-bistabil 22
5.2 JK-bistabil 23
5.3 T-bistabil 24
5.4 D-bistabil 25
6. Programsko oruđe LG-simulator 27
6.1 Glavni izbornik 27
6.2 Prostor sa sklopovima 29
6.3 Tiskana ploča 30
7. Simulacije 32
7.1 Generator parnoga pariteta 32
7.2 4-bitno binarno zbrajalo 33
7.3 4-bitno binarno brojilo 33
7.4 Dekoder 34
8. Zaključak 35
Bibliografija 36

1. Uvod

U ovom radu bit će prikazan kratki pregled povijesti nastanka logičkih sklopova, zatim će biti objašnjen način rada i poraba osnovnih i složenih logičkih sklopova te njihova primjena u suvremenim digitalnim elektroničkim sustavima. Rad navedenih kombinacijskih i sekvencijskih sklopova bit će simuliran u računalnoj aplikaciji izrađenoj u programskom jeziku četvrte generacije C# rabeći razvojni okvir WPF namijenjen za razvoj aplikacija na operacijskom sustavu Windows.

Brojni problemi iz svakodnevnice, posebice iz područja tehničkih znanosti mogu se riješiti porabom Booleove algebre koja će biti objašnjena u nastavku. Štoviše na svaki sustav u kojem svaka varijabla ima upravo dva stanja primjenjiva je Booleova algebra. Ta dva stabilna stanja prevodimo na jednostavno „da“ ili „ne“ , odnosno u slučaju računala na „1“ ili „0“. Upravo oni čine osnovu logičkih sklopova, posljedično i svih računala. [1]

Kao informatičari, postoji zanemariva vjerojatnost da ćemo trebati oblikovati fizičku komponentu računala, logički strujni krug ili neki drugi dio računalnoga *hardwarea*. Bez obzira na navedeno, znanje o principu rada suvremenih računala na najnižoj razini, motivacijama iza računalnoga oblikovanja i njegovu ostvaraju bit će od velike pomoći u shvaćanju funkcioniranja uređaja koji je iz temelja promijenio svijet i s kojim se većina svijeta služi svakodnevno. Razumijevanje rada Booleove logike i njezina utjecaja na rad računala će nam omogućiti, s logičke i programerske strane, puno efikasniju porabu svakoga računalnoga sustava te je upravo to motivacija za pisanje ovoga rada. [2]

2. Digitalna logika

Pojam „digitalne logike“ implicira da su informacije u sustavu prikazane brojevima ograničenim u vrijednosti. Takav sustav funkcionira pouzdanije i efikasnije ako obrađuje samo dvije vrijednosti. Digitalni sustavi koji imaju samo dvije vrijednosti nazivaju se binarnim sustavima te se odnose na sklopove koji imaju samo dva moguća stanja u bilo kojem trenutku [3]. Primjerice tranzistor koji može biti zasićen ili ne provoditi. U elektronici učestalo se govori o voltažama te se ta dva stanja nazivaju „*High*“ i „*Low*“. Predstavljaju „Da“ i „Ne“ stanja u Booleovoj logici i prikazuju bitove podataka kao što su:

- Bit broja
- Je li se neki događaj dogodio?
- Postoji li signal?

Upravo generiranje digitalnih izlaza iz digitalnih ulaza jest glavni zadatak digitalne elektronike. U uvodnom poglavlju navedeno je da se veliki broj problema iz svakodnevnice, a posebice iz područja tehničkih znanosti može riješiti porabom nekoga oblika Booleove logike, u ovom slučaju digitalne logike. Takve probleme svrstavamo u dva razreda:

- Problemi za koja nisu potrebna prethodna stanja
- Problemi za koja je potrebno znanje o prethodnim stanjima

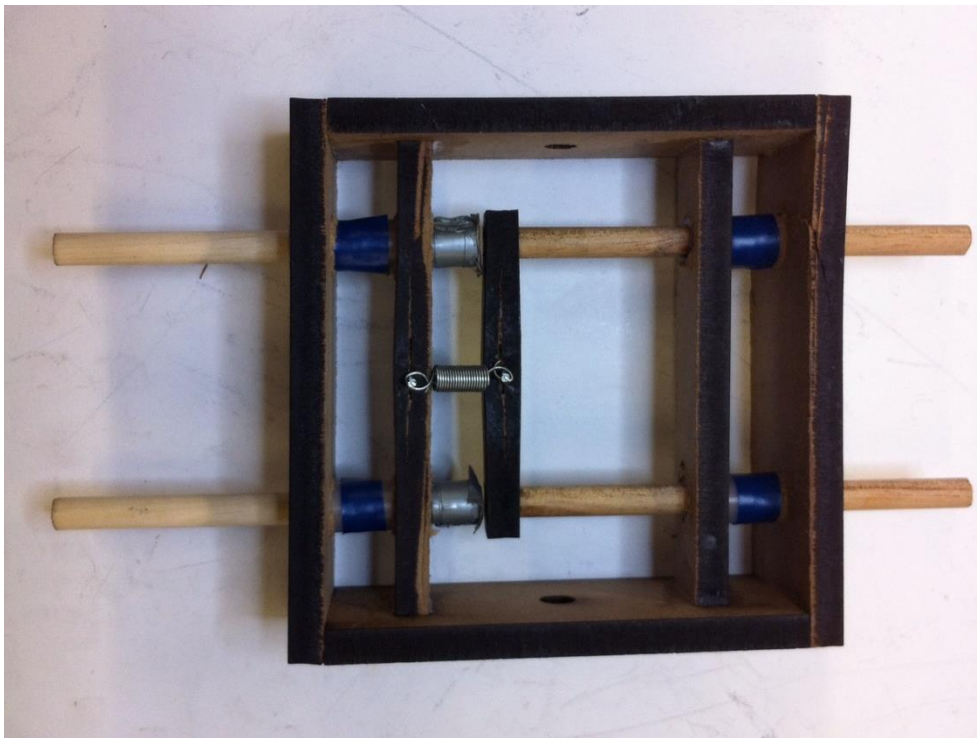
Primjerice uspoređivanje dva broja ili zbrajanje bita pariteta spada u prvu skupinu problema, dok držanje podatka o broju ponavljanja broja u sekvenci spada u potonju skupinu [4]. Zbog postojanja različitih razreda problema, potrebne su nam dvije vrste logičkih sklopova:

- Kombinacijski
- Sekvencijski

koji će biti pobliže prikazani u sljedećim poglavljima.

2.1 Povijest nastanka logičkih sklopova

Suvremena računala sadrže desetke milijuna logičkih sklopova ostvarenih pomoću tranzistora mikroskopskih veličina koji se iz godine u godinu smanjuju [5]. S toga je teško zamisliti da je Charles Babbage 1837. godine oblikovao prva logička vrata koja su bila potpuno mehanička. Porabio ih je u računalu opće namjene „*Analytical engine*“ gdje je rabio logička vrata kao mehaničke zupčanike za rješavanje polinomne jednadžbe.

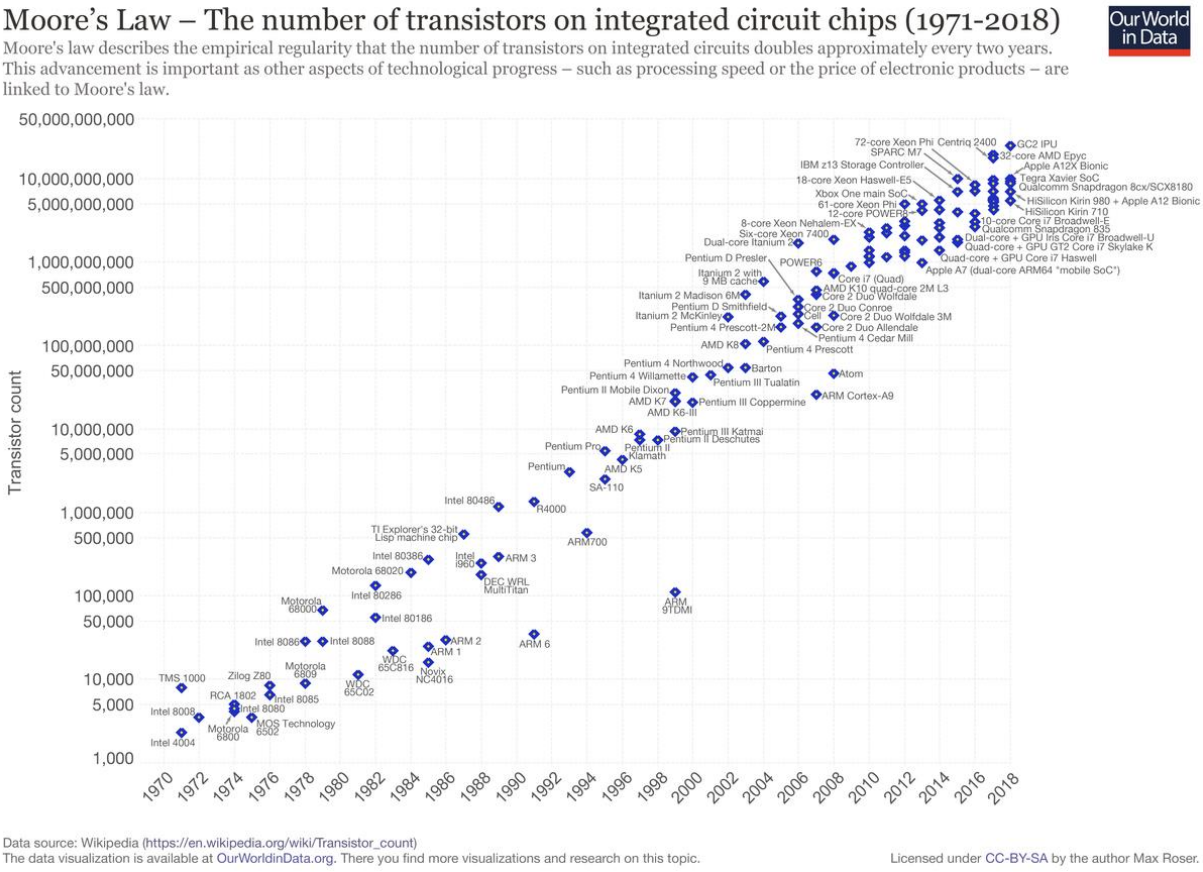


Slika 1: Mehanički logički sklop [6]

Prvi patent za strujni krug koji sadržava logička vrata registrirao je Almon Strowgera 1891. godine. Razvoju prvih računala pridonio je i Nikola Tesla koji je nakon 1898. godine imao više patenta za slične unaprijeđene uređaje. Teslini uređaji su imali funkcionalna vrata „I“, te će kasnije biti rabljeni kao podloga za razvitak uređaja na daljinsko upravljanje. [7]

Prvi digitalni i programirani strujni krugovi s logičkim sklopovima rabili su vakuumske cijevi. Prva poznata računala temeljena na toj tehnologiji su „ENIAC“ i „Colossus“. S obzirom da su sadržavali tisuće vakuumski cijevi, težili su desetke tona. Iz toga razloga njihova poraba i rasprostranjenost bila je vrlo ograničena. Unatoč tomu što su prvi oblikovani više od 100 godina unatrag, logički sklopovi široj javnosti postaju dostupni tek 1960-tih, u vidu logike rezistor-tranzistor i tranzistor-tranzistor koja se nalazila u prvim osobnim računalima.

Od tih preteča modernih računala, tehnologija je naprjedovala nevjerojatnom brzinom. S toga su danas u porabi vrlo sitni strujni krugovi sa velikim brojem vratâ koji su u bližoj prošlosti bili nezamislivi. [8]



Slika 2: Broj tranzistora u ICC kroz vrijeme [9]

2.2 Osnovna logička vrata

Logički sklopovi donose nizove odluka kako bi dali ili dobili logičke odgovore na problem za dani skup ulaznih uvjeta. Za donošenje takvih odluka rabe se 3 osnovna logička vrata

- I-vrata
- ILI-vrata
- NE-vrata

[3].

Navedena vrata dovoljna su za potpunu primjenu algebre sustava koju je George Boole razvio u 19. stoljeću. Njezina prva važnija poraba dogodila se 1938. godine kada je Shannon pokazao da se koncepti Booleove algebre mogu rabiti u teoriji elektroničkoga prekidanja.

U Booleovoj algebri simbol + znači ILI (engl. *OR*), a simbol · znači I (engl. *AND*):

$A + B = Y$ (Ako je A istinito ILI ako je B istinito tada je Y istinito.)

$A \cdot B = Y$ (Ako je A istinito I ako je B istinito tada je Y istinito.)

Kompleksniji izrazi se ostvaruju kombinacijom navedenih osnovnih logičkih izraza [10].

2.2.1 Logička I-vrata

Logički sklop „I“ obavlja logičku operaciju konjunkcije. Može imati 2 ili više ulaza, dok na izlazu daje samo jedno moguće stanje i zadovoljava definiciju:

izlaz je u stanju 1 ako i samo ako su svi ulazi u stanju 1.

$Y = A \cdot B \cdot ... \cdot N$

koja se čita: *Y je A i B i . . . i N*

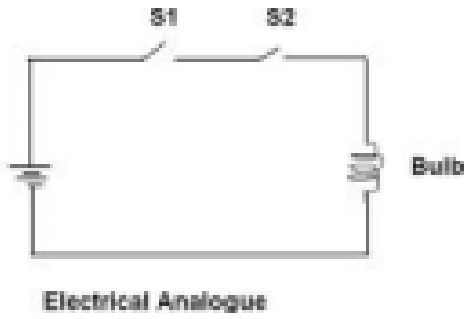
[10]

SIMBOL	TABLICA ISTINE		
	B	A	Q
	0	0	0
	0	1	0
	1	0	0
	1	1	1

Slika 3: Izgled i tablica istine I-vrata

Električni protulik:

Kada su oba prekidača zatvorena struja prolazi kroz žarulju te je izlaz 1. Ako je bilo koji od prekidača otvoren onda krug postaje otvoren i izlaz je 0 [3].



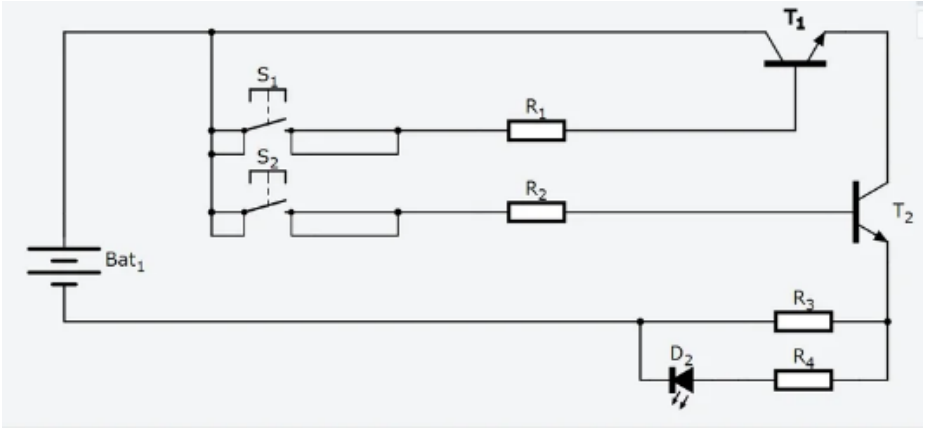
Slika 4: Prikaz električnoga protulika [3]

Poraba:

ILI-vrata često se rabe kao *Enable-vrata* i *Inhibit*-vrata kako bi se omogućio ili onemogućio prolazak podatka s jednoga mjesta u sklopu na drugo. Primjer uređaja koji se služi navedenim vratima jest samoposlužni uređaj za kavu s kojim se svakodnevno susrećemo. Da bi takav uređaj korisniku poslužio kavu potrebno je ispuniti dva uvjeta:

- Ubačena je dovoljna količina novca
- Pritisnuta je tipka za odabir napitka

Samo u slučaju da su oba uvjeta ispunjena izlaz iz aparata bit će napitak.



Slika 5: Shematski prikaz vrata Enable/Inhibit [11]

2.2.2 Logička ILI-vrata

Logički sklop „ILI“ obavlja logičku operaciju disjunkcije. Može imati 2 ili više ulaza, dok na izlazu daje samo jedno moguće stanje i zadovoljava definiciju:

izlaz je u stanju 1 ako je jedan ili više ulaza u stanju 1.

$Y = A + B + ... + N$

koja se čita: *Y* je *A ili B ili . . . ili N*

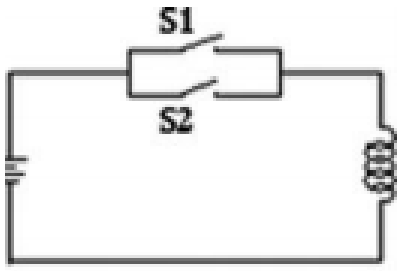
[10]

SIMBOL	TABLICA ISTINE		
	B	A	Q
	0	0	0
	0	1	1
	1	0	1
	1	1	1

Slika 6: Izgled i tablica istine ILI-vrata

Električni protulik:

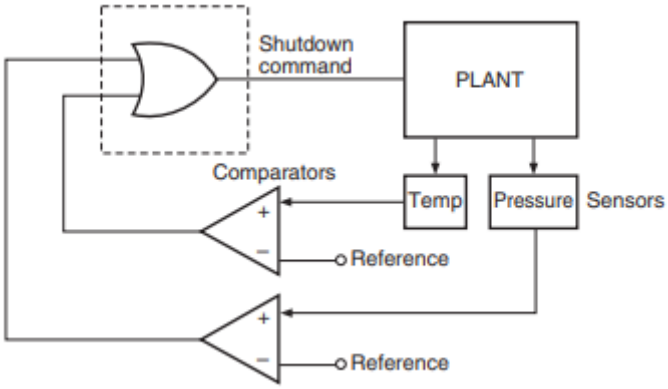
Kada su oba prekidača otvorena, struja prolazi kroz žarulju i izlaz je 1. Ako je bilo koji od prekidača zatvoren struja prolazi kroz kratki spoj i izlaz je 0 [3]



Slika 7: Prikaz električnoga protulika [3]

Poraba:

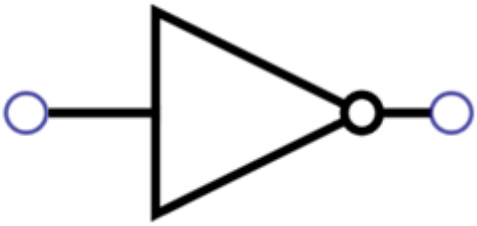
ILI-vrata mogu se rabiti u svim situacijama kada se mora izvršiti radnja pojavom jednoga ili više događaja. Primjerice kada u proizvodnom pogonu prelazak postavljenoga ograničenja jednoga ili više parametara treba rezultirati radnjom koja će zaštititi pogon [12]. Npr. pregrijavanje stroja trebalo bi rezultirati njegovim gašenjem. Shematski prikaz takve situacije prikazan je na sljedećoj slici.



Slika 8: Primjena ILI-vrata [12]

2.2.3 Logička NE-vrata

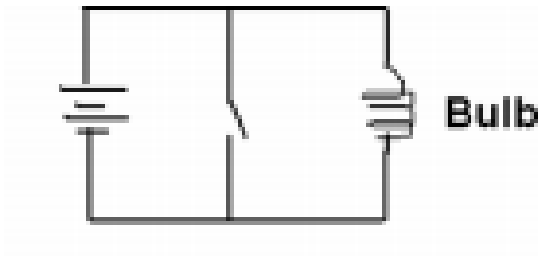
Logički sklop „NE“ (invertor) obavlja logičku operaciju nijekanja. Ima upravo jedan ulaz, dok na izlazu daje samo jedno moguće stanje koje će biti komplement ulaza. Ako je ulaz 0, izlaz je jedan, ako je ulaz 1 izlaz je 0.

SIMBOL	TABLICA ISTINE	
	A	Q
	0	1
	1	0

Slika 9: Izgled i tablica istine NE-vrata

Električni protulik:

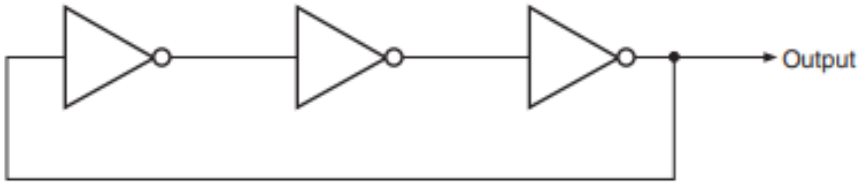
Kada je prekidač otvoren, žaruljica svijetli, ulaz = 0, izlaz = 1. Kada je prekidač zatvoren, cijela struja će proći kroz kratki spoj, zbog toga će izlaz biti 1 [3].



Slika 10: Prikaz električnoga protulika [3]

Poraba:

Invertori su često rabljeni za izgradnju oscilatora za generiranje signala takta. Ovi generatori sata pružaju dobru stabilnost i operacije u širokom rasponu napona napajanja i širokom frekvencijskom rasponu. Odlikuju ih niska potrošnja i jednostavno sučelje prema drugim logičkim obiteljima.



Slika 11: Oscilator kvadratnoga vala [12]

2.3 Izvedeni logički sklopovi

Spajanjem osnovnih logičkih sklopova nastaju izvedeni logički sklopovi koji imaju vrlo široku primjenu. Kombiniranjem logičkih funkcija (nijekanje, konjunkcija i disjunkcija) nastaju logički sklopovi:

- NI (engl. *NAND*)
- NILI (engl. *NOR*)
- isključivo-ILI (engl. *XOR*)
- isključivo-NILI (engl. *XNOR*)

2.3.1 Logički sklopovi NI i NILI

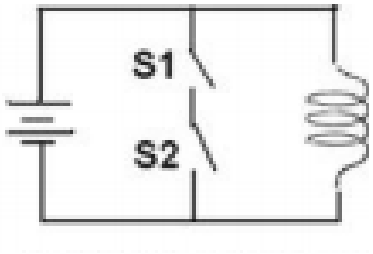
Funkcija invertora (engl. *NOT*) može se kombinirati s logičkim sklopovima „I“ i „ILI“ kako bi se tvorili izvedeni logički sklopovi „NI“ I „NILI“. Izlaz NILI-vrata bit će 1 kada su svi ulazi jednaki 0, a izlaz će biti 0 kada je bilo koji od ulaza 1. Analogno tomu izlaz NI-vrata bit će 0 kada su svi ulazi 1, a izlaz će biti 1 kad je bilo koji od ulaza 1.

SIMBOL	TABLICA ISTINE		
	B	A	Q
	0	0	1
	0	1	1
	1	0	1
	1	1	0

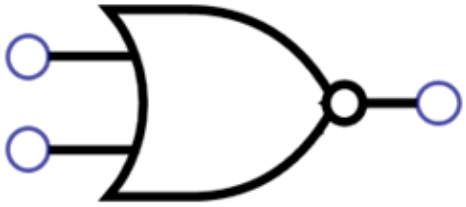
Slika 12: Izgled i tablica istine NI-vrata

Električni protulik NI-vrata:

Kada su oba prekidača zatvorena, struja prolazi kroz kratki spoj i ne dolazi do žarulje, zbog toga je izlaz 0. Kada je bilo koji od prekidača otvoren, struja prolazi kroz žarulju i daje izlaz 1. [3].



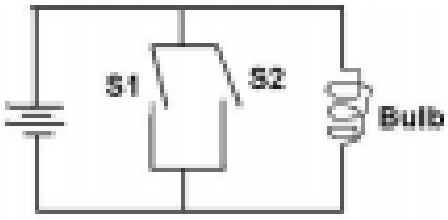
Slika 13: Električni protulik NI-vrata [3]

SIMBOL	TABLICA ISTINE		
	B	A	Q
	0	0	1
	0	1	0
	1	0	0
	1	1	0

Slika 14: Izgled i tablica istine NILI-vrata

Električni protulik NILI-vrata:

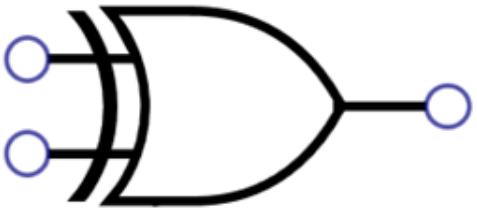
Kada su oba prekidača otvorena tada struja prolazi kroz žarulju i izlaz je 1. Ako je bilo koji od prekidača zatvoren, tada struja prolazi kroz kratki spoj i ne dolazi do žarulje i iz toga razloga je izlaz 0 [3].



Slika 15: Električni protulik NILI-vrata [3]

2.3.2 Logički sklop isključivo-ILI

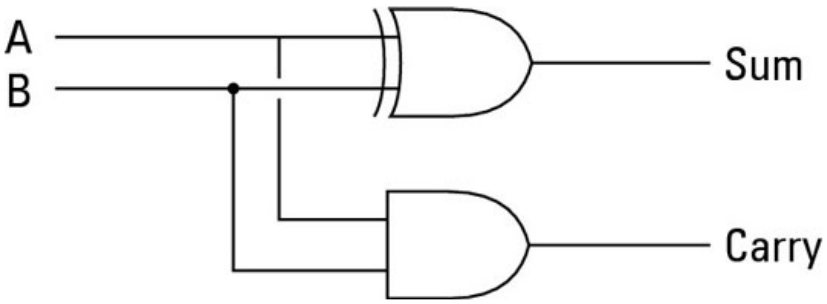
Isključivo-ILI-vrata su vrata s dva ulaza i jednim izlazom. Dobivaju se kombinacijom osnovnih logičkih vrata kako bi se stvorila složenija vrata koja su vrlo rabljena u izradi modernih aritmetičko-logičkih jedinica i sklopova za prepoznavanje pogrješaka. Njihov izlaz je logička jedinica kada su ulazi različiti, te logička ništica kada su ulazi jednaki.

SIMBOL	TABLICA ISTINE		
	B	A	Q
	0	0	0
	0	1	1
	1	0	1
	1	1	0

Slika 16: Izgled i tablica istine isključivo-ILI-vrata

Poraba:

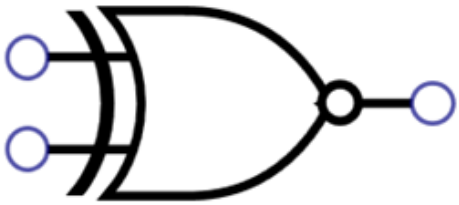
Jedna od najučestalijih poraba isključivo-ILI-vrata je zbrajanje dva binarna broja. Kako bi se omogućila operacija zbrajanja potrebno je navedena vrata kombinirati s I-vratima. Takav sklop ima dva izlaza. Jedan predstavlja zbroj, a drugi prijenos koji je važan kada se više zbrajala rabi zajedno kako bi se zbrojili binarni brojevi veći od jednoga bita. [13]



Slika 17: Sklop za zbrajanje dva binarna broja [13]

2.3.3 Logički sklop isključivo-NILI

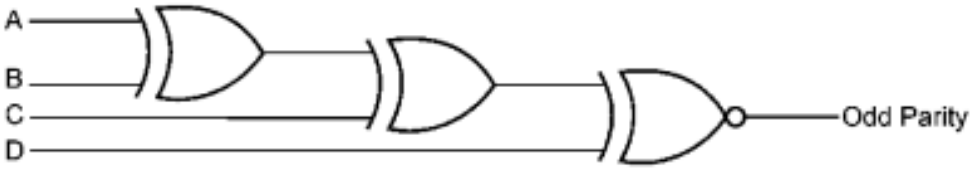
Isključivo-NILI-vrata su vrata dobivena komplementom izlaza isključivo-ILI-vrata, kombinacijom isključivo-ILI-vrata i NE-vrata. Izlaz isključivo-NILI-vrata s dva ulaza je logička jedinica kada su oba ulaza jednaka, a logička ništica kada su ulazi različiti. Zbog načina rada često se nazivaju i „vrata jednakosti“. [12]

SIMBOL	TABLICA ISTINE		
	B	A	Q
	0	0	1
	0	1	0
	1	0	0
	1	1	1

Slika 18: Izgled i tablica istine isključivo-NILI-vrata

Poraba:

Logička vrata isključivo-ILI i isključivo-NILI obično se rabe u logičkim sklopovima za generiranje i provjeru pariteta. Kombinacijom spomenutih vrata moguće je provjeriti i generirati parni i neparni paritet. [12]



Slika 19: Provjera neparnoga pariteta pomoću isključivo-NILI-vrata [12]

3. Osnovni kombinacijski sklopi

3.1 Dekoder 2/4

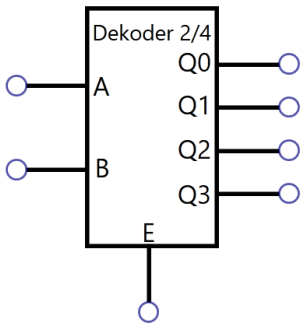
Dekoder je kombinacijski sklop koji dekodira podatke na *n* ulaza u najviše 2^n izlaza. Dekoder 2/4 sastoji se od dvije ulazne linije i 4 izlazne linije. U svakom trenutku samo jedan izlaz je aktivan dok drugi izlazi zadržavaju logičku ništicu. Osim navedenih ulaza dekoder sadrži i ulaz omogući (engl. *Enable)*. U slučaju da je njegova vrijednost 0 svi izlazi će biti 0, a u drugom slučaju izlazi će biti određeni prema tablici istine prikazanoj na slici 20. [14]

OMOGUĆI	ULAZI		IZLAZI			
E	**A_1**	**A_0**	**Y_3**	**Y_2**	**Y_1**	**Y_0**
0	x	x	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

Slika 20: Tablica istine dekodera 2/4

Poraba:

Jedna od poraba dekodera je dobivanje dekadske znamenke iz binarnih brojeva. Binarnom broju potrebne su 4 znamenke kako bi predstavio dekadske znamenke 0-9. Iz toga razloga se dekoder za pretvorbu sastoji od 4 ulazne linije i 10 izlaznih linija koje predstavljaju dekadske znamenke.



Slika 21: Izgled dekodera 2/4

3.2 Koder 4/2

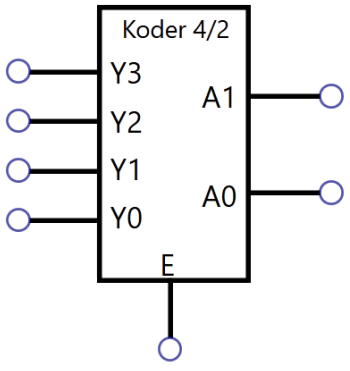
Koder je kombinatorna logička funkcija koja ima 2^n ulaznih linija i *n* izlaznih linija. Izlazne linije generiraju binarni kod za mogućih 2^n ulaznih linija. [12]

ULAZI				IZLAZI	
Y_3	Y_2	Y_1	Y_0	A_1	A_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Slika 22: Tablica istine za koder 4/2 [15]

Poraba:

Koderi imaju brojne porabe. Jedna od njih je smanjivanje broja žica potrebnih u određenom logičkom sklopu. Primjerice, pretpostavimo da mikroprocesor treba pročitati 104 tipke standardne QWERTY tipkovnice gdje u svakom trenutku najviše jedna tipka može biti pritisnuta. Jedan način za realizaciju jest da se žicama poveže svaka od 104 tipke direktno na ulaze računala. Takvo rješenje je vrlo nepraktično. Porabom dekodera svaku tipku moguće je kodirati u standardni ASCII-kod od 7 bitova (0-127) koji bi predstavljao svako slovo tipkovnice. Na taj način umjesto 104 veze na računalo bilo bi potrebno ostvariti samo 7 veza.



Slika 23: Izgled kodera 4/2

3.3 Poluzbrajalo

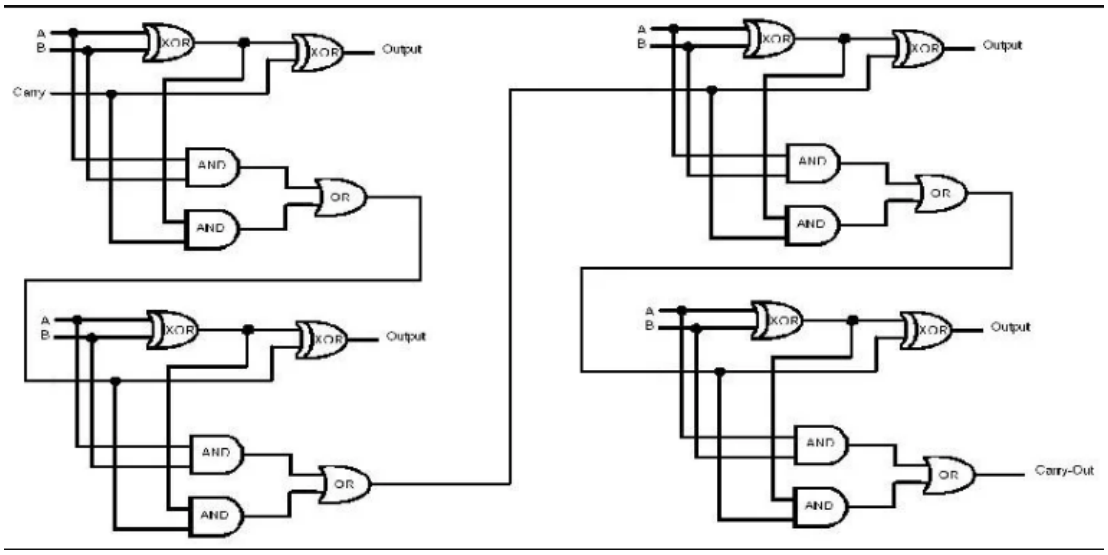
Poluzbrajalo je aritmetički logički sklop rabljen za zbrajanje dva bita. Takav sklop ima dva ulaza koji predstavljaju bitove za zbrajanje te dva izlaza, jedan koji daje zbroj i drugi koji daje prijenos.

ULAZI		IZLAZI	
A	B	C (Carry)	S (Sum)
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Slika 24: Tablica istine poluzbrajala [16]

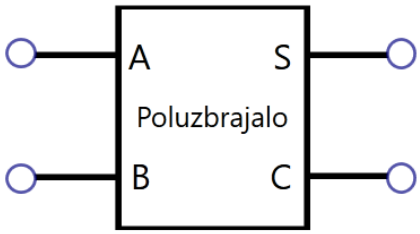
Poraba:

Poluzbrajalo je jedan od osnovnih gradivnih blokova digitalne elektronike. Ima brojne primjene u suvremenim digitalnim elektroničkim sustavima, a najčešće su integracija u kalkulatorima, procesorima i aritmetičko-logičkim jedinicama. Osim toga rabe se za izgradnju aritmetičkih blokova više razine poput potpunih zbrajala, *substractora, multipliera, dividera* itd.



Slika 25: Shema 4-bitnoga kalkulatora [17]

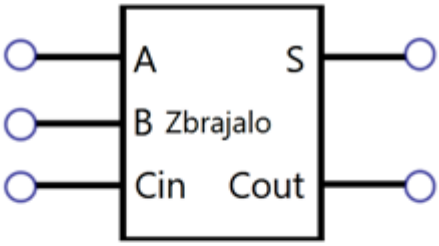
Na slici 25. prikazana je shema 4-bitnoga kalkulatora izgrađenoga rabeći poluzbrajala. Prema istom principu moguće je izgraditi i *N*-bitna zbrajala rabeći istu logiku.



Slika 26: Izgled poluzbrajala

3.4 Zbrajalo

Zbrajalo je aritmetički logički sklop koji zbraja 3 bita, producira zbroj i prijenos. Neophodan je kod zbrajanja binarnih brojeva s velikom količinom bitova. Savladava ograničenja prethodno opisanoga poluzbrajala, rabljenoga za zbrajanje 2 bita. Unatoč tomu jedno potpuno zbrajalo sposobno je zbrajati samo brojeve veličine jednoga bita. Kaskadni uređaj navedenih zbrajala može biti rabljeno za izgradnju zbrajala sposobnoga za zbrajanje binarnih brojeva s većom količinom bitova. [12]

SIMBOL	TABLICA ISTINE				
	C-in	B	A	Sum	C-out
	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

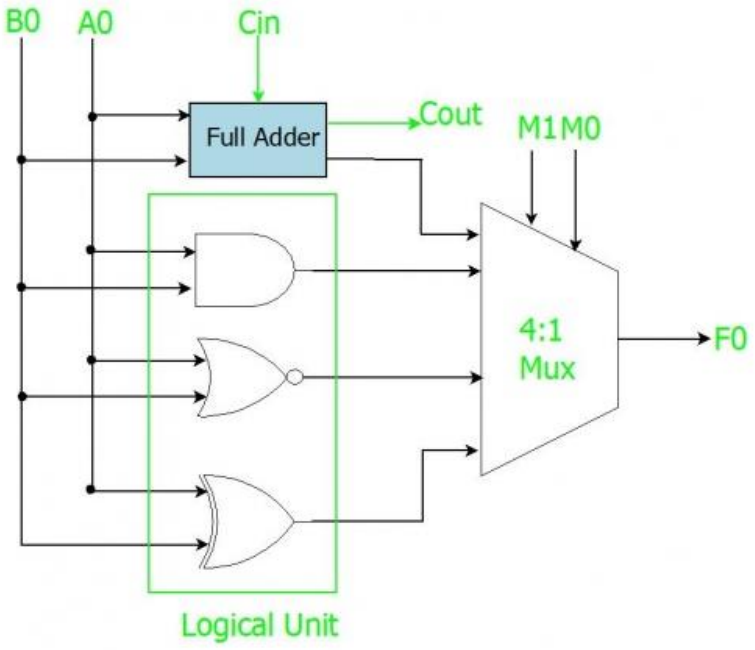
Slika 27: Izgled i tablica istine potpunoga zbrajala [18]

Poraba:

Potpuno zbrajalo rabi se kao dio većih logičkih sklopova kao što su:

- Zbrajalo širećega prijenosa *(*engl. *ripple carry adder)*
- Aritmetičko-logička jedinica
- Generiranje memorijske adrese unutar računala
- Pokazivanje programskoga brojila na sljedeću instrukciju
- Kompleksne izračune kod grafički orijentiranih aplikacija
  [19]

Prethodno je navedeno da se potpuno zbrajalo rabi kod izrade aritmetičko-logičkih jedinica. Rabeći jedno potpuno zbrajalo u kombinaciji s ostalim logičkim vratima moguće je izgraditi jednobitnu ALU sposobnu za zbrajanje jednoga bita i 3 logičke operacije I, NILI i isključivo-ILI. Shematski prikaz takve aritmetičko-logičke jedinice prikazan je na slici 28. U takvu sklopu potpuno zbrajalo izvodi aritmetičku operaciju zbrajanja jednoga bita.



Slika 28: 1-bitna ALU [20]

3.5 Demultipleksor 1/4

Demultipleksor je kombinacijski logički sklop s jednom ulaznom linijom, 2^n izlaznih linija i *n* linija odabira. Njegova zadaća jest usmjeravanje informacija s ulazne linije na neku od izlaznih linija. Linija na koju će informacije biti usmjerene odlučena je statusom bita linija odabira te je prethodno opisani dekoder posebni slučaj demultipleksora bez ulazne linije. [12]

LINIJE ODABIRA		IZLAZI			
S_1	S_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	I
0	1	0	0	I	0
1	0	0	I	0	0
1	1	I	0	0	0

Slika 29: Tablica istina za demultipleksor 1/4 [21]

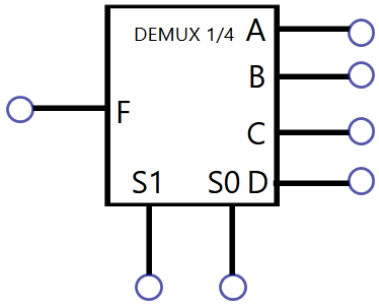
Poraba:

Demultipleksori se rabe u nekoliko područja gdje je neophodno spajanje jedne ulazne linije na nekoliko odredišta. Neke od najistaknutijih poraba su:

- Komunikacijski sustavi
- Aritmetičko-logičke jedinice
- Serijski u paralelni konverter

[22]

Komunikacijski sustavi rabe demultipleksore za provođenje procesa prijenosa podataka. Demultipleksor prima izlazne signale od multipleksora i na prijamniku ih pretvara u izvorni oblik.



Slika 30: Izgled demultipleksora 1/4

3.6 Multipleksor 4/1

Multipleksor je kombinacijski logički sklop s više ulaznih linija, jednom izlaznom linijom i više linija odabira. Odabire binarnu informaciju s ulaznih linija te ovisno o logičkom statusu linija odabira preusmjerava informaciju na izlazne linije. Ako postoji *n* ulaznih linija maksimalni broj izlaznih linija je 2^n iz čega multipleksor i dobiva naziv [12]. Kao i kod prethodno opisanih sklopova u ovom poglavlju, porabom osnovnoga multipleksora 4/1 moguće je ostvariti veći multipleksor. Primjerice za oblikovanje multipleksora 16/4 potrebna su 4 paralelno spojena osnovna multipleksora čiji su izlazi spojeni na još jedan osnovni 4/1-multipleksor.

LINIJE ODABIRA		IZLAZI
S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

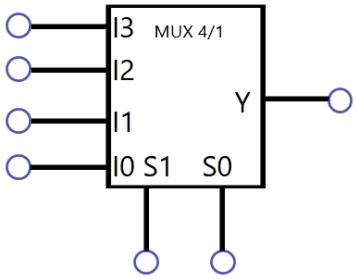
Slika 31: Tablica istine za multipleksor 4/1 [23]

Poraba:

Multipleksori se rabe u raznim aplikacijama gdje se više podataka može prenositi rabeći samo jednu liniju. Neke od najčešćih primjena su:

- Komunikacijski sustavi
- Računalna memorija
- Telefonske mreže

U računalnoj memoriji multipleksori se rabe za zadržavanje velike količine memorije u računalu, ali i za smanjenje broja bakrenih vodova potrebnih za povezivanje memorije s ostalim dijelovima računala. [24]

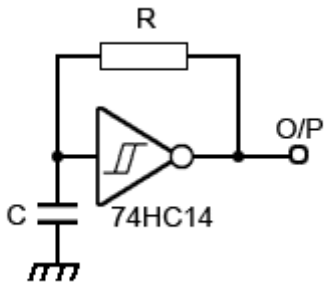


Slika 32: Izgled multipleksora 4/1

4. Dodatni logički sklopi

4.1 Generator signala takta

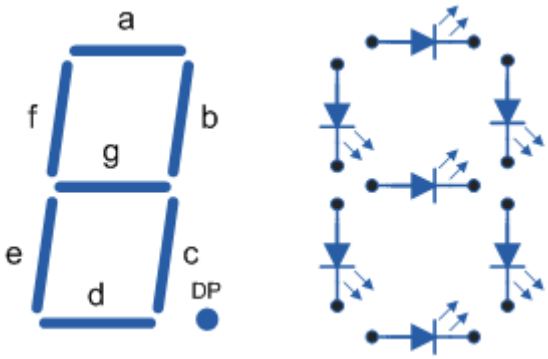
Generator signala takta je elektronički logički sklop koji proizvodi periodične, titrajuće signale koji služe za sinkronizaciju logičkih sklopova. Oscilira između „visokoga“ i „niskoga“ stanje te služi kao metronom koji koordinira radnje. Većina sekvencijskih sklopova opisanih u sljedećem poglavlju upravljani su upravo signalima takta, te su neizostavan dio svakoga modernoga digitalnoga sklopa [40].



Slika 33: Jednostavni oscilator [40]

4.2 Digitalni dekoder

Digitalni dekoder logički je sklop koji pretvara jedno digitalno obličje u drugo. Najčešći njegov ostvaraj je 7-segmentni indikator koji služi za prikazivanje stanja digitalnoga instrumenta u brojkama, slovima ili alfanumeričkim znakovima. Navedeni sklop sastoji se od sedam pojedinačno obojenih LED segmenata. Da bi se moglo upravljati pojedinim segmentom potrebno je svakomu dodijeliti oznaku [41].



Slika 34: Oznaka segmenata [41]

Primjerice kako bi prikazali broj 4 potrebno je osvijetliti segmente *a, b, c, d i g.*

5. Osnovni sekvencijski sklopi

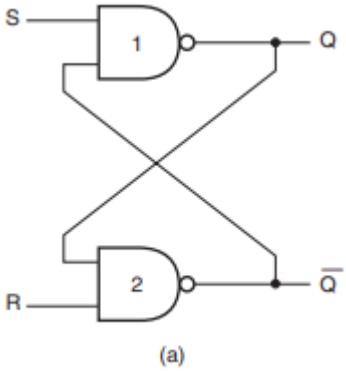
Bistabili su logički sklopovi sa dva stabilna stanja koji mogu biti rabljeni za pohranu binarnoga podataka (jedan bit). Pohranjeni podatci mijenjaju se promjenom ulaza. Temeljni su građevni blokovi sekvencijske logike, rabljeni u modernim računalima, komunikaciji i brojnim drugim sustavima. Najčešće se rabe kao elementi za pohranu podataka [25]. Jedno od dva moguća stanja predstavlja „jedinicu“, a drugo „ništicu“. Takvo pohranjivanje može biti rabljeno za pohranu stanja, te su upravo takvi sklopovi opisani kao sekvencijska logika u elektronici. Zbog svoje važnosti bit će opisana 4 tipa bistabila:

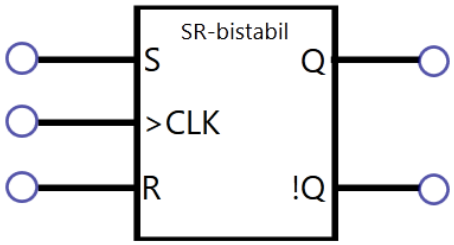
- JK-bistabil
- SR-bistabil
- D-bistabil
- T-bistabil

5.1 SR-bistabil

Oba stanja bistabila su stabilna. Sklop ostaje u određenom stanju sve dok nešto ne promijeni status izlaza. Prethodno je navedeno da bistabili imaju dva stanja „0“ i „1“, s obzirom da bilo koje od ta dva stanja može biti zadržano neodređeno dugo govorimo da sklop ima memoriju. Sposoban je pohraniti binarnu znamenku tj. jedan bit digitalne informacije. Izlazi su komplementi, te se najčešće obilježavaju s Q i zanijekano Q. [12]

SR-bistabil je najosnovniji od svih. Slova “R” i “S” predstavljaju *“Reset” i “Set”* tj. “Poništi” i “Postavi”. Kada je sklop u “Postavi” stanju izlaz mu je “1”, a kada je u “Poništi” stanju izlaz mu je 0. Izlaz zanijekano Q je u svakom trenutku komplement izlaza Q. Glavni nedostatak ovoga sklopa je neodređeno stanje kada su oba ulaza postavljena na logičku jedinicu. Zbog toga je nastao JK-bistabil koji rješava navedeni problem te se iz toga razloga on rabi umjesto SR-bistabila.





Slika 35: Izgled SR-bistabila

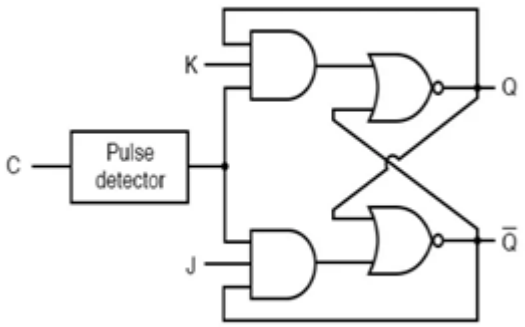
Slika 36: Ostvaraj SR-bistabila [12]

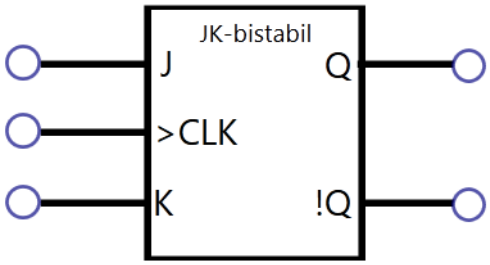
S	R	Q	Q'
0	0	0	1
0	1	0	1
1	0	1	0
1	1	∞	∞

Slika 37: Tablica istine za SR-bistabil [25]

5.2 JK-bistabil

JK-bistabil u osnovi je preinaka prethodno opisanoga SR-bistabila s bitnom razlikom da za razliku od njega nema nepravilnih stanja (stanje kada su oba ulaza pozitivna). To je postignuto tako da su I-vrata s dva ulaza zamijenjena I-vratima s tri ulaza gdje treći ulaz prima povratne informacije od Q i ne-Q izlaza. Na taj način J-ulaz je aktivan samo kada je sklop u stanju *reset*, dok je K-ulaz aktivan kada je bistabil u stanju *set*. [26]





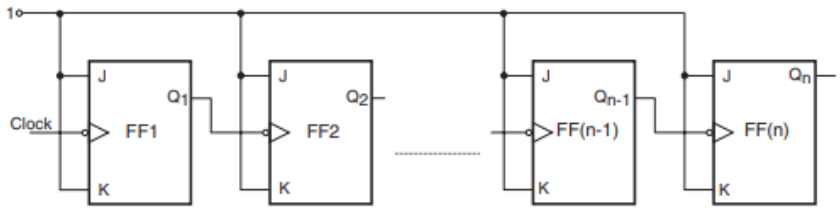
Slika 38: Izgled JK-bistabila

Slika 39:JK-bistabil ostvaraj [17]

Zbog navedenih značajka porabom samo JK-bistabila moguće je izgraditi bilo koji drugi bistabil te se zbog toga još naziva i univerzalni bistabil.

Poraba:

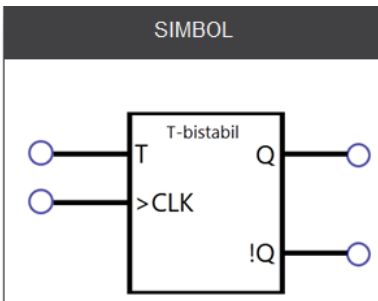
Šireće brojilo (*engl. ripple counter)* je kaskadno uređenje bistabila gdje izlaz jednoga pokreće *Enable*-ulaz drugoga, a samo prvi bistabil ima *Enable*-ulaz spojen na vanjski sat [12]. Ovakav sklop se rabi kao osnova za izgradnju složenijih brojila i moguće ga je izgraditi porabom JK-bistabila.



Slika 40: Shematski prikaz n-bitnoga binarnoga asinkronoga brojila [12]

5.3 T-bistabil

Izlaz T-bistabila mijenja stanje svaki put kada je aktiviran na ulazu, izlaz postaje 1 ako je bio 0 i 0 ako je bio 1. Zbog takvoga načina rada je i dobio naziv *Toggle* (prebaciti).

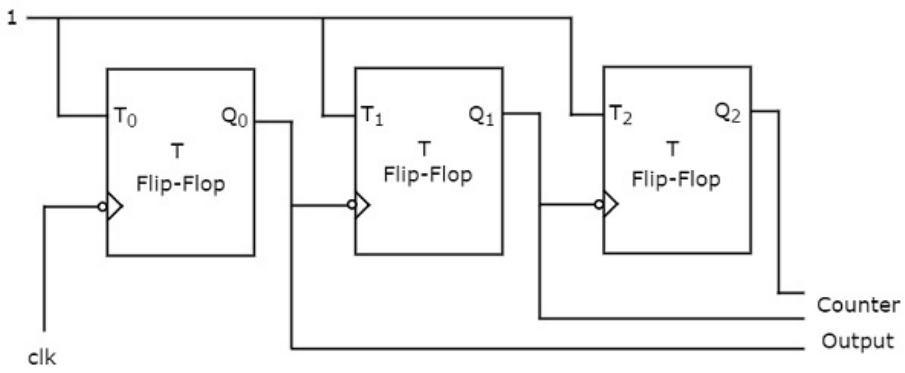


SIMBOL	ULAZI	TRENUTNO STANJE	SLJED. STANJE
	T	Q t	Q $t+1$
	0	0	0
	0	1	1
	1	0	1
	1	1	0

Slika 41: Izgled i tablica istine T-bistabila

Poraba:

N-bitno binarno brojilo sastoji se od *n* T-bistabila. 3-bitno asinkrono brojilo sastoji se od 3 T-bistabila i ulazi svih bistabila su spojeni na logičku jedinicu. Vanjski takt sata je spojen samo na prvi T-bistabil.



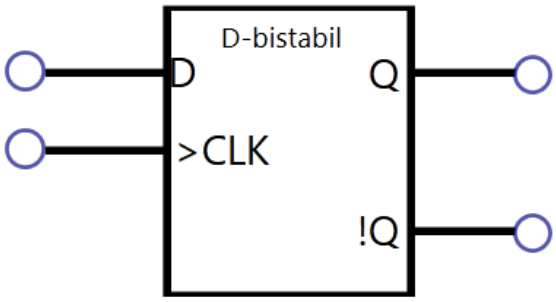
Slika 42: Shematski prikaz 3-bitnoga binarnoga zbrajala [27]

5.4 D-bistabil

Glavni nedostatak SR-bistabila je neodređeno stanje kada su oba ulaza postavljena na logičku jedinicu. Spajanje invertora između ulaza rješava navedeni problem te se taj preinačeni SR-bistabil naziva D-bistabil koji je najvažniji od svih bistabila koji imaju dodatni ulaz na koji se dovode sinkronizacijski impulsi konstantne frekvencije tako da bistabil mijenja stanje na promjenu okidnoga impulsa (sinkroni bistabili) [28]. Navedeni dodatni ulaz naziva se omogući (engl. *enable*) te se ulazna vrijednost preslikava na izlaz samo kada je on aktivan. Zbog toga se D-bistabil može rabiti kao privremena memorija za informaciju veličine jednoga bita [12].

Clk	D	Q		Description
↓ » 0	X	Q	$\overline{Q}$	Memory no change
↑ » 1	0	0	1	Reset Q » 0
↑ » 1	1	1	0	Set Q » 1

Slika 44: Tablica istine D-bistabila

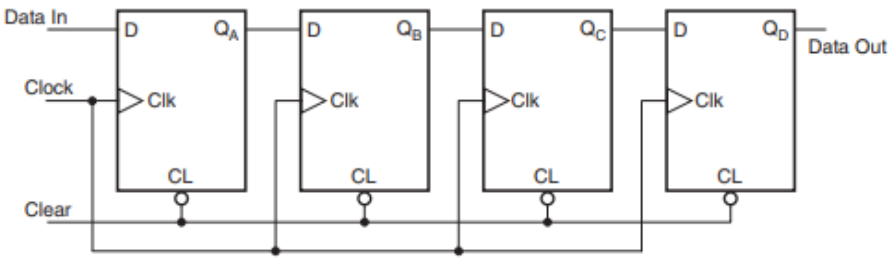


Slika 43: Izgled D-bistabila

Poraba:

Posmačni registar je sklop koji podatke nazočne na ulazu posmiče na izlaz na svaki takt sata. Takav sklop se sastoji od nekoliko D-bistabila gdje je svaki zadužen za jedan bit podatka. Broj D-bistabila potrebnih za izgradnju posmačnoga registra ovisi o broju pohranjenih bitova. Posmačni registri rabe se za pohranu i premještanje podataka te zbog toga imaju široku

primjenu u kalkulatorima i računalima. Primjerice za pohranu binarnih brojeva prije nego su zbrojeni. [29]



Slika 45: Shematski prikaz 4-bitnoga posmačnoga registra [12]

6. Programsko oruđe LG-simulator

Nakon pokretanja programa otvara se intuitivno korisničko sučelje. Njegovi dijelovi obilježeni su na slici 46.

Slika 46: Korisničko sučelje LG-simulatora

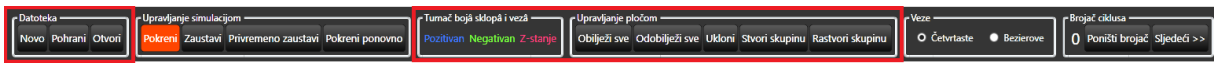
Pojedinačni elementi sučelja obilježeni su pravokutnicima:

1. Glavni izbornik
2. Radna površina (tiskana ploča)
3. Prostor sa sklopovima

6.1 Glavni izbornik

Glavni izbornik jednostavnoga je oblikovanja poznatoga svim korisnicima modernih računala. Iako fizički nije podijeljen njegove funkcionalnosti možemo svrstati u:

1. Funkcionalnosti za upravljanje datotekama i sučeljem
2. Funkcionalnosti za simulaciju



Slika 47: Podjela funkcionalnosti glavnoga izbornika

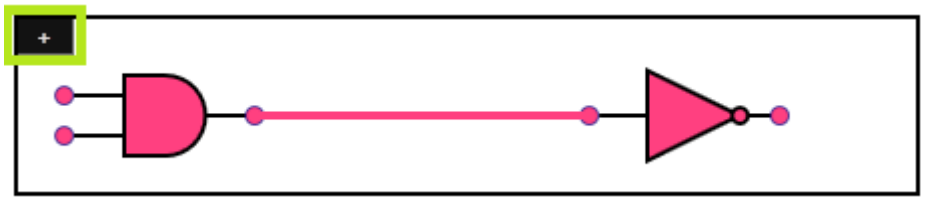
Funkcionalnostima za upravljanje datotekama i sučeljem pripadaju dijelovi glavnoga izbornika obilježeni crvenim pravokutnicima, dok potonjoj skupini pripadaju neobilježeni dijelovi sučelja.

Glavni izbornik sadrži osnovne funkcionalnosti pohrane i otvaranja pohranjenih datoteka. Datoteke se pohranjuju u deklarativnom jeziku XAML, zbog toga zauzimaju vrlo malo prostora te su dostupne na svim verzijama operacijskoga sustava Windows. Što omogućuje svim korisnicima da pohrane trenutnu simulaciju kako bi spriječili gubitak podataka.

```
<?xml version="1.0" encoding="utf-8"?>
<Root>
  <LogicGates>
    <LogicGate>
      <Id>0</Id>
      <Left>224.75</Left>
      <Top>283.31272727272722</Top>
      <Width>150</Width>
      <Height>150</Height>
      <zIndex>0</zIndex>
      <Type>EncoderFourTwo</Type>
      <IsZstate>true</IsZstate>
    </LogicGate>
  </LogicGates>
  <Connections />
</Root>
```

Slika 48: Primjer pohranjene datoteke

Osim navedenih pod osnovne funkcionalnosti možemo svrstati i funkcionalnosti za upravljanje pločom koje korisniku omogućuju uklanjanje i stvaranje skupine logičkih sklopova. Stvaranje skupine sklopova omogućuje veću preglednost ploče i organizaciju sklopova u logičke cjeline. Za izvršenje stvaranja skupine potrebno je odabrati željene sklopove te nakon odabira pritisnuti tipku “Stvori skupinu”. Odabirom skupine i pritiskom na tipku “Rastvori skupinu” poništava se skupina.



Slika 49: Izgled maksimizirane grupe

Stvorenu skupinu moguće je minimizirati i maksimizirati pritiskom na tipku u gornjem lijevom kutu skupine. Minimiziranoj skupini moguće je dodati naziv kako bi korisnik u svakom trenutku znao o kojoj skupini je riječ.

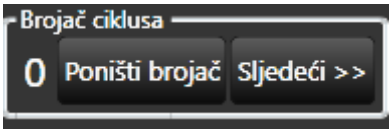
Slika 50: Izgled minimizirane skupine

U glavnom izborniku još se nalaze i tipke za upravljanje ciklusima. Na desnoj strani glavnoga izbornika nalazi se brojač ciklusa koji ima 3 funkcionalnosti:

- Drži podatak o broju izvedenih ciklusa
- Prelazi na sljedeći ciklus
- Poništava brojač ciklusa na 0

Ciklus predstavlja vremensku jedinicu T. U 0T i parnim ciklusima signali takta koji se nalaze na radnoj ploči imaju pozitivan izlaz, dok u neparnim ciklusima signali takta imaju negativan izlaz. Navedena funkcionalnost omogućuje korisniku simuliranje rada logičkih sklopova kroz vremenske jedinice. Glavno sučelje sadrži i dodatne tipke za upravljanje frekvencijama sata:

- Pokreni → pokreće sat te se svaku sekundu izvršava jedan ciklus
- Zaustavi → poništava brojač i zaustavlja sat
- Privremeno zaustavi → zaustavlja sat i ne poništava brojač
- Pokreni ponovni → poništava brojač i ponovno ga pokreće



Slika 51: Brojač ciklusa

6.2 Prostor sa sklopovima

Prostor sa sklopovima sadrži logičke sklopove raspoređene po skupinama kojima pripadaju:

- Osnovni logički sklopovi
- Kombinatorni logički sklopovi
- Dodatni logički sklopovi
- Sekvencijski logički sklopovi

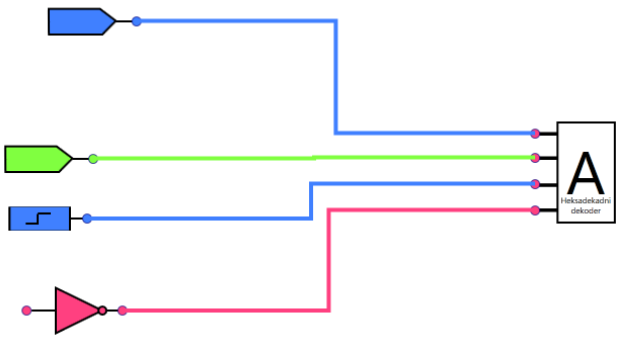
Prelaskom mišem preko sklopa ispisuje se njegov naziv, a za dodavanje sklopa na tiskanu ploču potrebno je pritisnuti lijevom tipkom miša na njega te ga odvući na tiskanu ploču na željeno mjesto.

6.3 Tiskana ploča

Na tiskanoj ploči korisnik modelira logičke sklopove rabeći sve prethodno navedene funkcionalnosti programa. Logički sklopovi dodaju se povlačenjem s „prostora za sklopove“. Tako dodani sklop ispunjen je ružičastom bojom koja predstavlja stanje kada sklop nema žice spojene na sve ulaze. U tom slučaju izlaz sklopa je nepoznat i to stanje naziva se stanje Z. Osim navedenoga postoje još dva stanja:

- Pozitivno → predstavljeno modrom bojom
- Negativno → predstavljeno zelenom bojom

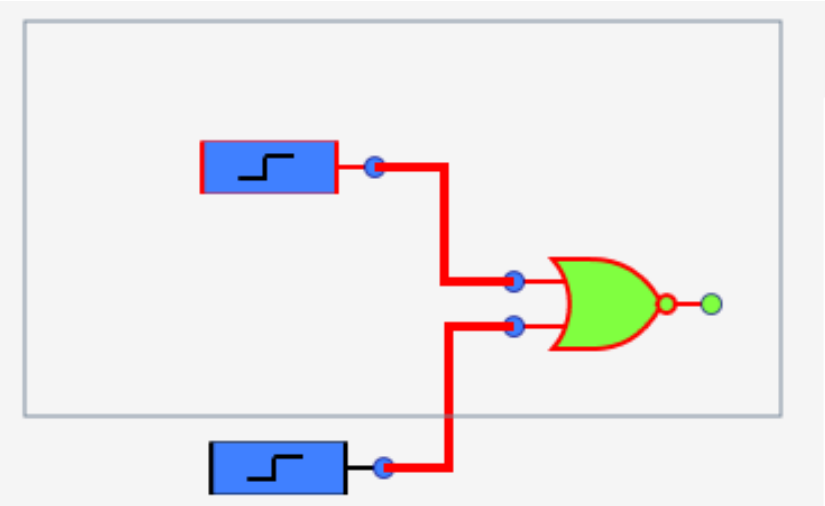
Izuzev samih sklopova i veze koje dolaze iz njihovih izlaza obojene su istom bojom kao i oni i predstavljaju ista prethodno navedena stanja. Iznimke ovom načinu predstavljanja stanja su sklopovi koji nemaju izlaze ili koji imaju više izlaza. Takvi sklopovi će u svakom trenutku biti ispunjeni bijelom bojom.



Slika 52: Sklopovi u različitim stanjima

Navigacija radnom površinom omogućena je porabom pomične vrpce na osi x i y, te klizača za *zoom*. U skladu sa svim ostalim funkcionalnostima u sučelju za olakšanu navigaciju radnom vrpcom omogućeni su i tipkovnički prečaci.

Sklope na tiskanoj ploči moguće je obilježiti pritiskom lijeve tipke miša na željeni sklop, pritiskom lijeve tipke miša na tiskanu ploču i povlačenjem iscrtanoga pravokutnika preko željenih sklopa ili pak korištenjem tipkovničkoga prečaca Ctrl+A. Obilježeni sklopi iscrtani su crvenom bojom, te su i sve njihove ulazne i izlazne veze također obilježene.



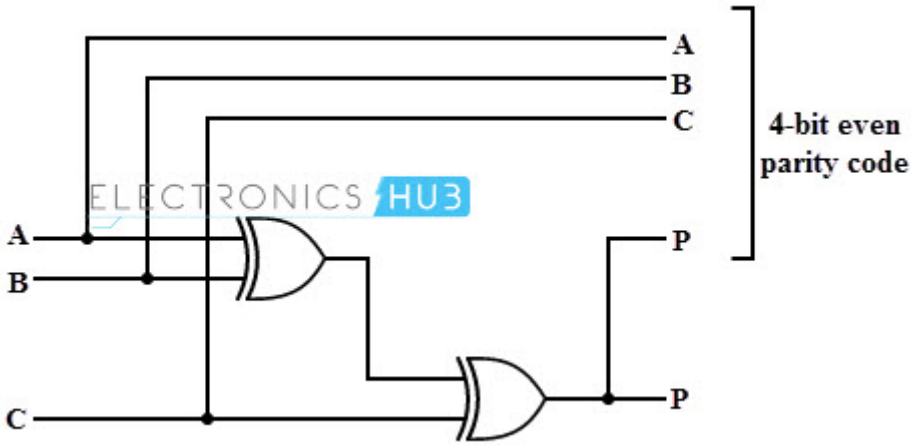
Slika 53: Obilježavanje sklopa

7. Simulacije

U ovom poglavlju bit će simuliran rad logičkih sklopova navedenih u prethodnim poglavljima. Simulacije će biti provedene porabom LG-simulatora, a bit će simulirani odabrani prethodno opisani kombinacijski i sekvencijski sklopovi.

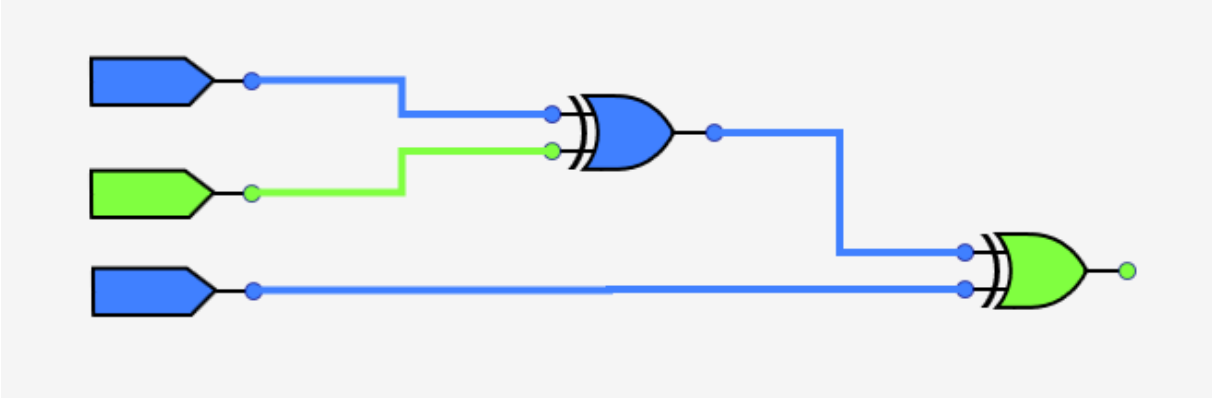
7.1 Generator parnoga pariteta

Generator pariteta kombinacijski je sklop koji prima *n* ulaznih podataka i generira dodatni bit koji treba biti prenesen s ulaznim podatcima. Taj dodatni bit naziva se bit pariteta. Kod sheme parnoga pariteta, bit pariteta jednak je „1“, ako postoji paran broj logičkih jedinica u ulaznim podatcima, a ako postoji neparni broj logičkih jedinica na ulazima onda je bit pariteta jednak 0 [30].



Slika 54: Shematski prikaz generatora parnoga pariteta [30]

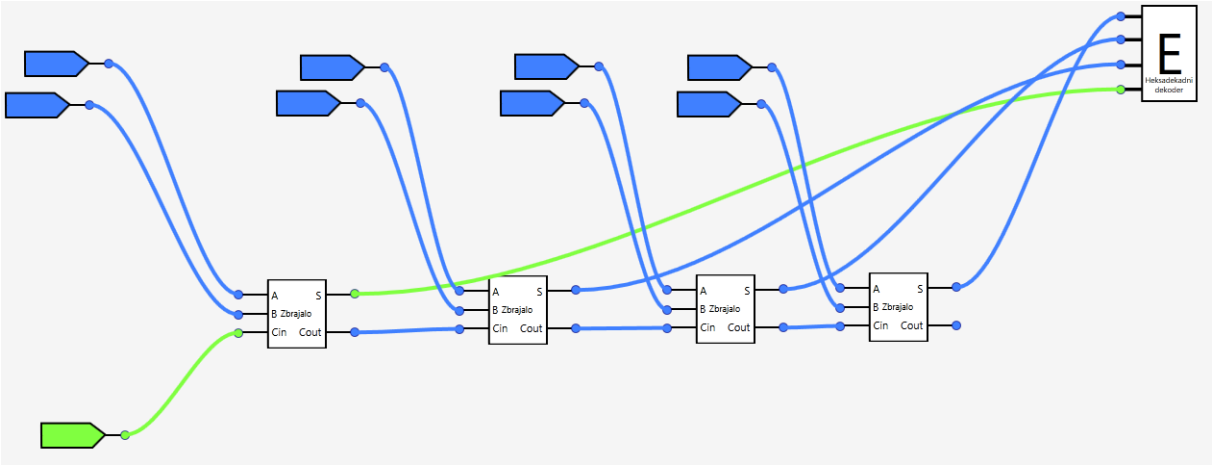
Porabom 2 isključivo-ILI-vrata izgrađen je 4-bitni generator bita pariteta prikazan na slici 54.



Slika 55: Generator parnoga pariteta u LG-simulatoru

7.2 4-bitno binarno zbrajalo

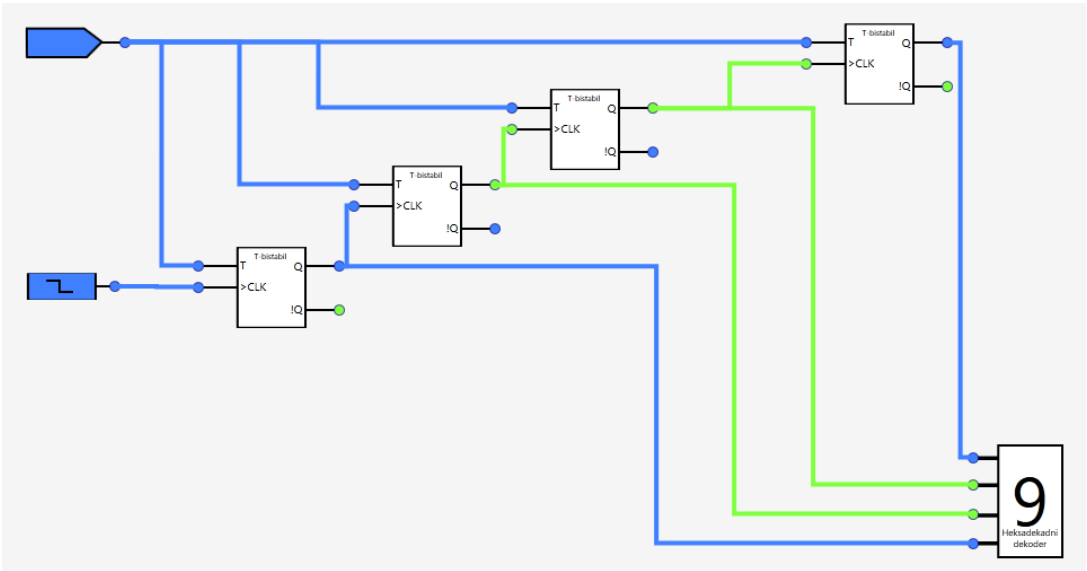
Na slici 56 prikazano je 4-bitno zbrajalo izgrađeno porabom 4 potpuna zbrajala. Prvi binarni broj su ulazi A0-A3, a drugi binarni broj su ulazi B0-B3. Izlazi su spojeni na heksadekadni dekoder kako bi se vidio zbroj danih binarnih brojeva.



Slika 56: 4-bitno binarno zbrajalo u LG-simulatoru

7.3 4-bitno binarno brojilo

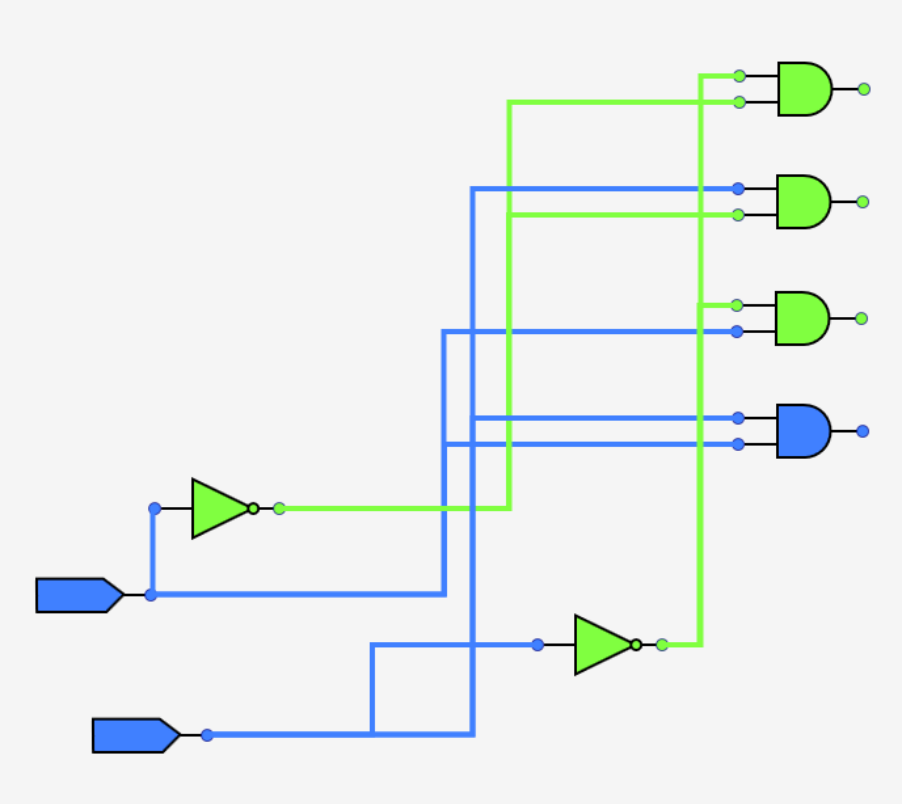
Brojila su sekvencijski sklopovi rabljeni za brojenje signala te su najčešća primjena bistabila. Na slici 57 prikazano je 4-bitno asinkrono brojilo izgrađeno porabom T-bistabila.



Slika 57: 4-bitno brojilo u LG-simulatoru

7.4 Dekoder

Dekoder 2/4 moguće je konstruirati rabeći 4 vrata „I“ te 2 vrata „NE“, na slici 58 prikazana je njegova konstrukcija porabom navedenih vrata.



Slika 58: Dekoder 2/4

8. Zaključak

Unutar modernoga računala nalaze se milijuni sklopova kroz koje informacije prolaze prema definiranim protokolima. Takva računala su temelj digitalne ere u kojoj se nalazimo, zbog toga je poznavanje njihova rada neophodno za svakoga pojedinca koji se služi informacijsko-komunikacijskim tehnologijama.

Računala su vrlo složeni strojevi, unatoč njihovoj raširenosti razumijevanje načina njihova rada nepoznanica je većini ljudi. Upravo je demistificiranje osnovnih principa djelovanja modernih računala glavna motivacija za pisanje ovoga rada. U prethodnim poglavljima opisani su najrabljeniji osnovni, kombinacijski i sekvencijski logički sklopovi koji tvore osnovu svakoga računala. Prikazana je njihova poraba u suvremenim digitalnim elektroničkim sustavim te je razvijeno radno okruženje za grafičko slaganje kombinacija opisanih logičkih sklopova.

Rast industrije informacijskih tehnologija je nezaustavljiv, a potreba za informatičkim stručnjacima je sve veća. Porabom razvijene aplikacije moguće je na jednostavan način prikazati rad modernih računala na najnižoj razini, a grafički prikaz djelovanja osnovnih komponenti olakšava njihovo razumijevanje, te učenje čini zabavnijim i bržim.

Bibliografija

[1] B. L. Koriolan Gilezan, Bulova algebra i primene, Beograd: Matematički institut, 1977..

[2] C. Shannon, »computerscience.jbpub.com,« 1. 7. 2003.. [Na internetu]. Dostupno: http://computerscience.jbpub.com/ecoa/2e/Null03.pdf [pristupano 19.08.2020.]

[3] S. K. Sarkar , A. K. De i S. Sarkar, Foundation of Digital Electronics and Logic Design, Jenny Stanford Publishing, 2014.

[4] P. Horowi tz i H. Winfield, The Art Of Electronics - 2nd Edition, New York: Press Syndicate of the University of Cambridge, 1989.

[5] A. L. Robinson, »One billion transistors on a chip?; the annual rate of increase in numbers of transistors on a chip is slowing as theoretical limits are neared, but there may be a billion by the year 2000,« *Science(Vol. 223),* p. 3, 1984 1 20.

[6] »Mehanički logički sklop [Slika] (bez dat.),« [Na internetu]. Dostupno: https://mechalogic.wordpress.com/2013/01/26/43/. [Pristupano 1 8 2020].

[7] sterchie, »www.scribd.com,« 27. 8. 2020.. [Na internetu]. Dostupno: https://www.scribd.com/doc/24926600/HISTORY-the-Earliest-Logic-Gates-Were-Made?fbclid=IwAR0qbidvFnJmTBejmfe_PUoKYP1VyqXZbEZ6JZ26iJPm9xZeNrpk87IClqo [pristupano 26.08.2020.].

[8] Nexperia, »allaboutcircuits.com,« 27. 8. 2020.. [Na internetu]. Dostupno: https://www.allaboutcircuits.com/industry-articles/from-ancient-binary-to-silicon-chips-logic-through-history/ [pristupano 10.08.2020.].

[9] »Broj tranzistora u ICC kroz vrijeme [Slika] (bez dat.),« [Na internetu]. Dostupno: https://en.wikipedia.org/wiki/Transistor_count. [Pristupano 1.8.2020].

[10] A. Hamzić, »Osnove poluvodičke digitalne elektronike,« Fizički odsjek, Prirodoslovno-matematički fakultet, Zagreb, 2010.

[11] »Shematski prikaz vrata Enable/Inhibit [Slika] (bez dat.),« [Na internetu]. Dostupno: https://www.instructables.com/id/AND-Gate-for-Logic-Circuit-Applications/. [Pristupano 1.9.2020].

[12] A. K. Maini, »Digital Electronics Principles, Devices and Applications,« John Wiley & Sons, West Sussex, 2007.

[13] D. Lowe, »Electronics Logic Gates: XOR and XNOR Gates,« [Na internetu]. Dostupno: https://www.dummies.com/programming/electronics/components/electronics-logic-gates-xor-and-xnor-gates/ [pristupano 29.08.2020.].

[14] Administrator, »Types of Binary Decoders,Applications,« 4 7 2015. [Na internetu]. Dostupno: https://www.electronicshub.org/binary-decoder/#:~:text=In%20a%202%2Dto%2D4,binary%20inputs%20A%20and%20B.. [pristupano 1.09.2020.]

[15] »Tablica istine za koder 4/2 [Slika] (bez dat.),« [Na internetu]. Dostupno: https://www.tutorialspoint.com/digital_circuits/digital_circuits_encoders.htm. [Pristupano 25.8.2020].

[16] »Tablica istine poluzbrajala [Slika] (bez dat.),« [Na internetu]. Dostupno: https://www.gatevidyalay.com/tag/applications-of-half-adder/#:~:text=Half%20Adder%20is%20a%20combinational,outputs%20(sum%20and%20carry). [Pristupano 20.8.2020].

[17] »Shema 4-bitnoga kalkulator [Slika] (bez dat.),« [Na internetu]. Dostupno: https://www.instructables.com/id/4-Bit-Binary-Adder-Mini-Calculator/. [Pristupano 3.9. 2020].

[18] »Tablica istine potpunoga zbrajala [Slika] (bez dat.),« [Na internetu]. Dostupno: https://www.geeksforgeeks.org/full-adder-in-digital-logic/#:~:text=Full%20Adder%20is%20the%20adder,as%20S%20which%20is%20SUM. [Pristupano 28.8.2020].

[19] H. AG, »What are the applications of a full adder circuit,« 30 11 2016. [Na internetu]. Dostupno: https://www.quora.com/What-are-the-applications-of-a-full-adder-circuit [pristupano 5.08.2020.].

[20] »1-bitna ALU [Slika] (bez dat.),« [Na internetu]. Dostupno: https://exploreembedded.com/wiki/ALU_in_Detail. [Pristupano 23.8.2020].

[21] »Tablica istine za demultipleksor 1/4 [Slika] (bez dat.),« [Na internetu]. Dostupno: http://electronics-course.com/demux. [Pristupano 2.9.2020].

[22] »Demultiplexer: What is it?,« 1 4 2020. [Na internetu]. Dostupno: https://www.electrical4u.com/demultiplexer/ [pristupano 14.08.2020.].

[23] »Tablica istine za multipleksor 4/1 [Slika] (bez dat.),« [Na internetu]. Dostupno: https://www.tutorialspoint.com/digital_circuits/digital_circuits_multiplexers.htm. [Pristupano 1.9.2020].

[24] D. J. Ellison, »Multiplexers/Demultiplexers,« DeVry University.

[25] T. Cafe, »Basics and Overview of Flip Flops,« 17 8 2017. [Na internetu]. Dostupno: https://www.electronicsforu.com/resources/learn-electronics/flip-flop-rs-jk-t-d [pristupano 17.08.2020.].

[26] T. R. Kuphaldt, Lessons in Electric Circuits, Volume IV - Digital, 2007.

[27] »Shematski prikaz 3-bitnoga binarnoga zbrajala [Slika] (bez dat.),« [Na internetu]. Dostupno: https://www.tutorialspoint.com/digital_circuits/digital_circuits_counters.htm. [Pristupano 3.9.2020].

[28] »The D-type Flip flop,« [Na internetu]. Dostupno: https://www.electronics-tutorials.ws/sequential/seq_4.html. [Pristupano 3.9.2020].

[29] »The Shift Register,« [Na internetu]. Dostupno: https://www.electronics-tutorials.ws/sequential/seq_5.html. [Pristupano 1.9.2020].

[30] »Parity Generator and Parity Check,« 7 7 2015. [Na internetu]. Dostupno: https://www.electronicshub.org/parity-generator-and-parity-check/. [Pristupano 6.9. 2020].

[31] »https://www.allaboutcircuits.com/industry-articles/from-ancient-binary-to-silicon-chips-logic-through-history/,« [Na internetu] [pristupano 7.09.2020.].

[32] U. Peruško, Digitalna elektronika, Školska knjiga.

[33] Uroš Peruško, Digitalna elektronika, Školska knjiga, Zagreb, 1991..

[34] V. Krajči, »www.veleri.hr,« 27. 8. 2020.. [Na internetu]. Dostupno: https://www.veleri.hr/files/datotekep/nastavni_materijali/k_telematika_1/Vjezba_0_a.pdf [pristupano 27.08.2020.].

[35] Edrawsoft, »www.edrawmax.com,« 27. 7. 2020.. [Na internetu]. Dostupno: https://www.edrawmax.com/logic-gate/ [pristupano 27.07.2020.].

[36] G. Smiljanić, Osnove digitalnih računala, Zagreb: Školska knjiga, 1990..

[37] »Isključivo-ILI-vrata tablica istine i prikaz [Slika] (bez dat.),« [Na internetu]. Dostupno: https://www.electronics-tutorials.ws/logic/logic_7.html. [Pristupano 29.8.2020].

[38] »Tablica istine i prikaz isključivo-ILI-vrata [Slika] (bez dat.),« [Na internetu]. Dostupno: (https://www.electronics-tutorials.ws/logic/logic_8.html. [Pristupano 2.9.2020].

[39] »Tablica istine T-bistabila [Slika] (bez dat.),« [Na internetu]. Dostupno: https://circuitdigest.com/electronic-circuits/t-flip-flop-truth-table-working#:~:text=02%2C%202017%203-,T%20Flip%2DFlop%3A%20Circuit%2C%20Truth%20Table%20and%20Working,reset%20which%20is%20ultimately%20binary.. [Pristupano 3.9.2020].

[40] Clock circuits [Na internetu].

Dostupno https://learnabout-electronics.org/Digital/dig51.php [pristupano 10.09.2020]

[41] Display Decoder [Na internetu].

Dostupno https://www.electronics-tutorials.ws/combination/comb_6.html

[pristupano 10.09.2020]