Laboratorinis darbas. Kombinacinės logikos schemos

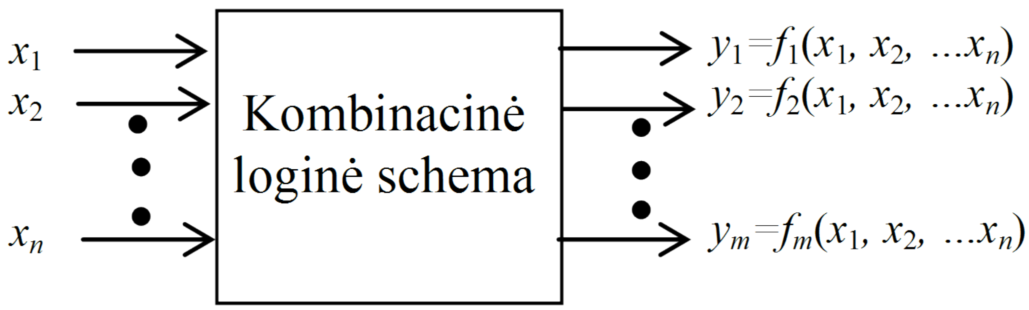
Turinys

1. Tikslas

Įsisavinti Bulio funkcijų minimizavimą ir kombinacinių loginių schemų projektavimą bei modeliavimą.

2. Teorija

Loginės schemos skirstomos į kombinacines ir trigerines (literatūroje dar vadinamas nuosekliosiomis, sekvencinėmis) schemas. Kombinacinėmis vadinamos loginės schemos, sudarytos iš sujungtų loginių elementų, neturinčios kilpų ir atminties elementų, o išvesties signalų vektoriaus Y reikšmes vienareikšmiškai nustato tuo metu veikiančių įvesčių signalų vektoriaus reikšmės X. Kombinacinės schemos dažnai vadinamos loginėmis schemomis be grįžtamųjų ryšių. Kombinacinė schema pavaizduota 1 pav.



1 pav. Kombinacinė schema

Skaitmeninių schemų funkcionavimui aprašyti plačiai taikoma Bulio algebra. Tai algebra $(B, \cdot, \cup, \neg, 0, 1)$, kurią sudaro baigtinė elementų aibė B, jos elementai gali įgyti reikšmes 0 arba 1, ir trys apibrėžtos operacijos su elementais:

- IR operacija (Bulio daugyba arba konjunkcija) $\cdot$,
- ARBA operacija (Bulio sudėtis arba disjunkcija) $\cup$,
- NE operacija (neigimas arba inversija) $\neg$.

Taigi kiekviena loginė schema realizuoja loginę (Bulio) funkciją (išvesties y reikšmę), kurios argumentai yra Bulio kintamieji (įvesčių x reikšmės). Bulio funkcijos vaizduojamos:

- teisingumo lentelėmis, kai Bulio funkcija turi n kintamųjų ir aprašoma visomis galimomis kintamųjų 2^n kombinacijų reikšmėmis;

- analitinėmis išraiškomis, kai funkcijos išreiškiamos formulėmis;
- skaitmenine forma, kai užrašomos įvesčių reikšmės, su kuriomis funkcija įgauna reikšmę 1;
- Karno (Veičo) diagramomis, kai Bulio funkcijos vaizduojamos grafiniu būdu.

Pavyzdys. Sudarykime trijų kintamųjų funkcijos $y = f(x_1, x_2, x_3)$, įgaunančios vienetinę reikšmę, kai nelyginis įėjimų skaičius įgyja reikšmę 1, teisingumo lentelę (1 lentelė).

1 lentelė. Funkcijos $y = f(x_1, x_2, x_3) = (1, 2, 4, 7)$ teisingumo lentelė

x_1	x_2	x_3	y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Lentelėje apimčiai sumažinti dažnai pateikiamos tik tos kintamųjų reikšmių kombinacijos, kurioms esant funkcija lygi 1. Tai pavaizduota 2 lentelėje.

2 lentelė. Funkcijos $y = f(x_1, x_2, x_3) = (1, 2, 4, 7)$ trumpa teisingumo lentelė

x_1	x_2	x_3	y
0	0	1	1
0	1	0	1
1	0	0	1
1	1	1	1

Pasinaudoję teisingumo lentele, parašykime pateiktos funkcijos analitinę išraišką – formulę. Paprasčiausiu atveju funkciją galima parašyti kaip konjunkcijų, atitinkančių tas kintamųjų reikšmes, su kuriomis funkcija lygi vienetui, disjunkciją:

$$y = f(x_1, x_2, x_3) = \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cup \overline{x_1} \cdot x_2 \cdot \overline{x_3} \cup x_1\overline{x_2} \cdot \overline{x_3} \cup x_1 \cdot x_2 \cdot x_3\,. \tag{1}$$

Kiekviena konjunkcijų grupė vadinama mintermu. Gautąją funkciją galima dar supaprastinti, iškeliant bendrus mintermų narius prieš skliaustus:

$$y = f(x_1, x_2, x_3) = \overline{x_1}(\overline{x_2} \cdot x_3 \cup x_2 \cdot \overline{x_3}) \cup x_1(\overline{x_2} \cdot \overline{x_3} \cup x_2 \cdot x_3) \ , \tag{2}$$

arba naudojantis Bulio algebros taisyklėms pakeisti ją kita išraiška:

$$y = f(x_1, x_2, x_3) = \overline{x_1}(x_2 \oplus x_3) \cup x_1(\overline{x_2 \oplus x_3}) = x_1 \oplus x_2 \oplus x_3 \ . \tag{3}$$

Parašykime pateiktos funkcijos skaitmeninę formą (teisingumo lentelėje funkciją atitinkantys numeriai):

$$y = f(x_1, x_2, x_3) = (1, 2, 4, 7) \ . \tag{4}$$

Tą pačią Bulio funkciją galima išreikšti įvairiai. Paprastai Bulio funkcija yra minimizuojama, tai yra gaunama funkcijos analitinė išraiška, turinti mažiau operacijų, konjunkcijų ar konjunkcijos raidžių. Tai leidžia realizuoti schemą mažesniu loginių elementų kiekiu. Nedidelėms funkcijoms minimizuoti gerai tinka vadinamasis bandymų ir klaidų metodas, kai minimizuojama remiantis projektuotojo žiniomis ir patirtimi. Minimizuoti naudojama ši procedūra:

- naudojantis aksiomomis ir teoremomis, išplėstinę disjunkcinę normaliąją formą galima pertvarkyti ir gauti paprastesnę disjunkcinę normaliąją formą, turinčią mažiau konjunkcijų ar mažiau raidžių konjunkcijose;
- grupuojant konjunkcijas, iškeliant bendras raides prieš skliaustus, galima gauti trumpesnę funkcijos išraišką;
- greta pagrindinių operacijų (disjunkcijos, konjunkcijos ir inversijos) naudojant sudėtingesnes operacijas (sumą moduliu du ir pan.), funkcijos išraišką taip pat galima sutrumpinti.

Norint minimizuoti funkciją gali būti naudojamos Karno diagramos. Jos sudaromos teisingumo lentelės rezultatus perrašant į dvimatę lentelę, kurios eilutės ir stulpeliai numeruojami Grėjaus kodu. Kiekviena ląstelė atitinka vieną įvesčių kombinaciją. Taikant šį metodą, efektyviai naudojami projektuotojo gebėjimai vizualiai atpažinti būdingą vienetų išdėstymą diagramoje. Randamos optimalios '1' ar '0' grupės, žyminčios kanoninės formos termus. Šie termai naudojami užrašant minimalią Bulio išraišką, kuri aprašo reikiamus loginius elementus. Funkcijai $y = f(x_1, x_2, x_3) = (1, 2, 4, 7)$ Karno diagrama pavaizduota 3 lentelėje (panaudojus teisingumo lentelę į atitinkamus langelius surašomos vienetinės funkcijos reikšmės).

3 lentelė. Funkcijos $y = f(x_1, x_2, x_3) = (1, 2, 4, 7)$ Karno diagrama

x_1x_2 \ x_3	0	1
00		1
01	1	
11		1
10	1	

Kombinacinės loginės schemos yra sudarytos iš elementariųjų loginių elementų. Jos projektuojamos tokia tvarka:

1. Nustatomos loginės funkcijos ir jų argumentai (loginiai kintamieji).

2. Sudaroma loginė funkcija, kurią turi vykdyti projektuojamoji schema, loginė funkcija gali būti pateikta teisingumo lentele arba loginės funkcijos kanoninės disjunkcinės formos išraiška.

3. Minimizuojant loginės funkcijas vienu iš žinomų metodų (Karno diagramų, Kvaino ir Maklaskio ir t. t.), gaunama minimali loginės funkcijos išraiška. Šią išraišką galima pertvarkyti taip, kad ją būtų patogiau realizuoti loginiais elementais.

4. Atliekama sintezė: paruošta loginė funkcija realizuojama norimais loginiais elementais. Gaunama kombinacinė loginė schema. Atsižvelgiant į greitaveikos ir galios suvartojimo reikalavimus gali būti taikomi specialūs automatizuoti kombinacinių schemų sintezės metodai, leidžiantys gauti schemas su optimaliu integrinių grandynų kiekiu.

5. Susintezavus kombinacinę schemą sudaromas kontrolinis testas, tai yra kintamųjų seka, kurią perdavus į suprojektuotos schemos įvestis galima patikrinti, ar ji realizuoja reikiamą funkciją.

Pavyzdys. Turime funkciją $y = f(x_1, x_2, x_3, x_4)$, kurios teisingumo lentelė pavaizduota 4. Lentelėje surašome tik funkcijos reikšmes, lygias vienetui.

4 lentelė. Funkcijos $y = f(x_1, x_2, x_3, x_4)$ teisingumo lentelė

x_1	x_2	x_3	x_4	y
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	1	1	1
1	1	0	0	1
1	1	1	0	1
1	1	1	1	1

Naudodamiesi lentele užrašome schemos funkciją normaliąja disjunkcine forma:

$$\begin{aligned} y = {} & \overline{x_1}\,\overline{x_2}x_3x_4 \cup \overline{x_1}x_2\overline{x_3}\,\overline{x_4} \cup \overline{x_1}x_2\overline{x_3}x_4 \cup \\ & \cup \overline{x_1}x_2x_3\overline{x_4} \cup \overline{x_1}x_2x_3x_4 \cup x_1\overline{x_2}x_3x_4 \cup \\ & \cup x_1x_2\overline{x_3}\,\overline{x_4} \cup x_1x_2x_3\overline{x_4} \cup x_1x_2x_3x_4\ . \end{aligned} \tag{5}$$

Pasinaudoję teisingumo lentele sudarome Karno diagramą. Kadangi yra 4 įvesties signalai, jie gali įgyti 16 skirtingų reikšmių, todėl pilnoji teisingumo lentelė turi 16 eilučių, ir Karno diagrama turi 16 langelių. Dėl šios priežasties diagrama vaizduojama 4×4 lentele. Eilučių ir stulpelių reikšmės sužymimos Grėjaus kodu. Grėjaus kodas užtikrina, kad šalia esantys langeliai skiriasi tik per vieną kintamąjį. Jei iš konkrečios įvesčių signalų kombinacijos išvestyje gaunamas 1, tame Karno lentelės langelyje pažymimas 1. Sudarius Karno lentelę, ieškoma supaprastinta kanoninė užrašytos informacijos išreiškimo forma. Greta esantys vienetai lentelėje sugrupuojami laikantis tam tikrų taisyklių:

- Mintermų grupės Karno diagramoje turi būti stačiakampio formos. Stačiakampio plotas gali būti tik dvejeto laipsnis (1,2,4,8...).
- Mintermų grupės turi būti kiek įmanoma didesnės.
- Grupės gali dengti viena kitą, kad kiekviena jų būtų kiek įmanoma didesnė.
- Lentelė yra toroidiškai sujungta. Tai reiškia, kad grupių stačiakampiai gali pereiti lentelės kraštus. Kairiojo krašto langeliai yra gretimi dešiniojo krašto langeliams, viršutiniai langeliai – apatiniams. Pvz., visi keturi lentelės kampai laikomi gretimais.

Sugrupuokime Karno lentelės elementus į grupes. Optimalios grupės šioje lentelėje pavaizduotos skirtingomis spalvomis. Kai kuriose vietose jos dengia viena kitą. Elementai, priklausantys kelioms grupėms, pažymėti tarpinėmis spalvomis. Raudonąją grupę sudaro 4×1 dydžio stačiakampis, mėlynąją – 1×4 dydžio stačiakampis, žaliąją – 2×2 dydžio stačiakampis (nes lentelės kraštai – gretimi).

x_1x_2 \ x_3x_4	00	01	11	10
00			1	
01	1	1	1	1
11	1		1	1
10			1	

Grupių mintermai randami nustatant, kurie kintamieji nesikeičia konkrečioje grupėje. Pvz., žaliojoje grupėje:

- kintamasis x_1 grupėje neišlaiko pastovios būsenos (kaitaliojasi iš 0 į 1), todėl į mintermą neįrašomas;
- kintamasis x_2 visai grupei vienodas ir lygus 1, todėl į mintermą, vaizduojantį žaliąją grupę, įtraukiamas x_2;
- kintamasis x_3 taip pat kaitaliojasi, todėl nerašomas;
- kintamasis x_4 visai grupei vienodas ir lygus 0, todėl į mintermą įrašomas jo komponentas $ne - x_4$ ir įrašoma $\overline{x_4}$.

Taigi žaliosios grupės mintermas yra $x_2\overline{x_4}$.

Mėlynojoje grupėje x_1 ir x_2 nesikeičia, o x_3 ir x_4 kaitaliojasi. x_1 visuomet lygus 0, todėl jį rašome neigiamą $\overline{x_1}$. Taigi šios grupės mintermas yra $\overline{x_1}x_2$.

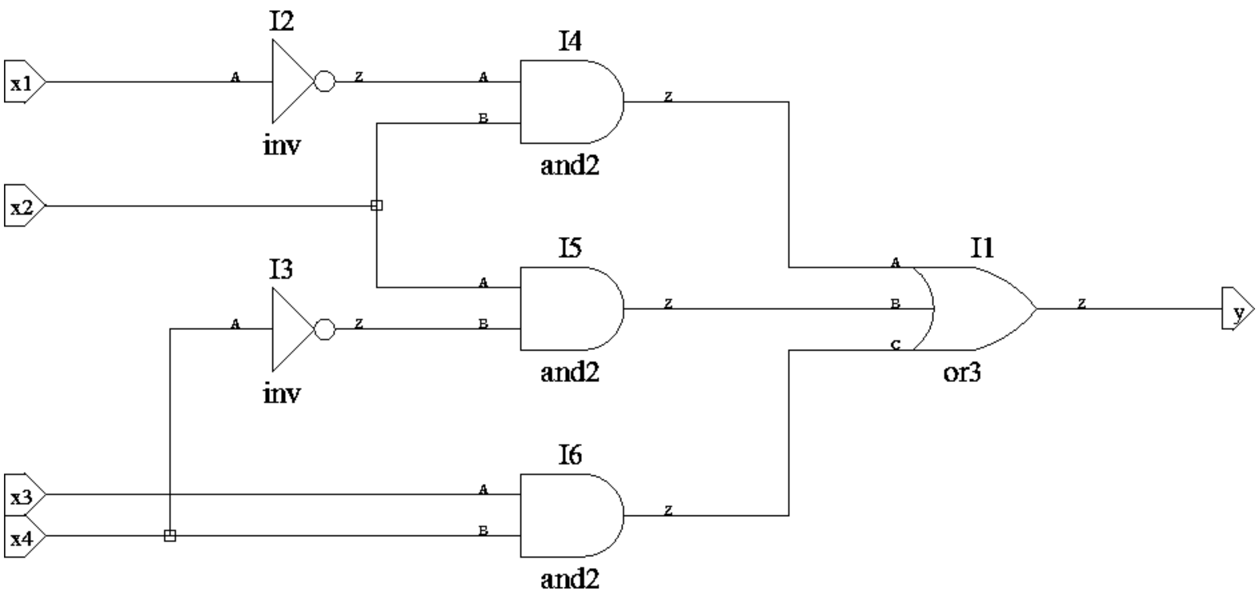
Raudonojoje grupėje x_3 ir x_4 nesikeičia ir yra lygus 1. x_1 bei x_2 kaitaliojasi, todėl gauname x_3x_4 .

Taigi pasinaudoję Karno diagrama (5) funkciją minimizavome:

$$y = x_2\overline{x_4} \cup \overline{x_1}x_2 \cup x_3x_4 \,. \tag{6}$$

Šią minimizuotą išraišką galima gauti ir nenaudojant Karno diagramos, o taikant Bulio algebros aksiomas, tačiau jei termų skaičius didelis, tai tampa labai sudėtinga.

Šią išraišką (kaip ir kiekvieną išraišką, užrašytą normaliąja disjunkcine forma) galima realizuoti naudojantis trijų tipų elementais: IR, ARBA ir NE. Kombinacinė schema, realizuojanti funkciją, pavaizduota 2 pav.



2 pav. Kombinacinė schema, gauta naudojant IR, ARBA, NE elementus

(6) funkciją užrašome su skliaustais:

$$y = x_2(\overline{x_1} \cup \overline{x_4}) \cup x_3 x_4 \,. \tag{7}$$

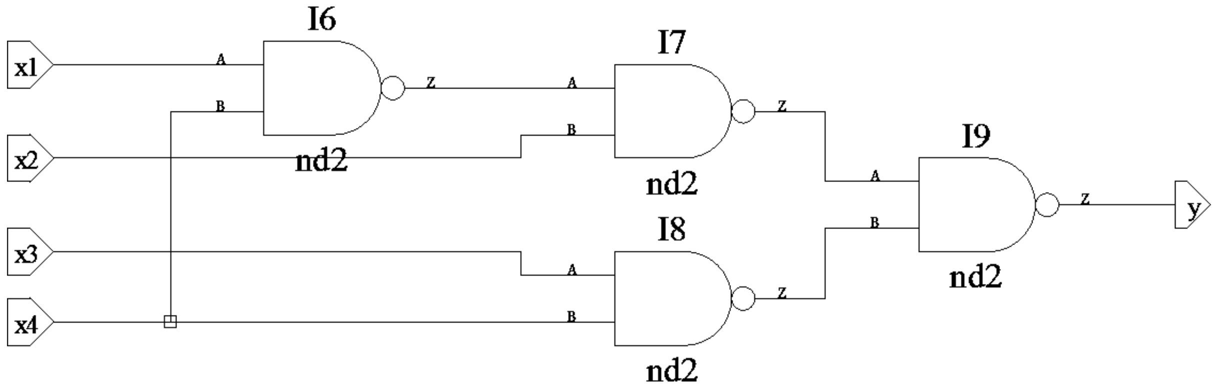
De Morgano dėsnis leidžia bet kokią funkciją išreikšti tik neigimo ir konjungcijos operacijomis, t. y. minimaliojoje elementų bazėje. De Morgano taisyklė užrašoma taip:

$$a \cup b = \overline{\overline{a} \cdot \overline{b}} \,. \tag{8}$$

Pritaikę De Morgano dėsnį (7) formulei, gauname:

$$y = \overline{\overline{x_2 \cdot \overline{x_1 x_4}} \cdot \overline{x_3 x_4}} \,. \tag{9}$$

Taip pertvarkytą funkciją lengva realizuoti naudojant tik IR-NE elementus. Kombinacinė schema, gauta naudojant IR-NE loginius elementus, parodyta 3 pav.



3 pav. Kombinacinė schema, gauta naudojant IR-NE loginius elementus

Matome, kad 2 pav. pateiktoje schemoje naudojami trijų tipų elementai, o 3 pav. pateiktai schemai reikia tik vieno tipo elementų.

Ar kombinacinė schema veikia gerai, galima patikrinti atlikus kontrolinį testą: schemos įvestims pateikus argumentų kombinacijų sekas ir patikrinus, ar schemos išvestys įgyja teisingumo lentelėje nurodytas reikšmes.

3. Laboratorinio darbo užduotis

Užduočių variantų lentelėje duotos funkcijos, kurių argumentų konjunkcijos pateiktos skaičiais. Kiekvienas studentas gauna jam priklausančio varianto numerį. Atlikti užduočiai reikia:

1. Užrašyti pateiktą funkciją normaliąja disjunkcine forma;
2. Minimizuoti pateiktą funkciją;
3. Realizuoti šią funkciją trimis būdais:
   (a) naudojant IR, ARBA, NE elementus,
   (b) naudojant tik IR-NE arba ARBA-NE ir NE elementus,
   (c) naudojant multiplekserį ir reikiamus IR, ARBA, NE, IR-NE, ARBA-NE elementus;
4. Patikrinti suprojektuotų schemų funkcionavimą;
5. Paruošti laboratorinio darbo ataskaitą. Ataskaitoje pateikti funkcijos minimizavimo rezultatus, realizuotas schemas bei šių schemų modeliavimo rezultatus.

4. Pavyzdys

Funkcijos argumentų konjunkcijos pateiktos šiais skaičiais:

$$f = 1, 2, 3, 9, 17, 20, 21, 22, 23, 25, 33, 41, 49, 57, 61, 63. \tag{10}$$

1. Užrašome šią funkciją tobula normaliąja disjunkcine forma (kiekvieno skaičiaus dvejetainis kodas atvaizduos mintermą):

$$\begin{aligned} f = {} & \overline{x_1x_2x_3x_4x_5}x_6 \cup \overline{x_1x_2x_3x_4}x_5\overline{x_6} \cup \overline{x_1x_2x_3x_4}x_5x_6 \cup \overline{x_1x_2}x_3\overline{x_4x_5}x_6 \cup \\ & \cup \overline{x_1}x_2\overline{x_3x_4x_5}x_6 \cup \overline{x_1}x_2\overline{x_3}x_4\overline{x_5x_6} \cup \overline{x_1}x_2\overline{x_3}x_4\overline{x_5}x_6 \cup \overline{x_1}x_2\overline{x_3}x_4x_5\overline{x_6} \cup \\ & \cup \overline{x_1}x_2\overline{x_3}x_4x_5x_6 \cup \overline{x_1}x_2x_3\overline{x_4x_5}x_6 \cup x_1\overline{x_2x_3x_4x_5}x_6 \cup x_1\overline{x_2}x_3\overline{x_4x_5}x_6 \cup \\ & \cup x_1x_2\overline{x_3x_4x_5}x_6 \cup x_1x_2x_3\overline{x_4x_5}x_6 \cup x_1x_2x_3x_4\overline{x_5}x_6 \cup x_1x_2x_3x_4x_5x_6. \end{aligned} \tag{11}$$

2. Sudarome Karno lentelę ir ją užpildome funkcijos reikšmėmis.

$x_4x_5x_6$ / $x_1x_2x_3$	000	001	011	010	110	111	101	100
000		1	1	1				
001		1						
011		1						
010		1			1	1	1	1
110		1						
111		1				1	1	
101		1						
100		1						

Pasinaudoję Karno lentele gausime minimizuotą funkcijos išraišką (į grupes jungiami mintermai nuspalvinti):

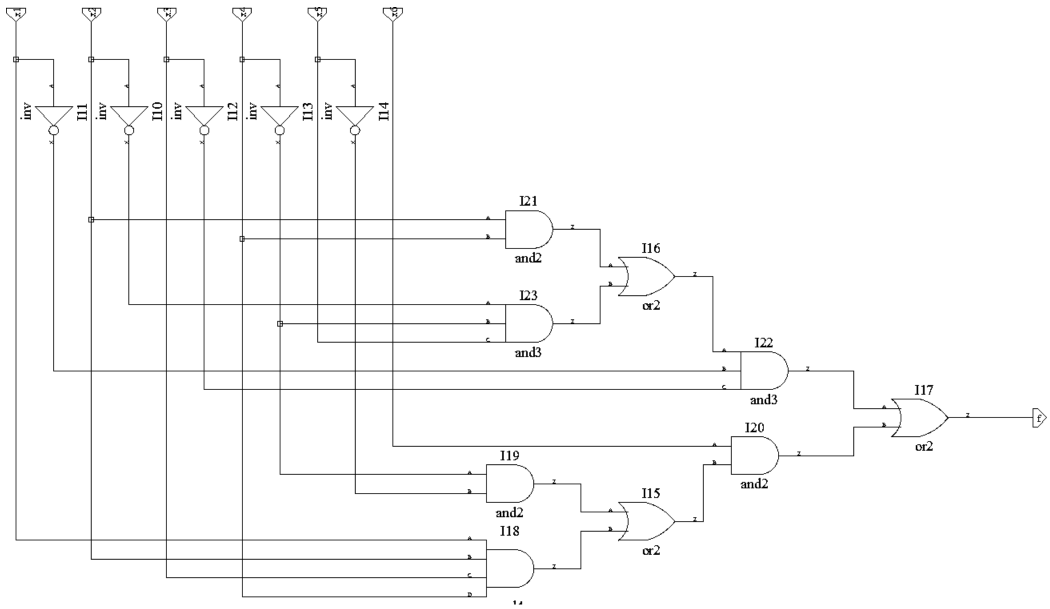
$$f = \overline{x_1x_2x_3x_4}x_5 \cup \overline{x_4x_5}x_6 \cup \overline{x_1}x_2\overline{x_3}x_4 \cup x_1x_2x_3x_4x_6. \tag{12}$$

Minimizuotą funkciją galima supaprastinti, iškeliant reikiamus kintamuosius prieš skliaustus.

$$f = \overline{x_1}\overline{x_2}\overline{x_3}\overline{x_4}x_5 \cup \overline{x_4}\overline{x_5}x_6 \cup \overline{x_1}x_2\overline{x_3}x_4 \cup x_1x_2x_3x_4x_6 =$$
$$\overline{x_1}\overline{x_3}(\overline{x_2}\overline{x_4}x_5 \cup x_2x_4) \cup x_6(\overline{x_4}\overline{x_5} \cup x_1x_2x_3x_4). \qquad (13)$$

3. Schemos realizacija:

   (a) Šios funkcijos realizacija naudojant IR, ARBA, NE elementus parodyta 4 pav.



4 pav. Funkcijos realizacija IR, ARBA, NE elementais

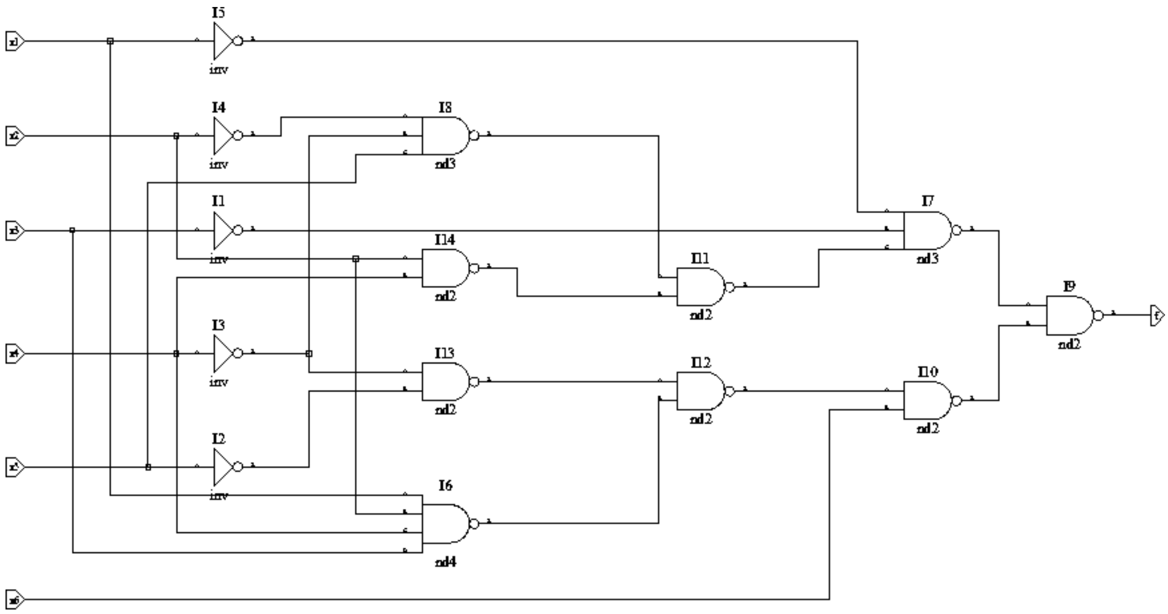
   (b) Projektuojant antrąją schemą, kai naudojami tik IR-NE arba ARBA-NE ir NE elementai, realizuojamoji funkcija užrašoma taip:

$$f = \overline{x_1}\overline{x_3}(\overline{x_2}\overline{x_4}x_5 \cup x_2x_4) \cup x_6(\overline{x_4}\overline{x_5} \cup x_1x_2x_3x_4). \qquad (14)$$

Pertvarkome funkcijos išraišką naudodamiesi De Morgano dėsniu $a \cup b = \overline{\overline{a} \cdot \overline{b}}$. Atlikę reikiamus pertvarkymus gausime:

$$
\begin{aligned}
f &= \overline{x_1x_3}(\overline{x_2x_4}x_5 \cup x_2x_4) \cup x_6(\overline{x_4x_5} \cup x_1x_2x_3x_4) = \\
&= \overline{x_1x_3} \cdot \overline{\overline{\overline{x_2x_4}x_5} \cdot \overline{x_2x_4}} \cup x_6 \cdot \overline{\overline{\overline{x_4x_5}} \cdot \overline{x_1x_2x_3x_4}} = \\
&= \overline{\overline{\overline{x_1x_3} \cdot \overline{\overline{\overline{x_2x_4}x_5} \cdot \overline{x_2x_4}}} \cdot \overline{x_6 \cdot \overline{\overline{\overline{x_4x_5}} \cdot \overline{x_1x_2x_3x_4}}}}.
\end{aligned}
\tag{15}
$$

Realizuojame nagrinėjamą funkciją naudodami tik IR-NE elementus, kaip parodyta 5 pav.



5 pav. Funkcijos realizacija IR-NE elementais

(c) Projektuojant schemą, naudojančią multiplekserį, reikia:

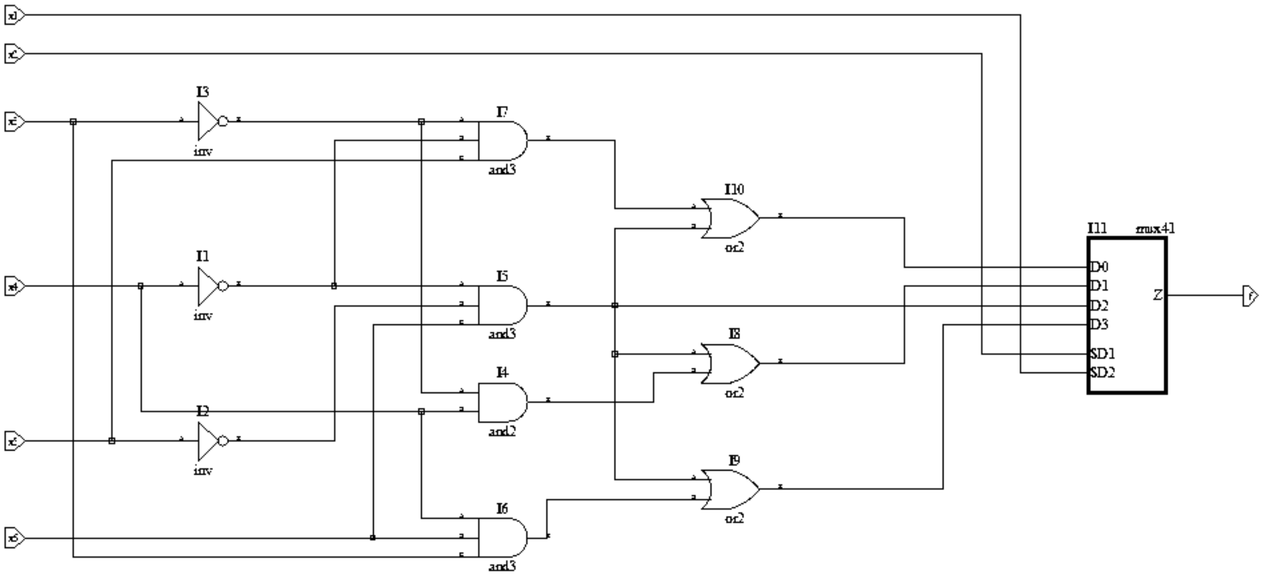
- iš normaliosios disjunkcinės išraiškos išskirti adresinius kintamuosius, kurių skaičius priklauso nuo to, kiek adresų turi pasirinktas multiplekseris. Šiame pavyzdyje kaip adresiniai kintamieji panaudoti x_1 ir x_2;
- šie kintamieji iškeliami prieš skliaustus ir gaunama:

$$
\begin{aligned}
f = {} & \overline{x_1x_2}(\overline{x_3x_4x_5}x_6 \cup \overline{x_3x_4}x_5\overline{x_6} \cup x_3\overline{x_4x_5}x_6 \cup \overline{x_3x_4}x_5x_6) \cup \\
& \overline{x_1}x_2(\overline{x_3x_4x_5}x_6 \cup \overline{x_3}x_4\overline{x_5x_6} \cup \overline{x_3}x_4\overline{x_5}x_6 \cup \overline{x_3}x_4x_5\overline{x_6} \cup \overline{x_3}x_4x_5x_6 \cup x_3\overline{x_4x_5}x_6) \cup \\
& x_1\overline{x_2}(\overline{x_3x_4x_5}x_6 \cup x_3\overline{x_4x_5}x_6) \cup \\
& x_1x_2(x_3\overline{x_4x_5}x_6 \cup x_3x_4\overline{x_5}x_6 \cup \overline{x_3x_4x_5}x_6 \cup x_3x_4x_5x_6).
\end{aligned}
\tag{16}
$$

- iškėlus adresinius kintamuosius prieš skliaustus, skliaustuose gautos liekamosios funkcijos, priklausančios nuo likusiųjų kintamųjų, minimizuojamos;
- realizuotos liekamosios funkcijos prijungiamos prie atitinkamų multiplekserio įvesčių.

Pastabos:

- Liekamųjų funkcijų sudėtingumas priklauso nuo pasirinktų adresinių kintamųjų.
- Jeigu kuri nors liekamoji funkcija gaunama lygi nuliui arba vienetui, atitinkama multiplekserio įvestis prijungiama prie loginio nulio arba loginio vieneto. Funkcijos realizacija su multiplekseriu pateikta 6 pav.

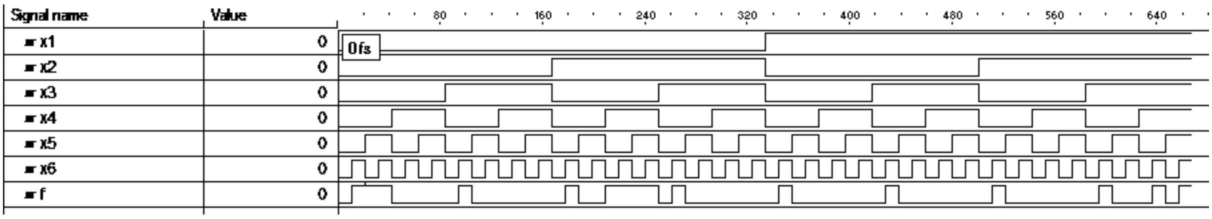


6 pav. Funkcijos realizacija su multiplekseriu

4. Testams generuoti panaudosime keletą VHDL kalbos procesų. Procesų VHDL kalbos teste kiekis neribotas, t. y. jų galima sukurti tiek, kiek reikia. Visi procesai vykdomi lygiagrečiai (vienu metu). Procesas yra begalinis: įvykdžius paskutinį priskyrimą vėl vykdomas pirmasis. Pavyzdyje testas sudarytas taip, kad būtų perrenkamos visos galimos įvesčių kombinacijos. Tam naudojami 6 procesai, atskirai valdantys kiekvieną įvestį. Pabaigoje testavimas stabdomas specialia komanda *assert*.

```
      -- Add your stimulus here ...
x6p: process
begin
      x6 <= '0';
      wait for 10 ns;
      x6 <= '1';
      wait for 10 ns;
end process;
x5p: process
begin
      x5 <= '0';
      wait for 20 ns;
      x5 <= '1';
      wait for 20 ns;
end process;
x4p: process
begin
      x4 <= '0';
      wait for 40 ns;
      x4 <= '1';
      wait for 40 ns;
end process;
x3p: process
begin
      x3 <= '0';
      wait for 80 ns;
      x3 <= '1';
      wait for 80 ns;
end process;
x2p: process
begin
      x2 <= '0';
      wait for 160 ns;
      x2 <= '1';
      wait for 160 ns;
end process;
x1p: process
begin
      x1 <= '0';
      wait for 320 ns;
      x1 <= '1';
      wait for 320 ns;
      assert false report "Pabaiga" severity failure;
end process;
```

Suvedus **.vhd* failą ir sukompiliavus testinius vektorius, gaunamos laikinės diagramos, parodytos 7 pav.



7 pav. Schemos modeliavimo laikinės diagramos

Matome, kad kai įvesčių reikšmės įgauna vieną iš užduotyje nurodytų reikšmių, išvesties f reikšmė įgyja vienetinę reikšmę. Visais kitais atvejais išvesties f reikšmė lygi 0. Iš laikinės diagramos matome, kad suprojektuota schema veikia tinkamai.