

**Kauno technologijos universitetas**

Informatikos fakultetas

Kombinacinės logikos schemos

P175B100 Skaitmeninės logikos pradmenų pirmas laboratorinis darbas

**Projekto autorius**

Gustas Klevinskas

**Akademinė grupė**

IFF-8/7

**Vadovai**

Doc. Tomas Adomkus

Kaunas, 2019

Turinys

[Įvadas 3](#_Toc3405876)

[Pagrindinė dalis 4](#_Toc3405877)

[Pirmoji schema (IR, ARBA, NE elementai) 5](#_Toc3405878)

[Antroji schema (tik ARBA-NE ir NE elementai) 7](#_Toc3405879)

[Trečioji schema (naudojant multiplekserį) 9](#_Toc3405880)

[Išvados 11](#_Toc3405881)

# Įvadas

Darbo tikslas – įsisavinti Būlio funkcijų minimizavimą ir kombinacinių loginių schemų projektavimą bei modeliavimą.

Užduotys:

1. Užrašyti pateiktą funkciją normaliąja disjunkcine forma;
2. Minimizuoti pateiktą funkciją;
3. Realizuoti šią funkciją trimis būdais:
   1. naudojant IR, ARBA, NE elementus;
   2. naudojant tik IR-NE arba ARBA-NE ir NE elementus, panaudojant De Morgano dėsnį;
   3. naudojant multiplekserį ir reikiamus IR, ARBA, NE, IR-NE, ARBA-NE elementus.
4. Patikrinti suprojektuotų schemų funkcionavimą.

# Pagrindinė dalis

Skaičių seka, kurią turės patikrinti mano schema: 0, 2, 11, 12, 18, 20, 22, 23, 24, 26, 28, 29, 30, 36, 37, 39, 45, 53 (323 užduotis).

Sutrumpinta teisingumo lentelė:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | x1 | x2 | x3 | x4 | x5 | x6 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 1 | 0 |
| 11 | 0 | 0 | 1 | 0 | 1 | 1 |
| 12 | 0 | 0 | 1 | 1 | 0 | 0 |
| 18 | 0 | 1 | 0 | 0 | 1 | 0 |
| 20 | 0 | 1 | 0 | 1 | 0 | 0 |
| 22 | 0 | 1 | 0 | 1 | 1 | 0 |
| 23 | 0 | 1 | 0 | 1 | 1 | 1 |
| 24 | 0 | 1 | 1 | 0 | 0 | 0 |
| 26 | 0 | 1 | 1 | 0 | 1 | 0 |
| 28 | 0 | 1 | 1 | 1 | 0 | 0 |
| 29 | 0 | 1 | 1 | 1 | 0 | 1 |
| 30 | 0 | 1 | 1 | 1 | 1 | 0 |
| 36 | 1 | 0 | 0 | 1 | 0 | 0 |
| 37 | 1 | 0 | 0 | 1 | 0 | 1 |
| 39 | 1 | 0 | 0 | 1 | 1 | 1 |
| 45 | 1 | 0 | 1 | 1 | 0 | 1 |
| 53 | 1 | 1 | 0 | 1 | 0 | 1 |

Čia skirtingomis spalvomis sužymėtos skirtingos multiplekserio įvestys ir tos grupės adresiniai kintamieji (D0 – žaliai, D1 – geltonai, D2 – mėlynai, D3 – raudonai).

## Pirmoji schema (IR, ARBA, NE elementai)

Iš teisingumo lentelės gauti mintermai:

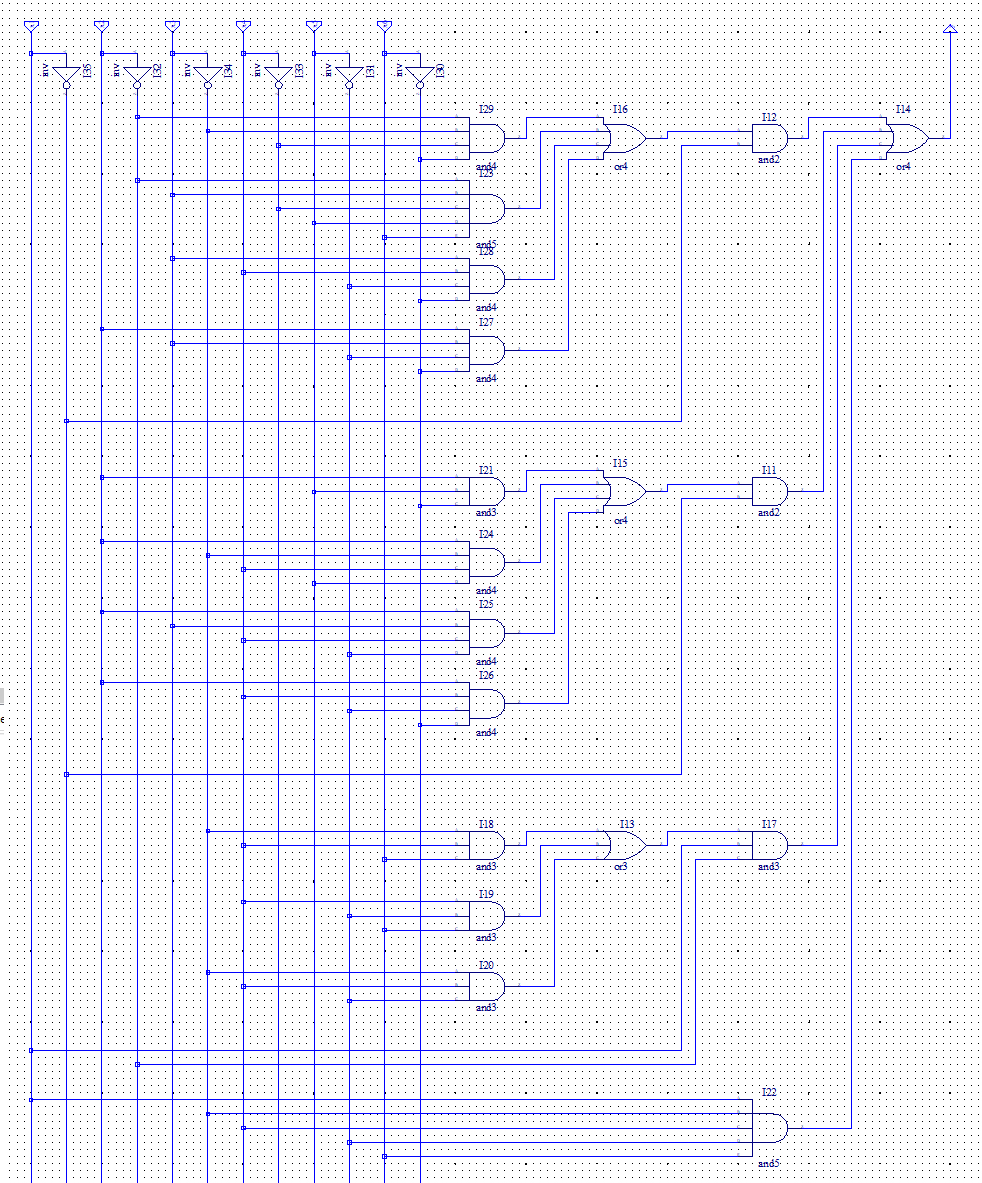
Karno diagrama:

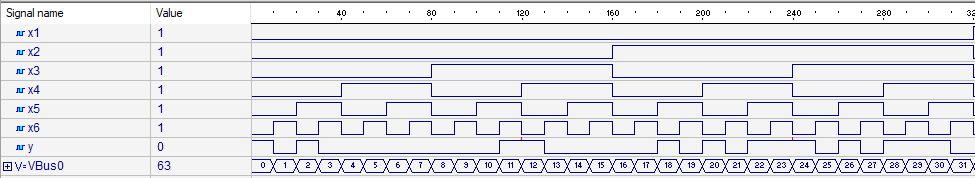
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| x4x5x6  x1x2x3 | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| 000 | 1 |  |  | 1 |  |  |  |  |
| 001 |  |  | 1 |  |  |  |  | 1 |
| 011 | 1 |  |  | 1 | 1 |  | 1 | 1 |
| 010 |  |  |  | 1 | 1 | 1 |  | 1 |
| 110 |  |  |  |  |  |  | 1 |  |
| 111 |  |  |  |  |  |  |  |  |
| 101 |  |  |  |  |  |  | 1 |  |
| 100 |  |  |  |  |  | 1 | 1 | 1 |

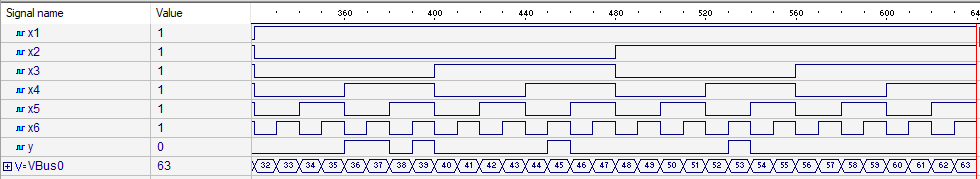
Iš Karno diagramos gauti mintermai:

Kadangi pasitaikė vienas elementas, kuris nesusiprastino Karno diagramoje ir liko sudarytas iš šešių signalų, mintermus reikia minimizuoti. Po iškėlimo prieš skliaustus mintermai atrodo taip:

Realizuota schema ir testai:



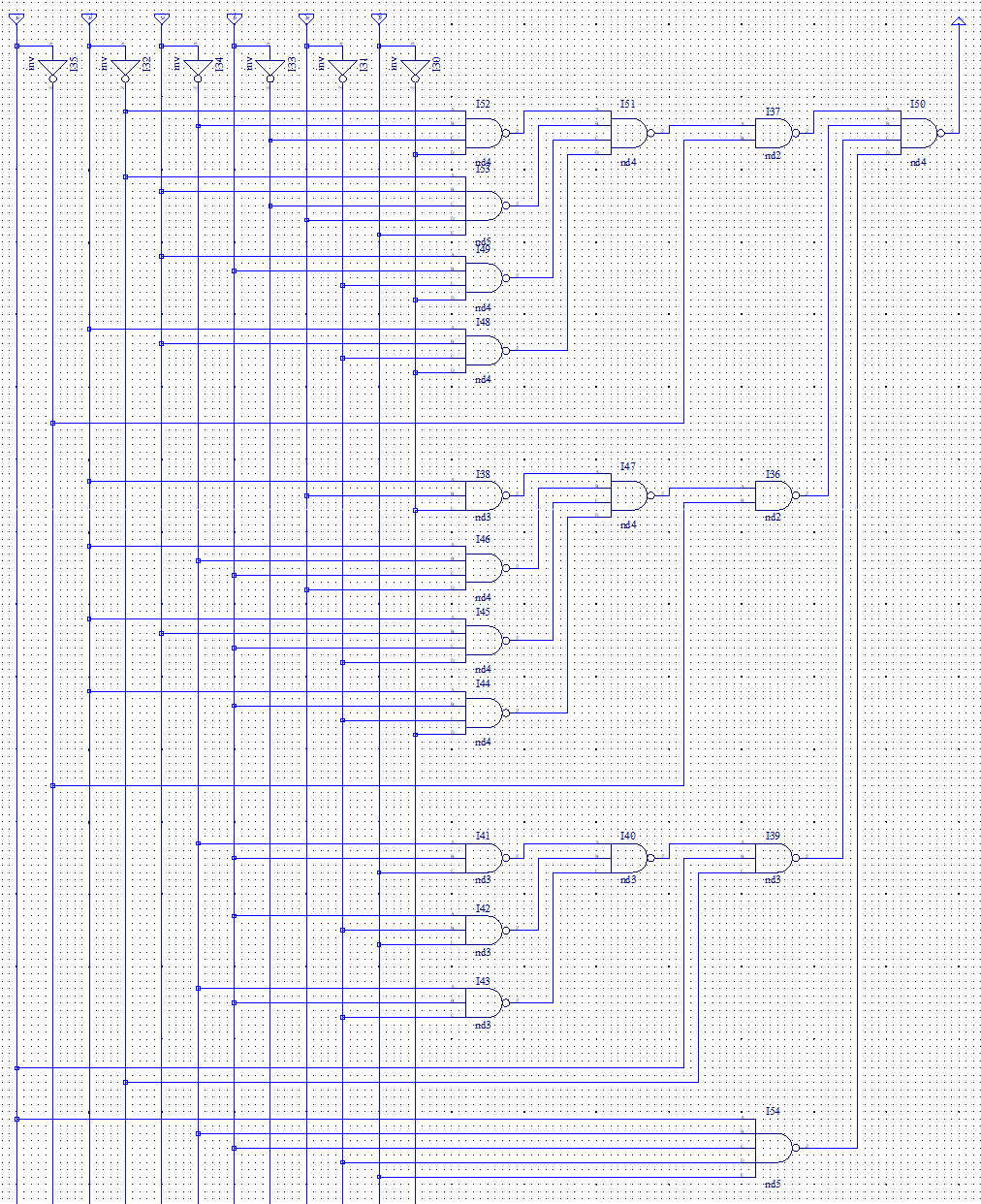


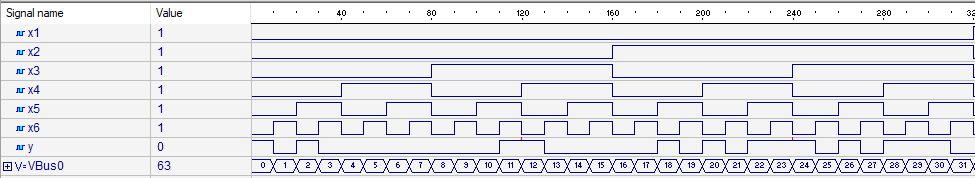


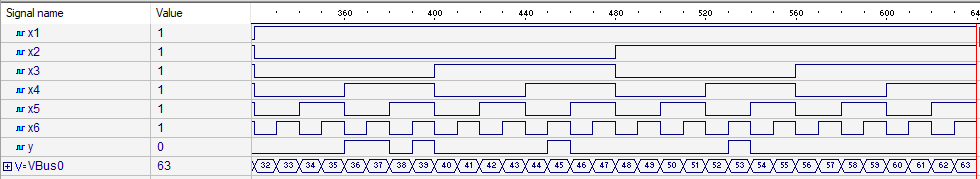
## Antroji schema (tik ARBA-NE ir NE elementai)

Pirmiausia, pasinaudojus De Morgano dėsniu, gauname mintermus, kuriuose realizuojama tik loginė daugyba:

Realizuota schema ir testai:







## Trečioji schema (naudojant multiplekserį)

Realizuojant schemą su multiplekseriu reikia išskirti adresinius kintamuosius. Aš parinkau x1 ir x2 kaip adresinius. Šiuos kintamuosius reikia išsikelti prieš skliaustus. Tuomet mano nauja išraiška atrodys taip:

Sudarome keturias Karno diagramas multiplekserio įvestims:

lentelė:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x5x6  x3x4 | 00 | 01 | 11 | 10 |
| 00 | 1 |  |  | 1 |
| 01 |  |  |  |  |
| 11 | 1 |  |  |  |
| 10 |  |  | 1 |  |

lentelė:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x5x6  x3x4 | 00 | 01 | 11 | 10 |
| 00 |  |  |  |  |
| 01 | 1 | 1 | 1 |  |
| 11 |  | 1 |  |  |
| 10 |  |  |  |  |

lentelė:

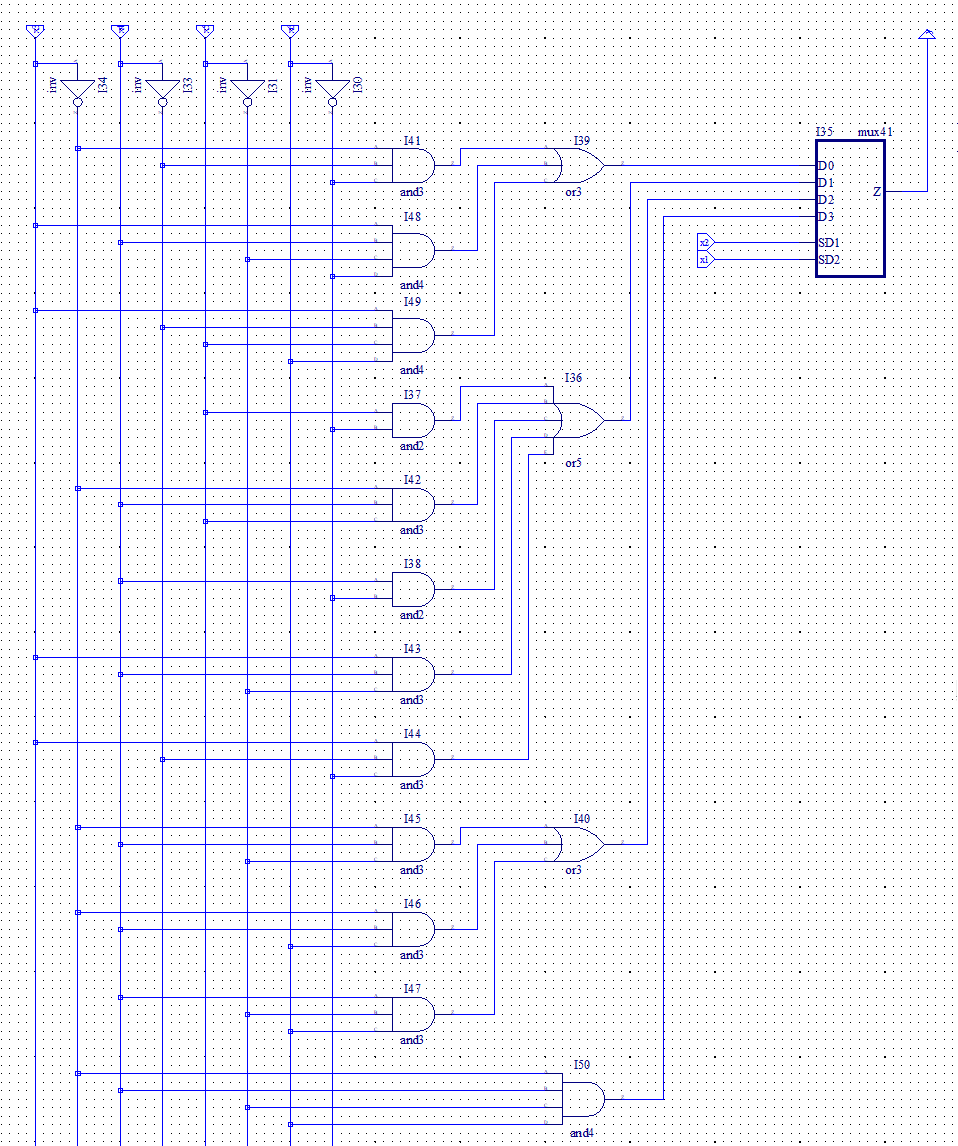
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x5x6  x3x4 | 00 | 01 | 11 | 10 |
| 00 |  |  |  | 1 |
| 01 | 1 |  | 1 | 1 |
| 11 | 1 | 1 |  | 1 |
| 10 | 1 |  |  | 1 |

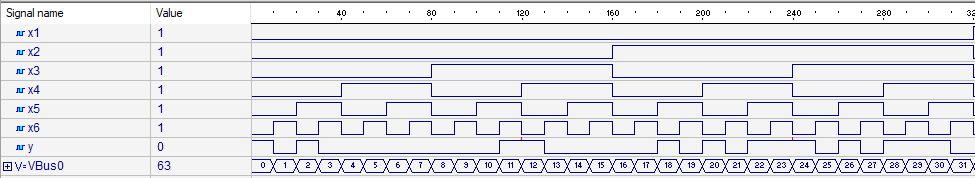
lentelė:

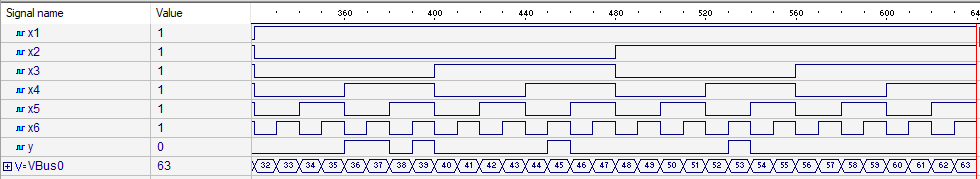
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x5x6  x3x4 | 00 | 01 | 11 | 10 |
| 00 |  |  |  |  |
| 01 |  | 1 |  |  |
| 11 |  |  |  |  |
| 10 |  |  |  |  |

Pasinaudojant Karno diagramomis, suprastiname kiekvieną grupę:

Realizuota schema ir testai:







# Išvados

Darbas buvo atliktas sėkmingai; testuojant kiekvieną schemą ji grąžino reikiamus rezultatus.

Tą pačią schemą galima realizuoti ne tik vienu būdų. Ją dar galima suprojektuoti pasinaudojus tik vieno tipo vartais. Tai reikalauja minimalių pakeitimų schemoje, tačiau produktas žymiai pigesnis, nes naudojami tik vieno tipo (šiuo atveju NAND) elementai. Taip pat schemą galima realizuoti pasinaudojus multiplekseriu, kas drastiškai sumažina schemos sudėtingumą.