

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

МЕТОДИЧНІ ВКАЗІВКИ
до виконання лабораторних робіт з дисципліни
"Комп'ютерна електроніка"
для студентів всіх форм навчання
спеціальності 123 "Комп'ютерна інженерія"
Частина 2: Логічні основи комп'ютерної електроніки

Методичні вказівки до виконання лабораторних робіт з дисципліни "Комп'ютерна електроніка" для студентів всіх форм навчання спеціальності 123 "Комп'ютерна інженерія". Частина 2: Логічні основи комп'ютерної електроніки / Укл. С.С. Грушко., Т.В. Голуб, І.Я. Зеленьова - Запоріжжя: НУ«Запорізька політехніка», 2023. – 42с.

Укладачі:	к.т.н., доцент каф. КСМ	С.С. Грушко
	к.т.н., доцент каф. КСМ	Т.В. Голуб
	к.т.н., доцент каф. КСМ	І.Я. Зеленьова

Рецензент:	к.т.н., доцент каф. КСМ	С.Ю. Скрупський

Відповідальний за випуск:	к.т.н., доцент каф. КСМ	Т.В. Голуб

Затверджено
на засіданні кафедри КСМ
Протокол № 8 від 27.04.2023

Рекомендовано до видання
на засіданні НМК КНТ
Протокол № 10 від 24.05.2023

ЗМІСТ

1	ВИВЧЕННЯ ОБОЛОНКИ ELECTRONICS WORKBENCH	4
1.1	Основні теоретичні відомості	4
1.1.1	Інтерфейс програмного комплексу EWB	4
1.1.2	Компоненти	4
1.1.3	Основні компоненти	5
1.1.4	Прилади для проведення вимірювань.....	6
1.1.5	Моделювання схем	8
1.2	Лабораторне завдання.....	10
1.3	Контрольні питання	11
2	ВИВЧЕННЯ СХЕМ БУЛЕВИХ ФУНКЦІЙ ДВОХ ЗМІННИХ ...	13
2.1	Основні теоретичні відомості	13
2.1.1	Основні поняття і визначення алгебри логіки	13
2.1.2	Основні логічні операції	14
2.1.3	Закони і правила алгебри логіки	16
2.2	Аналітичний запис функцій алгебри логіки	17
2.3	Лабораторне завдання.....	19
2.4	Порядок виконання роботи	20
2.5	Приклад виконання роботи	21
2.6	Зміст звіту	27
2.7	Контрольні питання	27
3	МІНІМІЗАЦІЯ ФУНКЦІЙ АЛГЕБРИ ЛОГІКИ.....	29
3.1	Основні теоретичні відомості	29
3.1.1	Алгебраїчне спрощення логічних функцій	29
3.1.2	Метод невизначених коефіцієнтів.....	31
3.1.3	Метод мінімізації логічних функцій за допомогою карт Карно	34
3.2	Лабораторне завдання.....	40
3.3	Контрольні питання	42
	РЕКОМЕНДОВАНА ЛІТЕРАТУРА.....	42

1 ВИВЧЕННЯ ОБОЛОНКИ ELECTRONICS WORKBENCH

Мета роботи – ознайомитися з інтерфейсом застосунку Electronics Workbench, освоїти основні принципи моделювання схем.

1.1 Основні теоретичні відомості

1.1.1 Інтерфейс програмного комплексу EWB

Програма ELECTRONICS WORKBENCH (EWB) є засобом для програмної розробки, тестування та імітації електронних схем. Застосунок не вимагає ліцензії. Посилання для завантаження в [1].

Інтерфейс користувача складається з панелі меню, панелі інструментів і робочої області.

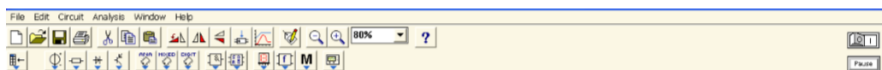


Рисунок 1.1 – Панель інструментів EWB

Панель меню складається з наступних компонентів: меню роботи з файлами (File), меню редагування (Edit), меню роботи з ланцюгами (Circuit), меню аналізу схем (Analysis), меню роботи з вікнами (Window), меню виклику довідки (Help). Панель інструментів складається з "швидких кнопок", що мають аналоги в меню, кнопок запуску і остановки схем, набору радіоелектронних аналогових і цифрових деталей, індикаторів, елементів управління і інструментів. Робочий простір (Workspace) - це найбільша центральна область - місце для безпосереднього моделювання схеми.

1.1.2 Компоненти

Інтерфейс EWB подібний реальному робочому місцю розробника електронних схем. Найбільша центральна область - робочий простір (Workspace) - місце для побудови та тестування схем. Над робочим простором знаходяться меню програми і панель компонентів, що включає в себе іконки приладів і перемикач живлення.

Отже, для операцій з компонентами на загальному полі EWB

виділені дві області: панель компонентів і поле компонентів.

Панель компонентів складається з піктограм полів компонентів, поле компонентів - з умовних зображень компонентів. У бібліотеки елементів програми EWB входять аналогові, цифрові і цифро-аналогові компоненти. Всі компоненти можна умовно розбити на наступні групи:

- базові компоненти;
- джерела;
- лінійні компоненти;
- ключі;
- нелінійні компоненти;
- індикатори;
- логічні компоненти;
- вузли комбінаційного типу;
- вузли послідовного типу;
- гібридні компоненти.

1.1.3 Основні компоненти

Кранка (Connector). Застосовується для з'єднання провідників і створення контрольних точок. До кожного вузла може приєднуватися не більше чотирьох провідників.

Заземлення (Ground). Не всі схеми потребують заземлення для моделювання, проте будь-яка схема, яка містить:

- операційний підсилювач;
- трансформатор;
- керований джерело;

- осцилограф, повинна бути обов'язково заземлена, інакше прилади не будуть проводити вимірювання або їх показання виявляться неправильними.

Джерело напруги +5 В (+ Vcc Voltage Source). Використовуючи цей джерело напруги, можна встановлювати фіксований потенціал вузла 5 В або рівень логічної одиниці.

Джерело сигналу «логічна одиниця» (Pull-Up Resistor). За допомогою цього джерела встановлюють рівень логічної одиниці в вузлі схеми.

Ключ, керований клавішею (Time-Delay Switch). Ключі можуть бути замкнуті або розімкнуті за допомогою клавіш на клавіатурі. Ім'я

керуючої клавіші можна ввести з клавіатури в діалоговому вікні, що з'являється після подвійного клацання мишею на зображенні ключа. Використовувані клавіші-ключі:

- букви від A до Z;
- цифри від 0 до 9;
- клавіша Enter на клавіатурі;
- клавіша «пробіл».

1.1.4 Прилади для проведення вимірювань

Осцилограф (Oscilloscope), імітований програмою EWB, являє собою аналог двулучевого осцилографа і має дві модифікації: просту і розширену. Розширена модифікація за своїми можливостями наближається до кращих цифрових осцилографів. Через те, що розширена модель займає багато місця на робочому полі, рекомендується починати дослідження простою моделлю, а для докладного дослідження процесів використовувати розширену модель. Ви можете підключити осцилограф до вже включеної схеми або під час роботи схеми переставити виводи до інших точок - зображення на екрані осцилографа зміниться автоматично. Подвійним клацанням миші по зображенню відкривається зображення передньої панелі простої моделі осцилографа з кнопками управління, інформаційними полями і екраном.

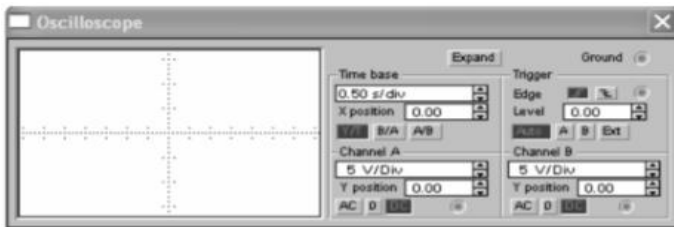


Рисунок 1.2 – Зображення передньої панелі простої моделі осцилографа

Для проведення вимірювань осцилограф потрібно налаштувати, для чого слід задати:

- розташування осей, за якими відкладається сигнал;
- потрібний масштаб розгортки по осях;
- зміщення початку координат по осях;

- режим роботи по входу: закритий чи відкритий;
- режим синхронізації: внутрішній або зовнішній.

Функціональний генератор (Function Generator) є ідеальним джерелом напруги, який виробляє сигнали синусоїдальної, прямокутної або трикутної форми. Середній вивід генератора при підключенні до схеми забезпечує загальну точку для відліку амплітуди змінної напруги. Для відліку напруги відносно нуля загальний вивід заземлюють. Крайні правий і лівий виводи служать для подачі змінної напруги на схему. Напруга на правому виводі змінюється в позитивному напрямку щодо спільного виводу, напруга на лівому виводі - в негативному. При подвійному натисканні мишею по зображенню функціонального генератора відкривається його збільшене зображення.

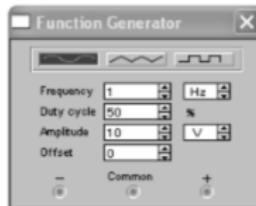


Рисунок 1.3 – Вікно регулювання функціонального генератора

Генератор слів (Word Generator) використовується для завдання цифрових послідовностей. На схему виводиться зменшене зображення генератора слів. На шістнадцять виходів в нижній частині генератора паралельно подаються біти слова, що генерується. На вихід тактового сигналу подається послідовність тактових імпульсів із заданою частотою. Вхід синхронізації використовується для подачі синхронізуючого сигналу від зовнішнього джерела.

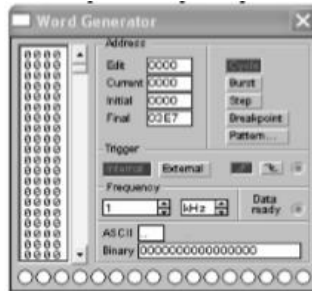


Рисунок 1.4 – Вікно регулювання генератора слів

Логічний аналізатор (Logic Analyzer) підключається до досліджуваної схеми за допомогою виводів в його лівій частині. Одночасно можуть спостерігатися сигнали в шістнадцяти точках схеми. Правий нижній затиск використовується для подачі синхронізуючих імпульсів. На схему виводиться зменшене зображення логічного аналізатора.

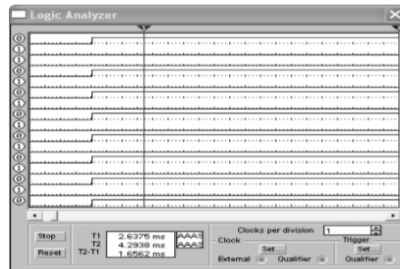


Рисунок 1.5 – Вікно логічного аналізатора

1.1.5 Моделювання схем

EWB дозволяє будувати аналогові, цифрові і цифро-аналогові схеми різного ступеня складності. Досліджувана схема збирається на робочому полі при одночасному використанні миші і клавіатури. При побудові та редагуванні схем виконуються наступні операції:

- вибір компонента з бібліотеки компонентів;
- виділення об'єкта;
- переміщення об'єкта;
- копіювання об'єктів;
- видалення об'єктів;
- поєднання компонентів схеми провідниками;

- установка значень компонентів;
- підключення приладів.

Якщо схема не поміщається на екрані монітора, будь-яку її ділянку можна переглянути за допомогою лінійок прокрутки, розташованих праворуч і під робочим полем. Після побудови схеми і підключення приладів аналіз її роботи починається після **натискання вимикача** в правому верхньому куті вікна програми.

Зробити паузу при роботі схеми можна натисканням кнопки Pause під вимикачем. Відновити процес можна повторним натисканням кнопки Pause. Повторне натискання вимикача в правому верхньому куті припиняє роботу схеми.

Вибір потрібного компонента проводиться з поля компонентів, потрібне поле компонентів вибирається натисканням лівої кнопки миші на одній з піктограм панелі компонентів. При цьому в полі компонентів з'являються зображення відповідних компонентів. Після вибору поля компонентів потрібний компонент за допомогою миші переміщається на робоче поле.

Об'єкт можна повертати на кут, кратний 90° . Для цього об'єкт потрібно попередньо виділити, а потім вибрати команду Rotate з меню Circuit або натиснути $\text{Ctrl} + R$. При цьому об'єкт повернеться на 90° за годинниковою стрілкою. При повороті групи компонентів на 90° повертається кожен компонент, а не вся група цілком.

Копіювання об'єктів здійснюється за допомогою команди Copy з меню Edit або натисканням $\text{Ctrl} + C$. Перед копіюванням об'єкт потрібно виділити. Після виконання команди виділений об'єкт копіюється в буфер пам'яті комп'ютера. Для вставки вмісту буфера на робоче поле потрібно вибрати команду Paste з меню Edit або натиснути $\text{Ctrl} + V$. Після виконання команди вміст буфера з'явиться на робочому полі і буде виділено кольором.

Видалення об'єкта здійснюється командами Cut (на робочій панелі кнопка) і Delete. Відмінність полягає в тому, що при виконанні команди Cut об'єкт видаляється в буфер і може бути потім вставлений назад на робоче поле, а при виконанні команди Delete об'єкт видаляється зовсім. Перед видаленням об'єкт також повинен бути виділений.

Для з'єднання компонентів провідниками потрібно підвести покажчик миші до виводу компонента. При цьому на виводі компонента з'явиться велика чорна точка. Натиснувши ліву кнопку

миші, перемістіть її покажчик до виводу компонента, з яким потрібно з'єднатися, і відпустіть кнопку миші. Виводи компонентів з'єднуються провідником.

Після того, як схема побудована, можна вставити в неї додаткові компоненти. Для цього потрібно мишею перемістити компонент в потрібну точку схеми і, помістивши його над провідником, відпустити кнопку миші. Компонент автоматично вставиться в ланцюг.

Більш детально оболочка EWB, її елементи, прилади та їх налаштування описані у навчальному посібнику, що знаходиться у файлі ***Modeling Electronic Circuit Tutorial.pdf***.

1.2 Лабораторне завдання

1.2.1 Зібрати схему 1 і отримати осцилограму за рис. 1.6. Для цього в схемі використати тринанізтори n-р-п типу, функціональний генератор встановити в режим генерації прямокутної форми.

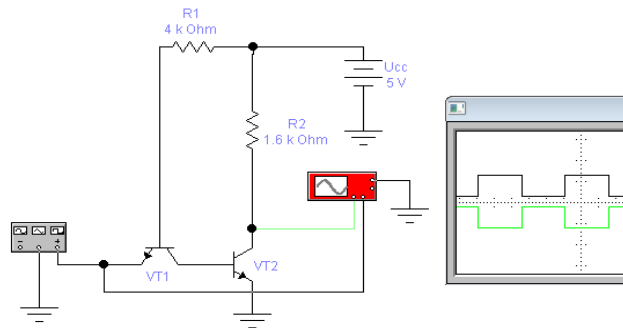


Рисунок 1.6 - Схема 1 для виконання лабораторного завдання

1.2.2 Зібрати схему 2 за рис. 1.7. Підключити входи схеми до генератора слів, а виходи – до логічного аналізатора. Налаштувати частоти генератора і аналізатора таким чином, щоб отримати часову діаграму. Задати значення для генератора слів згідно варіанту, наведеному в таблиці 1.1.

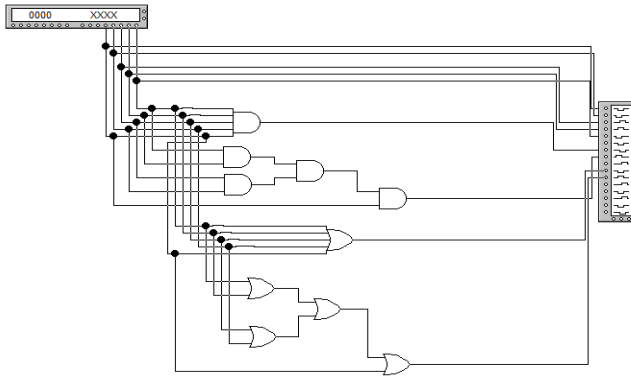


Рисунок 1.7 – Схема 2 для виконання лабораторного завдання

Таблиця 1.1 – Завдання для виконання налаштування схеми 2

Варіант №	Значення
1, 14	11111; 11011; 10110; 11; 101; 11101
2, 15	0; A; AD; 65; 15; FF
3, 16	11000; 11010; 10000; 111; 11101; 101
4, 17	B; 58; 94; DE; 83; F3
5, 18	101; 1111; 11001; 11; 10001; 11011
6, 19	65; 12; B5; 77; 58
7, 20	11110; 11; 11001; 11100; 101; 1110
8, 21	78; A5; FF; C; 82; 71
9, 22	1011; 11100; 11; 1000; 10001; 1100
10, 23	CF; 74; 11; 92; 20; 7B
11, 24	11001; 11010; 10; 1001; 11101; 1
12, 25	99; 8C; 4E; 23; 70; AC
13, 26	1101; 10; 11111; 11011; 111; 1000

1.3 Контрольні питання

1.3.1 Для чого призначена програма ELECTRONICS WORKBENCH? З яких елементів складається інтерфейс програми?

1.3.2 Як ви розумієте поняття «компонент»? В чому різниця між панеллю компонентів та полем компонентів?

1.3.3 Дайте коротку характеристику основних компонентів.

1.3.4 Дайте коротку характеристику основних приладів для проведення вимірювань.

1.3.5 Яким чином можна вставити необхідний елемент на робоче поле, виконати його поворот, з'єднати з іншим елементом?

2 ВИВЧЕННЯ СХЕМ БУЛЕВИХ ФУНКЦІЙ ДВОХ ЗМІННИХ

Мета роботи - вивчити булеві функції однієї і двох змінних, навчитися створювати **математичні моделі функцій алгебри логіки**, мінімізувати їх і перетворювати для реалізації в різних логічних базисах.

2.1 Основні теоретичні відомості

2.1.1 Основні поняття і визначення алгебри логіки

Алгебра логіки (Булева логіка, двійкова логіка, двійкова алгебра) — розділ математичної логіки, що вивчає логічні операції над висловлюваннями.

Основи алгебри логіки були сформульовані британцем Джорджем Булем в 1847 році. Пізніше її розвивали Чарлз Пірс, П. С. Порецький, Бертран Рассел, Давид Гільберт та ін.

Відтоді ця система застосовується для вирішення широкого спектру проблем математичної логіки та теорії множин, й особливо актуальна при конструюванні цифрової електроніки (початок використання алгебри логіки для синтезу перемикальних схем був покладений в 1938 році роботами відомого американського вченого Клода Шеннона).

Під *висловлюванням* розуміються ствердження, щодо змісту яких можна стверджувати, істинні вони або помилкові. Наприклад, висловлювання «кіт — тварина» - істинно, «всі кути прямі» — помилкове.

Для позначення істинності вводиться символ «I» а помилковості - символ "П". Потім замість цих символів стали вживати числа $\{1,0\}$, як двійкові змінні, тобто логічна «1» (лог.1), логічний «0» (лог.0).

Істинність висловлювання встановлюється на основі аналізу його змісту і обставин, якими керуються при його тлумаченні.

Абсолютно істинне висловлювання — це таке висловлювання, яке не може бути помилковим при будь-яких умовах. Позначається — Const 1.

Наприклад: «Земля обертається».

Абсолютно помилкове висловлювання — це таке висловлювання, яке не може бути істинним при будь-яких умовах. Позначається — Const 0.

Наприклад: «Земля — супутник Марса».

Абсолютно істинні (помилкові) висловлювання називаються **логічними константами**.

В алгебрі висловлювань розглядаються питання, пов'язані з **утворенням складних висловлювань**. Якщо в нас є кілька простих висловлювань, то за допомогою логічних зв'язувань і заперечень із них можна утворити нові складні висловлювання. Звичайно, простота і складність висловлювань відносні.

Наприклад: «На вулиці світить сонце», «В аудиторії йде заняття». Із цих висловлювань можна утворити більше десяти складних, у тому числі і безглузді висловлювання, наприклад: «Якщо в аудиторії йдуть заняття, то на вулиці світить сонце».

В алгебрі висловлювань **досліджується питання про істинність складного висловлювання** залежно від істинності вхідних у нього простих висловлювань і, з цього погляду, досліджуються різні логічні зв'язування.

Поряд із простими словесними висловлюваннями стали використовувати **літерні позначення змінних висловлювань**, тобто такі змінні $x, y, z, x_1, x_2, \dots, x_n$, значеннями яких можуть бути символи з множини $\{0,1\}$ і які позначають заздалегідь задані індивідуальні висловлювання.

Наприклад: $x_1=1$ може позначати висловлювання: "Поданий сигнал керування x_1 з рівнем напруги +4,5 В", що відповідає лог.1. $x_1=0$ може позначати висловлювання: «Поданий сигнал керування x_1 з рівнем напруги, який менше або дорівнює 0,3 В», що відповідає лог.0, і так далі: x_2 – «Поданий сигнал керування x_2 », x_3 – «Поданий сигнал керування x_3 ».

2.1.2 Основні логічні операції

При утворенні із простих висловлювань складних, велику роль відіграють сполучні зв'язування, що визначають зміст і логіку цілого ствердження. Наприклад: «Я піду в парк **і** зустріну друга», «Я піду в парк **або** зустріну друга», «**Якщо** я піду в парк, **то** зустріну друга», «Я піду в парк, **якщо і тільки якщо** зустріну друга», «Я піду в парк **тоді і тільки тоді**, коли зустріну друга».

В алгебрі логіки сполучні зв'язування, крім зв'язку змінних, визначають логічні операції.

Операція заперечення – $\neg$, в алгебрі логіки позначається рискою над змінною (формулою): $\bar{x}$.

Запереченням називається така логічна операція між вхідною логічною змінною x і вихідною логічною змінною y , при якій y істинна тільки тоді, коли x помилкова, і навпаки, y помилкова тільки тоді, коли істинна x .

За допомогою логіко-математичної символіки логічна функція НІ записується як $y = \bar{x}$.

Операція логічного множення декількох змінних – І, кон'юнкція. Синоніми: збіг, добуток, перетинання.

Логіко-математична символіка для двох змінних має кілька видів:

$$y = x_1 \cdot x_2, y = x_1 x_2, y = x_1 \wedge x_2, y = x_1 \& x_2, y = x_1 \cap x_2.$$

Операцією логічного множення декількох вхідних змінних x_i називається така вихідна функція y , яка істинна тільки тоді, коли одночасно істинні всі змінні x_i .

Операція логічного додавання декількох змінних – АБО, диз'юнкція (поділ, логічна сума, збірка).

Логіко-математична символіка для двох змінних має кілька видів:

$$y = x_1 + x_2, y = x_1 \vee x_2, y = x_1 \cup x_2.$$

Операцією логічного додавання кількох змінних x_i називається така функція y , що істинна тоді, коли істинна хоча б одна змінна x_i .

Операція еквівалентності (рівнозначність, взаємозалежність). Логіко-математична символіка для двох змінних має кілька видів зображень:

$$y = x_1 \sim x_2, y = x_1 \equiv x_2, y = x_1 \wedge x_2, y = x_1 \leftrightarrow x_2.$$

Операцією еквівалентності називається така функція декількох змінних x_i , функція y якої істинна тоді, коли всі змінні x_i або істинні або помилкові.

Операція імплікація (ділення із заборорою, проходження, селекція). Логіко-математична символіка для двох змінних має вигляд:

$$y = x_1 \rightarrow x_2, y = x_1 \supset x_2, y = x_1 > x_2.$$

Операцією імплікації називається така функція $y(x_1, x_2)$, що помилкова тоді і тільки тоді, коли x_1 істинна, а x_2 – помилкова.

Операція штрих Шеффера (заперечення кон'юнкції,

несумісність, логічне І-НЕ). Ця функція названа по імені німецького математика Шеффера, що на основі цієї функції створив свою алгебру логіки. Логіко-математична символіка:

$$y = x_1 / x_2, y = \overline{x_1 \wedge x_2}, y = \overline{x_1 \& x_2}$$

Операцією штрих Шеффера називається функція $y(x_1, x_2)$ яка помилкова тільки тоді, коли x_1 і x_2 істинні.

Операція стрілка (функція) Пірса (функція Вебба, заперечення диз'юнкції, логічне АБО-НЕ).

Логіко-математична символіка:

$$y = x_1 \downarrow x_2, y = x_1 \circ x_2, y = \overline{x_1 \vee x_2}.$$

Операцією стрілка Пірса (Вебба) називається функція $y(x_1, x_2)$, яка істинна тільки тоді, коли змінні x_1 і x_2 помилкові.

2.1.3 Закони і правила алгебри логіки

Алгебра Буля, заснована на логічних операціях кон'юнкції, диз'юнкції і заперечення, базується на наступних основних законах:

- закон комутативності;
- закон асоціативності;
- закон дистрибутивності;
- закон ідемпотентності;
- закон де Моргана.

У табл. 2.1 наведена інтерпретація цих законів для операцій кон'юнкції, диз'юнкції і заперечення щодо декількох змінних.

Таблиця 2.1- Основні закони алгебри логіки

Закон	Логічного додавання	Логічного множення
Комута- тивності	$x_1 + x_2 = x_2 + x_1$	$x_1 \cdot x_2 = x_2 \cdot x_1$
Асоціа- тивності	$(x_1 + x_2) + x_3 = x_1 + (x_2 + x_3)$	$(x_1 \cdot x_2) \cdot x_3 = x_1 \cdot (x_2 \cdot x_3)$
Дистрибу- тивності	$(x_1 + x_2) \cdot x_3 = x_1 \cdot x_3 + x_2 \cdot x_3$	$x_1 \cdot x_2 + x_3 = (x_1 + x_3) \cdot (x_2 + x_3)$
Ідемпотен- ції	$x + x = x$	$x \cdot x = x$
Закон де Моргана	$\overline{x_1 + x_2 + \dots + x_n} = \bar{x}_1 \cdot \bar{x}_2 \cdot \dots \cdot \bar{x}_n$	$\overline{x_1 \cdot x_2 \cdot \dots \cdot x_n} = \bar{x}_1 + \bar{x}_2 + \dots + \bar{x}_n$

Наслідок закону де Моргана	$x_1 + x_2 + \dots + x_n = \overline{\overline{x_1} \cdot \overline{x_2} \cdot \dots \cdot \overline{x_n}}$	$x_1 \cdot x_2 \cdot \dots \cdot x_n = \overline{\overline{x_1} + \overline{x_2} + \dots + \overline{x_n}}$
----------------------------	---	---

Наслідок закону де Моргана отримано з закону де Моргана шляхом загального заперечення лівої і правої частини рівності.

Довести ці співвідношення можна, наприклад, за допомогою складання таблиці істинності для правої і лівої частини рівнянь.

Використовуючи основні закони, одержимо ряд очевидних правил (див. табл. 2.2).

Таблиця 2.2 - Правила перетворень ЛФ

Правило	Диз'юнкція	Кон'юнкція
Інверсії	$0 = \overline{1}$	$1 = \overline{0}$
Незмінності	$x + 0 = x$	$x \cdot 1 = x$
Універсальності	$x + 1 = 1$	$x \cdot 0 = 0$
Повторення	$x + x = x$	$x \cdot x = x$
Доповнення	$x + \overline{x} = 1$	$x \cdot \overline{x} = 0$
Поглинання	$x + xy = x$	$x(x+y) = x$
Подвійного заперечення	$\overline{\overline{x}} = x$	
Склеювання	$xy + x\overline{y} = x$ $x + \overline{x}y = x + y$ $\overline{x} + x\overline{y} = \overline{x} + y$	$(x+y)(x+\overline{y}) = x$ $(x+y)(x+z) = x + yz$ $x(\overline{x}+y) = xy$ $\overline{x}(x+y) = \overline{x}y$

2.2 Аналітичний запис функцій алгебри логіки

Логічні функції мають різні форми подання: словесне; графічне; табличне; алгебраїчне; алгоритмічною мовою (наприклад, VHDL). Як приклад розглянемо функцію $y(x)$ від двох змінних x_1 і x_0 , задану словесним описом: $y(x)=1$, якщо змінні не рівні й $y(x)=0$, якщо $x_1 = x_0$. Таку ЛФ називають функцією суми за модулем два.

Після словесного опису функції $y(x)$ переходимо до її табличного подання. Табличне подання значень ЛФ для всіх наборів вхідних змінних виконується таблицею відповідності або таблицею функціонування. Головне на цьому етапі визначитись та позначити на яких наборах вхідних змінних функція приймає значення $y(x)=1$, а на яких $y(x)=0$ (табл. 2.3).

Таблиця 2.3 – Таблиця відповідності логічного елемента сума за модулем два

x_1	x_0	$y(x)$
0	0	0
0	1	1
1	0	1
1	1	0

Для подання функцій алгебри логіки дуже часто використовують аналітичний запис функцій у вигляді логічних формул, що одержують з таблиць відповідності за певними правилами їх запису.

Розглянемо деякі визначення.

Диз'юнктивний терм (макстерм) – терм, що зв'язує всі змінні, подані в прямій або інверсній формі, знаком диз'юнкції. Наприклад: $f(x) = \bar{x}_1 + x_2 + x_3 + x_4$.

Кон'юнктивний терм (мінтерм) – терм, що зв'язує всі змінні, подані в прямій або інверсній формі, знаком кон'юнкції. Наприклад: $f(x) = \bar{x}_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_4$.

Рангом терму r називається число, що визначає кількість змінних, що входять у даний терм. Наприклад, для мінтерму $\bar{x}_1 x_2 x_3 \bar{x}_4$, $r = 4$; макстерму $\bar{x}_1 + x_2 + x_3$, $r = 3$.

У загальному випадку, перехід від табличного подання функції до її логічної формули може здійснюватися через запис досконалої диз'юнктивної нормальної форми або досконалої кон'юнктивної нормальної форми ЛФ.

Диз'юнктивною нормальною формою (ДНФ) ЛФ в булевій логіці називається нормальна форма, в якій булева формула має вид диз'юнкції декількох кон'юнктив (де кон'юнктами називаються кон'юнкції декількох символів або їх заперечень).

Досконалою диз'юнктивною нормальною формою (ДДНФ) називається такий аналітичний запис функції, до якого входять:

- мінтерми у вигляді логічного добутку (кон'юнкції) усіх змінних i -го двійкового набору, які входять у прямому виді, якщо значення цієї змінної в наборі дорівнює 1, і в інверсному виді, якщо її значення дорівнює 0;

- диз'юнктивні операції, які поєднують усі мінтерми ДДНФ ЛФ;
 - значення усіх наборів, які входять до ЛФ, мають бути тільки логічною одиницею (звідси й назва «мінтерм»).

Наприклад, для операції сума за модулем два (табл. 9.2) ДДНФ буде мати вигляд: $y(x) = \bar{x}_1 \cdot x_0 + x_1 \cdot \bar{x}_0$.

Кон'юнктивною нормальною формою (КНФ) ЛФ в булевій логіці називається нормальна форма в якій булева формула має вид кон'юнкції декількох диз'юнктив.

Досконалою кон'юнктивною нормальною формою (ДКНФ) називається такий аналітичний запис функції, до якого входять:

- макстерми у вигляді логічної суми (диз'юнкції) усіх змінних i -го двійкового набору, які входять у прямому виді, якщо значення цієї змінної в наборі дорівнює 0, і в інверсному виді, якщо її значення дорівнює 1;
- кон'юнктивні операції, які поєднують усі макстерми ДКНФ ЛФ;
- диз'юнктивні операції, для запису макстерму.

Для операції сума за модулем два ДКНФ буде мати вигляд:

$$y(x) = (\bar{x}_1 + \bar{x}_0) \cdot (x_1 + x_0).$$

2.3 Лабораторне завдання

Індивідуальне завдання студента включає синтез функцій алгебри логіки (ФАЛ) і моделювання роботи схеми двох логічних функцій з табл. 2.3, що задані за варіантом (табл. 2.4).

Якщо не отримано інших вказівок, то одна функція після спрощення подається у базисі Пірса, інша – у базисі Шеффера.

До кожної ФАЛ будуються 2 схеми. Одна з використанням віртуальних елементів, що мають спрощені умовні графічні позначення, наступна – з використанням аналогів реальних мікросхем.

При правильно виконаному завданні обидві схеми повинні працювати ідентично.

Таблиця 2.3 – Назви логічних функцій (Y_1 - Y_{16})

X_1	0 0 1 1	Позначення	Назва
X_2	0 1 0 1		
Y_1	0 0 0 0	0	Константа 0
Y_2	0 0 0 1	$x_1 x_2$	Кон'юнкція, логічне множення, І
Y_3	0 0 1 0	$x_1 \leftarrow x_2$	Заперечення імплікації (заборона x_2)

Y_4	0 0 1 1	x_1	Повторення першого аргументу
Y_5	0 1 0 0	$x_2 \leftarrow x_1$	Заперечення оберненої імплікації
Y_6	0 1 0 1	x_2	Повторення другого аргументу
Y_7	0 1 1 0	$x_1 \oplus x_2$	Сума за модулем 2, виключне АБО
Y_8	0 1 1 1	$x_1 + x_2$	Диз'юнкція, логічне додавання, АБО
Y_9	1 0 0 0	$x_1 \downarrow x_2$	Стрілка Пірса
Y_{10}	1 0 0 1	$x_1 \sim x_2$	Еквіваленція
Y_{11}	1 0 1 0	$\bar{x}_2$	Заперечення другого аргументу
Y_{12}	1 0 1 1	$x_2 \rightarrow x_1$	Обернена імплікація
Y_{13}	1 1 0 0	$\bar{x}_1$	Заперечення першого аргументу
Y_{14}	1 1 0 1	$x_1 \rightarrow x_2$	Імплікація
Y_{15}	1 1 1 0	x_2 / x_1	Штрих Шеффера
Y_{16}	1 1 1 1	1	Константа 1

Таблиця 2.4 – Завдання для виконання лабораторної роботи №2

Варіант №	Значення	Варіант №	Значення
1	Y_1, Y_{11}	14	Y_{14}, Y_8
2	Y_2, Y_{12}	15	Y_{15}, Y_9
3	Y_3, Y_{13}	16	Y_{16}, Y_{10}
4	Y_4, Y_{14}	17	Y_1, Y_{15}
5	Y_5, Y_{15}	18	Y_2, Y_{16}
6	Y_6, Y_{16}	19	Y_3, Y_{14}
7	Y_7, Y_1	20	Y_4, Y_{13}
8	Y_8, Y_2	21	Y_5, Y_{12}
9	Y_9, Y_3	22	Y_6, Y_{11}
10	Y_{10}, Y_4	23	Y_7, Y_3
11	Y_{11}, Y_5	24	Y_8, Y_1
12	Y_{12}, Y_6	25	Y_9, Y_4
13	Y_{13}, Y_7	26	Y_{10}, Y_8

2.4 Порядок виконання роботи

2.4.1 Записати таблицю істинності (ТІ) заданої функції і згідно неї записати ДДНФ і ДКНФ функції.

2.4.2 Провести спрощення (мінімізацію) обох рівнянь, користуючись правилами, що наведені в табл. 2.1 і 2.2. Кінцевий вираз обох рівнянь повинен співпадати, це свідчить про правильність спрощення.

2.4.3 Подати спрощенні ФАЛ у заданому базисі (Пірса, Шеффера або ін.), для чого за необхідністю можна скористатись законом де Моргана.

2.4.4 За отриманим рівнянням, що є математичною моделлю (виразом) ФАЛ побудувати у середовищі EWB логічну схему функції з віртуальних логічних елементів. Перевірити правильність роботи схеми підключивши до неї інструменти формування та візуалізації сигналів.

2.4.5 Побудувати аналогічну схему, використовуючи аналоги реальних схем з бібліотеки моделей EWB. Впевнитися в тому, що обидві схеми працюють ідентично.

2.5 Приклад виконання роботи

Запишемо ТІ, задану викладачем у такому вигляді:

Таблиця 2.5 – Таблиця істинності схеми

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

За ТІ запишемо ДДНФ функції (табл. 2.6).

Таблиця 2.6 – Визначення аргументів ДДНФ функції

x_1	x_2	y	ДДНФ
0	0	1	$\overline{x_1} \cdot \overline{x_2}$
0	1	0	
1	0	0	
1	1	1	$x_1 x_2$

Та проведемо її мінімізацію:

$$y = \overline{x_1} \cdot \overline{x_2} + x_1 x_2$$

За ТІ запишемо ДКНФ функції (таб. 2.7)

Таблиця 2.7 – Визначення аргументів ДКНФ функції

x_1	x_2	y	ДКНФ
-------	-------	-----	------

0	0	1	
0	1	0	$(x_1 + x_2)$
1	0	0	$(x_1 + x_2)$
1	1	1	

Та проведемо її мінімізацію:

$$y = (x_1 + x_2)(\overline{x_1} + \overline{x_2}) = x_1 \overline{x_1} + x_1 \overline{x_2} + \overline{x_1} \cdot \overline{x_2} + x_2 \overline{x_2} = x_1 \overline{x_2} + \overline{x_1} \cdot \overline{x_2}$$

Мінімізовані функції ДДНФ та ДКНФ співпадають, що означає коректність їх побудови.

Перетворимо мінімізоване рівняння у вигляд, що відповідає базису Пірса або базису Шеффера.

Базис Пірса ($y = \overline{x_1 + x_2}$)

Перетворення виконуються за використання наслідку закону де Моргана для логічного множення.

$$y = \overline{x_1 \cdot x_2 + x_1 \cdot x_2} = \overline{x_1 + x_2 + x_1 + x_2} = \overline{x_1 + x_2 + x_1 + x_2} = \overline{x_1 + x_2 + x_1 + x_2}$$

Примітка. Приклад заміни елементів виразу для наведеного вище перетворення:

Дано: $f = \overline{x_1 \cdot x_2}$

$$z_1 = \overline{x_1},$$

Нехай виконаємо заміну:

$$z_2 = \overline{x_2}$$

Тоді:

$$f = z_1 \cdot z_2 = \overline{z_1 + z_2}$$

Відновлення:

$$f = \overline{x_1 + x_2} = \overline{x_1 + x_2}$$

Таким чином, в базисі Пірса рівняння має вигляд:

$$y = \overline{x_1 + x_2 + x_1 + x_2}$$

Базис Шеффера ($y = \overline{x_1 \cdot x_2}$)

Перетворення виконуються за використання наслідку закону де Моргана для логічного додавання.

$$y = \overline{x_1 \cdot x_2} + x_1 \cdot x_2 = \overline{\overline{x_1 \cdot x_2} \cdot x_1 \cdot x_2}$$

Примітка. Приклад заміни елементів виразу для наведеного вище перетворення:

Дано:

$$f = \overline{x_1 \cdot x_2} + x_1 \cdot x_2$$

Нехай виконаємо заміну:

$$z_1 = \overline{x_1 \cdot x_2};$$

$$z_2 = x_1 \cdot x_2$$

Тоді:

$$f = z_1 + z_2 = \overline{\overline{z_1 \cdot z_2}}$$

Відновлення:

$$f = \overline{\overline{x_1 \cdot x_2} \cdot x_1 \cdot x_2}$$

Таким чином, в базисі Шеффера рівняння має вигляд:

$$y = x_1 x_2 \cdot \overline{x_1 \cdot x_2}.$$

За отриманим рівнянням побудуємо логічну схему.

Далі наведено покроковий розподіл етапів виконання логічних операцій у рівнянні, які в подальшому реалізуються за допомогою логічних елементів.

Базис Пірса

На рисунку 2.1 наведено етапи виконання логічних операцій у рівнянні, які в подальшому реалізуються за допомогою логічних елементів. Однаковим кольором помічено оперції, які можна виконувати паралельно (одночасно).

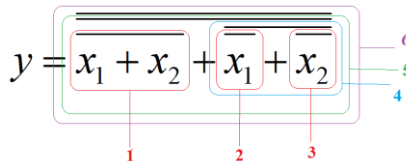


Рисунок 2.1 – Етапи виконання логічних операцій в рівнянні за базисом Пірса

Побудова схеми на основі логічних операцій, виділених на попередньому кроці, наведені на рисунку 2.2.

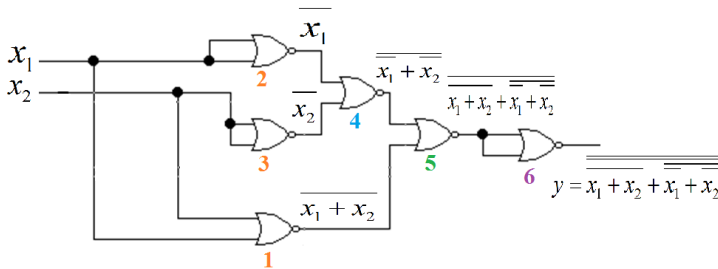


Рисунок 2.2 – Побудова логічної схеми за рівнянням у базисі Пірса

Базис Шеффера

На рисунку 2.3 наведено етапи виконання логічних операцій у рівнянні, які в подальшому реалізуються за допомогою логічних елементів. Однаковим кольором помічено оперції, які можна виконувати паралельно (одночасно).

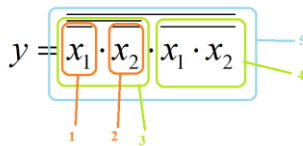


Рисунок 2.3 – Етапи виконання логічних операцій в рівнянні за базисом Шеффера

Побудова схеми на основі логічних операцій, виділених на попередньому кроці, наведені на рисунку 2.4.

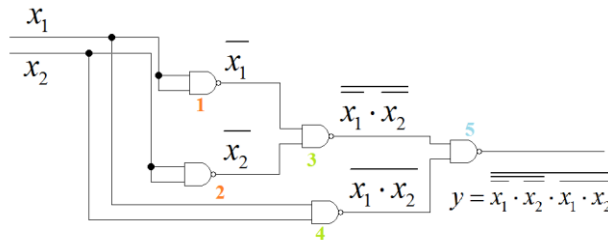


Рисунок 2.4 – Побудова логічної схеми за рівнянням у базисі Шеффера

Тестування розроблених за отриманим рівнянням логічних схем та схему із використанням аналогів реальних мікросхем наведені на рисунках 2.5 – 2.7.

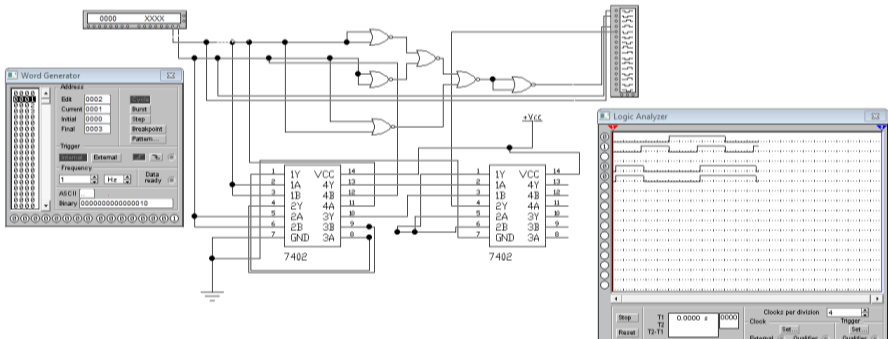


Рисунок 2.5 – Схема ФАЛ у базисі Пірса

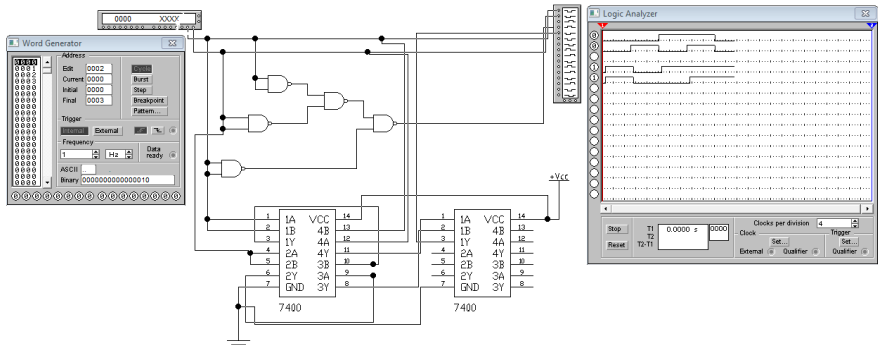


Рисунок 2.6 – Схема ФАЛ у базисі Шеффера

Зверніть увагу, що схеми можна будувати, використовуючи один прилад формування сигналу, та один прилад візуалізації сигналу.

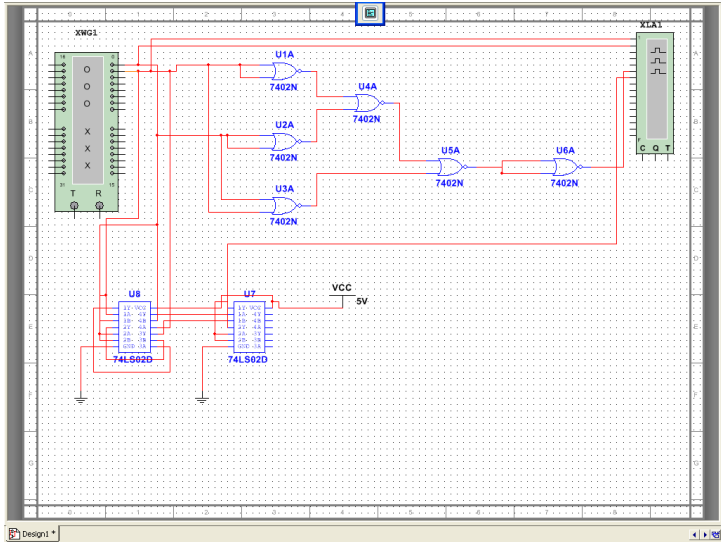


Рисунок 2.7 – Приклад побудови схеми ФАЛ у базисі Пірса в програмному середовищі Multisim.

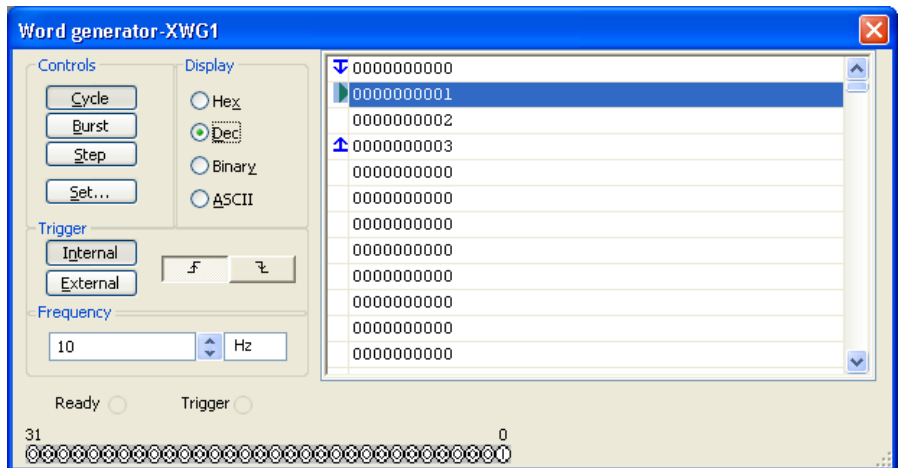


Рисунок 2.8 – Налаштування генератора слів

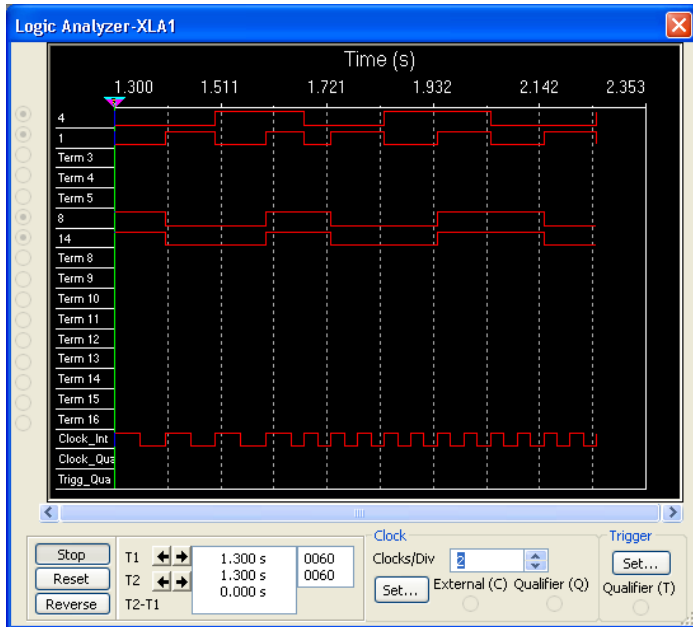


Рисунок 2.9 – Графік, отриманий в логічному аналізаторі

2.6 Зміст звіту

2.6.1 Короткий теоретичний огляд по елементарним двовходовим логічним схемам, їх ТІ.

2.6.2 Етапи аналізу, синтезу і мінімізації отриманих ДНФ (КНФ) до мінімізованих ДНФ і КНФ з поясненнями.

2.6.3 Роздруківки графічних матеріалів, що переконливо і повно ілюструють функціонування схеми відповідно до ТІ.

2.6.4 Визначення типу використаних мікросхем, їх призначення, особливості з погляду виконуваних функцій, назви серій цифрових інтегральних схем.

2.6.5 Висновки за отриманими даними з аналізом функціональних та часових характеристик.

2.7 Контрольні питання

2.7.1 Дайте визначення логічних функцій: інверсія, диз'юнкція, кон'юнкція, штрих Шеффера, стрілка Пірса, еквівалентність, константа «1» («0»), імплікація, сума за модулем 2, ін.

2.7.2 Зобразіть умовні графічні позначення логічних елементів відомих типів.

2.7.3 Запишіть ТІ заданих логічних елементів.

2.7.4 Визначте вихідний стан логічних елементів при заданих вхідних сигналах та навести ТІ.

2.7.5 Визначте якому логічному елементу належить ТІ.

2.7.6 Які логічні елементи можна використати для отримання функції інверсії?

2.7.7 Що таке ДДНФ і ДКНФ, для чого вони використовуються та чим відрізняються від ДНФ і КНФ?

3 МІНІМІЗАЦІЯ ФУНКЦІЙ АЛГЕБРИ ЛОГІКИ

Мета роботи - вивчити основні методи спрощення алгебраїчних функцій, навчитися застосовувати їх для мінімізації виразів.

3.1 Основні теоретичні відомості

3.1.1 Алгебраїчне спрощення логічних функцій

Алгебраїчне спрощення логічних виразів в першу чергу полягає у використанні рівностей і законів булевої алгебри. Розглянемо основні прийоми спрощення логічних функцій (ЛФ).

Групування членів з подальшим використанням тотожностей полягає у використанні закону асоціативності для правильного групування членів, після чого до згрупованих членів застосовуються відомі тотожності типу $x_1 + 1 = 1$, які дозволяють скоротити вираз. Наприклад:

$$f(x_1, x_2, x_3) = x_1 + \overline{x_1}x_3 + x_1x_2x_3 = x_1(1 + \overline{x_3} + x_2x_3) = x_1.$$

Для створення додаткових груп та отримання потрібних тотожностей можна вводити повторювані члени (тому що $x_1 + x_1 = x_1$).

Існує ще одна тотожність, яку зазвичай не відносять до базових, але вона часто дуже корисна при спрощенні рівнянь:

$$\overline{x_1} + x_1x_2 = \overline{x_1} + x_2.$$

Приведення виразу в канонічну форму з подальшим спрощенням - вираз, записаний у диз'юнктивній формі, можна привести до канонічного виду за допомогою множення на множники типу $\overline{x_1} + x_1$. Члени канонічного виразу можуть бути перегруповані так, що в результаті виходить спрощений вираз. Таким чином, спочатку вирази перетворюються в канонічний вигляд. Потім вони групуються з метою спрощення. При необхідності для формування груп можна ввести повторювані члени.

Наприклад:

$$\begin{aligned} f(x_1, x_2, x_3) &= \overline{x_1}x_2 + \overline{x_1}x_2x_3 + \overline{x_1}x_2x_3 + \overline{x_1}x_2x_3 = \\ &= \overline{x_1}x_2(\overline{x_3} + x_3) + \overline{x_1}x_2x_3 + \overline{x_1}x_2x_3 + \overline{x_1}x_2x_3 = \\ &= \overline{x_1}x_2x_3 + \overline{x_1}x_2x_3 + \overline{x_1}x_2x_3 + \overline{x_1}x_2x_3 + \overline{x_1}x_2x_3 = \end{aligned}$$

$$\begin{aligned}
 &= \overline{x_1 x_2 (x_3 + x_3)} + \overline{x_1 x_3 (x_2 + x_2)} + \overline{x_1 x_2 (x_3 + x_3)} = \\
 &= \overline{x_1 x_2} + \overline{x_1 x_3} + \overline{x_1 x_2}.
 \end{aligned}$$

Цей метод дозволяє мінімізувати вирази у диз'юнктивній формі. Для спрощення виразів в кон'юнктивній формі необхідно спочатку отримати інверсну функцію (вона, звісно, буде мати диз'юнктивну форму). Після цього функція мінімізується та з використанням теореми де Моргана переводиться в кон'юнктивну форму.

Використання закону де Моргана. Закон де Моргана є чи не основним способом зменшення логічних виразів. Зазвичай, він більш ефективний у тих випадках, коли доводиться мати справу зі складними функціями, що включають багаторівневі інверсії. Багатократне використання теореми де Моргана дозволяє значно зменшити кількість рівнів.

Щоб запам'ятати порядок дій при використанні теореми де Моргана, потрібно засвоїти просте **правило**: якщо над виразом, що включає члени, зв'язані оператором (І або АБО), є загальна довга риска інверсії, то можна розбити її на дві частини, одночасно змінивши оператор між членами (оператор І змінюється на АБО, а оператор АБО – на І). Аналогічно, якщо над різними членами виразу, зв'язаними оператором, знаходяться декілька окремих символів інверсії, то можна об'єднати їх та змінити оператор.

Спрощення функції

$$f(x_1, x_2, x_3) = \overline{(x_1 x_2 x_4 + x_2 x_3 x_4) x_2 x_3 + x_1 x_3}$$

можна почати з «розбиття» довгого знаку інверсії на два:

$$= \left(\overline{(x_1 x_2 x_4 + x_2 x_3 x_4) x_2 x_3} \right) \left(\overline{x_1 x_3} \right).$$

Продовжуючи так само, отримаємо

$$\begin{aligned}
 &= \left(\overline{x_1 x_2 x_4 + x_2 x_3 x_4} + \overline{x_2 x_3} \right) \left(\overline{x_1 + x_3} \right) = \\
 &= \left(\left(\overline{x_1 x_2 x_4} \right) \left(\overline{x_2 x_3 x_4} \right) + \left(\overline{x_2 + x_3} \right) \right) \left(\overline{x_1 + x_3} \right) = \\
 &= \left(\left(\overline{x_1 + x_2 + x_4} \right) \left(\overline{x_2 x_3 x_4} \right) + \overline{x_2 + x_3} \right) \left(\overline{x_1 + x_3} \right).
 \end{aligned}$$

Перемножуючи члени, які записані у дужках (з врахуванням того, що $\overline{x_2 x_2} = 0$ та $\overline{x_4 x_4} = 0$), легко отримати

$$= \left(\overline{x_1 x_2 x_3 x_4} + \overline{x_2 + x_3} \right) \left(\overline{x_1 + x_3} \right).$$

Продовжуючи перемноження (пригадаємо, що $\overline{\overline{x_3 x_3}} = \overline{x_3}$), отримаємо:

$$= x_1 x_2 + x_1 \overline{x_3} + \overline{x_1} x_2 \overline{x_3} x_4 + x_2 \overline{x_3} + \overline{x_3}.$$

Нарешті, після завершального перегрупування:

$$= x_1 x_2 + \overline{x_3} (x_1 + x_1 x_2 x_3 + x_2 + 1) = x_1 x_2 + \overline{x_3}.$$

Закон де Моргана може застосовуватися до різних членів виразів, та способи його використання змінюються в залежності від виду виразу. Наприклад, в одному виразі можна використати як об'єднання символів інверсії, так і їх розділення, але вирішити, що краще в кожному конкретному випадку, досить непросте завдання. Також слід пам'ятати, що сам закон де Моргана не є самодостатнім – він завжди працює разом з базовими тотожностями булевої алгебри.

3.1.2 Метод невизначених коефіцієнтів

Метод невизначених коефіцієнтів полягає в знаходженні ненульових коефіцієнтів для логічних функцій, поданих в диз'юнктивній нормальній формі (ДНФ).

Визначення. Будь-яку ДНФ можливо подати у вигляді диз'юнкції кон'юнкцій з коефіцієнтом k , який приймає значення 0 або 1. Так, для функції трьох змінних загальна форма ДНФ з коефіцієнтами має вигляд:

$$\begin{aligned} f(x_1, x_2, x_3) = & k_1^1 x_1 + k_1^0 \overline{x_1} + k_2^1 x_2 + k_2^0 \overline{x_2} + k_3^1 x_3 + k_3^0 \overline{x_3} + \\ & + k_{12}^{11} x_1 x_2 + k_{12}^{10} x_1 \overline{x_2} + k_{12}^{01} \overline{x_1} x_2 + k_{12}^{00} \overline{x_1} \overline{x_2} + k_{13}^{11} x_1 x_3 + k_{13}^{10} x_1 \overline{x_3} + \\ & + k_{13}^{01} \overline{x_1} x_3 + k_{13}^{00} \overline{x_1} \overline{x_3} + k_{23}^{11} x_2 x_3 + k_{23}^{10} x_2 \overline{x_3} + k_{23}^{01} \overline{x_2} x_3 + k_{23}^{00} \overline{x_2} \overline{x_3} + \\ & + k_{123}^{111} x_1 x_2 x_3 + k_{123}^{110} x_1 x_2 \overline{x_3} + k_{123}^{101} x_1 \overline{x_2} x_3 + k_{123}^{100} x_1 \overline{x_2} \overline{x_3} + k_{123}^{011} \overline{x_1} x_2 x_3 + \\ & + k_{123}^{010} \overline{x_1} x_2 \overline{x_3} + k_{123}^{001} \overline{x_1} \overline{x_2} x_3 + k_{123}^{000} \overline{x_1} \overline{x_2} \overline{x_3}. \end{aligned}$$

Коефіцієнти k з різними індексами є невизначеними. Якщо k приймає значення 0, то в нуль перетворюється і мінтерм, якщо $k=1$, то цей мінтерм може увійти у МДНФ.

Приклад:

Для прикладу візьмемо функцію:

$$f(x_1, x_2, x_3) = \bar{x}_1 x_2 x_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 x_2 x_3.$$

Її значення на наборах змінних приведені в табл. 3.1.

Таблиця 3.1 – Пошук ненульових коефіцієнтів. Ті функції

№ функції	Набори $x_1 x_2 x_3$	Значення	Невизначені коефіцієнти мінтерми ДНФ							
			k_1	k_2	k_3	k_{12}	k_{13}	k_{23}	k_{123}	
f_0	0 0 0	0	0	0	0	00	00	00	000	
f_1	0 0 1	0	0	0	1	00	01	01	001	
f_2	0 1 0	0	0	1	0	01	00	10	010	
f_3	0 1 1	1	0	1	1	01	01	11	011	
f_4	1 0 0	0	1	0	0	10	10	00	100	
f_5	1 0 1	1	1	0	1	10	11	01	101	
f_6	1 1 0	1	1	1	0	11	10	10	110	
f_7	1 1 1	1	1	1	1	11	11	11	111	

На етапі 1 викреслюємо всі мінтерми, для яких функція має значення нуль (табл. 3.2). Це означає, що невизначені коефіцієнти при них теж повинні бути рівними нулю.

Таблиця 3.2 – Пошук ненульових коефіцієнтів. Етап 1

№ функції	Набори	Значення	Невизначені коефіцієнти мінтерми ДНФ							
	$x_1 \ x_2 \ x_3$		k_1	k_2	k_3	k_{12}	k_{13}	k_{23}	k_{123}	
f_0	0 0 0	0	0	0	0	00	00	00	000	
f_1	0 0 1	0	0	0	1	00	01	01	001	
f_2	0 1 0	0	0	1	0	01	00	10	010	
f_3	0 1 1	1	0	1	1	01	01	11	011	
f_4	1 0 0	0	1	0	0	10	10	00	100	
f_5	1 0 1	1	1	0	1	10	11	01	101	
f_6	1 1 0	1	1	1	0	11	10	10	110	
f_7	1 1 1	1	1	1	1	11	11	11	111	

На етапі 2 викреслюємо мінтерми, для яких функція дорівнює одиниці, але коефіцієнти раніше були визначені як нульові (табл. 3.3).

Після окреслення нульових коефіцієнтів залишиться система мінтермів з ненульовими коефіцієнтами (табл.3.4).

Таблиця 3.3 – Пошук ненульових коефіцієнтів. Етап 2

№ функції	Набори $x_1 x_2 x_3$	Значення	Невизначені коефіцієнти мінтерми ДНФ						
			k_1	k_2	k_3	k_{12}	k_{13}	k_{23}	k_{123}
f_0	0 0 0	0	0	0	0	00	00	00	000
f_1	0 0 1	0	0	0	1	00	01	01	001
f_2	0 1 0	0	0	1	0	01	00	10	010
f_3	0 1 1	1	0	1	1	01	01	11	011
f_4	1 0 0	0	1	0	0	10	10	00	100
f_5	1 0 1	1	1	0	1	10	11	01	101
f_6	1 1 0	1	1	1	0	11	10	10	110
f_7	1 1 1	1	1	1	1	11	11	11	111

Таблиця 3.4 – Пошук ненульових коефіцієнтів. Результат

№ функції	Набори $x_1 x_2 x_3$	Значення	Невизначені коефіцієнти мінтерми ДНФ						
			k_1	k_2	k_3	k_{12}	k_{13}	k_{23}	k_{123}
f_0	0 0 0	0	0	0	0	00	00	00	000
f_1	0 0 1	0	0	0	1	00	01	01	001
f_2	0 1 0	0	0	1	0	01	00	10	010
f_3	0 1 1	1	0	1	1	01	01	11	011
f_4	1 0 0	0	1	0	0	10	10	00	100
f_5	1 0 1	1	1	0	1	10	11	01	101
f_6	1 1 0	1	1	1	0	11	10	10	110
f_7	1 1 1	1	1	1	1	11	11	11	111

На основі таблиці 3.4 будеться система рівностей:

- 1) $k_{23}^{11}x_2x_3 + k_{123}^{011}x_1x_2x_3 = 1$,
- 2) $k_{13}^{11}x_1x_3 + k_{123}^{101}x_1x_2x_3 = 1$,
- 3) $k_{12}^{11}x_1x_2 + k_{123}^{110}x_1x_2x_3 = 1$,
- 4) $k_{12}^{11}x_1x_2 + k_{13}^{11}x_1x_3 + k_{23}^{11}x_2x_3 + k_{123}^{111}x_1x_2x_3 = 1$.

Виберемо із коефіцієнтів кожного рівняння ті, які мають найменший ранг (меншу кількість змінних). Інші викреслимо. Це робиться на підставі того, що диз'юнкція буде дорівнювати одиниці, якщо хоча б один мінтерм буде дорівнювати одиниці.

Тоді одержимо систему:

$$1) k_{23}^{11}x_2x_3=1,$$

$$2) k_{13}^{11}x_1x_3=1,$$

$$3) k_{12}^{11}x_1x_2=1,$$

$$4) k_{12}^{11}x_1x_2=1.$$

Після склеювання однакових коефіцієнтів (п.3 та 4 системи)

$$k_{12}^{11}x_1x_2 + k_{12}^{11}x_1x_2 = k_{12}^{11}x_1x_2$$

одержимо МДНФ функції $f(x_1, x_2, x_3)$:

$$f(x_1, x_2, x_3) = x_1x_2 + x_1x_3 + x_2x_3.$$

3.1.3 Метод мінімізації логічних функцій за допомогою карт Карно

Одним із способів графічного подання булевих функцій є карти Карно. Їх різновид – карти Вейча. Карти будують як розгорнення кубів на площині. При цьому, вершини куба подаються клітинами карти, координати яких збігаються з координатами мінтермів відповідних вершин куба. Карта заповнюється так само, як таблиця істинності: значення 1 вказується в клітині, що відповідає набору (мінтерму), на якому функція має значення $f(x)=1$. Значення $f(x)=0$ звичайно на картах не відображається. Умовне розміщення координат змінних і приклади мінімізації функцій логіки на картах для двох, трьох і чотирьох змінних показані на рис. 9.1.

Правила мінімізації за картами Карно. Спосіб мінімізації логічних функцій (ЛФ) за допомогою карт Карно набув широкого застосування серед фахівців. Цьому сприяли досить висока формалізація дій, наочність графіки й відносно прості правила.

Як показано на рис. 3.1, карти Карно представляють прямокутником, на якому розміщують $2n$ клітин, де n – число змінних ЛФ. При заповненні клітин значеннями функцій використовують умовні координати розміщення змінних. Координати клітин звичайно позначають починаючи з лівого верхнього кута карти, використовуючи код Грея. Якщо порушити місця позначення координат, то мінімальна функція не буде відповідати істині. Кожній клітині відповідає свій мінтерм, позначений перетинанням змінних рядків і стовпців. Після заповнення клітин карт одиницями

визначають сусідні клітини.

Сусідні клітини карт Карно. Сусідніми клітинами вважають ті заповнені одиницями клітини, які мають спільні сторони між собою. Сусідніми вважають і клітини розташовані в протилежних крайніх рядках і (або) стовпцях. Справді, якщо уявно згорнути карту, то крайні протилежні рядки (стовпці) стануть сусідніми. Тому, наприклад, всі протилежні та кутові клітини карт для двох, трьох і чотирьох змінних вважаються сусідніми. Вони об'єднуються як сусідні карти.

При числі змінних більше чотирьох карти Карно складаються з декількох карт для чотирьох змінних, тому сусідніми клітками вважають ще й ті, які перебувають на однакових координатах у сусідніх картах.

Для мінімізації функцій із числом змінних більше 7 карти Карно практично не застосовуються.

Після визначення сусідніх клітин їх поєднують контурами об'єднання.

Правило об'єднання сусідніх клітин. Контур повинний містити тільки сусідні клітини, причому, кількість одиниць, включених до контуру, обмежується, і має відповідати одному з чисел: 2, 4, 8, 16, 32, 64, ..., $2n$. Ця умова обов'язкова.

Будь-який контур об'єднання може перетинатися з іншими. Тому, ті самі одиниці можуть бути включені одночасно в різні контури.

Загальне число контурів повинне бути як можна меншим, тому що кожний контур об'єднання – це новий мінтерм функції, чим менше їх буде, тим краще.

З іншого боку, загальне число об'єднаних одиниць повинне бути якомога більшим, тому що це зменшує число змінних у мінтермі, знижуючи його ранг. Завжди об'єднання в $2n$ клітин (сусідніх одиниць) виключає з мінтермів n змінних, де $n=1, 2, 3, 4, \dots$

Необхідно пам'ятати, що об'єднання сусідніх клітин – це процес, що потребує наявності досвіду з вибору найкращого варіанту. Тому, кожного разу при мінімізації необхідно переглядати кілька варіантів і вибирати кращий, який приводить до мінімальної функції. Після об'єднання сусідніх клітин переходимо до запису мінтермів для них.

Визначення простих імплікант. Кожний окремий контур

об'єднання відповідає мінімальному мінтерму, що інакше називають простою імплікантою. Для визначення й запису простої імпліканти прямо за картами Карно проводиться аналіз змінних по тим рядкам і стовпцям карти, які перетинають контур об'єднаних одиниць (вершин куба). Починають аналіз із першої змінної x_1 .

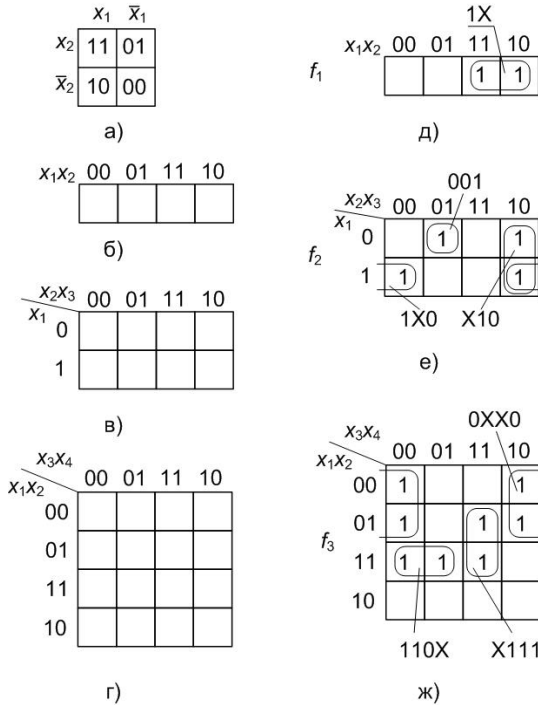
Якщо змінна x_1 у будь-якому рядку (стовпці), що проходить через контур об'єднання, не змінює свого значення на всіх координатах об'єднання, то у мінтерм об'єднання записується її значення. Якщо змінна x_1 у рядках (стовпцях), що проходять через контур, змінює своє значення на координатах об'єднання, то у мінтерм об'єднання записується так звана невизначена змінна x на тому місці, де повинне записуватися значення змінної x_1 (рис. 3.1 д, е, ж). Невизначена змінна x просто позначає місце склеєної змінної, яка у просту імпліканту не включається. Аналогічний аналіз проводиться послідовно для всіх змінних об'єднання, у результаті чого одержуємо просту імпліканту (мінтерм) об'єднання. На рис. 3.1 д, е, ж вони позначенні на виносках. Отримавши прості імпліканти для всіх об'єднань, записуємо мінімальну функцію, в яку входять усі прості імпліканти.

Це перший етап запису мінімальної логічної функції прямо з карти Карно. На другому етапі міняємо значення змінних на відповідні їм змінні за правилом: якщо значення змінної 0 – то пишемо $\overline{x_i}$, якщо 1 – пишемо x_i . На місці невизначеної змінної x не пишемо нічого, тобто це місце опускаємо. Тепер, з'єднавши всі мінімальні мінтерми знаком диз'юнкції, одержимо мінімальну диз'юнктивну нормальну форму логічної функції (МДНФ).

При побудові мінімальної форми для кон'юнкції карти Карно заповнюються 0, тобто мінтермами тих наборів, на яких функція дорівнює нулю. Об'єднання нулів і запис МДНФ для **оберненої функції**, тобто $\overline{y}$, виконують по аналогічним правилам. Потім, застосувавши до лівої й правої частин рівняння правило де Моргана й перетворивши диз'юнкцію правої частини в кон'юнкцію, одержимо мінімальну кон'юнктивну нормальну форму (МКНФ) логічної функції.

Таким чином, мінімальне покриття функції вибирається

інтуїтивно, шляхом аналізу різних варіантів об'єднань сусідніх клітин карти Карно для функції, що мінімізується.



$$f_1 = x_1 \bar{x}_2 + x_1 x_2 = x_1$$

$$f_2 = \bar{x}_1 \bar{x}_2 x_3 + x_1 \bar{x}_2 \bar{x}_3 + \bar{x}_1 x_2 x_3 + x_1 x_2 \bar{x}_3 = \bar{x}_1 \bar{x}_2 x_3 + x_1 \bar{x}_3 + x_2 \bar{x}_3$$

$$f_3 = \bar{x}_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 + x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 + \bar{x}_1 x_2 \bar{x}_3 \bar{x}_4 + x_1 x_2 \bar{x}_3 \bar{x}_4 + \bar{x}_1 x_2 x_3 \bar{x}_4 + x_1 x_2 x_3 \bar{x}_4 + x_1 x_2 \bar{x}_3 x_4 + x_1 x_2 x_3 x_4 = \bar{x}_1 \bar{x}_4 + x_1 \bar{x}_2 \bar{x}_3 + x_2 x_3 \bar{x}_4$$

Рисунок 3.1 – Карти Карно для двох (а, б), трьох (в) та чотирьох (г) змінних; приклади мінімізації ЛФ (д, е, ж)

Найкращим вважається таке покриття, що має мінімум контурів об'єднання. Якщо варіантів багато, то вибирається той, який дає мінімальну сумарну площу контурів (прямокутників).

Приклад.

Як приклад використання карт Карно для визначення

мінімальних ДНФ і КНФ розглянемо функцію, представлену на рис.3.2.

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

а)

		x_3x_4			
x_1x_2		00	01	11	10
00				0	0
01		0	0		
11		0			
10		0		0	0

б)

Рисунок 3.2 – Значення функції, задані картою Карно

Даний ДНФ (рис. 3.2 а) відповідає мінімальна форма виду:

$$f(x) = \overline{x_1} \overline{x_2} \overline{x_3} + x_1 x_3 x_4 + x_2 x_3.$$

Нульові значення цієї функції подані на рис. 3.2 б.

Позначені на карті клітини визначають функцію $\overline{f}(\delta)$, обернену до $f(\delta)$.

При мінімізації функції, отримаємо:

$$\overline{f}(x) = \overline{x_1} \overline{x_2} \overline{x_3} + x_1 x_3 x_4 + x_2 x_3.$$

Використовуючи правило де Моргана для правої частини рівняння, перетворимо функцію в КНФ:

$$\begin{aligned} f(x) &= \overline{x_1} \overline{x_2} \overline{x_3} + x_1 x_3 x_4 + x_2 x_3 = \\ &= (x_1 + \overline{x_2} + \overline{x_3})(\overline{x_1} + x_3 + x_4)(x_2 + \overline{x_3}). \end{aligned}$$

При п'яти і більше змінних будують комбіновану карту, що складається з сукупності простих, наприклад чотирьохвимірних. Тоді, спочатку знаходять мінімальні форми усередині чотирьохвимірних карт, а потім, розширюючи поняття сусідніх клітин, відшуковують мінімальні терми для сукупності карт. На рис. 3.3 показані карти для п'яти і шести змінних.

Використання невизначених наборів змінних при мінімізації логічних функцій в картах Карно. Невизначені набори змінних – це вершини кубів, значення яких несуттєві для карт Карно. Інакше кажучи, це ті набори змінних, які не використовуються при побудові цифрового автомату (ЦА). Звичайно їх клітини в картах Карно якимось позначають, наприклад, символом *. При побудові двійково-

десяткових кодів використовують десять наборів чотирьох змінних від 0000 до 1001, а інші шість не використані, тому ці вершини можуть мати різні значення, тобто вони невизначені.

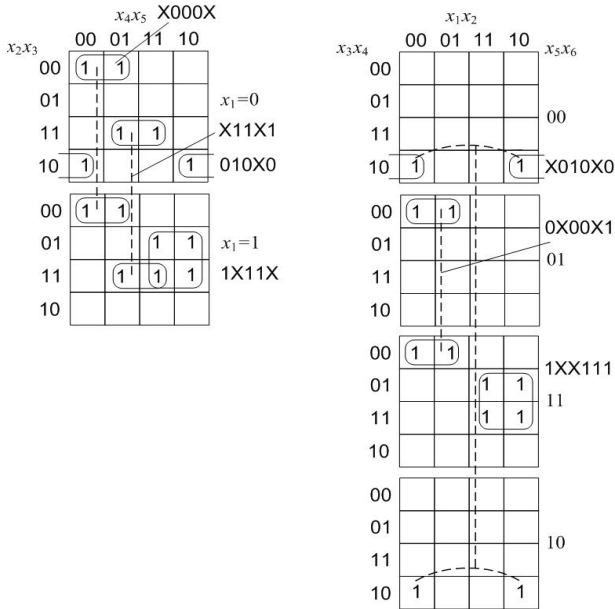


Рисунок 3.3 – Карти Карно для 5 і 6 змінних

При мінімізації невизначені вершини можна визначити самостійно на свій розсуд. Для цього вершини, відзначені символом *, змінюють, привласнюючи їм значення 1 або 0, підвищуючи ефективність мінімізації функції.

Дуже часто це спрощує процес мінімізації, тому що додавання, наприклад, одиниць до наборів, що вже визначені, дозволяє включати в контур покриття більше число одиниць, зменшуючи при цьому число змінних у МДНФ. Визначення невикористаних наборів при мінімізації функції нулями також спрощує мінімальну КНФ функції. Якщо функція має t невизначених наборів змінних, то може бути $2t$ варіантів рішення задачі її визначення. Бажано зупинитися на варіанті, що дає найбільший ефект при мінімізації.

На рис. 3.4 наведений приклад використання невизначених вершин для мінімізації функції

$$f(x) = \overline{x_1} \overline{x_2} x_3 + \overline{x_1} x_2 x_3 + x_1 x_2 x_3.$$

Приведено два варіанти рішення мінімізації функції. Приклад показує, що третій варіант мінімізації (функція f_2) кращий.

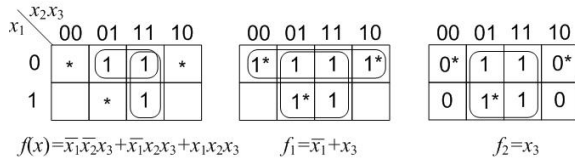


Рисунок 3.4 – Використання невизначених вершин у картах Карно

3.2 Лабораторне завдання

3.2.1 Мінімізувати функцію, задану варіантом, **методом алгебраїчного спрощення**. Побудувати в середовищі EWB схеми за повними та спрощеними виразами, порівняти результати. Всі розрахунки, схеми і часові діаграми навести у звіті.

Варіант №	Функція	Варіант №	Функція
1, 15	$f(x) = \bigvee_1(0,2,3,4,6)$	8, 22	$f(x) = \bigvee_1(0,1,3,4,6)$
2, 16	$f(x) = \bigvee_1(0,1,2,3,7)$	9, 23	$f(x) = \bigvee_1(0,1,3,4,7)$
а	$f(x) = \bigvee_1(2,3,4,6)$	10, 24	$f(x) = \bigvee_1(0,3,4,6)$
4, 18	$f(x) = \bigvee_1(0,2,3,4,6)$	11, 25	$f(x) = \bigvee_1(0,3,4,7)$
5, 19	$f(x) = \bigvee_1(2,3,5,6,7)$	12, 26	$f(x) = \bigvee_1(0,4,6,7)$
6, 20	$f(x) = \bigvee_1(2,3,4,7)$	13, 27	$f(x) = \bigvee_1(0,1,2,4,6)$
7, 21	$f(x) = \bigvee_1(0,1,6,7)$	14, 28	$f(x) = \bigvee_1(0,3,6,7)$

Примітка. $f(x) = \bigvee_1(0,2,3,4,6)$ означає, що в ТІ на відповідних позиціях вхідного двійкового коду, еквівалентних 0,2,3,4,6 в десятковому коді, вихідним значенням буде 1. У всіх інших позиціях – 0.

3.2.2 Мінімізувати функцію, задану варіантом **методом**

невизначених коефіцієнтів. Побудувати в середовище EWB схеми за повними та спрощеними виразами, порівняти результати. Всі розрахунки, схеми і часові діаграми навести у звіті.

Варіант №	Функція	Варіант №	Функція
1, 15	$f(x) = \bigvee_1(0,3,6,7)$	8, 22	$f(x) = \bigvee_1(0,1,6,7)$
2, 16	$f(x) = \bigvee_1(0,1,2,4,6)$	9, 23	$f(x) = \bigvee_1(2,3,4,7)$
3, 17	$f(x) = \bigvee_1(0,4,6,7)$	10, 24	$f(x) = \bigvee_1(2,3,5,6,7)$
4, 18	$f(x) = \bigvee_1(0,3,4,7)$	11, 25	$f(x) = \bigvee_1(0,2,3,4,6)$
5, 19	$f(x) = \bigvee_1(0,3,4,6)$	12, 26	$f(x) = \bigvee_1(2,3,4,6)$
6, 20	$f(x) = \bigvee_1(0,1,3,4,7)$	13, 27	$f(x) = \bigvee_1(2,3,4,6)$
7, 21	$f(x) = \bigvee_1(0,1,3,4,6)$	14, 28	$f(x) = \bigvee_1(0,2,3,4,6)$

3.2.2 Мінімізувати функцію, задану варіантом за допомогою карт Карно. Побудувати в середовище EWB схему за спрощеним виразом. Всі розрахунки, схеми і часові діаграми навести у звіті.

Варіант №	Функція	Варіант №	Функція
1	$f(x) = \bigvee_1(0,3,4,7,8,11,12,15)$	13	$f(x) = \bigvee_1(5,7,10,11,13,14,15)$
2	$f(x) = \bigvee_1(1,2,5,6,9,10,13,14)$	14	$f(x) = \bigvee_1(4,5,10,11,12,13,14,15)$
3	$f(x) = \bigvee_1(0,1,2,3,12,13,14,15)$	15	$f(x) = \bigvee_1(4,5,7,12,13,15)$
4	$f(x) = \bigvee_1(4,5,6,7,8,9,10,11)$	16	$f(x) = \bigvee_1(5,7,8,9,12,13,15)$
5	$f(x) = \bigvee_1(4,5,6,7,8,9,10,11)$	17	$f(x) = \bigvee_1(6,7,10,11,14,15)$
6	$f(x) = \bigvee_1(0,4,5,6,7,8,12)$	18	$f(x) = \bigvee_1(9,10,11,13,14,15)$
7	$f(x) = \bigvee_1(3,7,8,9,10,11,15)$	19	$f(x) = \bigvee_1(2,3,6,7,14,15)$
8	$f(x) = \bigvee_1(2,3,5,6,7,13,15)$	20	$f(x) = \bigvee_1(0,1,4,5,12,13)$
9	$f(x) = \bigvee_1(1,2,3,5,6,7)$	21	$f(x) = \bigvee_1(4,5,8,9,12,13)$

Варіант №	Функція	Варіант №	Функція
10	$f(x) = \bigvee_1(0,1,4,5,6,7,14,15)$	22	$f(x) = \bigvee_1(0,1,4,5,7,13,15)$
11	$f(x) = \bigvee_1(1,3,5,6,7,14,15)$	23	$f(x) = \bigvee_1(4,5,9,11,12,13,15)$
12	$f(x) = \bigvee_1(1,3,4,5,7,12,13)$	24	$f(x) = \bigvee_1(1,3,5,7,13,15)$

3.3 Контрольні питання

3.3.1 Перерахуйте основні прийоми та методи спрощення логічних функцій.

3.3.2 В чому полягає подібність та відмінність між групуванням членів з подальшим використанням тотожностей та приведенням виразу в канонічну форму з подальшим спрощенням?

3.3.3 В чому полягає процес використання закону де Моргана?

3.3.4 В чому полягає особливість використання методу невизначених коефіцієнтів?

3.3.5 В чому полягає особливість використання карт Карно для спрощення логічних функцій?

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Electronic Workbench [Електронний ресурс]– Режим доступу: <https://iowin.net/en/electronic-workbench/>

2. Бабич М.П. Комп'ютерна схемотехніка: Навчальний посібник [Текст] / М.П. Бабич, І.А. Жуков – Київ.: «МК-Прес», 2004. – 412 с.

3. Прикладна теорія цифрових автоматів. Частина 1 [Текст] / Р.К. Кудерметов, А.Н. Щербаков, М.П. Проскурін, С.С. Грушко. – ЗНТУ, 2010. – 196 с.

4. Матвієнко М.П. Комп'ютерна логіка [Текст] / М.П. Матвієнко. – Київ: Ліра-К, 2012. – 288 с.

5. Матвієнко М. П. М 33 Комп'ютерна логіка. Підручник [Текст] / М.П. Матвієнко – Вид. 2-ге перероб. та доп. – Київ : Видавництво Ліра-К, 2017. – 324 с

6. Дичка І.А. Основи прикладної теорії цифрових автоматів : підручник [Текст] / І. А. Дичка, В. П. Тарасенко, М. В. Онай. – Київ, КПІ ім. Ігоря Сікорського: Політехніка, 2019. – 508 с.