 UNIVERSITATEA TRANSILVANIA BRAŞOV

Facultatea de Inginerie Electrică şi Știința Calculatoarelor

Departamentul de Automatică Și Informatică Aplicată

**PROIECT ASCN**

***Îndrumător: Student:***

***Prof. dr. ing. Moldoveanu Florin*** Vodă Dumitru Andrei

Grupa: 4432

**Cuprins**

Enunţul proiectului ..........................................................................................................3

Introducere în circuitele logice combinaţionale.....................................................................4

Tabelul adevăr .................................................................................................................5

Formele canonice disjunctive şi conjunctive ale funcţiilor......................................................6

Formele minime disjunctive şi conjunctive prin metoda diagramelor Karnaugh....................... 7

Minimizarea funcţiilor *f*  prin metoda Quine-McCluskey .......................................................9

Implementarea funcţiei f, numai cu porţi SI-NU (TTL) .........................................................17

Implementarea funcţiei f, numai cu porţi SAU-NU (CMOS)................................................19

Implementare functiei f cu MUX de 4 căi în tehnologie CMOS ............................................ 22

Implementare functiei f cu MUX de 16 căi în tehnologie CMOS..........................................25

Implementare funcţiei f cu DMUX de 16 căi cu porţi SI-NU............................................... 26

Implementare funcţiei f cu DMUX de 16 căi cu porţi SI.......................................................28

Timpii de propagare „intrare-iesire” pentru toate schemele logice obţinute.............................29

Puterile disipate pe toate schemele logice obtinute...............................................................30

Concluzii finale...............................................................................................................31

Bibliografie....................................................................................................................32

# TEMĂ PROIECT Nr. 5

Să se proiecteze un circuit logic combinaţional cu „vot majoritar”,cu cinci intrări (circuit care furnizează la ieşire valoarea logică a majorităţii variabilelor de intrare). Proiectul va cuprinde următoarele puncte:

a) Să se exprime funcţia logică ce defineşte circuitul cu FCD (forma canonică disjuncti-vă), FCC (forma canonică conjunctivă), tabel de adevăr şi diagramă Karnaugh.

b) Să se obţină formele minime disjunctivă şi conjunctivă ale funcţiei logice asociate circuitului, utilizându-se metoda diagramelor Karnaugh, respectiv metoda Quine-McCluskey.

c) Să se implementeze funcţia logică, numai cu porţi logice ŞI-NU (porţile logice sunt realizate în tehnologia TTL).

d) Să se implementeze funcţia logică numai cu porţi logice SAU-NU (porţile logice sunt realizate în tehnologia CMOS).

e) Să se implementeze funcţia logică cu MUX-uri de 4 respectiv 16 căi (circuitele sunt realizate în tehnologia CMOS).

f) Să se implementeze funcţia logică cu DMUX-uri de 8 respectiv 16 căi şi porţi logice ŞI-NU în prima variantă, respectiv ŞI în a doua variantă (toate circuitele sunt realizate în tehnologia TTL)

g) Să se calculeze timpii de propagare „intrare-ieşire”, pentru toate schemele logice obţinute.

h) Să se calculeze puterile disipate pentru toate schemele logice obţinute.

i) Să se compare soluţiile de implementare obţinute.

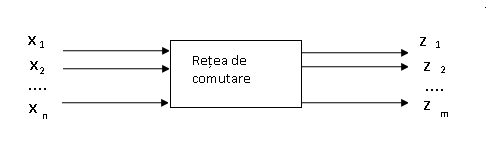
j) Se va face analiza, prin simulare, a tuturor schemelor logice obţinute utilizându-se pachetul de programe OrCAD.

Pe schemele logice obţinute se vor specifica tipul şi gradul de utilizare al fiecărui circuit integrat.

**Introducere**

Un circuit logic combinaţional (CLC) este un circuit de comutare care se caracterizează prin aceea că starea ieşirilor sale la un moment dat depinde numai de starea intrărilor sale la momentul considerat. Legătura între starea intrărilor şi starea ieşirilor circuitului este dată de funcţiile de transfer ale acestuia.

Schema bloc generală al unui CLC reprezentată printr-o rețea de comutare sau schema logica este următoarea:



x1,x2,...,xn -mărimi de intrare.

z1,z2,...,zm -mărimi de iesire.

Relaţii generale între aceste valori:

f(xi, …, xn) fi – functii logice

z1 = ……………. xi– variabile logice

zm = f(xi, …, xm)

Prin analiza unui CLC se înțelege determinarea expresiilor mărimilor de ieșire z1,z2,...,zm în funcție de mărimile de intrare x1,x2,...,xn si cunoscând configurația (numărul si tipul de elemente logice care intră în componență, modul de conectare, punctul în care se aplică variabilele de intrare, numărul de nivele logice etc.) si componența schemei logice.

Analiza rețelelor CLC realizate cu elemente logice de tip inversor (ŞI-NU, SAU-NU): numărul maxim de elemente logice aflate între intrarea si ieşirea rețelei determină numărul de nivele logice ale acestuia. Numerotarea lor se face de la ieșire către intrare. Un element logic ȘI-NU respectiv SAU-NU realizează operația SAU respectiv ȘI asupra variabilelor de intrare complementate dacă se află pe un nivel de inversare impar si operația ȘI respectiv SAU asupra variabilelor de intrare necomplementate dacă se află pe un nivel de inversoare impar.

Majoritatea sistemelor numerice necesită decodificarea informațiilor reprezentate prin cod mașină.Multe aplicații (de exemplu: multiplexarea datelor, multiplicarea proporțională, afișarea numerică, convertorul numeric analogic, adresarea memoriei) necesită decodificarea informației.

1. **Să se exprime funcţia logică ce defineşte circuitul cu FCD (forma canonică disjuncti-vă), FCC (forma canonică conjunctivă), tabel de adevăr şi diagramă Karnaugh.**

5 intrări -> 25 = 32 de combinaţii

1 iesire -> o singură funcţie

**Tabel de adevăr:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nr. | X1 | X2 | X3 | X4 | X5 | F |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| 3 | 0 | 0 | 0 | 1 | 1 | 0 |
| 4 | 0 | 0 | 1 | 0 | 0 | 0 |
| 5 | 0 | 0 | 1 | 0 | 1 | 0 |
| 6 | 0 | 0 | 1 | 1 | 0 | 0 |
| 7 | 0 | 0 | 1 | 1 | 1 | 1 |
| 8 | 0 | 1 | 0 | 0 | 0 | 0 |
| 9 | 0 | 1 | 0 | 0 | 1 | 0 |
| 10 | 0 | 1 | 0 | 1 | 0 | 0 |
| 11 | 0 | 1 | 0 | 1 | 1 | 1 |
| 12 | 0 | 1 | 1 | 0 | 0 | 0 |
| 13 | 0 | 1 | 1 | 0 | 1 | 1 |
| 14 | 0 | 1 | 1 | 1 | 0 | 1 |
| 15 | 0 | 1 | 1 | 1 | 1 | 1 |
| 16 | 1 | 0 | 0 | 0 | 0 | 0 |
| 17 | 1 | 0 | 0 | 0 | 1 | 0 |
| 18 | 1 | 0 | 0 | 1 | 0 | 0 |
| 19 | 1 | 0 | 0 | 1 | 1 | 1 |
| 20 | 1 | 0 | 1 | 0 | 0 | 0 |
| 21 | 1 | 0 | 1 | 0 | 1 | 1 |
| 22 | 1 | 0 | 1 | 1 | 0 | 1 |
| 23 | 1 | 0 | 1 | 1 | 1 | 1 |
| 24 | 1 | 1 | 0 | 0 | 0 | 0 |
| 25 | 1 | 1 | 0 | 0 | 1 | 1 |
| 26 | 1 | 1 | 0 | 1 | 0 | 1 |
| 27 | 1 | 1 | 0 | 1 | 1 | 1 |
| 28 | 1 | 1 | 1 | 0 | 0 | 1 |
| 29 | 1 | 1 | 1 | 0 | 1 | 1 |
| 30 | 1 | 1 | 1 | 1 | 0 | 1 |
| 31 | 1 | 1 | 1 | 1 | 1 | 1 |

Tabelul de adevăr este cea mai completă reprezentare a unei funcţii booleene deoarece pentru fiecare combinaţie posibilă a valorilor argumentelor se indică valoarea funcţiei.

Pentru reprezentarea unei funcții booleene se folosesc două forme de baza, numite *forme canonice:*

* forma canonică conjunctivă (FCC), unde termenii funcției sunt legți între ei prin ŞI, iar variabilele din cadrul unui termen sunt legate prin SAU;
* forma canonică disjunctivă (FCD), unde termenii funcției sunt legați între ei prin SAU, iar variabilele din cadrul unui termen sunt legate prin ŞI.

**Formele canonice disjunctive**

Pentru obținea formei canonice disjunctive din tabelul de adevăr se ia in considerare combinațiile pentru care valoarea funcției are valoarea 1.

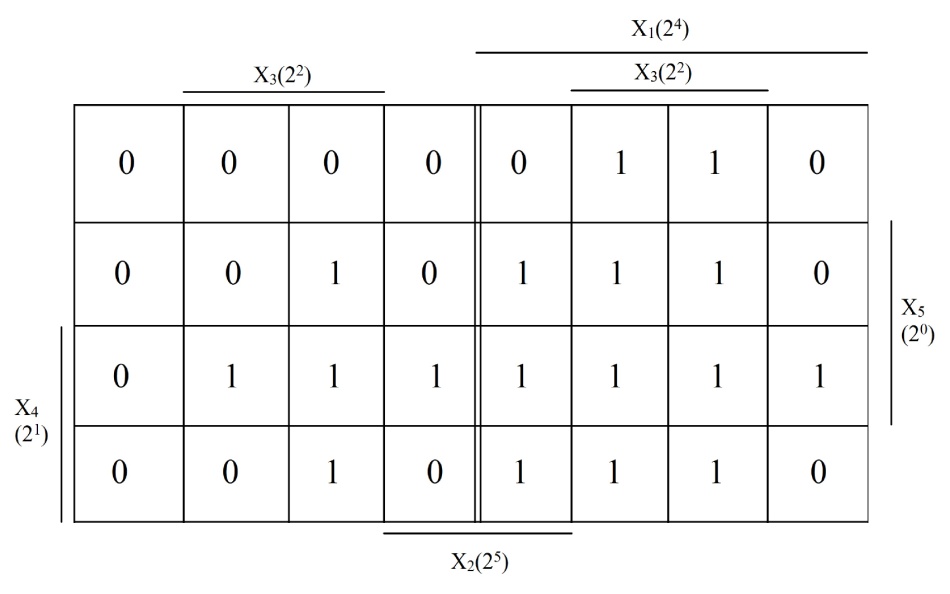


**Formele canonice conjunctive**

Pentru obținerea formelor canonice conjunctive din tabelul de adevăr se iau in considerare combinațiile pentru care functia are valoarea 0.



Diagrama Karnaugh:



1. **Să se obţină formele minime disjunctivă şi conjunctivă ale funcţiei logice asociate circuitului, utilizându-se metoda diagramelor Karnaugh, respectiv metoda Quine-McCluskey.**

**Diagrame Karnaugh**

Metoda diagramelor Karnaugh este o metodă grafo-analitică care pornește de la o formă canonică a funcției. Diagrama se prezinta sub forma unui pătrat sau dreptunghi având 2n locații.

În fiecare locație se va plasa un termen canonic așa incât in diagramă se vor găsi toți termenii canonici posibili.

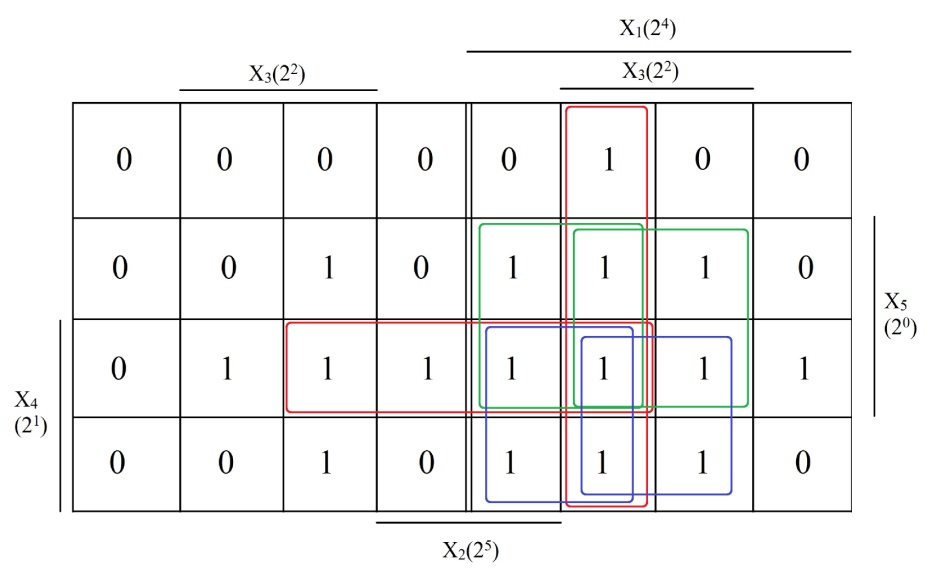
Diagrama K este construită astfel incât între două compartimente vecine pe linie sau pe coloană conțin termeni canonici care au proprietatea de adiacență. De asemenea se consideră vecine si au aceeași proprietate si locațiile situate la extremitățile liniilor si coloanelor.

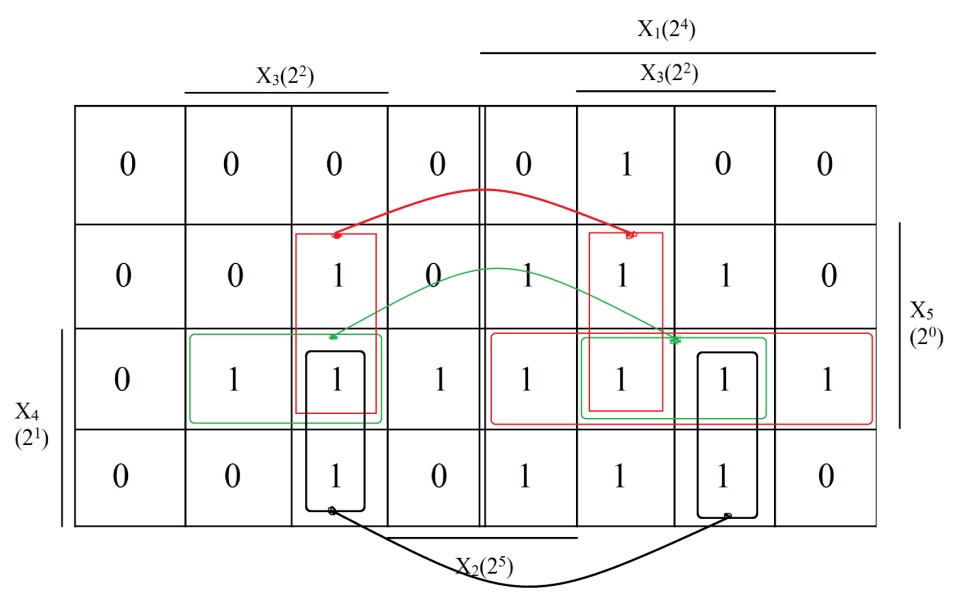
Diagramele K pentru mai mult de 4 variabile se obțin prin alăturarea pe linii respectiv coloane a unei variabile de 4 coloane considerate diagrame elementare.

În cazul nostru diagrama K are forma unui dreptunghi cu 32 locații.



fFMD  prin metoda diagramelor Karnaugh:

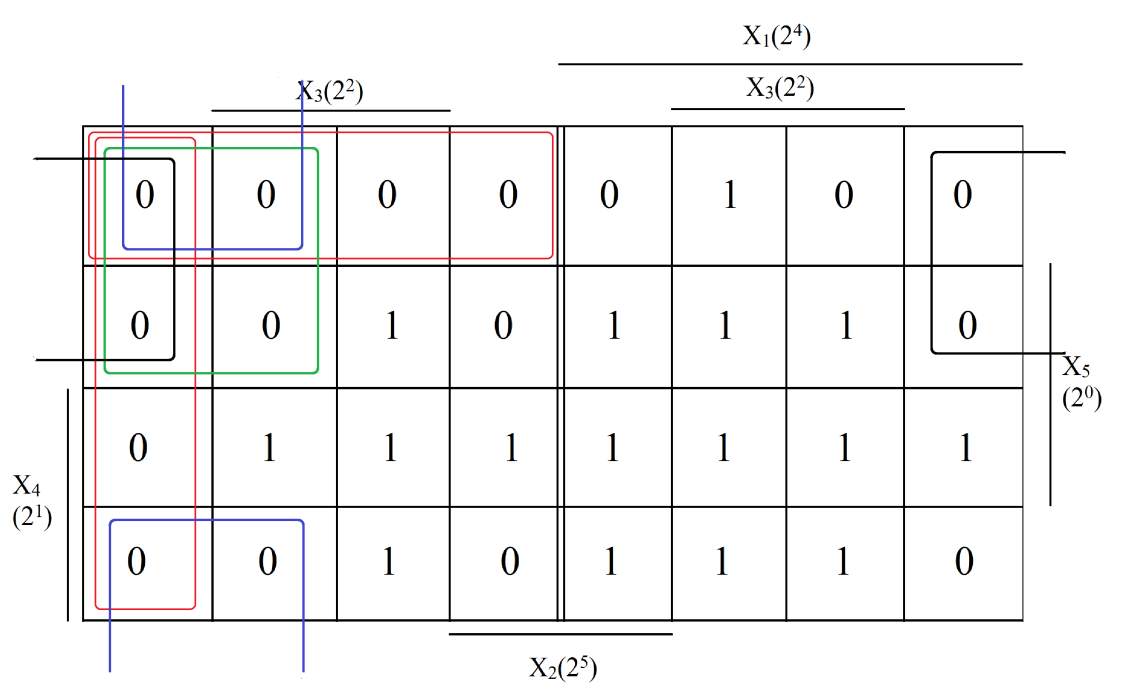


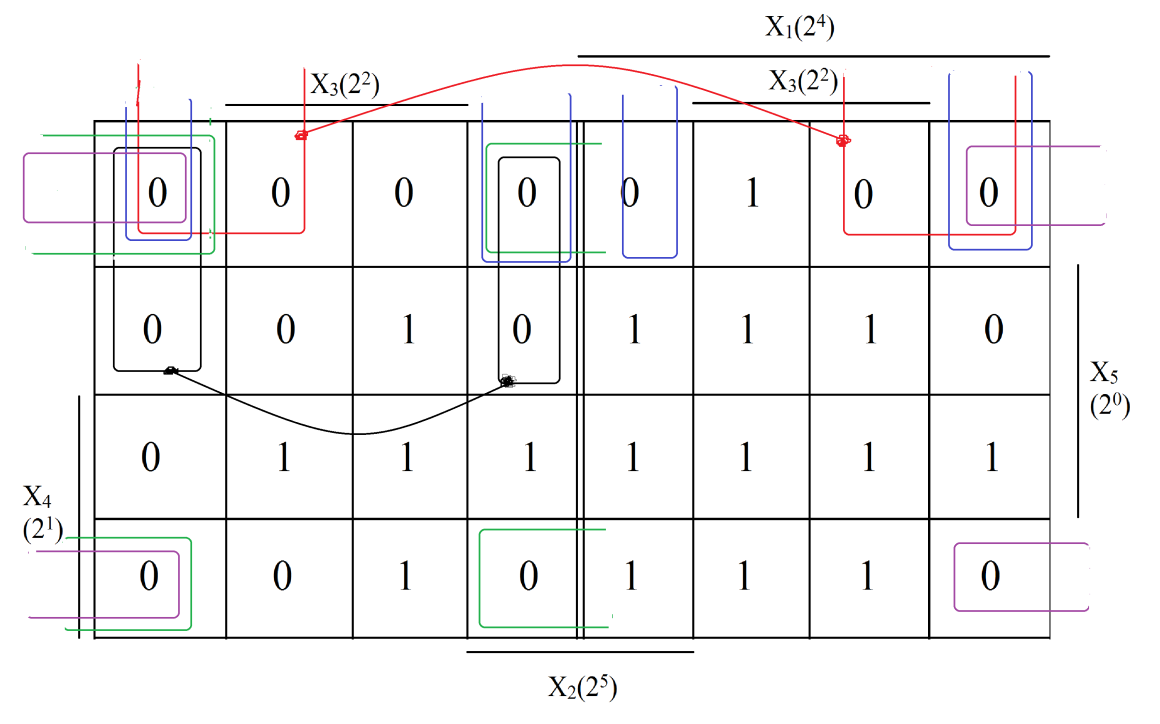




fFMC  prin metoda diagramelor Karnaugh:











Aplicam De Morgen:



**Forma minimă disjunctivă prin metoda Quine-McCluskey**

Aceasta metodă pornește de la forma canonică a funcției de minimizat.

Metoda are două etape:

* + se determină implicanții primi;
  + se selectează dintre implicanții primi obținuți doar aceia care acoperă toți termenii canonici ai funcției date si asigură realizarea acesteia la un cost minim.

Termenii canonici se compară în felul următor:

* + se compară fiecare termen canonic cu toți ceilalți;
  + când se găsesc doi termeni care au proprietatea de adiacență, variabila redundantă se elimina, obținându-se un termen elementar;
  + primul ciclu de comparații se consideră încheiat în momentul în care s-au comparat între ei toți termenii canonici, obținându-se toți implicanții primi posibili;
  + se compară între ei pe același criteriu termenii elementari obținuți;
  + se vor face atatea cicluri de comparație câte sunt necesare, pentru a nu mai exista termeni elementari cu proprietatea de adiacență.

Combinații indiferente se vor compara cu celelalte, dar nu se vor compara între ele.

Termenii elementari rămași necomparaţi în diverse etape sunt implicanți primi.

Selectarea implicanților primi care intră în FMD se face în felul următor:

* + la intersecția unei linii cu o coloană se va plasa un \* care va indica faptul că implicantul prim de pe linia respectivă va include termenul canonic de pe linia corespunzătoare.
  + dacă pe o linie apare un singur \*, implicantul prim de pe acea linie se numește implicant prim esential si va apărea în forma minimă disjunctivă în mod obligatoriu.



**Subcuburi 0-dimensionale**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| Grupa | Indici | X1 | X2 | X3 | X4 | X5 |  |
| 3 | 7 | 0 | 0 | 1 | 1 | 1 | ✓ |
|  | 11 | 0 | 1 | 0 | 1 | 1 | ✓ |
|  | 13 | 0 | 1 | 1 | 0 | 1 | ✓ |
|  | 14 | 0 | 1 | 1 | 1 | 0 | ✓ |
|  | 19 | 1 | 0 | 0 | 1 | 1 | ✓ |
|  | 21 | 1 | 0 | 1 | 0 | 1 | ✓ |
|  | 22 | 1 | 0 | 1 | 1 | 0 | ✓ |
|  | 25 | 1 | 1 | 0 | 0 | 1 | ✓ |
|  | 26 | 1 | 1 | 0 | 1 | 0 | ✓ |
|  | 28 | 1 | 1 | 1 | 0 | 0 | ✓ |
| 4 | 15 | 0 | 1 | 1 | 1 | 1 | ✓ |
|  | 23 | 1 | 0 | 1 | 1 | 1 | ✓ |
|  | 27 | 1 | 1 | 0 | 1 | 1 | ✓ |
|  | 29 | 1 | 1 | 1 | 0 | 1 | ✓ |
|  | 30 | 1 | 1 | 1 | 1 | 0 | ✓ |
| 5 | 31 | 1 | 1 | 1 | 1 | 1 | ✓ |

**Subcuburi 1-dimensionale**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Grupa | Indici | X1 | X2 | X3 | X4 | X5 |  |
| 3 | 7,15 | 0 | - | 1 | 1 | 1 | ✓ |
|  | 7,23 | - | 0 | 1 | 1 | 1 | ✓ |
|  | 11,15 | 0 | 1 | - | 1 | 1 | ✓ |
|  | 11,27 | - | 1 | 0 | 1 | 1 | ✓ |
|  | 13,15 | 0 | 1 | 1 | - | 1 | ✓ |
|  | 13,19 | - | 1 | 1 | 0 | 1 | ✓ |
|  | 14,15 | 0 | 1 | 1 | 0 | - | ✓ |
|  | 14,30 | - | 1 | 1 | 1 | 0 | ✓ |
|  | 19,23 | 1 | 0 | - | 1 | 1 | ✓ |
|  | 19,27 | 1 | - | 0 | 1 | 1 | ✓ |
|  | 21,23 | 1 | 0 | 1 | - | 1 | ✓ |
|  | 21,29 | 1 | - | 1 | 0 | 1 | ✓ |
|  | 22,23 | 1 | 0 | 1 | 1 | - | ✓ |
|  | 22,30 | 1 | - | 1 | 1 | 0 | ✓ |
|  | 25,27 | 1 | 1 | 0 | - | 1 | ✓ |
|  | 26,27 | 1 | 1 | 0 | 1 | - | ✓ |
|  | 26,30 | 1 | 1 | - | 1 | 0 | ✓ |
|  | 28,29 | 1 | 1 | 1 | 0 | - | ✓ |
|  | 28,30 | 1 | 1 | 1 | - | 0 | ✓ |
| 4 | 15,31 | - | 1 | 1 | 1 | 1 | ✓ |
|  | 23,31 | 1 | - | 1 | 1 | 1 | ✓ |
|  | 27,31 | 1 | 1 | - | 1 | 1 | ✓ |
|  | 29,31 | 1 | 1 | 1 | - | 1 | ✓ |
|  | 30,31 | 1 | 1 | 1 | 1 | - | ✓ |

**Subcuburi 2-dimensionale**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Grupa | Indici | X1 | X2 | X3 | X4 | X5 |  |
| 3 | 7,15,23,31 | - | - | 1 | 1 | 1 | ✓ |
|  | 7,23,15,31 | - | - | 1 | 1 | 1 | ✓ |
|  | 15,11,27,31 | - | 1 | - | 1 | 1 | ✓ |
|  | 11,27,15,31 | - | 1 | - | 1 | 1 | ✓ |
|  | 13,15,29,31 | - | 1 | 1 | - | 1 | ✓ |
|  | 13,29,15,31 | - | 1 | 1 | - | 1 | ✓ |
|  | 14,15,30,31 | - | 1 | 1 | 1 | - | ✓ |
|  | 14,30,15,31 | - | 1 | 1 | 1 | - | ✓ |
|  | 19,23,27,31 | 1 | - | - | 1 | 1 | ✓ |
|  | 19,27,23,31 | 1 | - | - | 1 | 1 | ✓ |
|  | 21,23,29,31 | 1 | - | 1 | - | 1 | ✓ |
|  | 21,29,23,31 | 1 | - | 1 | - | 1 | ✓ |
|  | 22,23,30,31 | 1 | - | 1 | 1 | - | ✓ |
|  | 22,30,23,31 | 1 | - | 1 | 1 | - | ✓ |
|  | 25,27,29,31 | 1 | 1 | - | - | 1 | ✓ |
|  | 25,23,27,31 | 1 | 1 | - | - | 1 | ✓ |
|  | 26,27,29,31 | 1 | 1 | - | - | 1 | ✓ |
|  | 26,30,27,31 | 1 | 1 | - | 1 | - | ✓ |
|  | 26,30,27,31 | 1 | 1 | - | 1 | - | ✓ |
|  | 28,29,30,31 | 1 | 1 | 1 | - | - | ✓ |
|  | 28,30,29,31 | 1 | 1 | 1 | - | - | ✓ |

**Tabelul acoperirilor:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| T.C.  I.P. | 7 | 11 | 13 | 14 | 15 | 19 | 21 | 22 | 23 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| x1x2x3 |  |  |  |  |  |  |  |  |  |  |  |  | \* | \* | \* | \* |
| x1x2x4 |  |  |  |  |  |  |  |  |  |  | \* | \* |  |  | \* | \* |
| x1x2x5 |  |  |  |  |  |  |  |  |  | \* |  | \* |  | \* |  | \* |
| x1x3x4 |  |  |  |  |  |  |  | \* |  |  |  |  |  |  | \* | \* |
| x1x3x5 |  |  |  |  |  |  | \* |  | \* |  |  | \* |  |  |  | \* |
| x1x4x3 |  |  |  |  |  | \* |  |  | \* |  |  | \* |  |  |  | \* |
| x2x3x4 |  |  |  | \* |  |  |  |  |  |  |  |  |  |  | \* | \* |
| x2x3x5 |  |  | \* |  | \* |  |  |  |  |  |  |  |  | \* |  | \* |
| x2x4x5 |  |  |  |  | \* |  |  |  |  |  |  | \* |  |  |  | \* |
| x3x4x5 | \* |  |  |  | \* |  |  |  | \* |  |  |  |  |  |  | \* |
|  | IPE |  | IPE | IPE |  | IPE | IPE | IPE |  | IPE | IPE |  | IPE |  |  |  |



**Forma minimă conjunctivă prin metoda Quine-McCluskey**

 **Subcuburi 0-dimensionale**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Grupa | Indici | X1 | X2 | X3 | X4 | X5 |  |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | ✓ |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 | ✓ |
|  | 2 | 0 | 0 | 0 | 1 | 0 | ✓ |
|  | 4 | 0 | 0 | 1 | 0 | 0 | ✓ |
|  | 8 | 0 | 1 | 0 | 0 | 0 | ✓ |
|  | 16 | 1 | 0 | 0 | 0 | 0 | ✓ |
| 2 | 3 | 0 | 0 | 0 | 1 | 1 | ✓ |
|  | 5 | 0 | 0 | 1 | 0 | 1 | ✓ |
|  | 6 | 0 | 0 | 1 | 1 | 0 | ✓ |
|  | 9 | 0 | 1 | 0 | 0 | 1 | ✓ |
|  | 10 | 0 | 1 | 0 | 1 | 0 | ✓ |
|  | 12 | 0 | 1 | 1 | 0 | 0 | ✓ |
|  | 17 | 1 | 0 | 0 | 0 | 1 | ✓ |
|  | 18 | 1 | 0 | 0 | 1 | 0 | ✓ |
|  | 20 | 1 | 0 | 1 | 0 | 0 | ✓ |
|  | 24 | 1 | 1 | 0 | 0 | 0 | ✓ |

**Subcuburi 1-dimensionale:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Grupa | Indici | X1 | X2 | X3 | X4 | X5 |  |
| 0 | 0,1 | 0 | 0 | 0 | 0 | - | ✓ |
|  | 0,2 | 0 | 0 | 0 | - | 0 | ✓ |
|  | 0,4 | 0 | 0 | - | 0 | 0 | ✓ |
|  | 0,8 | 0 | - | 0 | 0 | 0 | ✓ |
|  | 0,16 | - | 0 | 0 | 0 | 0 | ✓ |
| 1 | 1,3 | 0 | 0 | 0 | - | 1 | ✓ |
|  | 1,5 | 0 | 0 | - | 0 | 1 | ✓ |
|  | 1,9 | 0 | - | 0 | 0 | 1 | ✓ |
|  | 1,17 | - | 0 | 0 | 0 | 1 | ✓ |
|  | 2,3 | 0 | 0 | 0 | 1 | - | ✓ |
|  | 2,6 | 0 | 0 | - | 1 | 0 | ✓ |
|  | 2,10 | 0 | - | 0 | 1 | 0 | ✓ |
|  | 2,18 | - | 0 | 0 | 1 | 0 | ✓ |
|  | 4,5 | 0 | 0 | 1 | 0 | - | ✓ |
|  | 4,6 | 0 | 0 | 1 | - | 0 | ✓ |
|  | 4,12 | 0 | - | 1 | 0 | 0 | ✓ |
|  | 4,20 | - | 0 | 1 | 0 | 0 | ✓ |
|  | 8,9 | 0 | 1 | 0 | 0 | - | ✓ |
|  | 8,10 | 0 | 1 | 0 | - | 0 | ✓ |
|  | 8,12 | 0 | 1 | - | 0 | 0 | ✓ |
|  | 8,24 | - | 1 | 0 | 0 | 0 | ✓ |
|  | 16,17 | 1 | 0 | 0 | 0 | - | ✓ |
|  | 16,18 | 1 | 0 | 0 | - | 0 | ✓ |
|  | 16,20 | 1 | 0 | - | 0 | 0 | ✓ |
|  | 16,24 | 1 | - | 0 | 0 | 0 | ✓ |

**Subcuburi 2-dimensionale:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Grupa | Indici | X1 | X2 | X3 | X4 | X5 |  |
| 0 | 0,1,3,21 | 0 | 0 | 0 | - | - | ✓ |
|  | 0,1,4,5 | 0 | 0 | - | 0 | - | ✓ |
|  | 0,1,8,9 | 0 | - | 0 | 0 | - | ✓ |
|  | 0,1,16,17 | - | 0 | 0 | 0 | - | ✓ |
|  | 0,2,1,3 | 0 | 0 | 0 | - | - | ✓ |
|  | 0,2,4,6 | 0 | 0 | - | - | 0 | ✓ |
|  | 0,2,8,10 | 0 | - | 0 | - | 0 | ✓ |
|  | 0,2,16,18 | - | 0 | 0 | - | 0 | ✓ |
|  | 0,4,1,15 | 0 | 0 | - | 0 | - | ✓ |
|  | 0,4,2,6 | 0 | 0 | - | - | 0 | ✓ |
|  | 0,4,8,12 | 0 | - | - | 0 | 0 | ✓ |
|  | 0,4,16,20 | - | 0 | - | 0 | 0 | ✓ |
|  | 0,8,1,9 | 0 | - | 0 | 0 | - | ✓ |
|  | 0,8,2,10 | 0 | - | 0 | - | 0 | ✓ |
|  | 0,8,4,12 | 0 | - | - | 0 | 0 | ✓ |
|  | 0,8,16,24 | - | - | 0 | 0 | 0 | ✓ |
|  | 0,16,1,17 | - | 0 | 0 | 0 | - | ✓ |
|  | 0,16,2,18 | - | 0 | 0 | - | 0 | ✓ |
|  | 0,16,4,20 | - | 0 | - | 0 | 0 | ✓ |
|  | 0,16,8,24 | - | - | 0 | 0 | 0 | ✓ |

**Tabelul acoperirilor:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| T.C  I.P | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 8 | 9 | 10 | 12 | 16 | 17 | 18 | 20 | 24 |
|  | **\*** |  |  |  |  |  |  | **\*** |  |  |  | **\*** |  |  |  | **\*** |
|  | **\*** |  | **\*** |  | **\*** |  |  |  |  |  |  | **\*** |  |  | **\*** |  |
|  | **\*** |  | **\*** |  |  |  |  |  |  |  |  | **\*** |  | **\*** |  |  |
|  | **\*** | **\*** |  |  |  |  |  |  |  |  |  | **\*** | **\*** |  |  |  |
|  | **\*** |  |  |  | **\*** |  |  | **\*** |  |  | **\*** |  |  |  |  |  |
|  | **\*** |  | **\*** |  |  |  |  | **\*** |  | **\*** |  |  |  |  |  |  |
|  | **\*** | **\*** |  |  |  |  |  | **\*** | **\*** |  |  |  |  |  |  |  |
|  | **\*** |  | **\*** |  | **\*** |  | **\*** |  |  |  |  |  |  |  |  |  |
|  | **\*** | **\*** |  |  | **\*** | **\*** |  |  |  |  |  |  |  |  |  |  |
|  | **\*** | **\*** |  | **\*** |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | IPE |  | IPE |  | IPE | IPE |  | IPE | IPE | IPE |  |  | IPE | IPE | IPE |





Aplicam De Morgen:



1. **Să se implementeze funcţia logică, numai cu porţi logice ŞI-NU (porţile logice sunt realizate în tehnologia TTL).**

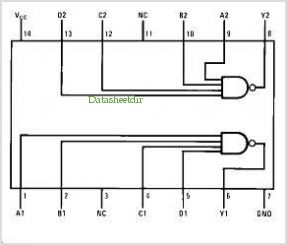
Implemantarea cu porți logice pornește de la una dintre formele minime ale funcției (FMD sau FMC).

Prin implementarea unei funcții booleene sau a unui sistem de funcții booleene se ințelege realizarea acesteia cu circuite integrete adică proiectarea si desenarea schemei logice utilizându-se simbolurile grafice asociate circuitelor integrate si calculul numărului necesar de circuite integrate.

TTL – Transistor transistor logic.

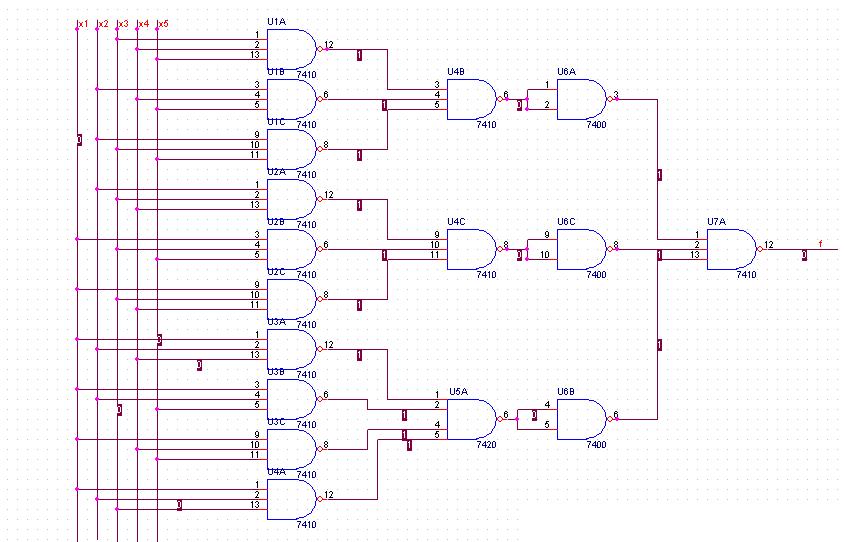
În cazul nostru se vor utiliza circuitele integrate 74LS00, 74LS10 si 74LS20 :

**** ****

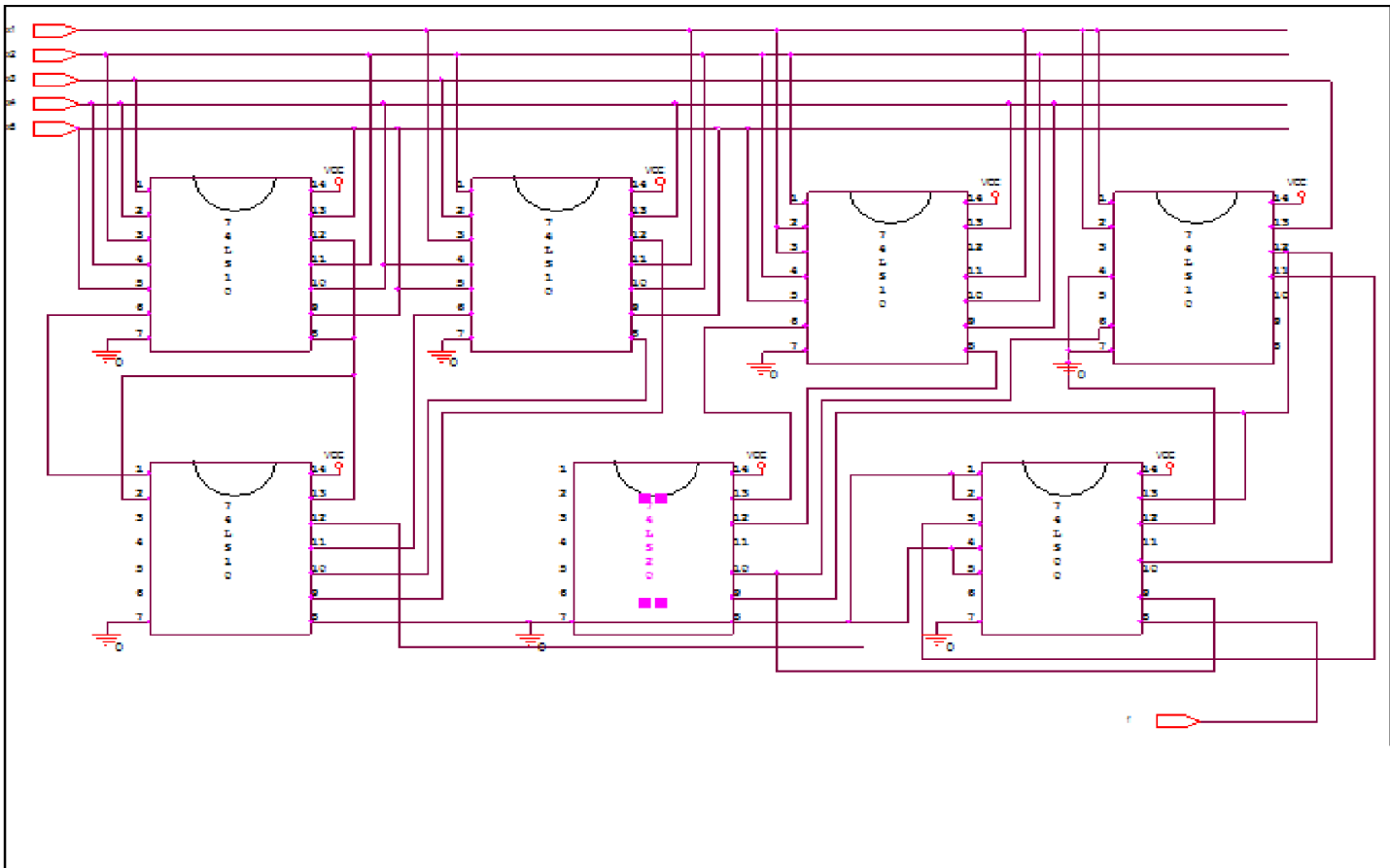


Implementarea functiei f:





Implementarea funcției Fa cu circuite integrate:



S-au folosit: 5\*74LS10 (-2)

1\*74LS20(-1)

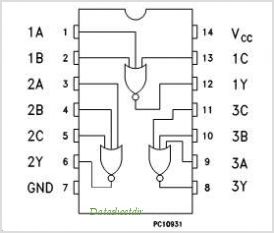
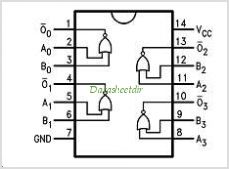
1\*74LS00(-)

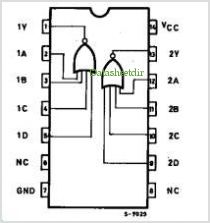
1. Să se implementeze funcţia logică numai cu porţi logice SAU-NU (porţile logice sunt realizate în tehnologia CMOS).

Implemantarea cu porți logice pornește de la una dintre formele minime ale funcției (FMD sau FMC).

Prin implementarea unei funcții booleene sau a unui sistem de funcții booleene se ințelege realizarea acesteia cu circuite integrete adică proiectarea si desenarea schemei logice utilizându-se simbolurile grafice asociate circuitelor integrate si calculul numărului necesar de circuite integrate.

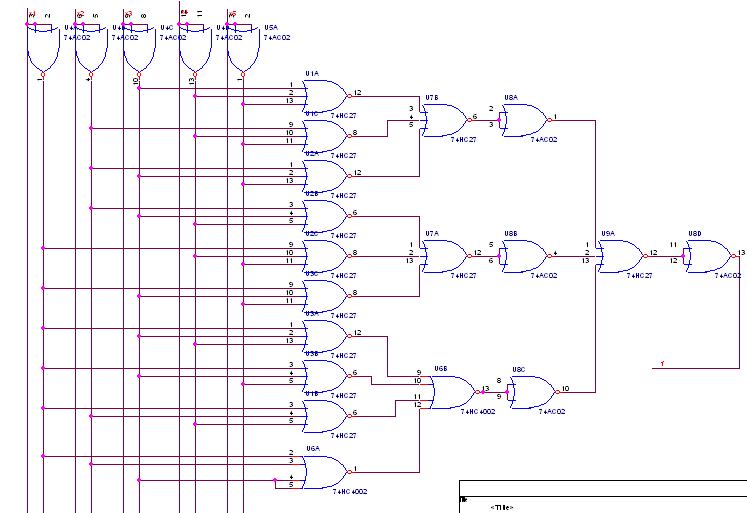
În cazul nostru se vor utiliza circuitele integrate 74AC02, 74HC4002N si 74HC27 :



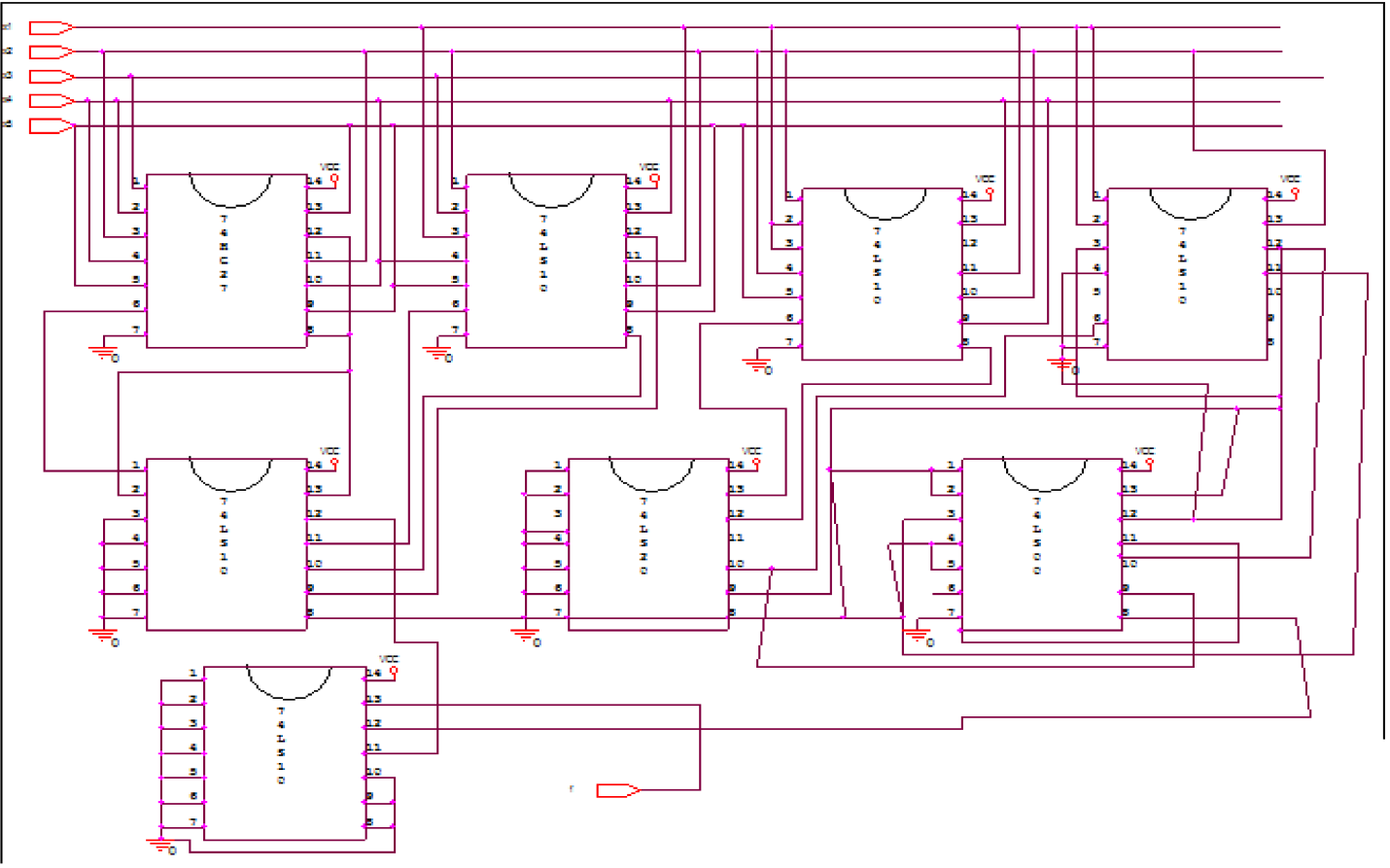


Implementarea funcției f :





Implementarea funcției f cu circuite integrate:



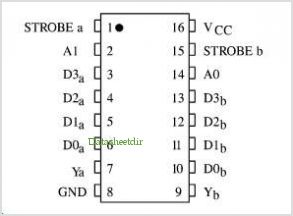
* + 5\*74AC02(-3)
  + 1\*74HC4002N(-)
  + 4\*74HC27(-)

1. **Să se implementeze ansamblul funcţiilor logice cu MUX-uri de 4 respectiv 16 căi (circuitele sunt realizate în tehnologia CMOS).**

Un circuit de multiplexare este un circuit logic combinational care, in cazul general, are 2n intrari de date (I2n-1 … I2 I1 I0), n intrari de selectie (S0 S1 … Sn+1) si o iesire (Z). Expresia iesirii Z la un moment dat este data de intrare Ik, k=0, … ,2n-1 unde k reprezinta echivalentul zecimal al numarului binar dat de starile 1 si 0 ale intrarilor de selectie: k=Sn-1, Sn-2, … , S1, S0.

Implementarea funcției fcu MUX de 4 căi

Circuite integrate folosite:



74HC153



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Echiv.zec. | X1 | X2 | X3 | X4 | X5 | Fa |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| 3 | 0 | 0 | 0 | 1 | 1 | 0 |
| 4 | 0 | 0 | 1 | 0 | 0 | 0 |
| 5 | 0 | 0 | 1 | 0 | 1 | 0 |
| 6 | 0 | 0 | 1 | 1 | 0 | 0 |
| 7 | 0 | 0 | 1 | 1 | 1 | 1 |
| 8 | 0 | 1 | 0 | 0 | 0 | 0 |
| 9 | 0 | 1 | 0 | 0 | 1 | 0 |
| 10 | 0 | 1 | 0 | 1 | 0 | 0 |
| 11 | 0 | 1 | 0 | 1 | 1 | 1 |
| 12 | 0 | 1 | 1 | 0 | 0 | 0 |
| 13 | 0 | 1 | 1 | 0 | 1 | 1 |
| 14 | 0 | 1 | 1 | 1 | 0 | 1 |
| 15 | 0 | 1 | 1 | 1 | 1 | 1 |
| 16 | 1 | 0 | 0 | 0 | 0 | 0 |
| 17 | 1 | 0 | 0 | 0 | 1 | 0 |
| 18 | 1 | 0 | 0 | 1 | 0 | 0 |
| 19 | 1 | 0 | 0 | 1 | 1 | 1 |
| 20 | 1 | 0 | 1 | 0 | 0 | 0 |
| 21 | 1 | 0 | 1 | 0 | 1 | 1 |
| 22 | 1 | 0 | 1 | 1 | 0 | 1 |
| 23 | 1 | 0 | 1 | 1 | 1 | 1 |
| 24 | 1 | 1 | 0 | 0 | 0 | 0 |
| 25 | 1 | 1 | 0 | 0 | 1 | 1 |
| 26 | 1 | 1 | 0 | 1 | 0 | 1 |
| 27 | 1 | 1 | 0 | 1 | 1 | 1 |
| 28 | 1 | 1 | 1 | 0 | 0 | 1 |
| 29 | 1 | 1 | 1 | 0 | 1 | 1 |
| 30 | 1 | 1 | 1 | 1 | 0 | 1 |
| 31 | 1 | 1 | 1 | 1 | 1 | 1 |

Configuraţia x1x2=00:

Pentru x3x4=00=> f=0, pentru ∀ x5 => 1C01=0

Pentru x3x4=01=> f=0, pentru ∀ x5 => 1C11=0

Pentru x3x4=10=> f=0, pentru ∀ x5 => 1C21=0

Pentru x3x4=11=> f=1, pentru x5=1 => 1C31=x5

Configuraţia x1x2=01:

Pentru x3x4=00=> f=0, pentru ∀ x5 => 2C02=0

Pentru x3x4=01=> f=1, pentru x5=1 => 2C12= x5

Pentru x3x4=10=> f=1, pentru x5=1=> 2C22 =x5

Pentru x3x4=11=> f=1, pentru ∀ x5 => 2C02=1

Configuraţia x1x2=10:

Pentru x3x4=00=> f=0, pentru ∀ x5 => 3C04=0

Pentru x3x4=01=> f=1, pentru x5=1 => 3C14=x5

Pentru x3x4=10=> f=1, pentru x5=1=> 3C24=x5

Pentru x3x4=11=> f=1, pentru x5=1 => 3C34=x5

Configuraţia x1x2=11:

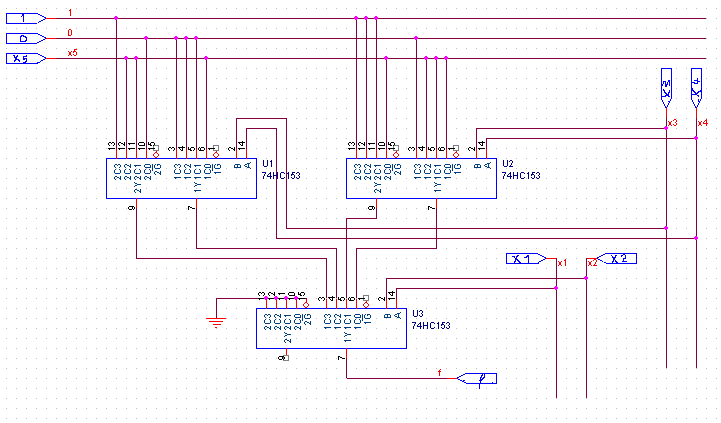
Pentru x3x4=00=> f=1, pentru x5=1=> 4C04=0

Pentru x3x4=01=> f=0, pentru ∀ x5 => 1C14=1

Pentru x3x4=10=> f=0, pentru ∀ x5 => 1C24=1

Pentru x3x4=11=> f=1, pentru x5=1 => 1C34=1

Implementarea funcţie f cu MUX de 4 căi cu porţi logice:

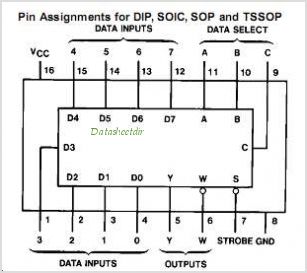


**Implementarea funcţiei logice f cu MUX de 16 căi**

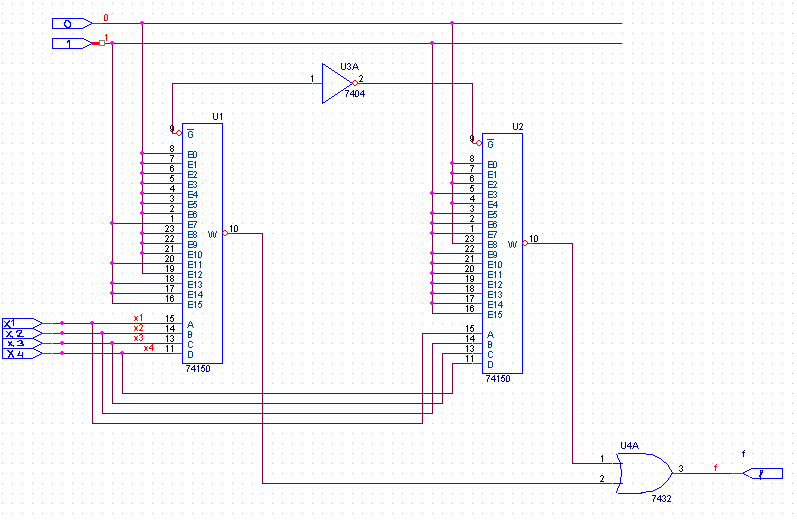
La implementarea cu MUX de 16 căi variabilele funcţiei se vor aplica pe intrările de date, deoarece circuitele au iesirile active in 0 logic de unde rezultă că pe intrarile de selectie se vor lega la 1 logic termenii cononici care nu sunt prezenti in expresia funcţiei iar termenii canonici prezenţi în expresia funcţiei se vor lega la 0 logic.

Pentru obţinerea unul MUX de 16 căi se vor utiliza doua multiplexoare de 8 căi (74hc151) conectaţi in paralel.

Circuit integrat folosit:



Implementarea funcţiei logice f cu 16 căi:



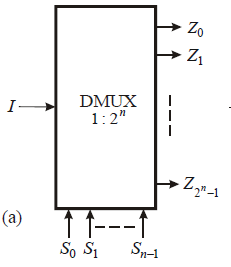
1. **Să se implementeze funcţia logică cu DMUX-uri de 8 respectiv 16 căi şi porţi logice ŞI-NU în prima variantă, respectiv ŞI în a doua variantă (toate circuitele sunt realizate în tehnologia TTL).**

Circuitele demultiplexoare (DMUX-urile) realizează operaţia inversă multiplexării şi

anume distribuie un semnal (0 sau 1) de pe o cale, pe mai multe căi, în funcţie de adresa

acelor căi, motiv pentru care se mai numesc şi circuite distribuitoare .

Demultiplexorul este un circuit logic combinaţional care are, în cazul general, o intrare de date *I*, *n* intrări de selecţie (adresă), *S*0 , *S*1,.., *Sn*−1 şi ieşiri,….



Orice funcţie logică de *n* variabile, dată sub FCD, se poate implementa cu ajutorul unui

DCD cu *n* intrări sau a unui DMUX 1: 2*n* , dacă la ieşirea acestor circuite termenii

canonici (nivelul logic ŞI), existenţi în tabelul de adevăr al funcţiei sunt introduşi într-o

poartă logică pentru realizarea nivelului logic SAU.

1. ŞI-NU, în acest caz se leagă la intrările porţilor ŞI-NU ieşirile DMUX corespunzătoare termenilor canonici prezentaţi în expresia funcţiei de implementat.
2. ŞI, în acest caz se leagă la intrările porţilor ŞI ieşirile DMUX corespunzătoare termenilor canonici care nu apar în expresia funcţiei (se vor considera funcţiile negate).

**Implementarea funcţiei logice f cu DMUX de 16 căi porţi SI-NU**





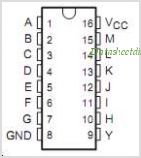
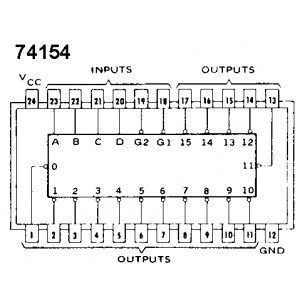




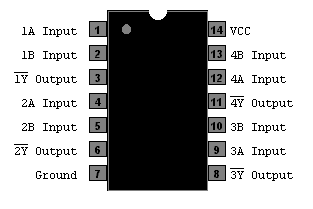
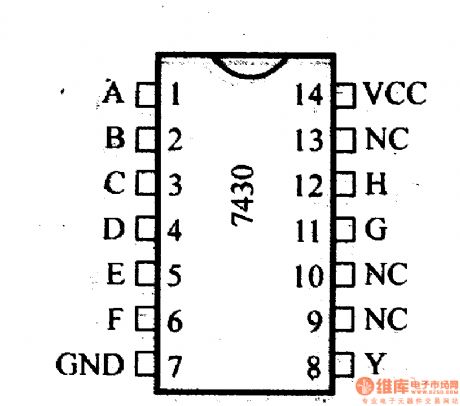


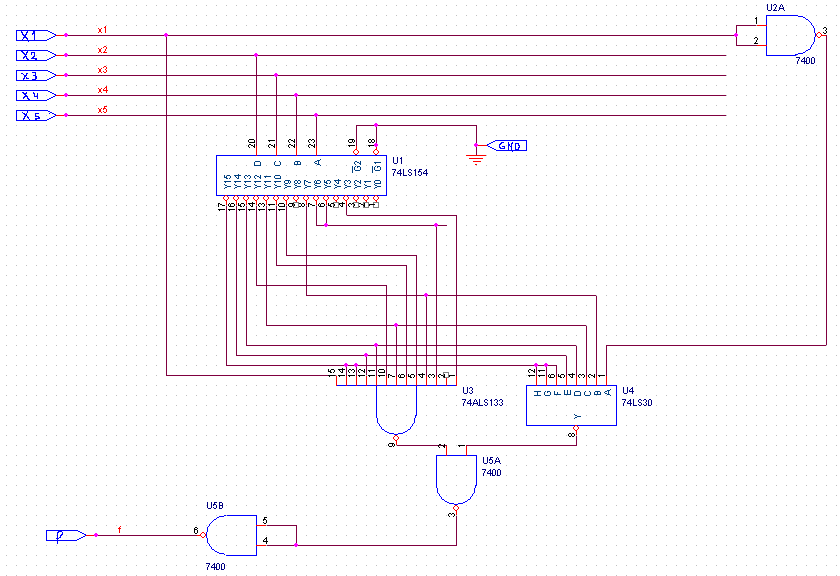
**Implementarea funcţiei f cu DMUX de 16 căi cu porţi SI-NU**

Circuite integrate folosite:



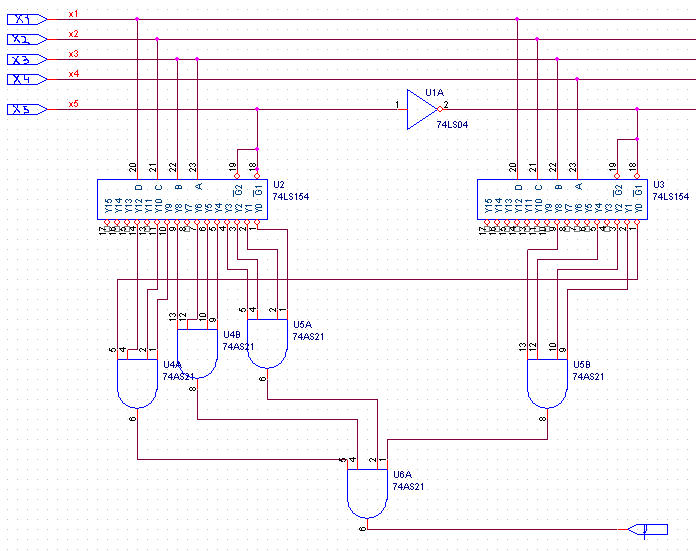
74LS154 74ALS133



****

**Implementarea functiei logice f cu DMUX de 16 cai cu poti SI**





1. **Sa se calculeze timpii de propagare „intrare-iesire”, pentru toate schemele logice obtinute.**

Timpii de propagare se vor calcula dupa formula:



Date de catalog pentru circuite folosite la T=25\* C si Vcc=5V.

Calculul timpilor de propagare pentru fiecare din schemele următoare:

* + Funcţia f cu porţi SI-NU(porţile sunt realizate în tehnologia TTL)

tp= 3\*tp 74ls10+tp 74ls00=3\*+ ==37,5 ns.

* + Funcţia f cu porţi SAU-NU (porţile sunt realizate in tehnologia CMOS)

tp=2\*tp 74hc02+3\*tp 74hc27=2\* + 3\*= 14+69=83 ns.

* + Funcţia f cu MUX de 4 căi(CMOS)

tP==3\* = 3\*=3\*38=114 ns.

* + Funcţia f cu MUX de 16 căi(CMOS)

tp =2\*td MUX+tp=2\* + =70+=70+18,5=88,5

* + Funcţia f cu DMUX de 16 căi cu porti SI-NU

tp=2\*tp 74ls00+tp 74ls133+tp 74ls154=11+7+25+14=57 ns.

* + Funcţia f cu DMUX de 16 căi cu porţi SI

TP=2\*TP 74LS21+tp 73ls154=25+28=53 ns.

1. **Să se calculeze puterile disipate pentru toate schemele logice obţinute.**

Se vor calcula puterile disipate pe fiecare circuit integrat cu formula:



* + Puterea disipata pentru implementarea funcţiei logice f folosind porţi SI-NU

Pd=3\*Pdcc 74ls10+Pdcc 74ls00=30+15=45 mW.

* + Puterea disipata pentru implementarea funcţiei logice f folosind SAU-NU

Pd=2\*Pdcc 74hc02+3\*Pdcc 74hc27=750+45=795 mW

* + Puterea disipata pentru implementarea funcţiei logice f cu 16 căi MUX

Pd=Pdcc 74hc32+Pdcc 74hc150=16+150=166 mW

* + Puterea disipata pentru implementarea funcţiei logice f cu 4 căi MUX

Pd=3\*Pdcc 74hc153=500mW

* + Puterea disipata pentru implementarea funcţiei logice f cu 16 căi DMUX cu porti SI-NU



* + Puterea disipata pentru implementarea funcţiei logice f cu 16 căi DMUX cu porţi SI



**Concluzii finale:**

Putem compara implementările obţinute din mai multe puncte de vedere.

Din punctul de vedere al complexităţii implementării schemei logice cea mai puţin complexă este schema cu un DMUX de 16 căi şi porţi ŞI-NU în tehnologie TTL (pag. 29) sau sau schema cu un MUX de 16 căi (pag. 24) şi probabil şi cele mai ieftină. La capitolul timp de propagare, schema cu MUX-uri de 16 căi CMOS este cea mai rapidă.(pag. 24) În schimb din punctul de vedere al consumul de energie, schemele cu MUX-uri sunt mari consumatoare de putere. Cea mai mică consumatoare, ţinând cont ca la unele subpuncte ale proiectului nu s-a implementat ansamblul funcţiilor ci numai una maxim două funcţii, este implementarea cu circuite CMOS (pag. 21 , MUX-urile în tehnologie CMOS ).

**Bibliografie:**

1. „CURS Circuite Logice şi Comenzi Secvenţiale”

Lector: Conf. dr. ing. Florin MOLDOVEANU

1. <http://www.ti.com> – pagina oficiala a Texas Instruments, Dallas SUA;

site utilizat pentru obţinerea caracteristicilor circuitelor integrate folosite în cadrul prezentului Proiect.

1. <http://www.google.ro> – motor de căutare internaţional

4. <http://www.datasheetcatalog.com/datasheets_pdf/> -pagină de căutare a diferitelor

date de catalog ale produselor diferitelor firme producătoare de circute integrate