

UVOD U ARHITEKTURU RAČUNARA
Letnji semestar 2013/2014
(Saša Malkov)

Autori:
Ludi Burekdžija
Anja Bukurov

2014

1 Osnovne zakonitosti algebre logike

Algebra logike predstavlja strukturu $\{S; \wedge, \vee, \neg\}$ gde je $S = \{\top, \perp\}$, \wedge binarna operacija konjunkcije, \vee binarna operacija disjunkcije i \neg unarna operacija negacije.

OSNOVNI ZAKONI

Zakon komutacije: $A \cdot B = B \cdot A$ $A + B = B + A$

Zakon asocijacije: $(A \cdot B) \cdot C = A \cdot (B \cdot C)$ $(A + B) + C = A + (B + C)$

Zakon distribucije: $A \cdot (B + C) = (A \cdot B) + (A \cdot C)$ $A + (B \cdot C) = (A + B) \cdot (A + C)$

Neutralni element: $1 \cdot A = A$ $0 + A = A$

Inverzni element: $A \cdot \neg A = 0$ $A + \neg A = 1$

2 Identiteti i pravila algebre logike

Nula-pravilo: $A \cdot 0 = 0$ $A + 1 = 1$

Idempotencija: $A \cdot A = A$ $A + A = A$

Apsorpcija: $A \cdot (A + B) = A$ $A + (A \cdot B) = A$

Dvostruka negacija: $\neg \neg A = A$

Brisanje zagrada: $(A \cdot B) \cdot C = A \cdot B \cdot C$ $(A + B) + C = A + B + C$

De Morganove teoreme: $\neg(A \cdot B) = \neg A + \neg B$ $\neg(A + B) = \neg A \cdot \neg B$

3 Šta je logička funkcija?

Neka su A_1, A_2, \dots, A_n logičke promenljive.

Svaka funkcija $f : (A_1, A_2, \dots, A_n) \rightarrow \{0, 1\}$ se naziva logička funkcija.

Svaka funkcija sa domenom S^n i kodomenom S naziva se logička funkcija.

Kako svaka od logičkih promenljivih koja je argument funkcije može imati vrednost 0 ili 1 (tj. netačno ili tačno), to je broj različitih funkcija od n argumenata jednak 2^{2^n} .

4 Logičke funkcije sa 1 argumentom

Argument	Vrednost		Naziv	Oznaka
	0	1		
A	0	1		
Funkcija				
f_{10}	0	0	nula funkcija	
f_{11}	0	1	identitet	
f_{12}	1	0	negacija	$\neg A$
f_{13}	1	1	jedinična funkcija	

5 Logičke funkcije sa 2 argumenta

Argument	Vrednost				Naziv	Oznaka
A	0	0	1	1		
B	0	1	0	1		
Funkcija						
f_{20}	0	0	0	0	nula funkcija	
f_{21}	0	0	0	1	konjunkcija	$A \wedge B$
f_{22}	0	0	1	0	negacija implikacije od A ka B	$A \wedge \neg B = \neg(A \Rightarrow B)$
f_{23}	0	0	1	1	prva projekcija	
f_{24}	0	1	0	0	negacija implikacije od B ka A	$\neg A \wedge B = \neg(B \Rightarrow A)$
f_{25}	0	1	0	1	druga projekcija	
f_{26}	0	1	1	0	ekskluzivna disjunkcija	$(A \wedge \neg B) \vee (\neg A \wedge B)$
f_{27}	0	1	1	1	disjunkcija	$A \vee B$
f_{28}	1	0	0	0	Pirsova (Lukaševičeva) funk.	$A \downarrow B$
f_{29}	1	0	0	1	ekvivalencija	$(A \Leftrightarrow B)$
f_{2A}	1	0	1	0	negacija druge projekcije	
f_{2B}	1	0	1	1	implikacija od B na A	$B \Rightarrow A$
f_{2C}	1	1	0	0	negacija prve projekcije	
f_{2D}	1	1	0	1	implikacija od A na B	$A \Rightarrow B$
f_{2E}	1	1	1	0	Šeferova funkcija	$A \uparrow B$
f_{2F}	1	1	1	1	Jedinična funkcija	

6 Pun sistem funkcija

Neka je f_i neka od prethodno definisanih funkcija algebre logike sa jednim ili dva argumenta. Skup $F = \{f_1, f_2, \dots, f_n\}$ funkcija algebre logike se naziva pun sistem funkcija ako se proizvoljna funkcija algebre logike može predstaviti pomoću funkcija iz ovog skupa.

Primeri: $\{\wedge, \vee, \neg\}$ $\{\vee, \neg\}$ $\{\wedge, \neg\}$

Pun sistem funkcija je skup funkcija na osnovu kojih mogu da se izvedu sve ostale funkcije. Ako se iz nekog sistema funkcija mogu izvesti sve funkcije punog sistema funkcija, onda je i taj sistem pun sistem funkcija.

7 Elementarna Konjunkcija i Disjunkcija

Elementarna konjunkcija je logički izraz koji ne sadrži operaciju disjunkcije. Elementarna disjunkcija je logički izraz koji ne sadrži operaciju konjunkcije.

8 Savršena Konjukcija i Disjunkcija

Savršena elementarna konjunkcija je elementarna konjunkcija koja sadrži sve promenljive (ili njihove negacije) iz skupa promenljivih od kojih se grade logički izrazi.

Savršena elementarna disjunkcija je elementarna disjunkcija koja sadrži sve promenljive (ili njihove negacije) iz skupa promenljivih od kojih se grade logički

izrazi.

9 Konjunktivne i Disjunktivne forme

Disjunktivna forma je logički izraz koji se sastoji od elementarnih konjunkcija međusobno povezanih operacijama disjunkcije.

Konjunktivna forma je logički izraz koji se sastoji od elementarnih disjunkcija međusobno povezanih operacijama disjunkcije.

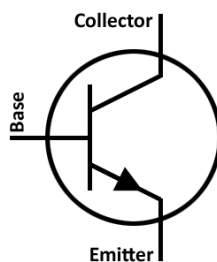
Savršena disjunktivna normalna forma je disjunktivna forma u kojoj su sve funkcije savršene elementarne konjunkcije.

Savršena konjunktivna normalna forma je konjunktivna forma u kojoj su sve funkcije savršene elementarne disjunkcije.

10 Šta je tranzistor? Nacrtati simbol i objasniti

Osnovna jedinica implementacije digitalnog računara je tranzistor. Sadrži:

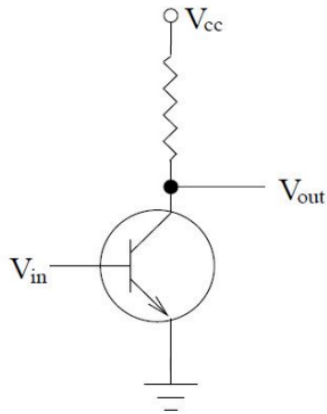
- Emiter - izvor elektrona (negativan kraj)
- Kolektor - sakupljač elektrona (pozitivan kraj)
- Baza - poput prekidača - visok potencijal (iznad 2V) omogućava protok nizak potencijal (ispod 0,8V) sprečava protok



Uobičajeno je:

- emiteri se vezuju za uzemljenje
- izvor napajanja obezbeđuje napon od +5V u odnosu na uzemljenje (tačan napon zavisi od implementacije)
- odsustvo napona predstavlja vrednost 0
- postojanje napona predstavlja vrednost 1

11 Kako se pomoću tranzistora implementira negacija?



Implementacija negacije (NE-element)

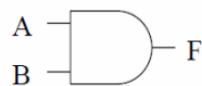
12 Šta je logičko kolo?

Logička kola su apstraktna digitalna kola koja implementiraju logičke funkcije. Predstavljaju apstrakciju električnih (optičkih i drugih) kola.

13 Šta je logički element?

Logički elementi su elementarna digitalna kola koja implementiraju elementarne logičke funkcije. Obično logički elementi implementiraju funkcije koje čine pun sistem funkcija.

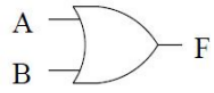
14 Uobičajeni logički elementi



AND gate

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

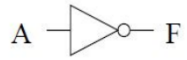
I - element



OR gate

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

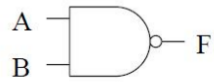
ILI - element



NOT gate

A	F
0	1
1	0

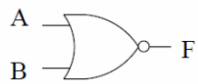
NE - element



NAND gate

A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

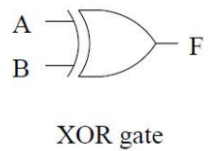
NI - element



NOR gate

A	B	F
0	0	1
0	1	0
1	0	0
1	1	0

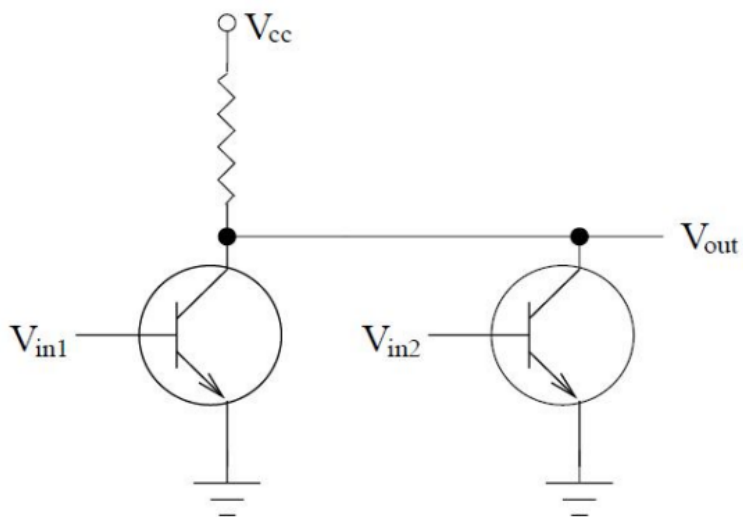
NILI - element



A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

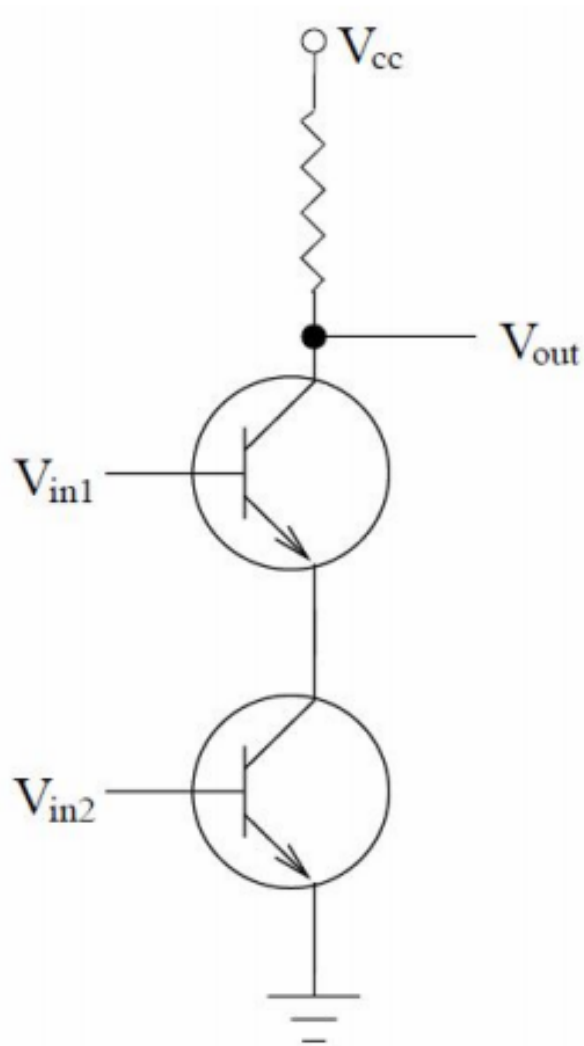
EILI - element

15 Implementirati pomoću tranzistora logički element NILI



NILI tranzistor

16 Implementirati pomoću tranzistora logički element NI



NI tranzistor

17 Koji su osnovni koraci projektovanja logičkog kola?

- Određivanje problema
- Izvođenje istinitosne tablice
- Izvođenje logičkog izraza - obično SDNF, neposredno iz tablice
- Uprošćavanje logičkog izraza - svođenje na minimalni oblik
- Oblikovanje logičkog kola

18 Šta je minimizacija logičkih funkcija? Koji su osnovni metodi minimizacije logičkih funkcija?

Minimizovanje logičke funkcije je pronalaženje njenog najjednostavnijeg zapisa. Osnovni metodi minimizacije su:

- Algebarske transformacije
- Karnoove mape
- Tablična metoda Kvin MekKlaskog

19 Objasniti način upotrebe Karnoovih mapa

Pravi se višedimenziona mapa. Na svaku dimenziju navode se najviše po dva argumenta funkcije. Vrednosti argumenata se navode takvim redom da se menja po tačno jedan bit (Hamingova distanca 1) - 00, 01, 11, 10.

Osnovu za uprošćavanje predstavlja pravilo:

$$A_1 A_2 \dots A_k \dots A_n + A_1 A_2 \dots A_{k'} \dots A_n = A_1 A_2 \dots A_{k-1} A_{k+1} \dots A_n$$

Bilo koja dva susedna kvadrata se razlikuju samo za vrednosti jedne promenljive koja se u tim kvadratima pojavljuje sa komplementiranim vrednostima. Ako je u oba kvadrata upisana 1, tada se iz izraza koji je rezultat njihove disjunkcije može, primenom distribucije i korišćenjem zakona o inverznom elementu, eliminisati ta promenljiva. Susednim se mogu smatrati i kvadrati na ivicama mape - kvadrati na vrhu i dnu svake svake od kolona i na levoj i desnoj strani svake od vrsta mape. Isti postupak se ože primeniti i na grupe od $2n$ susednih kvadrata.

Proces minimizacije se započinje posmatranjem što je moguće veće grupe. Odaberu se sve grupe koje su susedne. Ako neki od kvadrata koji sadrži 1 ostane nezaokružen, tada se posmatraju i manje grupe. Dozvoljeno je da jedan kvadrat koji sadrži 1 pripada većem broju grupa. Minimizovana verzija funkcije uključuje odgovarajući broj promenljivih na mesto svake od odabranih grupa, pri čemu se odabrana grupa koja je potpuno obuhvaćena drugim odabranim grupama ignoriše kao redundantna.

20 Kako se upotrebljavaju Karnoove mape u prisustvu nedefinisanih vrednosti?

U nekim slučajevima određena kombinacija vrednosti promenljivih se nikada ne pojavljuje, pa samim tim ne može ni da bude prisutna u rezultatu. Ove vrednosti se označavaju kao "nebitno".

Za svaku od ovih kombinacija u odgovarajući kvadrat se unosi slovo "n" koje može da se koristi ili kao 0 ili kao 1, po potrebi.

21 Šta je metod Kvin MekKlaskog? Koji su osnovni koraci?

Metod Kvin-MekKalskog ili tablični metod je metod minimizacije funkcija. Osnovni koraci su: pronalaćenje prostih implikanata, određivanje bitnih implikanata i uključivanje dodatnih prostih implikanata.

22 Objasniti korak pronalaženja prostih implikanata metoda Kvin MekKlaskog

Konstruiše se tabela tako što se u svaki red upise po jedan disjunkt SDNF. Redovi se grupišu prema broju promenljivih koje su komplementirane. Svaki term treba porediti sa termima iz prethodne grupe.

Upareni su samo ako se razlikuju samo po stanju jedne promenljive. Termini koji su upareni se štikliraju i upisuju u novu tabelu bez promenljive po kojoj se razlikuju

Postupak se ponavlja dokle god postoje termini koji mogu da se upare. Svi preostali, neoznačeni termini, iz svih tabela, su prosti implikanti.

23 Objasniti korak pronalaženja bitnih implikanata metoda Kvin MekKlaskog

Pravi se tabela u kojoj vrste odgovaraju prostim implikantima a kolone disjunktima polazne SDNF. Ako je prosti implikant sadržan u disjunktima onda se u tabeli, presek odgovarajuće vrste i kolone označi sa X.

Ako kolona sadrži samo jedno X ono se zaokružuje, to je bitan implikant. Ako u vrsti postoji još neko X osim zaokruženog, oko njega se nacrtava kvadrat. Ako u svakoj koloni postoji X sa krugom ili kvadratom onda je postupak završen.

Konjunkcija bitnih implikanata je rezultat. Ako postoji kolona u kojoj X nema krug ili kvadrat, onda se uključuju dodatni prosti implikanti.

24 U čemu je značaj i šta obuhvata korak uključivanja i isključivanja dodatnih prostih implikanata metoda Kvin MekKlaskog?

Ako bitni implikanti nisu dovoljni da pokriju rezultat uključuje se minimalan broj dodatnih prostih implikanata. Za svaki nepokriven disjunkt (kolonu) bira se po jedan prost implikant. Dodatni prosti implikant se bira tako da pokriva što je više moguće kolona kako bi njihov broj bio minimalan.

25 Kako se primenjuje metod Kvin MekKlaskog u prisustvu nebitnih slučajeva?

U prvom koraku se koriste kao da su jedinice na nedefinisanim mestima, a u drugom koraku se zanemaruju.

26 Šta je kombinatorna mreža?

Kombitarne mreže predstavljaju skup međusobno povezanih elemenata čiji je izlaz u nekom trenutku funkcija koja zavisi od vrednosti ulaza u tom istom trenutku.

27 Kako se definišu kombinatorne mreže?

U opštem slučaju kombinatorna mreža se sastoji od n binarnih ulaza i m binarnih izlaza. Može se opisati pomoću:

- tabele istinitosnih vrednosti koja sadrži svaku od 2^n mogućih kombinacija ulaznih vrednost
- povezanog skupa grafičkih simbola
- logičkih funkcija koje izražavaju vezu između ulaznih i izlaznih vrednosti

28 Navesti najvažnije vrste kombinatornih mreža

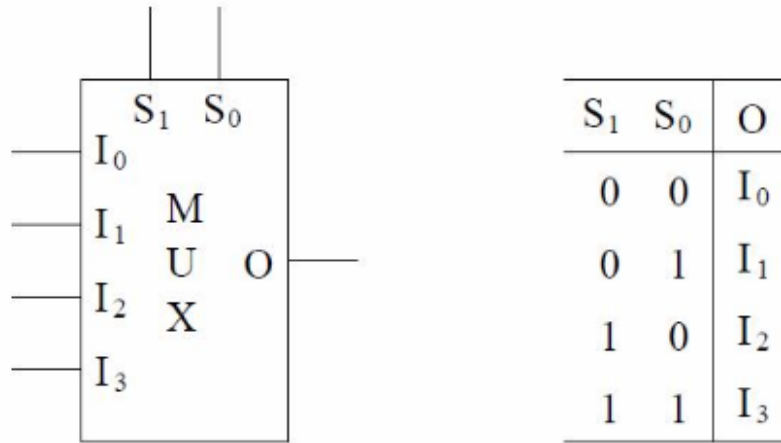
Najvažnije vrste kombinatornih mreža su:

- multipleksori
- demultipleksori
- dekoderi
- enkoderi
- komparatori
- sabirači
- programabilni niz logičkih elemenata

29 Šta je multipleksor?

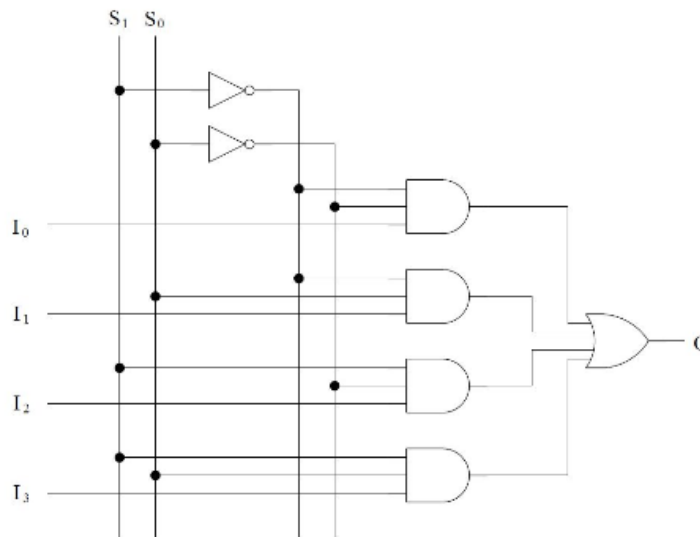
Multipleksor je kombinatorna mreža koja ima 2^n ulaza, n selektorskih ulaza i jedan izlaz. Vrednost izlaza odgovara vrednosti ulaza koja je određena vrednošću selektorskih ulaza.

30 Predstaviti grafičkim simbolom i tabelom multipleksor 4-1



Multipleksor

31 Nacrtati logičko kolo implementacije multipleksora 4-1



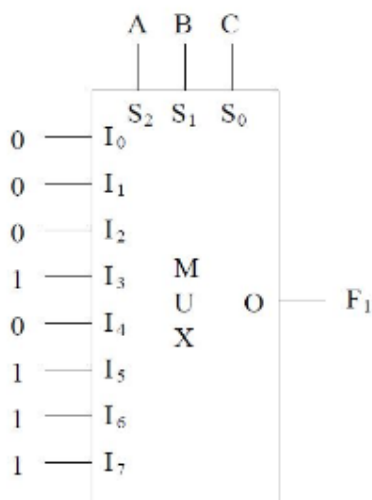
Implementacija multipleksora

32 Kako se multipleksor upotrebljava za implementaciju logičkih funkcija? Primer.

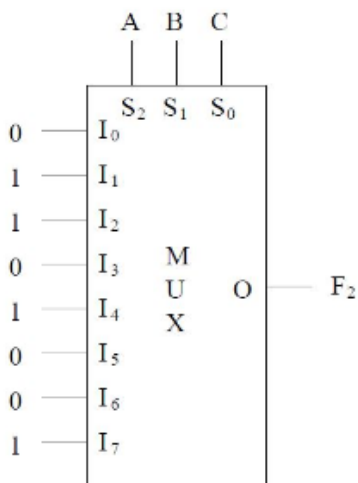
Osim osnovne primene kominovanja više ulaza u jedan izlaz (odabiranja), biranjem ulaznih vrednosti se multipleksorski mogu upotrebljavati za implementiranje funkcija od selektorskih ulaza.

I način: na ulaze se dovedu konstante vrednosti, tako da odgovaraju vrednostima funkcije za odgovarajuće selektorske ulaze.

II način: procesom redukcije se multipleksorom može implementirati funkcija sa $n+1$ argumenata (u nekim slučajevima može i više).



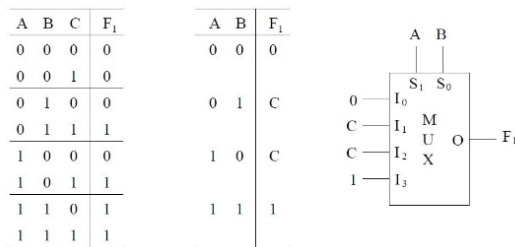
Izračunavanje zastupljenijeg bita na selektorskim ulazima



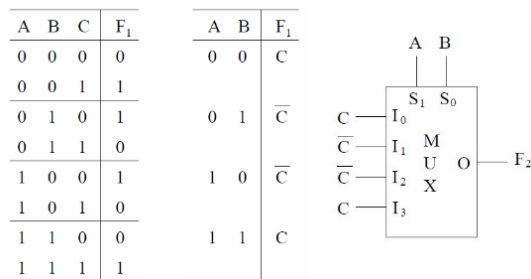
Izračunavanje parnosti za selektorske ulaze

33 Opisati metod redukcije pri implementiranju logičkih funkcija pomoću multipleksora. Primer.

Ideja redukcije je da se jedan od argumenata pretoči u rezultat funkcije tako što se prvo izabere jedan argument, npr. A_k . Zatim se prepoznaju slučajevi u kojima važi $F = A_k$ ili $F = A'_k$. U ostalim slučajevima se funkcija predstavi tako da ne zavisi od argumenta



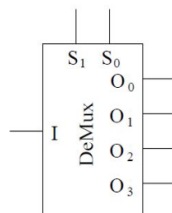
Funkcija računa većinski bit



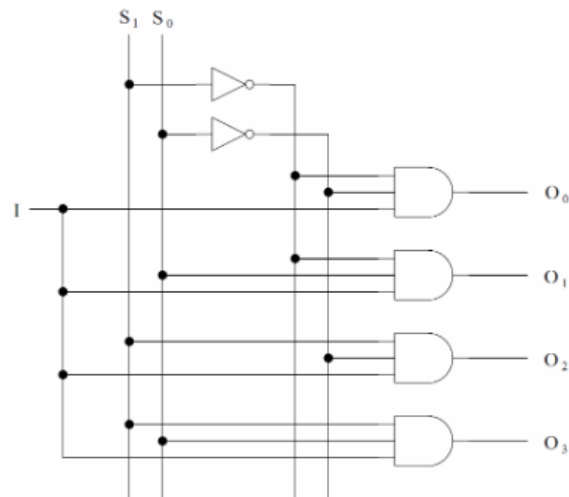
Funkcija računa parnost

34 Šta je demultipleksor? Predstaviti grafičkim simbolom primer demultipleksora.

Demultipleksori su kombinatorne mreže koje imaju $n+1$ ulaza, 1 ulaznu vrednost i 2^n izlaza. Obavljaju inverznu funkciju od multipleksora. Tačno na jedan izlaz se preslikava vrednost ulaza, a svi ostali uzimaju vrednost 0.



35 Nacrtati logičko kolo implementacije demultipleksora 1-4



Implementacija demultipleksora

36 Šta je dekođer?

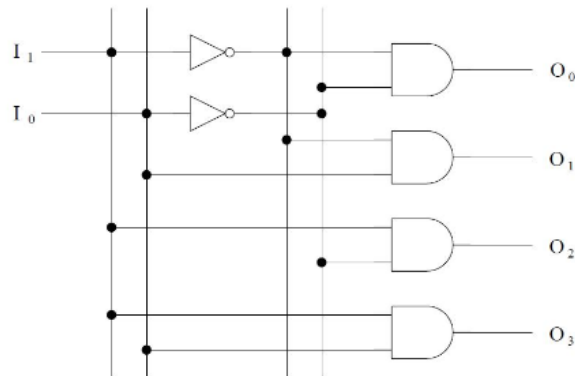
Dekoderi su kombinatorne mreže koje imaju n ulaza i najviše 2^n izlaza. U svakom trenutku aktivan je najviše jedan izlaz, u zavisnosti od ulaza.

37 Predstaviti grafičkim simbolom i tablicom dekođer 2-4

I_1	I_0	O_3	O_2	O_1	O_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

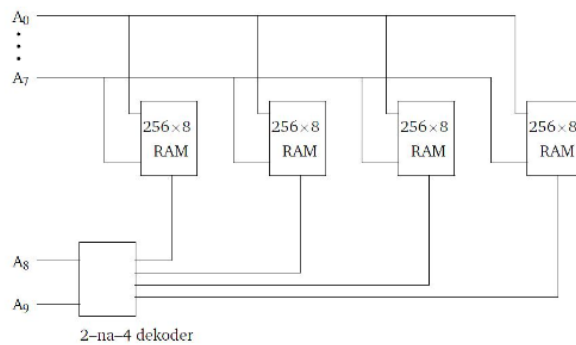
Na osnovu neoznačeno 2-bitnog celog broja bira odgovarajući od 4 ulaza

38 Nacrtati logičko kolo implementacije dekodera 2-4



Implementacija dekodera

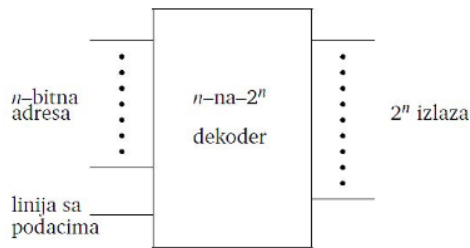
39 Kako se dekoderi 2-4 mogu implementirati za proširivanje adresnog prostora memorije? Nacrtati.



Dekodiranje adresa memorije veličine 1KB od četiri 256B (8-bitna) RAM čipa

40 Kako se dekoderi mogu upotrebljavati kao demultipleksori?

Dekoderu se uvode dodatni ulaz, i on predstavlja ulaz demultipleksora. Može da se tumači i kao kontrolni bit dekodera. Dekoder radi ako je ulaz aktivan, ako je neaktivan onda su svi ulazi neaktivni.



Dekoder kao demultipleksor

41 Nacrtati logičko kolo koje implementira dekodler 2-4

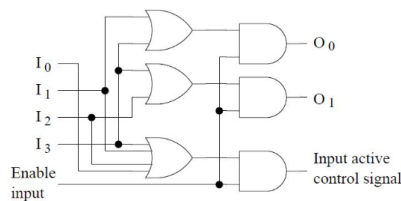
42 Šta je enkoder?

Enkoderi su kombinatorne mreže koje imaju $2n$ ulaza i n izlaza. Predstavljaju inverznu operaciju dekodera. U svakom trenutku je aktivan najviše jedan ulaz. Izlaz je određen aktivnom ulazom. Obično se dodaju kontrolni ulaz koji uključuje enkoder, kontrolni izlaz koji je aktivan ako su aktivni kontrolni ulaz i bar jedan ulazni bit.

43 Nacrtati tablicu vrednosti enkodera 4-2

Enable input	I_3	I_2	I_1	I_0	O_1	O_0	Input active control signal
0	X	X	X	X	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	1	1
1	0	1	0	0	1	0	1
1	1	0	0	0	1	1	1

44 Nacrtati logičko kolo enkodera 4-2



45 Šta je enkoder sa prioritetom?