
Brojači i konačni automati

Proučavanje sekvencijalnih mreža biće zaokruženo pregledom najvažnijih osobina brojača i konačnih automata. Formalno gledano, iako brojači predstavljaju podklasu konačnih automata, zbog njihovog značaja i raširenosti primene ovde će biti predstavljeni kao zasebna klasa sekvencijalnih mreža. Ono što je zajedničko za sve ove mreže je da se u njihovoj osnovi nalazi registar stanja, sastavljen od bistabilnih elemenata, najčešće flipflopova. Binarni broj koji je formiran stanjima tih bistabilnih elemenata se naziva stanjem kola. Prelazi između pojedinih stanja su određeni kombinacionom logikom, što će biti detaljnije razmatrano u nastavku.

Brojači

Brojači su sekvencijalne mreže čija sekvenca stanja predstavlja repetitivni ciklus. Podela brojača može se vršiti na osnovu više kriterijuma. Po načinu taktovanja flipflopova koji sačinjavaju registar stanja, brojači se dele na:

- **Redne (asinhronne)**, kod kojih se takt signal dovodi samo na prvi flipflop u nizu. Svaki sledeći flipflop je taktovan od strane izlaza prethodnog, što znači da se promena stanja obavlja redno.
- **Paralelne (sinhrone)**, kod kojih se svi flipflopovi okidaju istovremeno (sinhrono), zajedničkim takt signalom.

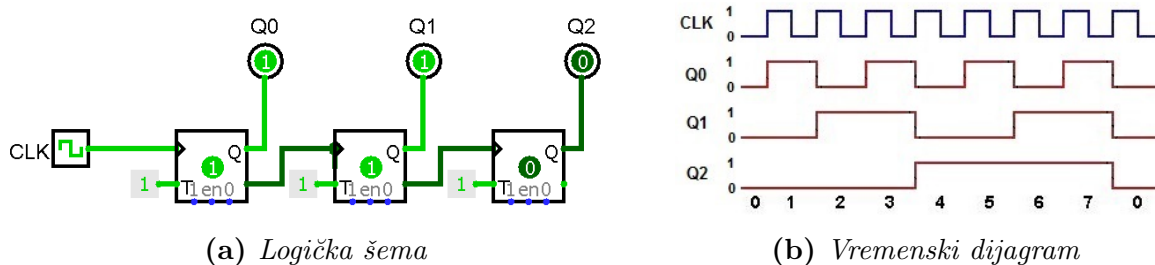
Kada je u pitanju sekvenca stanja u okviru ciklusa brojanja, brojači se dele na:

- **Brojače naviše**, kod kojih je sledeće stanje za 1 veće od prethodnog
- **Brojače naniže**, kod kojih je sledeće stanje za 1 manje od prethodnog
- **Brojače sa proizvoljnom sekvencom stanja**

Ukupan broj stanja u jednom ciklusu brojanja naziva se modul ili osnova brojanja. Pomoću n flipflopova moguće je konstruisati brojač modula $m \leq 2^n$. Ako je $m = 2^n$, kaže se da je ciklus brojanja potpun, a ako je $m < 2^n$, reč je o brojaču sa nepotpunim (skraćenim) ciklusom brojanja.

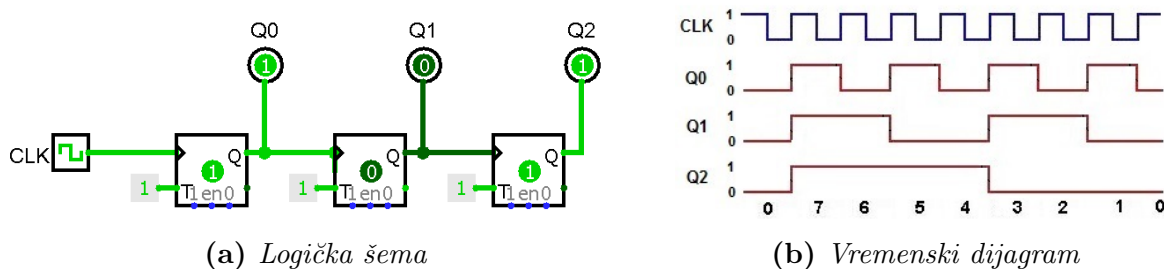
Redni brojači

Redni brojači se najčešće realizuju korišćenjem T-flipflopova, kojima je na T ulaz dovedena logička jedinica, čime se postiže promena stanja svaki put pri nailasku aktivne ivice takta (videti kombinacionu tabelu za T-ff). Na slici 1a prikazana je struktura rednog brojača modula 8. Signal takta se dovodi na prvi flipflop u nizu, čije stanje odgovara najmanje značajnom bitu brojača Q_0 . Vremenski dijagram na slici 1b pokazuje da se promena stanja tog flipflopa dešava na svaku rastuću ivicu takta. Takt ulaz sledećeg flipflopa vezan je na invertovani izlaz prethodnog ($\overline{Q_0}$). To znači da Q_1 menja stanje na rastuću ivicu signala $\overline{Q_0}$, odnosno u trenutku kada se Q_0 menja sa 1 na 0. Ovakva logika je primenjena i na ostale flipflopove u nizu, čime je postignuta logika brojanja naviše.



Slika 1: Redni brojač naviše modula 8

Konstrukcija rednog brojača naniže se razlikuje od brojača naviše po tome što se flipflop k taktuje neinvertvanim izlazom prethodnog flipflopa (Q_{k-1}). Šema ovakvog brojača prikazana je na slici 2a, a vremenski dijagram signala tokom jednog ciklusa brojanja na slici 2b.



Slika 2: Redni brojač naniže modula 8

Posmatranjem vremenskih dijagrama, moguće je uočiti jednu od najvažnijih karakteristika rednih brojača: perioda signala na izlazu svakog flipflopa traje koliko i dve periode signala koji se dovodi na taktni ulaz. Drugim rečima, signal na izlazu ima dvostruko manju frekvenciju od signala koji ga taktuje. Zbog toga, ovakvi brojači se često koriste kao *delitelji frekvencije*: redni brojač sa n flipflopova u nizu na izlazu poslednjeg flipflopa daje signal frekvencije $f = \frac{f_{clk}}{2^n}$, gde je f_{clk} frekvencija takta.

Vremenski dijagrami signala na slikama 1b i 2b prikazuju idealizovan slučaj, u kom su zanemarena kašnjenja propagacije kroz flipflopove. U praksi, pošto se promena stanja prenosi

redom sa jednog na drugi flipflop, ova kašnjenja se akumuliraju i ukupno kašnjenje je proporcionalno broju flipflopova u nizu, što ograničava maksimalnu dozvoljenu frekvenciju rada brojača. Drugi nepoželjan efekat predstavljaju tzv. parazitna stanja koja nastaju usled neistovremene promene stanja na flipflopovima i traju do uspostavljanja ispravnog stanja, u najgorem slučaju nakon promene stanja poslednjeg flipflopa u nizu. Ova ograničenja u radu se prevazilaze upotrebom paralelnih (sinhronih) brojača koji su u opštem slučaju složenije konstrukcije, ali imaju tu prednost da se promene stanja flipflopova odvijaju istovremeno.

Paralelni brojači

Projektovanje paralelnog brojača biće prikazano na primeru brojača unapred modula 8, realizovanog korišćenjem T-flipflopova¹. Za početak, određuje se potreban broj flipflopova da bi brojač zadatog modula mogao da bude realizovan. U ovom slučaju, potrebna su 3 flipflopa, koji će tokom jednog ciklusa brojanja prolaziti kroz ukupno $2^3 = 8$ stanja. Na slici 3a prikazana je tabela stanja/prelaza, koja se popunjava na sledeći način. Za svako stanje brojača $Q_2Q_1Q_0$, prvo se upisuje željeno naredno stanje² $Q'_2Q'_1Q'_0$, na osnovu željenog ponašanja brojača, a to je u ovom slučaju brojanje naviše. Zatim se određuje pobuda koju je potrebno dovesti na ulaze svih flipflopova kako bi bili ostvareni željeni prelazi između stanja, na osnovu eksitacione tabele T-flipflopa prikazane na slici 3b. U svrhu ilustracije ovog postupka, na slici je istaknut primer (polja u tabeli koja su označena žutom bojom): pri prelazu brojača iz stanja 3 u stanje 4, bit Q_2 prelazi iz stanja 0 u stanje 1. Da bi takav prelaz bio ostvaren, u datom slučaju je na ulaz T_2 potrebno dovesti 1, kao što je prikazano u eksitacionoj tabeli.

Trenutno stanje				Pobuda FF			Naredno stanje			
S_i	Q_2	Q_1	Q_0	T_2	T_1	T_0	Q'_2	Q'_1	Q'_0	S'_i
0	0	0	0	0	0	1	0	0	1	1
1	0	0	1	0	1	1	0	1	0	2
2	0	1	0	0	0	1	0	1	1	3
3	0	1	1	1	1	1	1	0	0	4
4	1	0	0	0	0	1	1	0	1	5
5	1	0	1	0	1	1	1	1	0	6
6	1	1	0	0	0	1	1	1	1	7
7	1	1	1	1	1	1	0	0	0	0

(a) Tabela stanja/prelaza

Q	Q'	T
0	0	0
0	1	1
1	0	1
1	1	0

(b) Eksitaciona tabela T-ff

Slika 3: Formiranje tabele stanja/prelaza paralelnog brojača modula 8 realizovanog T-flipflopovima

¹Generalno, pri realizaciji proizvoljnog paralelnog brojača može biti korišćen bilo koji tip flipflopa. Ovde je prikazan samo jedan od mogućih načina realizacije, pomoću T-ff.

²U tekstu posvećenom bistabilnim kolima, bilo je uobičajeno da se trenutno stanje bistabilnog elementa označava sa Q_n , a naredno sa Q_{n+1} . U ovom slučaju, pošto se istovremeno posmatraju stanja više flipflopova, takva notacija bi bila isuviše nezgrapna i mogla bi da dovede do zabune. Naime, u ovom kontekstu Q_n označava trenutno stanje n-tog flipflopa, a ne stanje flipflopa Q u trenutku n. U tu svrhu, ovde se koristi pojednostavljena notacija u kojoj se trenutno stanje n-tog flipflopa označava sa Q_n , a naredno stanje sa Q'_n .

Sledeći korak je određivanje pobude flipfopova. Pobudu obezbeđuje kombinatorna mreža čiji ulazi su promenljive stanja Q_2 , Q_1 i Q_0 , a izlazi se vode na ulaze flipfopova T_2 , T_1 i T_0 . Funkcije pobude se dobijaju minimizacijom, korišćenjem Karnovih tabela:

		Q_1Q_0			
		00	01	11	10
Q_2	0	0	0	1	0
	1	0	0	1	0

(a) $T_2 = Q_1 \cdot Q_0$

		Q_1Q_0			
		00	01	11	10
Q_2	0	0	1	1	0
	1	0	1	1	0

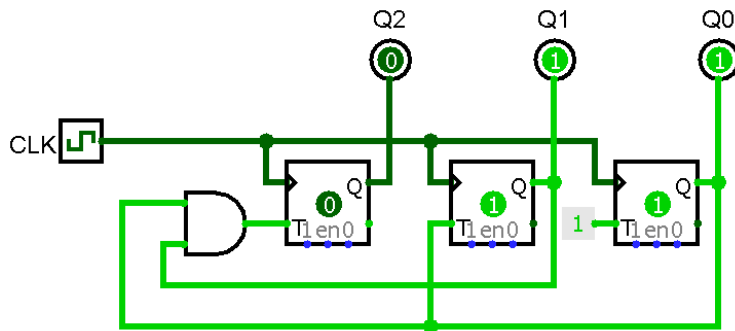
(b) $T_2 = Q_0$

		Q_1Q_0			
		00	01	11	10
Q_2	0	1	1	1	1
	1	1	1	1	1

(c) $T_2 = 1$

Slika 4: Određivanje funkcija pobude flipfopova

Na osnovu dobijenih funkcija pobude, crta se šema paralelnog brojača. Takt signal je zajednički, odnosno dovodi se istovremeno na sve flipfopove, što dovodi do njihovog sinhronog okidanja pri pojavi aktivne ivice takta. Šema ovako realizovanog brojača prikazana je na slici 5.



Slika 5: Logička šema paralelnog brojača modula 8

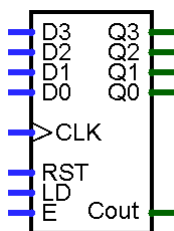
Zadatak Koristeći potreban broj JK-flipfopova i dodatnih logičkih kola, realizovati sinhroni brojač unazad modula 5.

Zadatak Koristeći potreban broj D-flipfopova i dodatnih logičkih kola, realizovati sinhroni brojač unapred modula 10.

Integrirani brojači i brojači sa skraćenim ciklusom

Budući da spadaju u najvažnije elemente složenih digitalnih sistema, brojači se često izrađuju kao integrisane komponente. Ovakvi brojači obično imaju dodatne kontrolne signale čijim korišćenjem je moguće uticati na ciklus brojanja. Signali 4-bitnog integrisanog brojača prikazanog na slici 6a su:

- $Q_3..Q_0$ - izlazi flipflopova koji predstavljaju stanje brojača
- $D_3..D_0$ - ulazi preko kojih je moguće izvršiti paralelni upis novog stanja u brojač
- CLK (engl. Clock) - ulaz za signal takta
- E (engl. Enable) - signal dozvole brojanja; kada je $E = 0$, brojač zadržava stanje u kom se zatekao, a kada je $E = 1$, brojač se inkrementira u svakoj periodi takta
- LD (engl. Load) - kada je ovaj signal aktivan (tj. $LD = 1$), u narednoj periodi takta se u brojač upisuje stanje koje je postavljeno na ulazima $D_3..D_0$
- RST (engl. Reset) - postavlja brojač u početno stanje $Q_3Q_2Q_1Q_0 = 0000_2$; reset može biti sinhroni (čeka sledeću aktivnu ivicu takta) ili asinhroni (brojač se resetuje odmah po aktiviranju RST signala)
- Cout (engl. Carry out) - ovaj signal se aktivira (tj. $C_{out} = 1$) kada se brojač nađe u poslednjem stanju $Q_3Q_2Q_1Q_0 = 1111_2$; koristi se pri kaskadiranju integriranih brojača, kako bi se realizovao brojač modula većeg od 16.



(a) Šematski simbol

E	CLK	RST	LD	$Q'_3Q'_2Q'_1Q'_0$	operacija
0	X	X	X	$Q_3Q_2Q_1Q_0$	zadržava stanje
1	↑	1	X	0 0 0 0	reset
1	↑	0	1	$D_3D_2D_1D_0$	paralelni upis
1	↑	0	0	$Q_3Q_2Q_1Q_0 + 1$	broji

(b) Funkcionalna tabela

Slika 6: 4-bitni integrirani brojač

Tabela na slici 6b prikazuje ponašanje 4-bitnog integriranog u zavisnosti od stanja kontrolnih signala. Signal dozvole E je najviši po prioritetu, pošto kada je jednak nuli onemogućava promenu stanja brojača. Brojač može da menja stanje u momentu nailaska aktivne (rastuće) ivice takta, pod uslovom da je $E = 1$, a funkcija mu je određena stanjima preostalih kontrolnih ulaza. Sledeći ulaz po prioritetu je RST koji dovodi do resetovanja brojača, ukoliko je aktivan u trenutku okidanja³. Najniži prioritet ima ulaz LD , koji dovodi do paralelnog upisa stanja dovedenog na ulaze $D_3..D_0$, što je takođe sinhrona operacija. Poslednji slučaj ($E = 1, RST = LD = 0$) predstavlja uobičajenu situaciju, kada brojač obavlja svoju primarnu funkciju inkrementirajući se u svakoj periodi takta.

³Dakle, u ovom slučaju u pitanju je sinhroni reset.

4-bitni integrisani brojač koji je upravo opisan može da prolazi kroz ukupno $2^4 = 16$ stanja. Ako su recimo na raspolaganju 2 takva brojača, njihovim kombinovanjem moguće je realizovati 8-bitni brojač maksimalnog modula $16^2 = 2^8 = 256$. To se postiže davanjem stalne dozvole brojanja brojaču čiji izlazi predstavljaju niža 4 bita 8-bitnog brojača ($Q_3..Q_0$). Izlaz *Cout* koji se aktivira kada je prvi brojač u stanju 1111_2 , vezuje se na signal dozvole brojanja drugog brojača, čiji izlazi predstavljaju viša 4 bita 8-bitnog brojača ($Q_7..Q_4$). Na ovaj način drugi brojač menja stanje tačno jednom na svakih 16 perioda takta prvog brojača, što je prikazano u tabeli 1. Podrazumeva se da je takt signal zajednički za oba integrisana brojača.

<i>stanje</i>	Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0	<i>Cout</i>
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0
...
15	0	0	0	0	1	1	1	1	1
16	0	0	0	1	0	0	0	0	0
17	0	0	0	1	0	0	0	1	0
...
31	0	0	0	1	1	1	1	1	1
32	0	0	1	0	0	0	0	0	0
...
255	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0

Tabela 1: Tabela stanja brojača modula 256

Zadatak Korišćenjem potrebnog broja 4-bitnih integrisanih brojača i dodatnih logičkih kola, realizovati brojač modula 100.

Rešenje:

Brojač modula 100 prolazi redom kroz stanja 0-99. Za realizaciju su potrebna 2 integrisana brojača, pošto je $16^1 = 16 < 100 \leq 16^2 = 256$. Dakle, potrebno je skratiti ciklus brojanja dobijenog 8-bitnog brojača sa 256 na 100. Ovo se postiže aktiviranjem signala za reset ($RST = 1$), u slučaju kada brojač dođe do poslednjeg stanja u željenom ciklusu, a to je 99:

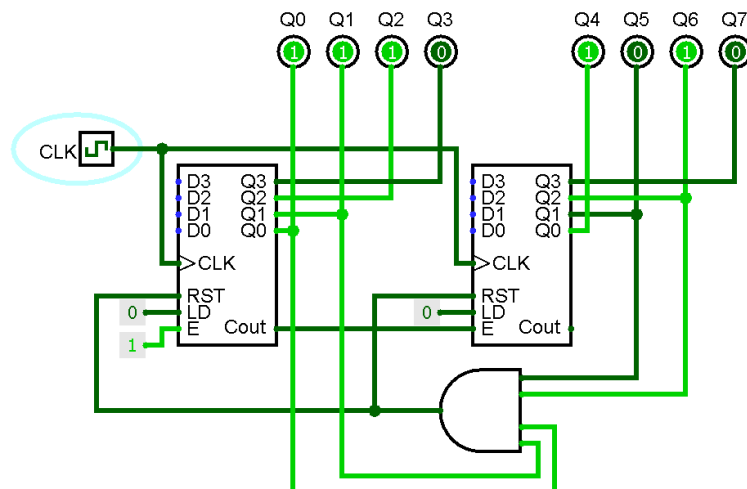
$$99_{10} = 0110\ 0011_2$$

Detekcija stanja 99, koja ujedno predstavlja uslov za resetovanje, vrši se detekcijom svih bita koji su u tom stanju jednaki jedinici⁴, a to su u ovom slučaju biti Q_6 , Q_5 , Q_1 i Q_0 . Na osnovu toga, dobija se funkcija koju je potrebno dovesti na ulaze za reset oba brojača:

$$RST = Q_6 \cdot Q_5 \cdot Q_1 \cdot Q_0$$

⁴Zašto je dovoljno detektovati samo one bite koji su jednaki jedinici? Evidentno, stanje 99 nije jedino u kojem važi uslov da je $Q_6 \cdot Q_5 \cdot Q_1 \cdot Q_0 = 1$, ali jeste prvo od svih takvih stanja. Pošto se nakon njegovog dostizanja brojač resetuje, ostala takva stanja jednostavno nikad neće biti dostignuta.

U ovom slučaju, ulazi za paralelni upis u brojače se ne koriste, pa se na njih dovodi neaktivno stanje (logička nula). Šema ovako realizovanog brojača prikazana je na slici 7.



Slika 7: Brojač modula 100

Zadatak Korišćenjem potrebnog broja 4-bitnih integrisanih brojača i dodatnih logičkih kola, realizovati brojač modula 10, koji broji od stanja 3 do 12.

Konačni automati

Za razliku od brojača kod kojih za svako trenutno stanje sledeće stanje unapred određeno, konačni automati predstavljaju širu klasu sekvencijalnih mreža kod kojih pored trenutnog stanja mreže, na sledeće stanje utiču i dodatni ulazni signali. U opštem slučaju, u strukturu konačnog automata spadaju 3 funkcionalna bloka:

- Blok za određivanje narednog stanja (ulazna kombinaciona mreža)
- Registar stanja (memorijski elementi)
- Blok za određivanje izlaznih promenljivih (izlazna kombinaciona mreža)

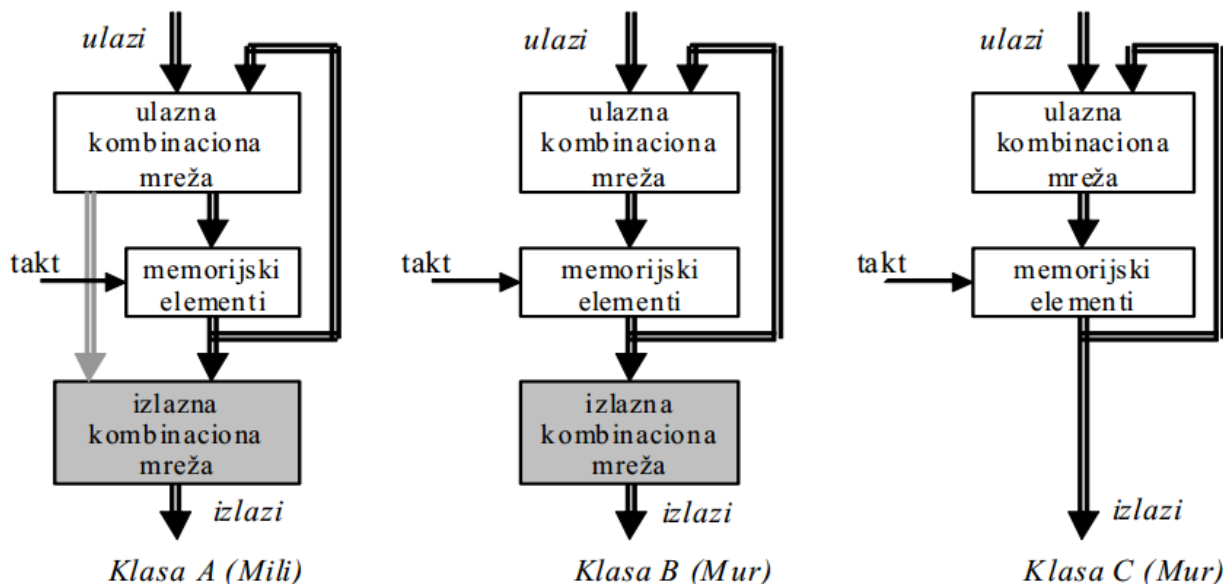
Blok za određivanje narednog stanja je kombinaciona mreža koja u zavisnosti od trenutnog stanja memorijskih elemenata i stanja na ulazima određuje u koje naredno stanje automat treba da pređe. Registar stanja je registar u kome se čuva stanje u kojem se automat trenutno nalazi. Blok za određivanje izlaznih promenljivih je kombinaciona mreža koja diktira stanja na izlazima.

U zavisnosti od načina na koji se formiraju izlazi, konačni automat može biti:

- Milijev (Mealy) automat (automat klase A)
- Murovi (Moore) automati (automati klase B i C)

Kod Murovih automata izlazi zavise isključivo od trenutnog stanja, a kod Milijevog izlazi zavise od trenutnog stanja i od ulaza. Zbog toga, kod Murovih automata stanja na izlazima menjaju se

isključivo sinhrono, zajedno sa stanjem celog automata. Sa druge strane, kod Milijevih atomata promene na izlazima mogu se dešavati asinhrono, u proizvoljnom vremenskom trenutku. Struktura sve 3 klase automata prikazana je na slici 8. Murov automat klase C je najjednostavniji i najčešće korišćen automat, zbog čega će njemu u nastavku biti posvećeno najviše pažnje⁵.



Slika 8: Struktura konačnih automata klase A, B i C

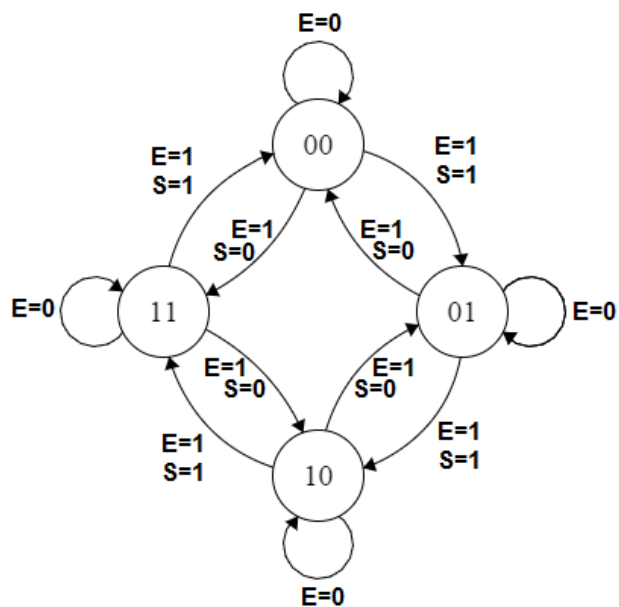
Ponašanje konačnih automata može se opisati na više načina, među kojima su najčešće korišćeni:

- Dijagram stanja (bubble dijagram)
- Tablica prelaza/izlaza
- Tekstualni opis

Projektovanje konačnog automata biće prikazano na primeru dvobitnog bidirekcionog (dvosmernog) brojača sa signalom dozvole brojanja. Registar stanja ovog automata ima 2 flipfopa, što znači da je ukupan broj stanja 4. Ovde će biti prikazana realizacija automata korišćenjem D-flipflopova. Kada je signal dozvole brojanja neaktivan ($E = 0$), automat zadržava stanje u kom se zatekao, a kada je aktivan ($E = 1$), brojač broji u smeru koji je određen ulazom S . Za $S = 1$, smer brojanja je unapred, odnosno sekvenca brojanja je $0 - 1 - 2 - 3$. Nasuprot tome, za $S = 0$, smer brojanja je unazad, pa je sekvenca brojanja $3 - 2 - 1 - 0$. Dijagram stanja ovakvog automata prikazan je na slici 9a. Stanja automata su predstavljena kružićima, a prelazi između njih su predstavljeni strelicama na koje se upisuju uslovi prelaza. Na osnovu dijagrama stanja, popunjava se tabela prelaza/izlaza prikazana na slici 9b. Pobuda koju je potrebno dovesti na flipfopove popunjava se na osnovu eksitacione tabele D-flipfopa⁶.

⁵Ovde je moguće primetiti da paralelni brojači koji su opisani u prethodnom odeljku po svojoj strukturi spadaju u konačne automate klase C, koji su još dodatno pojednostavljeni činjenicom da kod njih ne postoje ulazni signali.

⁶Eksitacija D-ff predstavlja najjednostavniji slučaj, pošto je logika rada ovog flipfopa takva da se sledeće stanje poklapa sa vrednošću koja je dovedena na ulaz (odnosno $Q' = D$).



(a) Dijagram stanja dvobitnog bidirekcionog brojača

Trenutno stanje		Ulazni signali		Naredno stanje		Pobuda FF	
Q_1	Q_0	E	S	Q'_1	Q'_0	D_1	D_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	1	1	1	1
0	0	1	1	0	1	0	1
0	1	0	0	0	1	0	1
0	1	0	1	0	1	0	1
0	1	1	0	0	0	0	0
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	0
1	0	0	1	1	0	1	0
1	0	1	0	0	1	0	1
1	0	1	1	1	1	1	1
1	1	0	0	1	1	1	1
1	1	0	1	1	1	1	1
1	1	1	0	1	0	1	0
1	1	1	1	0	0	0	0

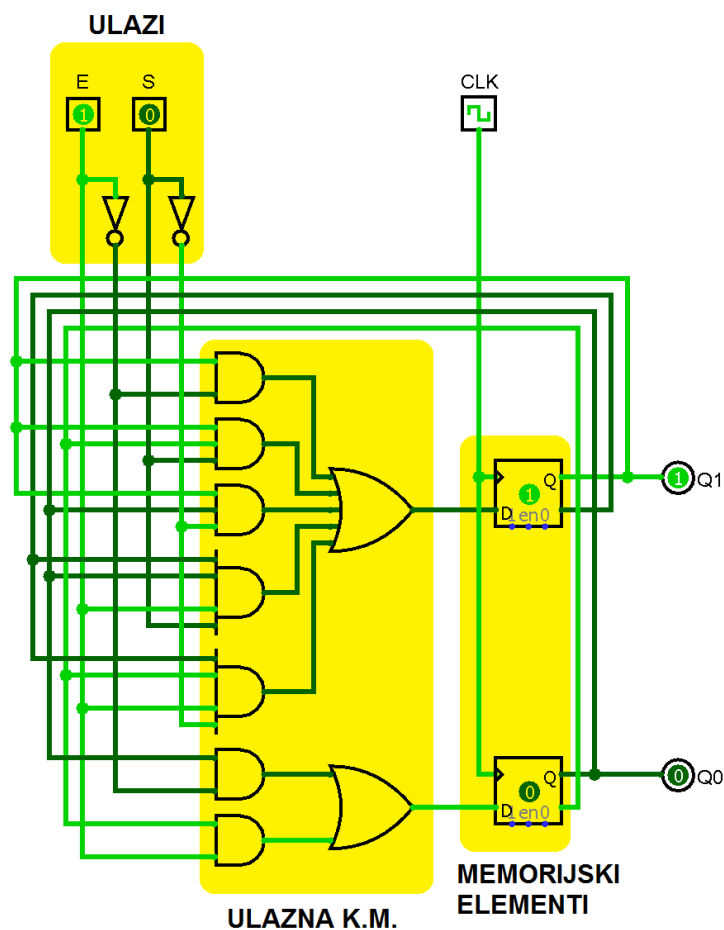
(b) Tabela prelaza/izlaza

Slika 9: 2-bitni bidirekcionni brojač

Na osnovu tabele prelaza/izlaza, minimizacijom se određuju funkcije pobude flipflova u zavisnosti od stanja automata i od ulaznih promenljivih. Ovaj korak zapravo predstavlja projektovanje ulazne kombinacione mreže. Dobijaju se funkcije pobude, na osnovu kojih se realizuje automat, kao što je prikazano na slici 10.

$$D_1 = Q_1 \cdot \bar{E} + Q_1 \cdot \bar{Q}_0 \cdot S + Q_1 \cdot Q_0 \cdot \bar{S} + \bar{Q}_1 \cdot Q_0 \cdot E \cdot S + \bar{Q}_1 \cdot \bar{Q}_0 \cdot E \cdot \bar{S}$$

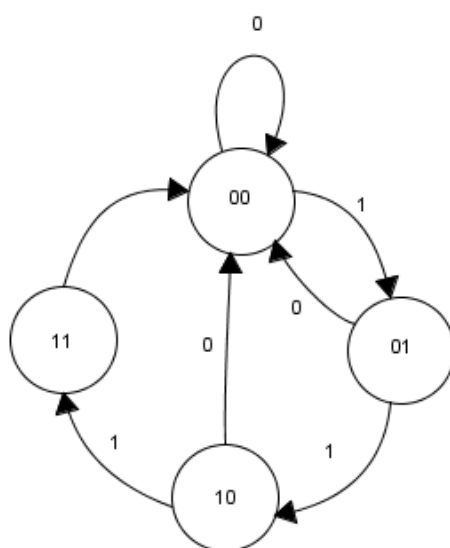
$$D_0 = Q_0 \cdot \bar{E} + \bar{Q}_0 \cdot E$$



Slika 10: Logička šema 2-bitnog bidirekcionog brojača

Zadatak Nacrtati šemu i simulirati rad automata čiji dijagram stanja je prikazan na slici 11. Pri realizaciji koristiti potrebna logička kola i:

- JK-flipflopove
- T-flipflopove
- D-Flipflopove



Slika 11: *Dijagram stanja automata*