

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського»

КОМП'ЮТЕРНА ЛОГІКА

ПРАКТИКУМ

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для студентів,
які навчаються за спеціальністю 123 «Комп'ютерна інженерія»,
спеціалізаціями «Комп'ютерні системи та мережі» та
«Технології програмування для комп'ютерних систем та мереж»*

Київ
КПІ ім. Ігоря Сікорського
2019

Комп'ютерна логіка: Практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 123 «Комп'ютерні системи та мережі», спеціалізацій «Комп'ютерні системи та мережі» та «Технології програмування для комп'ютерних систем та мереж» / В. І. Жабін, І. А. Клименко, В. В. Ткаченко; КПІ ім. Ігоря Сікорського. – Електронні текстові данні (1 файл: 1,02 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2019. – 97с.

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського
(протокол № 5 від 15.11.2018 р.)*

*за поданням Вченої ради факультету інформатики та обчислювальної техніки (протокол № 4 від
29.11.2018 р.)*

Електронне мережне навчальне видання

КОМП'ЮТЕРНА ЛОГІКА

ПРАКТИКУМ

Укладачі: Жабін Валерій Іванович, д-р. техн. наук, проф.
Клименко Ірина Анатоліївна, д-р. техн. наук, доцент.
Ткаченко Валентина Василівна, канд. техн. наук, доцент.

Відповідальний редактор Прізвище, ініціали, науковий ступінь, вчене звання

Рецензент: Дичка І.А., д.т.н., професор, декан факультету прикладної математики, науковий керівник кафедри програмного забезпечення комп'ютерних систем КПІ ім. Ігоря Сікорського

Навчальний посібник призначено для проведення лабораторних занять і виконання курсової роботи з дисципліни «Комп'ютерна логіка 1. Комп'ютерна логіка» для підготовки студентів за спеціальністю 123 Комп'ютерна інженерія спеціалізацій «Комп'ютерні системи та мережі» та «Технології програмування для логічних схем в заданому елементі базисі та їх дослідженню. Розглядаються прикладні питання теорії цифрових автоматів, методи синтезу логічних схем в заданому елементі базисі та їх дослідженню. Наведено тематика курсових робіт, склад, обсяг і структура курсової роботи, вказівки до виконання її розділів, список літератури.

© КПІ ім. Ігоря Сікорського, 2019

ЗМІСТ

ВСТУП.....	4
ПЕРЕЛІК СКОРОЧЕНЬ.....	6
1. ЛАБОРАТОРНА РОБОТА №1 СИНТЕЗ ПЕРЕМИКАЛЬНИХ ФУНКЦІЙ В РІЗНИХ АЛГЕБРАХ.....	7
2. ЛАБОРАТОРНА РОБОТА №2 МІНІМІЗАЦІЯ ПЕРЕМИКАЛЬНИХ ФУНКЦІЙ.....	15
3. ЛАБОРАТОРНА РОБОТА №3 МІНІМІЗАЦІЯ СИСТЕМ ПЕРЕМИКАЛЬНИХ ФУНКЦІЙ.....	30
4. ЛАБОРАТОРНА РОБОТА №4 МІНІМІЗАЦІЯ ЧАСТКОВО ВИЗНАЧЕНИХ ФУНКЦІЙ.....	37
5. ЛАБОРАТОРНА РОБОТА №5. СИНТЕЗ ЦИФРОВИХ АВТОМАТІВ НА ТРИГЕРАХ.....	48
6. ЛАБОРАТОРНА РОБОТА №6 СИНТЕЗ АВТОМАТІВ З ВИКОРИСТАННЯМ АПАРАТА ЧАСОВИХ ФУНКЦІЙ	56
7. КУРСОВА РОБОТА	
7.1. Завдання на курсову роботу	77
7.2. Зміст курсової роботи.....	78
7.3. Вимоги до оформлення курсової роботи.....	79
7.4. Література	
8. ПРОГРАМНИЙ КОМПЛЕКС ДЛЯ МОДЕЛЮВАННЯ ЛОГІЧНИХ СХЕМ.....	90
ДОДАТОК А. ТИТУЛЬНИЙ АРКУШ.....	97

ВСТУП

Дисципліна „Комп'ютерна логіка” відноситься до циклу професійної та практичної підготовки студентів спеціальності 6.050102 „Комп'ютерна інженерія”. Навчальний модуль „Комп'ютерна логіка – 1. Комп'ютерна логіка” присвячений вивченню логіческих основ побудови схем комп'ютерних систем. Виконання лабораторних робіт дозволяє розширити і закріпити теоретичні знання за курсом, опанувати навичками проектування і дослідження логічних схем.

Кожній лабораторній роботі повинна передувати самостійна підготовка, в процесі якої докладно вивчається опис лабораторної роботи, відповідний розділ конспекту лекцій і літературні джерела. В процесі підготовки складається звіт про лабораторну роботу, в якому повинні бути відображені виконані пункти теоретичного завдання, а також заготовлені необхідні для експериментальної частини лабораторної роботи таблиці, діаграми і т. ін.

Перед початком лабораторної роботи результати підготовки перевіряються викладачем. При цьому студент повинний сформулювати мету і порядок виконання лабораторної роботи, представити заготовлений звіт і відповісти на контрольні питання.

По результатам роботи в лабораторії студент представляє оформлений звіт, який повинний містити короткі теоретичні відомості, необхідні для виконання завдання і відповідей на контрольні питання, усі схеми, формули, таблиці, діаграми, графіки, отримані при виконанні завдання та в процесі експериментального дослідження схем, а також висновки по роботі. Залік по лабораторній роботі студент одержує після співбесіди по тематиці виконаної роботи.

Лабораторні роботи виконуються з використанням програмного моделюючого комплексу ПРОГМОЛС 2.0 (ПРОГрама МОделювання

Логічних Схем). Комплекс розроблено на кафедрі обчислювальної техніки НТУУ "КПІ" за участю студентів і призначений для моделювання процесів у комбінаційних і послідовносних схемах. Рекомендації щодо використання комплексу надані в розділі 8.

Курсова робота виконується за індивідуальним завданням. Вона призначена для розширення, закріплення і практичного застосування знань, умінь і навичок, отриманих студентом при вивченні курсу. У процесі виконання роботи студент повинен також вивчити процес створення проектно-конструкторської документації відповідно до діючих стандартів.

ПЕРЕЛІК СКОРОЧЕНЬ

ДДНФ	Досконала диз'юнктивна нормальна форма
СДНФ	Скорочена диз'юнктивна нормальна форма
ТДНФ	Тупикова диз'юнктивна нормальна форма
МДНФ	Мінімальна диз'юнктивна нормальна форма

1. ЛАБОРАТОРНА РОБОТА №1

СИНТЕЗ ПЕРЕМИКАЛЬНИХ ФУНКЦІЙ В РІЗНИХ АЛГЕБРАХ

Ціль роботи – вивчити методи синтезу комбінаційних схем в заданому елементному базисі, визначення складності і дослідження швидкодії комбінаційних схем.

Теоретичні відомості

Логічний елемент – це електронна схема, що реалізує певну перемикальну функцію.

Сукупність логічних елементів, призначена для перетворення двійкових змінних, називається *логічною схемою*.

Логічні схеми поділяються на послідовні і комбінаційні.

Комбінаційною називається логічна схема, в якій значення вихідних сигналів цілком визначаються значеннями вхідних сигналів, що діють в даний момент часу і не залежать від значень вхідних сигналів, що діяли в попередні моменти часу.

Вважають, що така схема має один стан. Поведінка комбінаційної схеми може бути описана системою перемикальних функцій.

Розрізняють задачі аналізу і синтезу комбінаційних схем.

Задача аналізу комбінаційної схеми зводиться до знаходження системи функцій, що відбивають логіку роботи цієї схеми.

Задача синтезу зворотна задачі аналізу, тобто припускає побудову схеми, використовуючи заданий базис логічних елементів.

Синтез комбінаційної схеми з одним виходом можна розбити на три етапи.

На першому етапі виконують мінімізацію перемикальної функції.

На другому етапі функцію записують у так званій операторній формі, тобто у вигляді суперпозиції операторів заданих логічних елементів.

Оператором логічного елемента називають функцію, що реалізує цей елемент. Якщо число входів у елементів досить, то одержання операторного запису функції зводиться до її представлення в одній з нормальних форм.

В базисі елементів І, АБО, НЕ, І-НЕ, АБО-НЕ таких форм вісім.

На прикладі функції $F(X, Y, Z) = \bar{X} \cdot Y \vee X \cdot \bar{Y} \vee \bar{Y} \cdot Z$ і її заперечення $\overline{F(X, Y, Z)} = X \cdot Y \vee \bar{X} \cdot \bar{Y} \cdot \bar{Z}$, покажемо одержання всіх нормальних форм.

Позначати нормальні форми будемо з використанням внутрішньої і зовнішньої функцій. Наприклад, у диз'юнктивної нормальної форми (ДНФ) внутрішньою є функція І, а зовнішньою – АБО, тобто ДНФ – форма типу І/АБО.

Взявши подвійне заперечення заданої функції і застосувавши кілька разів правило де Моргана, послідовно одержимо такі нормальні форми:

$$\begin{aligned}
 F(X, Y, Z) &= \bar{X} \cdot Y \vee X \cdot \bar{Y} \vee \bar{Y} \cdot Z = && \text{(форма І/АБО);} \\
 &= \overline{\overline{\bar{X} \cdot Y \vee X \cdot \bar{Y} \vee \bar{Y} \cdot Z}} = \\
 &= \overline{\overline{\bar{X}} \cdot \overline{\overline{Y}} \vee \overline{X} \cdot \overline{\overline{\bar{Y}}} \vee \overline{\bar{Y}} \cdot \overline{\overline{Z}}} = && \text{(форма І-НЕ/І-НЕ);} \\
 &= \overline{(X \vee \bar{Y}) \cdot (\bar{X} \vee Y) \cdot (Y \vee \bar{Z})} = && \text{(форма АБО/І-НЕ);} \\
 &= \overline{(X \vee \bar{Y}) \vee (\bar{X} \vee Y) \vee (Y \vee \bar{Z})}. && \text{(форма АБО-НЕ/АБО).}
 \end{aligned}$$

Виходячи з заперечення заданої функції, запишемо ще чотири нормальні форми:

$$\begin{aligned}
 F(X, Y, Z) &= \overline{\overline{X \cdot Y \vee \bar{X} \cdot \bar{Y} \cdot \bar{Z}}} = && \text{(форма І/АБО-НЕ);} \\
 &= \overline{X \cdot Y \cdot \bar{X} \cdot \bar{Y} \cdot \bar{Z}} = && \text{(форма І-НЕ/І);} \\
 &= \overline{(\bar{X} \vee \bar{Y}) \cdot (X \vee Y \vee Z)} = && \text{(форма АБО/І);} \\
 &= \overline{(\bar{X} \vee \bar{Y}) \cdot (X \vee Y \vee Z)} =
 \end{aligned}$$

$$= \overline{\overline{(X \vee Y)} \vee (X \vee Y \vee Z)} = \quad (\text{форма АБО-НЕ/АБО-НЕ}).$$

Нормальні форми дозволяють одержати комбінаційну схему з двома рівнями (каскадами) логічних елементів, якщо елементи мають необхідне число входів, а аргументи представлені прямими та інверсними значеннями.

Якщо число входів p елементів менше, ніж потрібно для реалізації нормальної форми, то для одержання операторної форми змінні поєднують у групи, що містять не більше p елементів, і використовують співвідношення виду:

$$X_1 \cdot X_2 \cdot \dots \cdot X_m = (X_1 \cdot \dots \cdot X_g) \cdot \dots \cdot (X_s \cdot \dots \cdot X_m);$$

$$X_1 \vee X_2 \vee \dots \vee X_m = (X_1 \vee \dots \vee X_g) \vee \dots \vee (X_s \vee \dots \vee X_m);$$

$$\overline{X_1 \cdot X_2 \cdot \dots \cdot X_m} = \overline{(X_1 \cdot \dots \cdot X_g) \cdot \dots \cdot (X_s \cdot \dots \cdot X_m)};$$

$$\overline{X_1 \vee X_2 \vee \dots \vee X_m} = \overline{(X_1 \vee \dots \vee X_g) \vee \dots \vee (X_s \vee \dots \vee X_m)},$$

де $g \leq p$ і $m-s+1 \leq p$.

Число груп змінних також не повинне перевищувати p . В протилежному випадку зазначені перетворення виконують стосовно груп змінних. Такі перетворення дозволяють представити задану функцію в операторній формі з урахуванням числа входів елементів. Схема, отримана по операторній формі, може містити більше двох рівнів.

На третьому етапі по операторних представленнях функцій складають комбінаційну схему. Задана система елементів може дозволити реалізувати кілька операторних представлень функції. Наприклад, при наявності елементів І, АБО та І-НЕ можна використовувати в якості вихідної одну з п'яти нормальних форм (І/АБО, І-НЕ/І-НЕ, АБО/І-НЕ, І-НЕ/І, АБО/І) для одержання відповідних операторних представлень з урахуванням числа входів елементів. Щоб вибрати одну схему з декількох можливих,

необхідно порівнювати їх по заданих параметрах (найбільш часто – по складності і швидкодії).

Існує кілька способів оцінки складності схем. Часто використовують оцінку по Квайну (K), яка визначається як сумарне число входів усіх логічних елементів. Складність можна також оцінити в числі логічних елементів (M) чи в числі умовних корпусів мікросхем, що визначається по формулі

$$N = \sum_{i=1}^r \frac{m_i \cdot n_i}{g},$$

де r – число типів мікросхем; m_i , n_i – кількість відповідно мікросхем i -го типу і виводів такої мікросхеми, g – число виводів умовного корпусу. Як умовний корпус в даній роботі використовується корпус мікросхеми на 14 виводів.

Параметри K і M доцільно використовувати при проектуванні інтегральних схем, тому що їх складність залежить від площі кристала, яка пропорційна числу логічних елементів і числу їхніх входів.

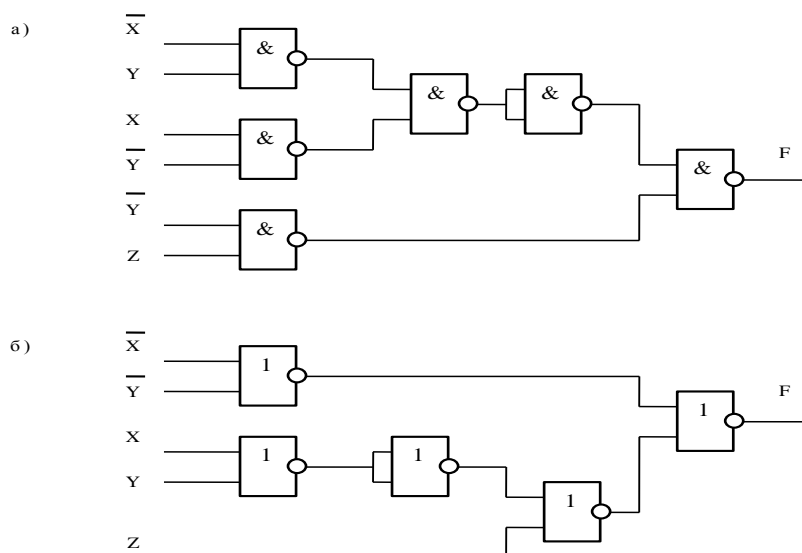


Рис. 1.1. Функціональні схеми реалізації функції: а – на елементах І-НЕ, б – на елементах АБО-НЕ

Оцінка N зручна при порівнянні складності пристроїв, побудованих на мікросхемах. Швидкодія комбінаційних схем залежить від часових параметрів логічних елементів t_{01} і t_{10} , що характеризують затримку сигналів (час переходу вихідного сигналу від одного логічного рівня до іншого). На практиці використовують звичайно усереднене значення часу затримки $t = (t_{01} + t_{10})/2$ чи максимальне – $t^* = \max(t_{01}, t_{10})$.

Для комбінаційних схем на однотипних елементах (приклад на рис.1.1) середній час затримки сигналів $T = Lt$, де L – рівень схеми, дорівнює числу елементів, що входять в максимальний по довжині ланцюжок елементів. Якщо використовуються елементи з різною затримкою, то в схемі визначається шлях, який вимагає максимального часу поширення сигналів.

З декількох можливих вибирають комбінаційну схему, що краще інших задовольняє заданим параметрам. Наприклад, при наявності елементів 2І-НЕ та 2АБО-НЕ розглянуту функцію можна представити в операторних формах І-НЕ/І-НЕ та АБО-НЕ/АБО-НЕ таким чином:

$$F(X, Y, Z) = \overline{\overline{X \cdot Y} \cdot \overline{X \cdot Y \cdot Y} \cdot Z} = \overline{\overline{X \cdot Y} \cdot \overline{X \cdot Y \cdot Y} \cdot Z};$$

$$F(X, Y, Z) = \overline{\overline{(X \vee Y)} \vee \overline{(X \vee Y \vee Z)}} = \overline{\overline{(X \vee Y)} \vee \overline{(X \vee Y \vee Z)}}.$$

Отриманим формам відповідають схеми на рис. 1.1.

Якщо елементи І-НЕ мають менший час затримки сигналів, ніж елементи АБО-НЕ, то схема на рис. 1.1,а більш швидкодіюча, але вона програє другій схемі (рис. 1.1,б) по складності (для першої схеми $K=12$, а для другої $K=10$).

Підготовка до роботи

1. Визначити свій варіант перемикальної функції. Для цього необхідно номер варіанта перевести в двійкову систему числення і записати шість його молодших розрядів у вигляді слова $h_6 h_5 h_4 h_3 h_2 h_1$. Значення h_i підставити в табл.1.1. Наприклад, якщо номер варіанта 19 (у двійковій системі 010 011), то $h_6=0, h_5=1, h_4=0, h_3=0, h_2=1, h_1=1$.

2. Знайти зроблену ДНФ функції і її заперечення. Представити функцію у всіх восьми нормальних формах.

3. Одержати операторні представлення функції, що можуть бути реалізовані на елементах, заданих табл. 1.2 (n – число логічних елементів в корпусі мікросхеми).

4. Вибрати операторні форми, що забезпечують одержання комбінаційної схеми з максимальною швидкістю і комбінаційною схемою з мінімальним числом умовних корпусів, тобто схему з кращим параметром T и схему з мінімальним значенням N . Усі мікросхеми в табл. 1.2 мають по 14 виводів. Побудувати зазначені комбінаційні схеми.

Порядок виконання роботи

1. Побудувати модель заданої комбінаційної схеми.
2. Переконатися в правильності функціонування моделі, визначити часові параметри комбінаційної схеми за допомогою часової діаграми.

Зміст звіту

Звіт повинний містити короткі теоретичні відомості, отримані формули, таблиці, малюнки і висновки за результатами роботи.

Табл. 1.1
Таблиця
істинності

x_3	x_2	x_1	y
0	0	0	h_6
0	0	1	h_5
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	h_4

Табл. 1.2
Варіанти систем елементів

h_3	h_2	h_1	Характеристики елементів		
			Тип	n	t
0	0	0	3І-НЕ	3	10
			3І	3	14
0	0	1	4І-НЕ	2	10
			2АБО	4	12
0	1	0	4І	2	14
			2АБО	4	12
0	1	1	3І	3	14
			2АБО	4	12
1	0	0	2АБО-НЕ	4	12
			4І	2	14
1	0	1	2І-НЕ	4	10
			2АБО	4	12
1	1	0	2АБО-НЕ	4	10
			3І	3	14
1	1	1	2І-НЕ	4	10
			2АБО-НЕ	4	12

Контрольні питання

1. Сформулювати визначення перемикальної функції, логічного елемента, комбінаційної схеми.
2. Основні властивості комбінаційних схем.
3. У чому сутність задач аналізу і синтезу комбінаційних схем?
4. Охарактеризувати основні етапи синтезу комбінаційних схем.
5. Що таке операторне представлення функції?
6. Як визначити складність і швидкодію комбінаційних схем?
7. Чим пояснюється можливість виникнення збоїв комбінаційних схем при збільшенні частоти подачі змінних на їх входи?

Література

1. Жабін В.І. Прикладана теорія цифрових автоматів: Навчальний посібник / В.І.Жабін, І.А.Жуков, І.А.Клименко, В.В.Ткаченко. – К.: Книжкове видавництво НАУ, 2009. – 364 с.
2. Майоров С.А. Принципы организации цифровых машин / С.А.Майоров, Г.И.Новиков. - Л.: Машиностроение, 1977. – 432 с.
3. Поспелов Д.А. Логические методы анализа и синтеза схем / Д.А. Поспелов. – М.: Энергия, 1974. – 367 с.
4. Савельев А.Я. Арифметические и логические основы цифровых автоматом: Учебник. / А.Я.Савельев. – М.: Высшая школа, 1980. – 255 с.
5. Самофалов К.Г. Прикладная теория цифровых автоматом / К.Г.Самофалов, А.М.Романкевич, В.Н.Валуйский и др. – К.: Вища школа, 1987. – 375 с.

2. ЛАБОРАТОРНА РОБОТА №2

МІНІМІЗАЦІЯ ПЕРЕМИКАЛЬНИХ ФУНКЦІЙ

Ціль роботи – вивчення методів мінімізації перемикальних функцій, знаходження операторних форм перемикальних функцій, побудова та дослідження параметрів логічних схем.

Теоретичні відомості

Функції f і φ називаються *еквівалентними*, якщо вони приймають однакові значення на всіх наборах аргументів.

Еквівалентні функції можуть відрізнятися формами представлення і ціною. Під *ціною* перемикальної функції розуміється кількість букв, що входять в її запис.

Проблема мінімізації зводиться до відшукування форми представлення функції з мінімальною ціною. Мінімізація дозволяє спростити схеми, що реалізують перемикальні функції.

В роботі методи мінімізації розглядаються щодо диз'юнктивних форм представлення функцій.

Метод мінімізації Квайна

Вихідною формою представлення функції для мінімізації по методу Квайна є досконала диз'юнктивна нормальна форма (ДДНФ).

Метод забезпечує одержання скороченої ДНФ (СДНФ), тобто сукупності всіх простих імплікант.

Метод базується на використанні співвідношення неповного склеювання

$$Ax \vee \overline{Ax} = Ax \vee \overline{Ax} \vee A$$

і співвідношення поглинання

$$BA \vee A = A,$$

де A і B – довільні кон'юнктивні терми, x – перемінна.

Етапи мінімізації:

- 1) запис функції у вихідній формі – ДДНФ;
- 2) застосування співвідношення склеювання послідовно до конститuent одиниці, потім до імплікант $n-1$ рангу, $n-2$ рангу і так далі, поки можливе формування нових імплікант;
- 3) виконання всіх можливих поглинань, в результаті чого визначаються всі прості імпліканти;
- 4) побудова таблиці покриття (імплікантної матриці) і знаходження тупікових ДНФ (ТДНФ);
- 5) вибір мінімальної ДНФ (МДНФ) з числа ТДНФ.

Приклад 2.1. Виконати мінімізацію функції, заданої табл. 2.1.

Табл. 2.1.

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

Представимо функцію в ДДНФ

$$y = \overline{x_3} \overline{x_2} \overline{x_1} \vee \overline{x_3} x_2 \overline{x_1} \vee \overline{x_3} x_2 x_1 \vee x_3 \overline{x_2} \overline{x_1} \vee x_3 \overline{x_2} x_1 .$$

Виконавши попарне склеювання конституент одиниці, одержуємо множину імплікант 2-го рангу:

$$\overline{x_3} x_1, \overline{x_2} x_1, \overline{x_3} x_2, \overline{x_2} x_1 .$$

Подальше склеювання імплікант неможливе. Тоді функцію можна записати у вигляді

$$y = \overline{x_3} \overline{x_2} \overline{x_1} \vee \overline{x_3} x_2 \overline{x_1} \vee \overline{x_3} x_2 x_1 \vee \overline{x_3} x_2 x_1 \vee x_3 \overline{x_2} \overline{x_1} \vee x_3 \overline{x_2} x_1 \vee x_3 x_1 \vee \overline{x_2} x_1 \vee \overline{x_3} x_2 \vee \overline{x_2} x_1 .$$

Після виконання поглинання, одержуємо СДНФ

$$y = \overline{x_3} x_1 \vee \overline{x_2} x_1 \vee \overline{x_3} x_2 \vee \overline{x_2} x_1 .$$

Будуємо таблицю покриття (табл. 2.2).

Табл. 2.2
Таблиця покриття функції

Імпліканти	Конституенти				
	$\overline{x_3} \overline{x_2} \overline{x_1}$	$\overline{x_3} x_2 \overline{x_1}$	$\overline{x_3} x_2 x_1$	$x_3 \overline{x_2} \overline{x_1}$	$x_3 \overline{x_2} x_1$
$\overline{x_3} x_1$	⊙		⊙		
$\overline{x_2} x_1$	∨			⊙	
$\overline{x_3} x_2$		∨	∨		
$x_2 \overline{x_1}$		⊙			⊙

Знаходимо ядро функції – сукупність імплікант відповідних однократно покритим конституентам. В даному випадку ядро складають імпліканти $\overline{x_2} x_1$ і $x_2 \overline{x_1}$.

Як правило, ядро варто доповнити ще декількома імплікантами, для одержання повного покриття всіх конституент вихідної функції. Різні

варіанти покриття є тупіковими ДНФ. Серед ТДНФ форма з мінімальною ціною буде мінімальної ДНФ (МДНФ).

Для розглянутої функції, існують дві рівноцінні ТДНФ:

$$y = \overline{x_3 x_1} \vee \overline{x_2 x_1} \vee \overline{x_3 x_2} \vee \overline{x_2 x_1};$$

$$y = \overline{x_2 x_1} \vee \overline{x_3 x_2} \vee \overline{x_2 x_1}.$$

В якості МДНФ вибираємо, наприклад,

$$y = \overline{x_3 x_1} \vee \overline{x_2 x_1} \vee \overline{x_3 x_2} \vee \overline{x_2 x_1}.$$

Метод мінімізації Квайна - Мак-Класки

Метод Квайна-Мак-Класки є модифікацією методу Квайна. Він ґрунтується на співвідношеннях неповного склеювання і поглинання, як і метод Квайна. Особливістю методу є використання цифрової форми запису перемикальних функцій. В цьому випадку зменшується число символів для представлення термів і число операцій в процесі мінімізації, що робить метод зручним при програмній реалізації.

Якщо використовувати геометричну інтерпретацію представлення перемикальних функцій, то кожен набір аргументів є n -мірним вектором (n – число аргументів) і визначає точку n -мірного простору. Сукупність усіх наборів представляє n -мірний куб. Конституентам відповідають вершини куба, а імплікантам – ребра і грані. Кожної конкретної функції відповідає певне просторове представлення.

Наприклад, для функції 3-х змінних (рис. 2.1) конституентам відповідають вершини 3-мірного куба, а імплікантам – ребра і грані. Терми n -го рангу називають 0-кубами, терми $(n-1)$ -го рангу – 1-кубами, $(n-2)$ -го рангу – 2-кубами і т.д.

Символом X в r -кубах відзначаються аргументи, по яких склеюються $(r-1)$ -куби. Множина кубів i -го рангу утворюють комплекс K^i .

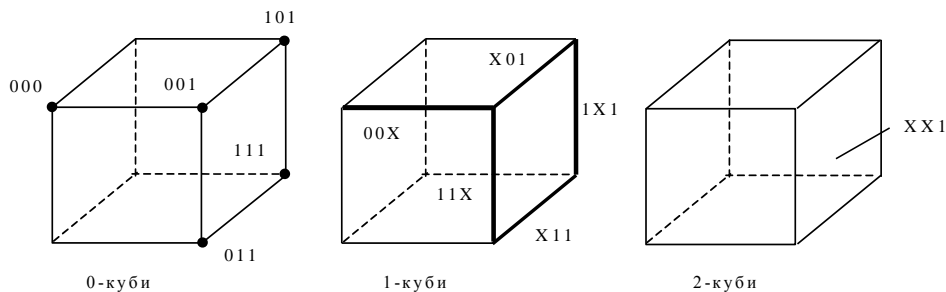


Рис. 2.1 Графічне представлення термів

Етапи мінімізації

1. Для функції виписують комплекс 0-кубів (K^0). Набори упорядковуються по кількості одиниць. Одержують групи без одиниць, з однією одиницею, із двома і т.д. В цьому випадку склеювання можливе тільки між сусідніми групами кубів.

2. Шляхом склеювання формують 1-куби, 2-куби і т.д., поки можливе склеювання. Кожен куб упорядковується аналогічно 0-кубу. При цьому в одну групу повинні входити куби, що мають не тільки однакове число одиниць, але й залежать від тих самих змінних.

3 Шляхом поглинання формується покриття Z , що відповідає скороченій ДНФ.

4 Будується матриця покрить, з якої визначають усі ТДНФ.

5 Серед ТДНФ відшукується МДНФ.

Приклад 2.2. Методом Квайна – Мак-Класки виконати мінімізацію функції, заданої табл. 2.3.

Табл. 2.3

Таблиця

істинності

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

$$K^0 = \begin{cases} \overline{000} \\ \overline{010} \\ \overline{100} \\ \overline{011} \\ \overline{101} \\ \overline{110} \end{cases}; \quad K^1 = \begin{cases} \overline{x00} \\ \overline{0x0} \\ \overline{1x0} \\ \overline{01x} \\ \overline{10x} \end{cases}; \quad K^2 = \{xx0\}; \quad Z = \begin{cases} 01x \\ 10x \\ xx0 \end{cases}.$$

Рис. 2.2 Формування Z-покриття

Виходячи з таблиці істинності функції, записуємо конституенти, поєднуючи їх у групи по кількості одиниць (K^0). Виконуючи склеювання, формуємо куби K^1 і K^2 . Після виконання поглинань, одержуємо Z-покриття (рис. 2.2). Будуємо таблицю покриття (табл. 2.4).

В даному випадку всі імпліканти входять в ядро функції. Отже, МДНФ має вигляд

$$y = \overline{x_3}x_2 \vee x_3\overline{x_2} \vee \overline{x_1}.$$

Метод невизначених коефіцієнтів

Будь-яку функцію можна представити у вигляді диз'юнкції всіх конститuent і всіх можливих імплікант, помножених на відповідний коефіцієнт, що може приймати значення 0 чи 1. (Метод може бути використаний у будь-якій алгебрі перемикальних функцій. Перетерплюють зміни тільки вихідні канонічні форми запису функцій і системи рівнянь для перебування коефіцієнтів).

Наприклад, при $n=2$ можна записати

$$y = k_2^1 x_2 \vee k_2^0 \overline{x_2} \vee k_1^1 x_1 \vee k_1^0 \overline{x_1} \vee k_{21}^{00} \overline{x_2} \overline{x_1} \vee k_{21}^{01} \overline{x_2} x_1 \vee k_{21}^{10} x_2 \overline{x_1} \vee k_{21}^{11} x_2 x_1.$$

Кожна функція визначається своїм набором значень коефіцієнтів. Для пошуку значень коефіцієнтів необхідно вирішити систему рівнянь:

$$y(0,0) = k_2^0 \vee k_1^0 \vee k_{21}^{00};$$

$$y(0,1) = k_2^0 \vee k_1^1 \vee k_{21}^{01};$$

$$y(1,0) = k_2^1 \vee k_1^0 \vee k_{21}^{10};$$

$$y(1,1) = k_2^1 \vee k_1^1 \vee k_{21}^{11}.$$

Всі ненульові коефіцієнти після процедури поглинань визначають сукупність простих імплікант, тобто дозволяють побудувати скорочену ДНФ.

Таблиця 2.4

Таблиця покриття функції

Імпліканти	Конституенти					
	000	010	100	011	101	110
01X		✓		✓		
10X			✓		✓	
XX0	✓	✓	✓			✓

Мінімізацію зручно виконувати за допомогою спеціальної таблиці, що після знаходження простих імплікант розглядається як таблиця покриття. За допомогою таблиці знаходять ТДНФ, а потім визначають МДНФ.

Етапи мінімізації

1. Складання таблиці коефіцієнтів.
2. Викреслювання нульових коефіцієнтів.
3. Виділення простих імплікант.
4. Знаходження покриття, що відповідають ТДНФ.
5. Вибір МДНФ.

Приклад 2.3. Виконати мінімізацію функції, заданої табл. 2.5.

Табл. 2.5

Таблиця
істинності

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

Складаємо таблицю коефіцієнтів (табл. 2.6). Викреслюємо в таблиці коефіцієнти, що знаходяться в рядках з нульовим значенням функції. Викреслені коефіцієнти мають нульові значення. Далі викреслюємо вже знайдені нульові коефіцієнти в інших рядках таблиці. Коефіцієнти, що

залишилися, поглинають у рядку праворуч від себе всі інші коефіцієнти, в індекси яких входять індекси даного коефіцієнта. Наприклад, k_{32}^{01} поглинає k_{321}^{010} . (Поглинені коефіцієнти в табл. 2.6 відзначені зірочкою). Коефіцієнти, що залишилися, $\{k_{32}^{01}, k_{32}^{10}, k_{31}^{01}, k_{31}^{10}, k_{21}^{01}, k_{21}^{10}\}$ визначають СДНФ.

$$y = \overline{x_3}x_2 \vee \overline{x_3}x_2 \vee \overline{x_3}x_1 \vee \overline{x_3}x_1 \vee x_2x_1 \vee x_2x_1.$$

Враховуємо в таблиці тільки ненульові коефіцієнти і розглядаємо її як таблицю покриття функції. Знаходимо дві ТДНФ, що містять по три імпліканти. Вони відповідають множині коефіцієнтів $\{k_{32}^{01}, k_{31}^{10}, k_{21}^{01}\}$ і $\{k_{32}^{10}, k_{31}^{01}, k_{21}^{10}\}$. Інші ТДНФ складаються з чотирьох імплікант.

Табл. 2.6
Таблиця коефіцієнтів

x_3	x_2	x_1	Коефіцієнти							y
0	0	0	k_3^0	k_2^0	k_1^0	k_{32}^{00}	k_{31}^{00}	k_{21}^{00}	k_{321}^{000}	0
0	0	1	k_3^0	k_2^0	k_1^1	k_{32}^{00}	k_{31}^{01}	k_{21}^{01}	k_{321}^{001} *	1
0	1	0	k_3^0	k_2^1	k_1^0	k_{32}^{01}	k_{31}^{00}	k_{21}^{10}	k_{321}^{010} *	1
0	1	1	k_3^0	k_2^1	k_1^1	k_{32}^{01}	k_{31}^{01}	k_{21}^{11}	k_{321}^{011} *	1
1	0	0	k_3^1	k_2^0	k_1^0	k_{32}^{10}	k_{31}^{10}	k_{21}^{00}	k_{321}^{100} *	1
1	0	1	k_3^1	k_2^0	k_1^1	k_{32}^{10}	k_{31}^{11}	k_{21}^{01}	k_{321}^{101} *	1
1	1	0	k_3^1	k_2^1	k_1^0	k_{32}^{11}	k_{31}^{10}	k_{21}^{10}	k_{321}^{110} *	1
1	1	1	k_3^1	k_2^1	k_1^1	k_{32}^{11}	k_{31}^{11}	k_{21}^{11}	k_{321}^{111}	0

В якості МДНФ, наприклад, вибираємо

$$y = \overline{x_3}x_2 \vee \overline{x_3}x_1 \vee x_2x_1.$$

Графічний метод мінімізації функцій

Існують два різновиди таблиць, що забезпечують одержання МДНФ, минаючи етапи формування скороченої і тупікової ДНФ.

На рис. 2.2 представлені діаграми Вейча і Карно для функцій 2, 3 і 4-х аргументів. Номера наборів показані всередині кліток.

Наочність методів зберігається при невеликій кількості аргументів.

Кожна клітинка відповідає конституенті. Прямокутник, що містить 2^k клітинок ($k=1, \dots, n-1$), відповідає імпліканті.

Обґрунтуванням графічного методу мінімізації є той факт, що поруч розташовані клітинки відповідають наборам аргументів, що відрізняється значенням однієї змінної i , таким чином, склеюються по Квайну. Чим більше клітинок містить прямокутник, тим менше букв входить у представлення імпліканти. Імпліканта містить тільки ті змінні, котрі приймають однакові значення для всіх клітинок прямокутника.

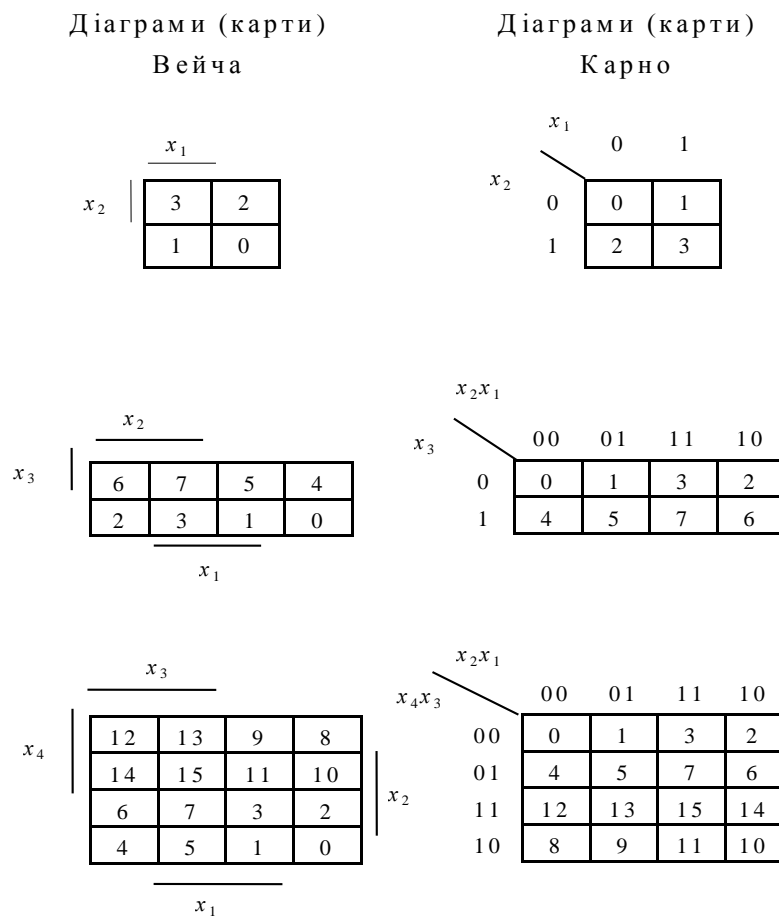


Рис. 2.3. Табличне подання функцій

Етапи мінімізації

1. Заповнення діаграми Вейча чи карти Карно. Значення функцій записують в клітинки, що відповідають номерам наборів.

2. Об'єднання одиниць в прямокутники з максимально можливою кількістю клітинок. Число клітинок повинне дорівнювати 2^k , наприклад 1,2,4,8, 16... При цьому кожна одиниця повинна входити як мінімум в один прямокутник.

3. *Визначення МДНФ. Сукупності простих імплікант, що входять у МДНФ, відповідає мінімальна множина прямокутників, що покривають всі одиниці.*

Приклад 2.4. Одержати МДНФ функцій трьох аргументів (табл. 2.7).

Виходячи з діаграм Вейча на рис. 2.5, записуємо МДНФ функцій:

$$y_1 = \overline{x_2 x_1} \vee \overline{x_3 x_2} \vee \overline{x_2 x_1};$$

$$y_2 = x_3 \vee x_2 x_1 \vee \overline{x_2 x_1};$$

$$y_3 = x_1 \vee \overline{x_3 x_2} \vee \overline{x_3 x_2}.$$

Графічний метод призначений для ручної мінімізації при невеликих n . При цьому відшукування імплікант не формалізованим, і успіх мінімізації цілком визначається кваліфікацією оператора.

Табл. 2.7

Таблиця істинності

x_3	x_2	x_1	y_1	y_2	y_3
0	0	0	0	1	0
0	0	1	1	0	1
0	1	0	1	0	1
0	1	1	1	1	1
1	0	0	0	1	1
1	0	1	1	1	1
1	1	0	1	1	0
1	1	1	0	1	1

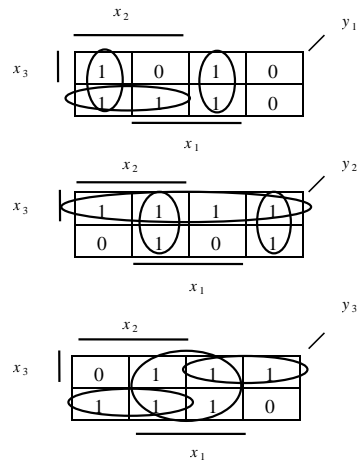


Рис 2.7 Діаграми Вейча

Підготовка до роботи

1. Визначити свій варіант перемикальних функцій, заданих табл. 2.8.
2. Для цього необхідно одержати дев'ять молодших розрядів номера залікової книжки студента, представленого в двійковій системі числення $(h_9 h_8 h_7 \dots h_1)$, а потім підставити h_i в табл. 2.8.

Табл. 2.8

Таблиця істинності

x_4	x_3	x_2	x_1	f_1	f_2	f_3	f_4
0	0	0	0	1	1	h_1	0
0	0	0	1	h_1	0	h_2	1
0	0	1	0	h_2	h_1	h_3	h_3
0	0	1	1	h_3	h_2	h_4	h_4
0	1	0	0	h_4	h_3	1	0
0	1	0	1	1	h_4	h_5	h_5
0	1	1	0	h_5	1	0	0
0	1	1	1	0	h_5	h_6	h_6
1	0	0	0	h_6	0	h_7	h_7
1	0	0	1	h_7	h_6	1	1
1	0	1	0	1	h_7	h_8	h_8
1	0	1	1	h_8	1	0	h_2
1	1	0	0	0	h_8	0	1
1	1	0	1	1	0	h_9	h_9
1	1	1	0	h_9	1	0	h_1
1	1	1	1	0	h_9	1	1

2. Виконати мінімізацію функцій чи їх заперечень (в залежності від заданої елементної бази) наступними методами:

- методом Квайна (для f_1);
- методом Квайна – Мак-Класки (f_2);
- методом невизначених коефіцієнтів (f_3);
- методом діаграм Вейча (f_4).

3. Одержати операторні представлення функцій. Для комбінаційних схем, що реалізують функції f_1 і f_2 використовувати елементи ЗІ-НЕ, а функцій f_3 і f_4 – елементи ЗАБО-НЕ.

4. Представити комбінаційні схеми, що відповідають отриманим операторним представленням функцій.

Порядок виконання роботи

1. Побудувати моделі комбінаційні схеми, вказаних викладачем.
2. Переконатися в правильності функціонування моделей, визначити параметри комбінаційних схем.

Зміст звіту

Звіт повинний містити короткі теоретичні відомості, отримані формули, таблиці, малюнки і висновки за результатами роботи.

Контрольні питання

1. Сформулювати визначення перемикальної функції, конституенти, імпліканти і простої імпліканти функції.
2. Що таке доконана, скорочена, тупікова і мінімальна ДНФ?
3. Дати визначення функціонально повної системи функцій.
4. В чому сутність проблеми мінімізації перемикальних функцій?

5. Охарактеризувати основні етапи мінімізації перемикальних функцій різними методами.
6. Як побудувати операторні форми представлення функцій?
7. Дайте порівняльну оцінку методам мінімізації функцій.

Література

2. Жабін В.І. Прикладна теорія цифрових автоматів: Навчальний посібник / В.І.Жабін, І.А.Жуков, І.А.Клименко, В.В.Ткаченко. – К.: Книжкове видавництво НАУ, 2009. – 364 с.
3. Савельев А.Я. Арифметические и логические основы цифровых автоматов: Учебник. / А.Я.Савельев. – М.: Высшая школа, 1980. – 255 с
4. Поспелов Д.А. Логические методы анализа и синтеза схем / Д.А. Поспелов. – М.: Энергия, 1974. – 367 с.
5. Самофалов К.Г. Прикладная теория цифровых автоматов / К.Г.Самофалов, А.М.Романкевич, В.Н.Валуйский и др. – К.: Вища школа, 1987. – 375 с.

3. ЛАБОРАТОРНА РОБОТА №3

МІНІМІЗАЦІЯ СИСТЕМ ПЕРЕМИКАЛЬНИХ ФУНКЦІЙ

Ціль роботи – вивчення методу мінімізації систем перемикальних функцій, дослідження часових параметрів схем.

Теоретичні відомості

При спільній мінімізації декількох функцій з використанням диз'юнктивних нормальних форм (ДНФ) необхідно усунути дублювання однакових імплікант в представленні функцій, що приводить до спрощення комбінаційної схеми.

В даній роботі розглядається мінімізація систем функцій методом Квайна (Квайна-Мак-Класки).

Вихідною формою для мінімізації функцій методом Квайна є досконала ДНФ (ДДНФ) системи функцій.

Перед мінімізацією кожену функцію необхідно представити в ДДНФ. В цьому випадку ДДНФ системи знаходиться як непересічна множина конститuent одиначі всіх функцій, причому кожній конститuentі приписується множина міток, що визначають її приналежність до функцій.

Етапи мінімізації

1. Склеювання термів.
2. Виконання поглинань.
3. Складання таблиці покриття.
4. Вибір форм представлення кожної функції.

Склеювання і поглинання виконуються за тими же правилами, що і при мінімізації окремих функцій (див. розділ 2). Відмінності методу мінімізації систем полягають в наступному:

- склеювання здійснюються тільки для тих кон'юнктивних термів, в яких маються однакові мітки. Отриманому в результаті склеювання терму привласнюється множина міток, що обумовлюється як перетинання множин міток термів, що склеюються;

-поглинання здійснюється тільки в тому випадку, коли множині міток у термів цілком збігаються.

При виборі остаточної форми представлення функцій з використанням таблиці покриття необхідно прагнути до того, щоб загальне число букв у покритті було мінімальним.

При використанні методу Квайна-Мак-Класки всі операції виконуються аналогічно, але в цифровій формі.

Приклад 3.1. Виконати мінімізацію системи функцій, заданої табл. 3.1.

Табл.3.1

Таблиця істинності

x_3	x_2	x_1	y_1	y_2	y_3
0	0	0	1	0	1
0	0	1	1	1	1
0	1	0	0	1	0
0	1	1	1	1	1
1	0	0	1	1	0
1	0	1	0	0	1
1	1	0	1	1	0
1	1	1	0	0	1

Скористаємося методом Квайна-Мак-Класки. Послідовно одержуємо комплекси кубів (рис. 3.1).

$$K^0 = \begin{cases} \overline{000}(1,3) \\ \overline{001}(1,2,3) \\ 010(2) \\ \overline{100}(1,2) \\ \overline{011}(1,2,3) \\ \overline{101}(3) \\ \overline{110}(1,2) \\ \overline{111}(3) \end{cases}; \quad K^1 = \begin{cases} \overline{X00}(1) \\ \overline{X01}(3) \\ \overline{X10}(2) \\ \overline{X11}(3) \\ 0X1(1,2,3) \\ 1X0(1,2) \\ \overline{1X1}(3) \\ 00X(1,3) \\ 01X(2) \end{cases}; \quad K^2 = \begin{cases} \overline{XX1}(3) \\ \overline{XX1}(3) \end{cases}.$$

Рис.3.1. Формування комплексів кубів

Визначаємо покриття, що відповідає скороченій ДНФ системи:

$$Z = \begin{cases} \overline{X00}(1) \\ \overline{X10}(2) \\ 0X1(1,2) \\ 1X0(1,2) \\ 00X(1,3) \\ 01X(2) \\ \overline{XX1}(3) \end{cases}$$

Складаємо таблицю покриття (табл. 3.2), на підставі якої знаходимо остаточні форми представлення функцій, що забезпечують їх спільну реалізацію з мінімальними апаратурними витратами.

Табл. 3.2
Таблиця покриття функцій

	y ₁					y ₂					y ₃				
	000	001	011	100	110	001	010	011	100	110	000	001	011	101	111
00X(1,3)	✓	✓									✓	✓			
X00(1)	✓			✓											
0X1(1,2,3)		✓	✓			✓		✓				✓	✓		
01X(2)							✓	✓							
X10(2)							✓				✓				
1X0(1,2)				✓	✓					✓	✓				
XX1(3)												✓	✓	✓	✓

$$y_1 = \overline{x_3} \overline{x_2} \vee \overline{x_3} x_1 \vee x_3 \overline{x_1}; y_2 = \overline{x_3} x_1 \vee \overline{x_3} x_2 \vee x_3 \overline{x_1}; y_3 = \overline{x_3} \overline{x_2} \vee x_1.$$

На рис. 3.2 приведена комбінаційна схема, що відповідає знайденим формам функцій.

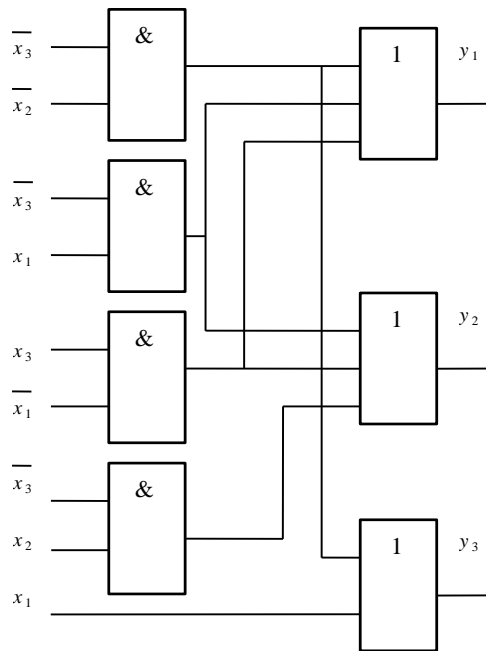


Рис. 3.2. Комбінаційна схема системи функцій

При побудові комбінаційних схем, що відповідають системам булевих функцій, виникає проблема забезпечення заданого коефіцієнта розгалуження по виходу елементів, що реалізують загальні імпліканти.

Якщо число входів, до яких повинний бути підключений вихід елемента, перевищує коефіцієнт розгалуження, то застосовують способи дублювання елемента чи посилення сигналу.

В першому випадку входи однакових елементів з недостатньою навантажувальною здатністю з'єднують паралельно, а в другому випадку до виходу елемента підключають повторювачі.

Підготовка до роботи

Табл. 3.3

Таблиця істинності

x_4	x_3	x_2	x_1	f_1	f_2	f_3
0	0	0	0	1	1	h_1
0	0	0	1	h_1	0	h_2
0	0	1	0	h_2	h_1	h_3
0	0	1	1	h_3	h_2	h_4
0	1	0	0	h_4	h_3	1
0	1	0	1	1	h_4	h_5
0	1	1	0	h_5	1	0
0	1	1	1	0	h_5	h_6
1	0	0	0	h_6	0	h_7
1	0	0	1	h_7		1
1	0	1	0	1	h_7	h_8
1	0	1	1	h_8	1	0
1	1	0	0	0	h_8	0
1	1	0	1	1	0	h_9
1	1	1	0	h_9	1	0
1	1	1	1	0	h_9	1

1. Визначити свій варіант системи перемикальних функцій (табл. 3.3).

Для цього необхідно одержати дев'ять молодших розрядів номера залікової книжки студента, представленого в двійковій системі числення ($h_9 h_8 h_7 \dots h_1$), а потім підставити h_i в табл. 3.3.

2. Виконати спільну мінімізацію функцій і їх заперечень методом Квайна чи Квайна-Мак-Класки.

3. Одержати операторні представлення функцій (два варіанти) для їхньої реалізації у формі І/АБО і формі І/АБО-НЕ. Можна використовувати елементи з будь-яким числом входів, не більш чотирьох.

4. Представити комбінаційні схеми, що відповідають отриманим операторним формам. При необхідності виконати розв'язку схем з урахуванням коефіцієнта розгалуження елементів по виходу, рівного двом.

Порядок виконання роботи

1. Побудувати моделі комбінаційні схеми, зазначених викладачем.
2. Переконайтеся в правильності функціонування моделей.
3. Визначити часові параметри схем за допомогою часових діаграм.

Зміст звіту

Звіт повинний містити короткі теоретичні відомості, отримані формули, таблиці, малюнки і висновки за результатами роботи.

Контрольні питання

1. Сформулюйте визначення доконаної ДНФ системи функцій.
2. Охарактеризуйте етапи мінімізації системи функцій.
3. Як виконуються операції склеювання і поглинання при мінімізації систем перемикальних функцій?
4. В чому сутність проблеми мінімізації систем перемикальних функцій?

5. Охарактеризуйте способи забезпечення заданого коефіцієнта розгалуження елементів по виходу. Вкажіть переваги і недоліки способів.

6. Як побудувати операторні форми представлення функцій?

Література

6. Жабін В.І. Прикладана теорія цифрових автоматів: Навчальний посібник / В.І.Жабін, І.А.Жуков, І.А.Клименко, В.В.Ткаченко. – К.: Книжкове видавництво НАУ, 2009. – 364 с.

7. Савельев А.Я. Арифметические и логические основы цифровых автоматов: Учебник. / А.Я.Савельев. – М.: Высшая школа, 1980. – 255 с

8. Поспелов Д.А. Логические методы анализа и синтеза схем / Д.А. Поспелов. – М.: Энергия, 1974. – 367 с.

9. Самофалов К.Г. Прикладная теория цифровых автоматов / К.Г.Самофалов, А.М.Романкевич, В.Н.Валуйский и др. – К.: Вища школа, 1987. – 375 с.

4. ЛАБОРАТОРНА РОБОТА №4

МІНІМІЗАЦІЯ ЧАСТКОВО ВИЗНАЧЕНИХ ФУНКЦІЙ

Ціль роботи – вивчення методів мінімізації частково визначених функцій, аналітичного одержання множини ТДНФ, дослідження параметрів комбінаційних схем.

Теоретичні відомості

В реальних системах можливі випадки, коли не всі набори змінних можуть подаватися на входи комбінаційної схеми, тобто існують заборонені вхідні комбінації змінних.

На заборонених наборах функція вважається невизначеною, що дає додаткові можливості для спрощення комбінаційної схеми. В таблиці істинності значення функції на таких наборах відзначаються символом, відмінним від 0 і 1, наприклад – прочерком. Довизначення функції на заборонених наборах необхідно робити таким чином, щоб забезпечити найбільш ефективну мінімізацію.

При використанні для мінімізації методу діаграм Вейча прочерки розглядають як одиниці в тих випадках, коли це приводить до збільшення розміру прямокутника, що відповідає імпліканті. В протилежному випадку вони розглядаються як нулі.

Приклад 4.1. Знайти МДНФ функції, заданої діаграмою Вейча (рис. 4.1).

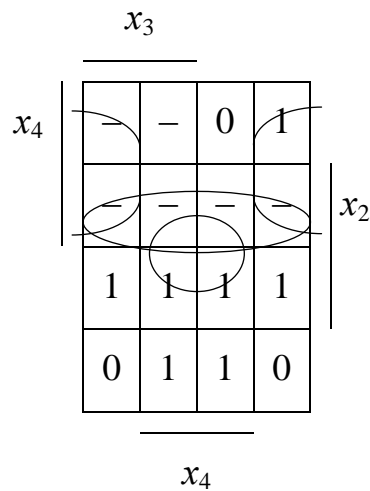


Рис. 4.1.
Діаграма
Вейча

Після відшукування простих імплікант, запишемо МДНФ

$$y = \overline{x_4} \overline{x_1} \vee x_2 \vee \overline{x_4} x_1.$$

При використанні аналітичних методів мінімізації функцій у її ДДНФ вводять всі конституенти заборонених наборів, але в таблицю покриття дані конституенти не включаються.

Приклад 4.2. Одержати МДНФ частково визначеної функції, заданої табл. 4.1 методом Квайна.

Табл. 4.1

Таблиця
істинності

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

Записуємо ДДНФ:

$$y = \overline{x_3} \overline{x_2} \overline{x_1} \vee \overline{x_3} \overline{x_2} x_1 \vee \overline{x_3} x_2 \overline{x_1}.$$

Після доповнення конституентами одиниці, що відповідають забороненим наборам, функція приймає вигляд

$$y = \overline{x_3} \overline{x_2} \overline{x_1} \vee \overline{x_3} \overline{x_2} x_1 \vee \overline{x_3} x_2 \overline{x_1} \vee \overline{x_3} x_2 x_1 \vee x_3 \overline{x_2} \overline{x_1} \vee x_3 \overline{x_2} x_1 \vee x_3 x_2 \overline{x_1} \vee x_3 x_2 x_1.$$

В результаті склеювання і поглинання одержуємо скорочену ДНФ функції, що довизначена, можливо, не оптимально.

$$y = \overline{x_2} \vee x_1 \vee x_3.$$

Для зручності мінімізації конституенти та імпліканти записуємо стовпцями. (рис. 4.2).

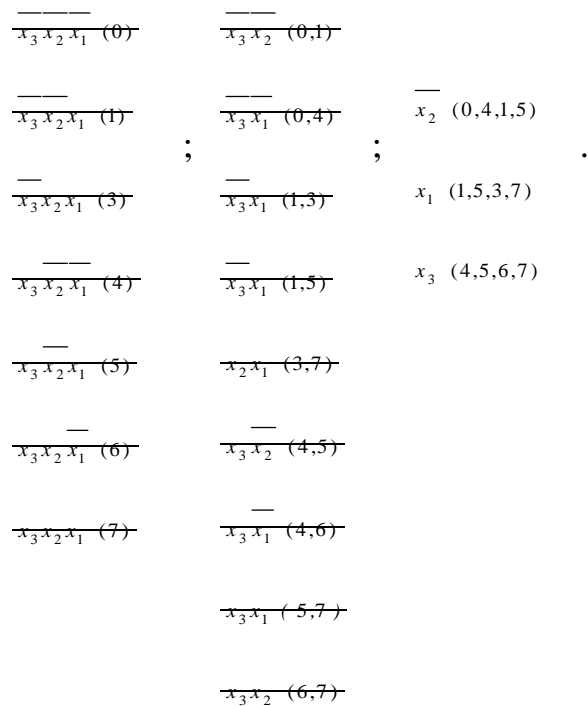


Рис. 4.2. Склеювання і поглинання термів

Будуємо таблицю покриття, в яку включаємо тільки три конституенти одиниці (табл. 4.2).

Табл. 4.2

Таблиця покриття

Імпліканти	Конституенти		
	$\overline{x_3 x_2 x_1}$	$\overline{x_3 x_2 x_1}$	$\overline{x_3 x_2 x_1}$
$\overline{x_2}$	∨	∨	
x_1		∨	
x_3			∨

Знаходимо ТДНФ функції, що одночасно є її МДНФ:

$$y = \overline{x_2} \vee x_3.$$

При мінімізації системи частково визначених функцій в її ДДНФ також вводяться конституенти, що відповідають забороненим наборам. На етапі вибору мінімального покриття ці конституенти не включаються в таблицю покриттів.

Приклад 4.3. Виконати мінімізацію методом Квайна-Мак-Класки, системи функцій, заданої табл. 4.3.

Табл. 4.3

Таблиця істинності

x_3	x_2	x_1	y_1	y_2	y_3
0	0	0	-	1	-
0	0	1	1	1	0
0	1	0	0	0	0
0	1	1	1	1	1
1	0	0	1	-	1
1	0	1	0	-	1
1	1	0	0	0	0
1	1	1	-	1	1

Випишемо 0-куби, що відповідають забороненим наборам і наборам, на яких функції приймають одиничне значення. При цьому відзначаємо приналежність кубів до заданих функцій. Виконуємо склеювання і поглинання (рис. 4.3).

$$K^0 = \left\{ \begin{array}{l} \underline{000(1,2,3)} \text{ ---} \\ \underline{001(1,2)} \\ \underline{100(1,2,3)} \text{ ---} \\ \underline{011(1,2,3)} \\ \underline{101(2,3)} \text{ ---} \\ \underline{111(1,2,3)} \end{array} \right\}; \quad K^1 = \left\{ \begin{array}{l} \underline{\underline{X00(1,2,3)}} \text{ ---} \\ \underline{\underline{X01(2)}} \text{ ---} \\ \underline{\underline{X11(1,2,3)}} \text{ ---} \\ \underline{0X1(1,2)} \text{ ---} \\ \underline{1X1(2,3)} \text{ ---} \\ \underline{00X(1,2)} \text{ ---} \\ \underline{10X(2,3)} \end{array} \right\}; \quad K^2 = \left\{ \begin{array}{l} \underline{X0X(2)} \\ \underline{\underline{X0X(2)}} \\ \underline{XX1(2)} \\ \underline{\underline{XX1(2)}} \end{array} \right\}; \quad Z = \left\{ \begin{array}{l} \underline{X00(1,2,3)} \\ \underline{X11(1,2,3)} \\ \underline{0X1(1,2)} \\ \underline{1X1(2,3)} \\ \underline{00X(1,2)} \\ \underline{10X(2,3)} \\ \underline{X0X(2)} \\ \underline{XX1(2)} \end{array} \right\}.$$

Рис. 4.3. Склеювання і поглинання термів

Складаємо таблицю покриття (табл. 4.4), на підставі якої знаходимо форми функцій вихідної системи, що забезпечують їхню спільну реалізацію.

Табл.4.4
Таблиця покриття функцій

	y_1			y_2				y_3			
	001	011	100	000	001	011	111	011	100	101	111
$a=00X(1,2)$	⊙			⊙	⊙						
$b=X00(1,2,3)$			⊙	∨					⊙		
$c=0X1(1,2)$	∨	∨			∨	∨					
$d=X11(1,2,3)$		⊙				⊙	⊙	⊙			∨
$e=10X(2,3)$									∨	∨	
$f=1X1(2,3)$							∨			⊙	⊙
$g=X0X(2)$				∨	∨						
$h=XX1(2)$					∨	∨	∨				

Для визначення покриття (для однієї функції чи системи функцій) як можливий варіант можна використовувати метод Петрика, що складається з виконання наступних етапів:

- визначення умов покриття імплікантами кожної константи одиниці окремо, використовуючи функцію АБО;
- складання умови одночасного покриття всіх конституент одиниці з використанням функції І;
- розкриття дужок в отриманому логічному вираженні за правилами булевої алгебри.

Кон'юнктивні терми, отримані в результаті виконання зазначених етапів, відповідають множинам імплікант, кожне з яких визначає можливе покриття. З отриманих варіантів покриття вибирають один відповідно до

цільової функції проектування (мінімальні апаратні витрати, максимальна швидкодія і т.ін.).

Позначимо для зручності виконання логічних перетворень імпліканти в табл. 4.4 буквами від a до h .

Умова покриття 001 має вигляд $(a \vee c)$, для 011 одержимо $(c \vee d)$ і т.ін. Покриття всіх конститuent визначається узагальненою умовою

$$(a \vee c)(a \vee d)b(a \vee b \vee g)(a \vee c \vee g \vee h)(c \vee d \vee h)(d \vee f \vee h)d(b \vee e)(e \vee f)(d \vee f)$$

Після поглинання диз'юнктивних термов одержуємо $(a \vee c)bd(e \vee f)$, а після розкриття дужок знаходимо $abde \vee cbde \vee abdf \vee c bdf$.

Отримані терми визначають чотири множини імпліканти, що покривають функції. Якщо вибрати множину $\{a, b, d, f\}$, то можна одержати функції:

$$y_1 = \overline{x_3}x_2 \vee x_2x_1 \vee \overline{x_2}x_1; \quad y_2 = \overline{x_3}x_2 \vee x_2x_1; \quad y_3 = x_2x_1 \vee \overline{x_2}x_1 \vee x_3x_1.$$

При перехідних процесах на виходах комбінаційних схем можуть формуватися помилкові (не передбачені таблицею істинності) короткочасні сигнали. Якщо такі сигнали неприпустимі (можуть привести до неправильного спрацьовування інших схем) то для їх усунення використовуються апаратні "фільтри", варіанти яких представлені на рис.4.4.

Табл. 4.5

Таблиця істинності

x_4	x_3	x_2	x_1	f_1	f_2	f_3
0	0	0	0	1	1	1
0	0	0	1	1	1	0
0	0	1	0	1	1	1
0	0	1	1	0	0	0
0	1	0	0	-	0	1
0	1	0	1	0	0	0
0	1	1	0	1	-	-
0	1	1	1	-	-	1
1	0	0	0	1	h_4	h_7
1	0	0	1	0	0	h_8
1	0	1	0	0	0	h_9
1	0	1	1	h_1	0	0
1	1	0	0	1	-	1
1	1	0	1	h_2	h_5	0
1	1	1	0	h_3	h_6	0
1	1	1	1	1	1	1

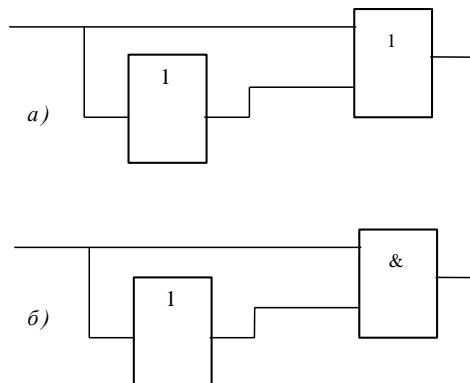


Рис. 4.4. Схеми для усунення короточасних сигналів: а – нулевого рівня;

б – одиничного рівня

Підготовка до роботи

1. Визначити свій варіант системи перемикальних функцій (табл. 4.5).

Для цього необхідно одержати дев'ять молодших розрядів номера залікової книжки студента, представленого в двійковій системі числення

$(h_9 h_8 h_7 \dots h_1)$, а потім підставити h_i в табл. 4.5.

2. Виконати окремо мінімізацію кожної функції методом Вейча.

3. Виконати спільну мінімізацію функцій методом Квайна.

4. Виконати спільну мінімізацію заперечення функцій методом Квайна-Мак-Класки.

5. Одержати представлення функцій у формі I-НЕ/I-НЕ і формі I-НЕ/I. Число входів елементів не повинне перевищувати чотирьох.

6. Представити комбінаційні схеми, що відповідають отриманим операторним формам. Оцінити можливість формування короткочасних помилкових сигналів в отриманих схемах. Показати способи усунення ризику збою в комбінаційних схемах.

Порядок виконання роботи

1. Побудувати моделі комбінаційні схеми, вказаних викладачем.

2. Переконатися в правильності функціонування моделей, усунути при необхідності короткочасні помилкові вихідні сигнали.

3. Визначити часові параметри схем за допомогою часових діаграм.

Зміст звіту

Звіт повинний містити короткі теоретичні відомості, отримані формули, таблиці, малюнки і висновки за результатами роботи.

Контрольні питання

1. В чому складається особливість мінімізації частково визначених функцій?
2. Охарактеризувати основні етапи спільної мінімізації систем частково визначених перемикальних функцій.
3. Як одержати операторні форми представлення функцій?
4. Чим пояснюється можливість виникнення збоїв комбінаційних схем?
5. Як оцінити апаратурні витрати і швидкодію комбінаційних схем?
6. Як забезпечити заданий коефіцієнт розгалуження елементів по виходу при побудові комбінаційних схем з багатьма виходами?
7. Як усунути короточасні помилкові сигнали на виходах комбінаційної схеми при перехідних процесах?
8. Поясніть етапи знаходження мінімального покриття функції методом Петрика.
9. Що є вихідними даними для використання метода Петрика?

Література

10. Жабін В.І. Прикладана теорія цифрових автоматів: Навчальний посібник / В.І.Жабін, І.А.Жуков, І.А.Клименко, В.В.Ткаченко. – К.: Книжкове видавництво НАУ, 2009. – 364 с.

11. Пospelов Д.А. Логические методы анализа и синтеза схем / Д.А. Пospelов. – М.: Энергия, 1974. – 367 с.
12. Самофалов К.Г. Прикладная теория цифровых автоматов / К.Г.Самофалов, А.М.Романкевич, В.Н.Валуйский и др. – К.: Вища школа, 1987. – 375 с.

5. ЛАБОРАТОРНА РОБОТА №5.

СИНТЕЗ ЦИФРОВИХ АВТОМАТІВ НА ТРИГЕРАХ

Ціль роботи

Вивчити методи структурного синтезу керуючих автоматів із жорсткою логікою, одержати навички в їх налагодженні та експериментальному дослідженні.

Теоретичні відомості

Цифровий автомат, що має два і більше станів, є послідовною схемою. Ознакою такої логічної схеми є наявність петель. Під петлею розуміється шлях з виходу логічного елемента на його вхід безпосередньо або через інші елементи.

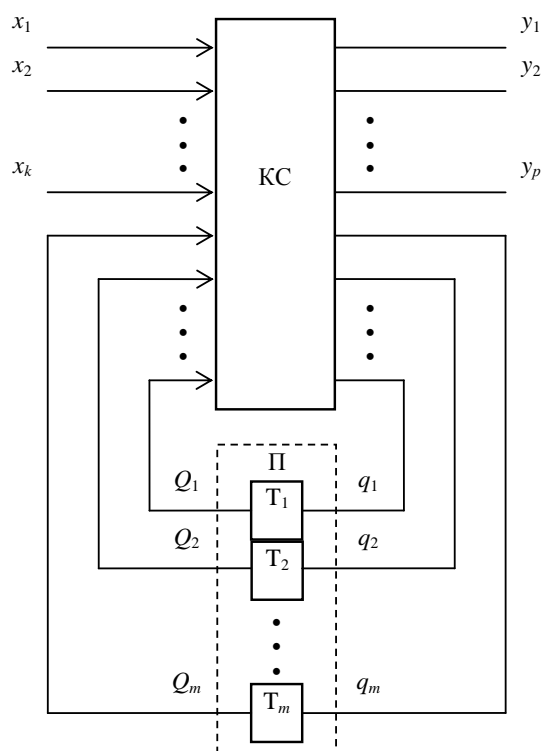


Рис.5.1 Структурна схема автомата

Автомат може виконувати функцію управління для пристроїв обробки інформації.

Послідовність мікрооперацій, виконання яких призводить до виконання операції, називають мікроалгоритмом даної операції.

Один з підходів теорії цифрових автоматів до побудови структурних автоматів полягає в представленні будь-якого автомата у вигляді композиції елементарних автоматів Мура, що мають назву тригерів (рис. 5.1).

Автомат містить комбінаційну схему (КС) і пам'ять (П), що складається з тригерів T_i . Входами КС є виходи Q_1, \dots, Q_m тригерів і входні сигнали (логічні умови) x_1, \dots, x_k , що формуються в операційному пристрої. КС виробляє керуючі сигнали y_1, \dots, y_p для операційного пристрою і функції збудження тригерів q_1, \dots, q_m , що визначають перехід автомата з одного стану в інший. Кожній з безлічі станів $\{a_1, \dots, a_m\}$ відповідає визначений набір значень Q_i .

Якщо вихідні сигнали залежать тільки від стану, в якому знаходиться автомат, його називають автоматом Мура. Закон функціонування такого автомата визначається функціями переходів і виходів відповідно

$$a^{s+1} = \delta(a^s, x^s),$$

$$y^{s+1} = \lambda(a^s),$$

де $s=0, 1, 2, \dots$ – моменти автоматного (дискретного) часу;

δ – функція переходів;

λ – функція виходів;

$a \in \{a_1, a_2, \dots, a_m\}$ – стан автомата;

$x = \{x_1, x_2, \dots, x_k\}$ – вектор значень вхідних сигналів;

$y = \{y_1, y_2, \dots, y_p\}$ – вектор вихідних сигналів автомата.

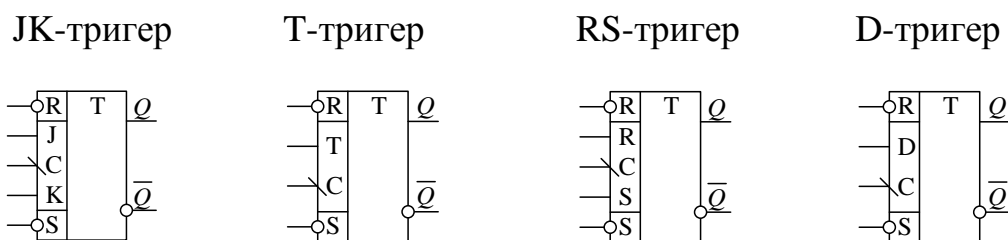
Автомат, вихідні сигнали якого залежать як від стану, так і від вхідних сигналів, називають автоматом Мілі. Його функціонування визначається виразами

$$a^{s+1} = \delta(a^s, x^s),$$

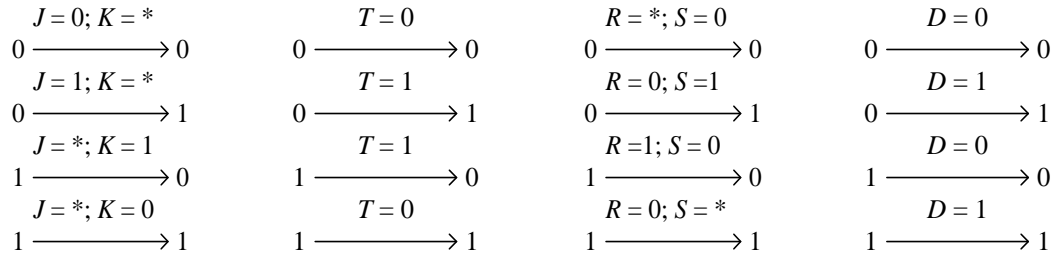
$$y^{s+1} = \lambda(a^s, x^s).$$

Можна виділити чотири основні функціональні типи тригерів: RS-тригери, JK-тригери, D-тригери і T-тригери. Умовні графічні позначення синхронних тригерів і системи підграфів переходів, що пояснюють спосіб зміни стану тригерів, зображені на рис. 5.2. Тригери мають тільки два стани: нульовий стан – при $Q = 0$ і $\bar{Q} = 1$, та одиничний стан – при $Q = 1$ і $\bar{Q} = 0$. Перехід тригерів з одного стану в інший визначається інформаційними сигналами, а момент переходу – перепадом синхросигналу С (в даному випадку перепад з 1 в 0). Асинхронні входи тригерів R і S дозволяють встановлювати початковий стан тригерів.

Вихідними даними для синтеза автомата є схема операційного пристрою, що дає можливість зробити перелік керуючих сигналів для кожного вузла, і змістовний мікроалгоритм операції, що включає опис мікрооперацій і логічних умов. Побудова схеми і розробка мікроалгоритму є взаємозалежними процесами.



a)



б)

Рис. 5.2. Синхронні тригери: а – умовні графічні означення; б – система підграфів переходів

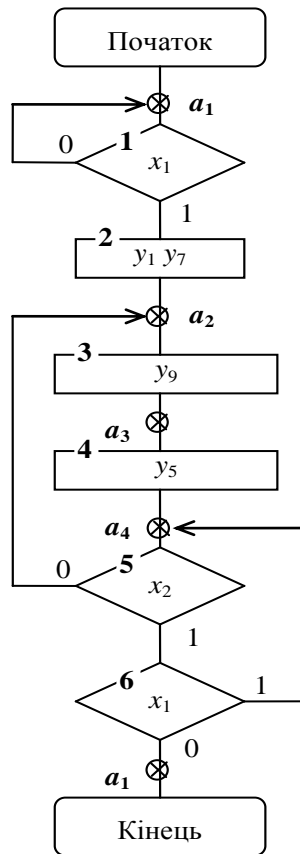


Рис. 5.3. Графічна схема закодованого мікроалгоритму автомата Мілі

Синтез автомата включає наступні етапи:

- 1) складання списку керуючих сигналів, що забезпечують виконання кожної мікрооперації;
- 2) визначення тривалості кожного керуючого сигналу (в числі тактів) і періоду тактуючих сигналів автомата;
- 3) одержання закодованого мікроалгоритму;
- 4) відмітка станів автомата;
- 5) складання графа автомата;
- 6) кодування станів автомата;
- 7) складання структурної таблиці автомата;
- 8) одержання МДНФ функцій збудження тригерів і керуючих сигналів;
- 9) представлення функцій збудження тригерів і керуючих сигналів в операторній формі;
- 10) побудова схеми керуючого автомата.

Розглянемо приклад синтезу автомата Мілі.

Розглянемо приклад синтезу автомата. Будемо вважати, що після виконання пунктів 1, 2 і 3 одержано закодований структурний мікроалгоритм (рис. 5.3), де y_i ($i=1,5,7,9$) – управляючі сигнали, що подаються на операційний пристрій, а x_i – логічні умови (зовнішні для автомата вхідні сигнали). Для переходу до графа автомата треба зробити розмітку станів. Розмітка станів для циклічного автомата Мілі здійснюється наступним чином.

Символом a_i відзначається вхід вершини (логічної або операторної), наступної за початковою, а також вхід кінцевої вершини; входи всіх

вершин, наступних за операторними, повинні бути відзначені різними символами a_i .

Відмітимо, що спосіб кодування впливає на правильність формування керуючих сигналів і складність автомата. Можливість формування сигналів, не передбачених графом автомата, при неоптимальному кодуванні станів обумовлена появою “гонок”, що пов'язано з розкидом часу переключення окремих тригерів автомата. Наприклад, при переході автомата зі стану 10 у стан 01 під час переключення тригерів можлива поява станів 00 або 11 (в залежності від того, який із тригерів раніш спрацює). Ці проміжні стани при використанні тригерів із внутрішньою затримкою не впливають на правильність переключення автомата, однак можуть привести до появи короткочасних помилкових керуючих сигналів. Для усунення цього недоліку можна використовувати протигончне сусіднє кодування. При сусіднім кодуванні перехід автомата з одного в будь-який інший припустимий для даного автомата стан здійснюється переключенням тільки одного тригера, внаслідок чого ”гонки” не виникають. В автоматах, що не допускають сусіднього кодування, необхідно вводити додаткові стани.

Таблиця 5.1

Таблиця кодування станів
автомата Мілі

Стан	Код стану	
	Q_1	Q_2
a_1	0	0
a_2	0	1
a_3	1	1
a_4	1	0

Крім станів, визначених таким чином, може виникнути необхідність введення додаткових станів. Наприклад, якщо мікрооперація виконується по перепаду керуючого сигналу і мікроалгоритм має петлю, яка охоплює операторну вершину з таким сигналом, або кілька вершин, що слідують підряд, то для забезпечення перепадів керуючого сигналу при кожному черговому виконанні мікрооперації необхідно вводити додаткові стани. В цих станах автомат не повинний виробляти керуючі сигнали. Додаткові стани можуть знадобитися для забезпечення протигончного кодування (див. далі).

Мікроалгоритм на рис. 5.3 відзначений чотирма різними станами (a_1, a_2, a_3, a_4). Граф розглянутого автомата Мілі зображений на рис. 5.4.

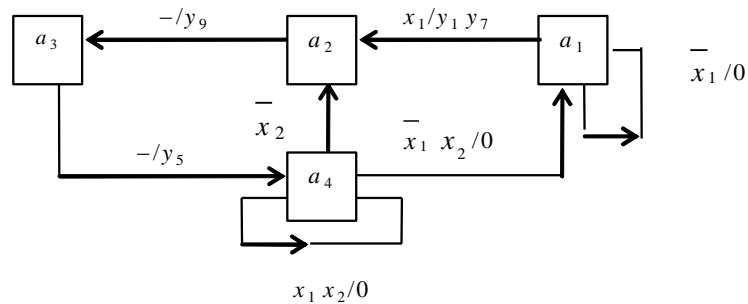


Рис. 5.4. Граф автомата Мілі

Число вершин графа дорівнює числу станів a_i автомата. Кожному переходові автомата з одного стану в інший відповідає дуга графа. Дугі приписується набір логічних умов, при якому здійснюється перехід автомата з одного стану в інший, а також набір керуючих сигналів, що відповідають даному переходові.

Кількість тригерів, необхідних для організації пам'яті автомата (див. рис. 5.1) визначається із співвідношення $m > \lceil \log_2 M \rceil$, де M – число станів автомата. Кожному станові a_i повинна відповідати одна визначена комбінація значень Q_1, \dots, Q_m . Для розглянутого приклада вибираємо коди

станів відповідно до табл. 5.1. Для організації пам'яті будемо використовувати JK-тригери.

Структурна таблиця автомата складається по його графу. Кожен рядок (табл. 5.2) відповідає визначеному переходові автомата з одного стану в інший. В ній записують вихідний стан, стан переходу, коди цих станів, значення логічних умов, що забезпечують перехід, необхідні значення керуючих сигналів і функцій збудження тригерів. Значення функції збудження визначаються відповідно до таблиці переходів тригера відповідного типу. В кожному рядку для i -го тригера розглядаються переходи $Q_i^S \rightarrow Q_i^{S+1}$ з попереднього (ПС) в наступний (НС) стан. Довільні значення (0 або 1) сигналів позначаються в таблиці знаком *. Збільшення невизначених станів, як правило, спрощує схему автомата.

Таблиця 5.2

Структурна таблиця автомата

ПС	Код ПС		НС	Код НС		Логічні умови		Керуючі сигнали				Функції збудження тригерів			
	Q_1^S	Q_2^S		Q_1^{S+1}	Q_2^{S+1}	x_1	x_2	y_1	y_7	y_5	y_9	J_1	K_1	J_2	K_2
a_1	0	0	a_1	0	0	0	*	0	0	0	0	0	*	0	*
a_1	0	0	a_2	0	1	1	*	1	1	0	0	0	*	1	*
a_2	0	1	a_3	1	1	*	*	0	0	0	1	1	*	*	0
a_3	1	1	a_4	1	0	*	*	0	0	1	0	*	0	*	1
a_4	1	0	a_3	1	0	1	1	0□	0	0	0	*	0	0	*
a_4	1	0	a_2	0	1	*	0	0	0	0	0	*	1	1	*
a_4	1	0	a_1	0	0	0	1	0	0	0	0	*	1	0	*

На підставі структурної таблиці автомата визначаємо МДНФ функцій збудження і керуючих сигналів. Аргументами функцій J_i , K_i та y_i є значення Q_1^S , Q_2^S , x_1 і x_2 .

Використовуючи діаграми Вейча (рис. 5.5), одержуємо:

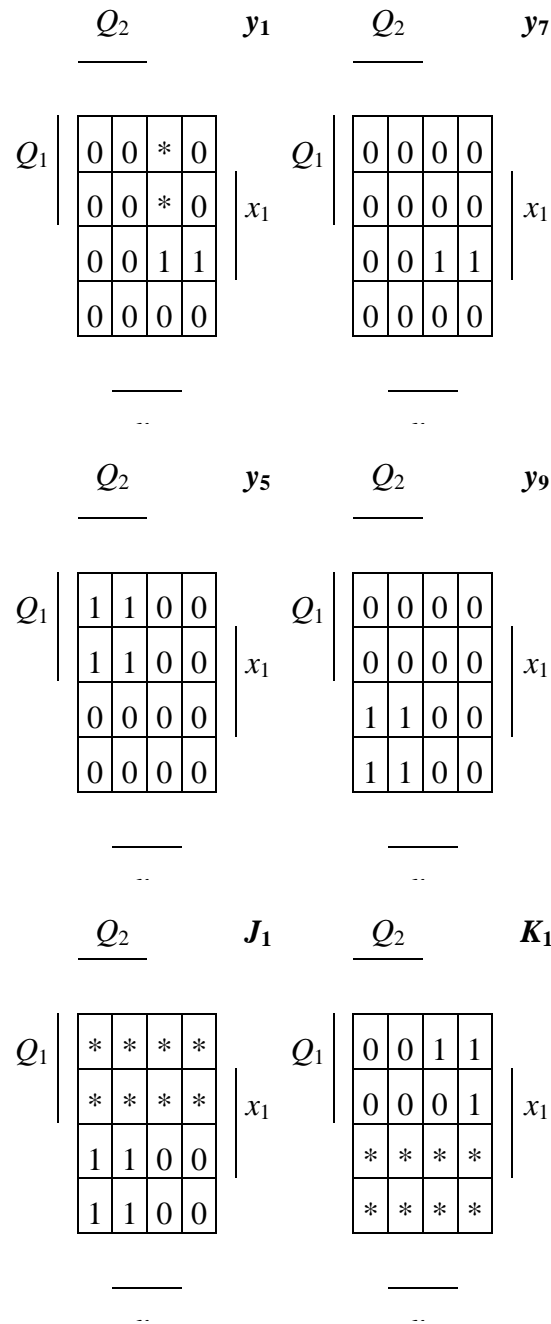
$$y_1 = y_7 = \overline{Q_1} \overline{Q_2} x_1 ; y_5 = Q_1 Q_2 ; y_9 = \overline{Q_1} Q_2 ;$$

$$K_2 = Q_1 ; J_1 = Q_2 ;$$

$$K_1 = \overline{Q_2} x_2 \vee \overline{Q_2} x_1 = \overline{Q_2} (x_2 \vee x_1) ;$$

$$J_2 = \overline{Q_1} x_1 \vee Q_1 x_2 .$$

Функціональна схема автомата зображена на рис. 5.6, де УПС – устанавлення початкового стану, ТС – тактуючі сигнали.



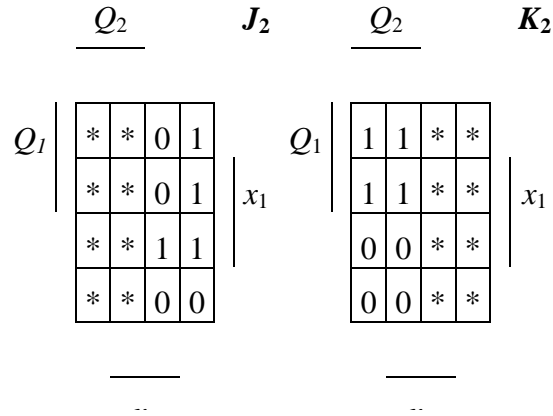


Рис. 5.5. Діаграми Вейча керуючих сигналів і функцій збудження тригерів

Відмінність синтезу автомата Мура полягає в наступному. Розмітка станів автомата в цьому випадку здійснюється іншим чином. Символом a_1 відзначаються початкова і кінцева вершини; всі операторні вершини відзначаються різними символами a_j .

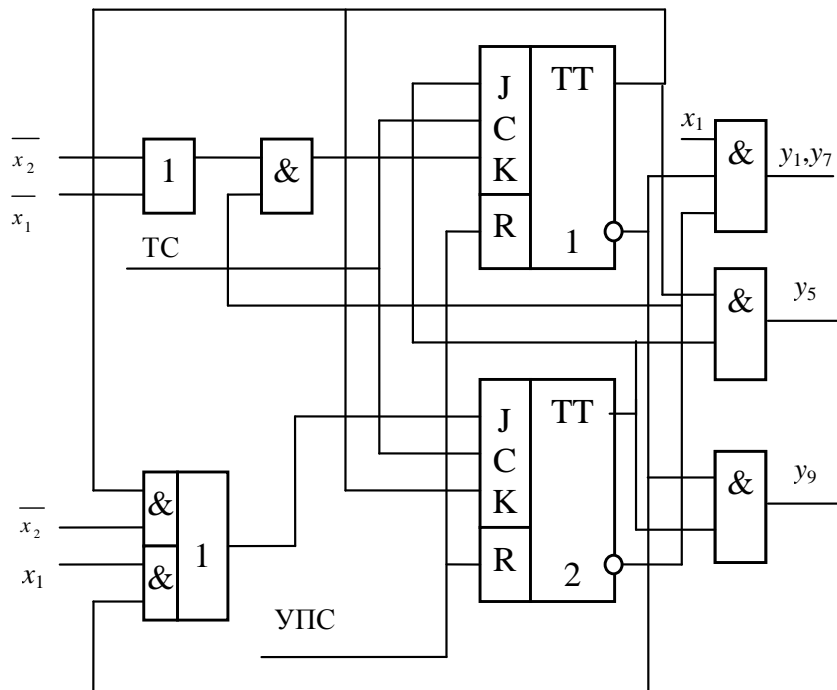


Рис. 5.6. Функціональна схема автомата

В даному випадку, як і при оцінці станів автомата Мілі, може виникнути необхідність введення додаткових станів.

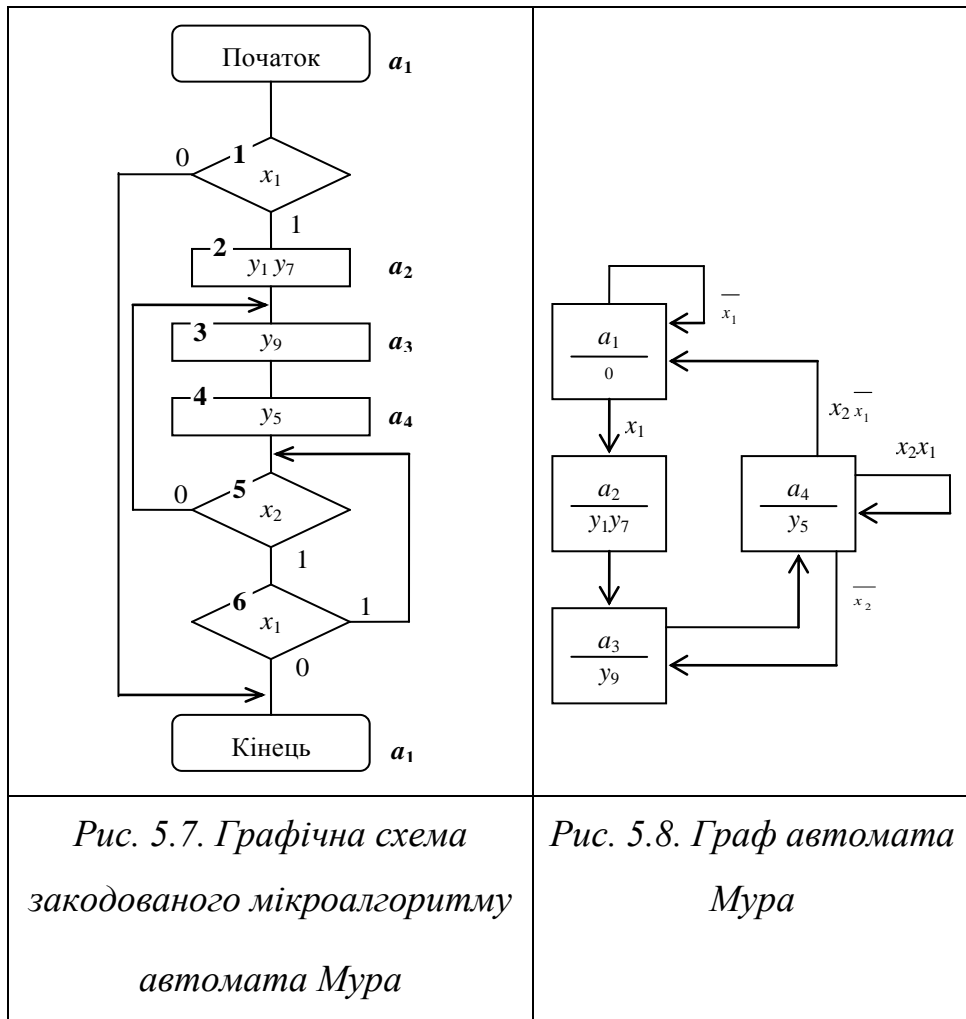
На графі автомата Мура дугам приписують набори логічних умов, що забезпечують відповідний перехід автомата. Керуючі сигнали записують у вершинах графа, тому що вони не залежать від логічних умов.

На рис. 5.7 і 5.8 показані закодований відмічений мікроалгоритм і граф автомата Мура.

Кодування станів автомата Мура можна виконувати так само, як і для автомата Мілі. Однак при відповідному кодуванні керуючі сигнали можна знімати і безпосередньо з виходів тригерів автомата Мура (комбінаційна схема для формування функцій y_j не потрібна). При цьому число тригерів повинне бути не менше числа керуючих сигналів, які не повторюють один одного і не є константами.

В цьому випадку відповідний рівень керуючих сигналів після спрацьовання тригерів устанавлюється швидше, ніж в автомата Мілі. Крім того при будь-якому переході виключається можливість короткочасного формування помилкових керуючих сигналів.

Приклад такого кодування станів автомата, що відповідає графові на рис. 5.8, приведений в табл. 5.3.



Таблиця 5.3

Таблиця кодування станів автомата Мура

Стан	Код станів		
	Q_1	Q_2	Q_3
a_1	0	0	0
a_2	1	0	0
a_3	0	0	1
a_4	0	1	0

В даному випадку автомат має три тригери, причому $y_1=Q_1$, $y_7=Q_1$, $y_5=Q_2$, $y_9=Q_3$.

Співпадаючі сигнали y_1 і y_7 знімаються з одного тригера.

Підготовка до роботи

1. Для визначення варіанту завдання перевести в двійкову систему числення десятковий номер залікової книжки студента і виділити дев'ять молодших розрядів одержаного числа $h_9, h_8, h_7, \dots, h_1$.

Згідно з табл. 5.4 і рис. 5.9 побудувати структурний мікроалгоритм за своїм варіантом. Для цього необхідно з'єднати послідовно зверху вниз фрагменти блок-схеми алгоритму (рис. 5.9) в порядку, зазначеному в таблиці варіантів (табл. 5.4). У відповідності з таблицею варіантів в кожному логічну вершину отриманої графічної схеми мікроалгоритму, починаючи з верхньої, переписати в зазначеному порядку по одному вхідному структурному сигналу. Потім в зазначеному порядку зверху вниз і зліва направо записати в операторні вершини вихідні структурні керуючі сигнали.

2. Виконати синтез і побудувати функціональну схему керуючого автомата в заданому елементному базисі. Тип тригерів, набір логічних елементів, які можна використовувати для побудови автомата, а також тип автомата зазначені в табл. 5.4.

3. Побудувати часову діаграму роботи автомата в умовних одиницях часу для кожної комбінації значень вхідних сигналів. Затримка сигналів логічними елементами складає 1, а тригерами – 6 умовних одиниць часу t . Визначити тривалість такту t .

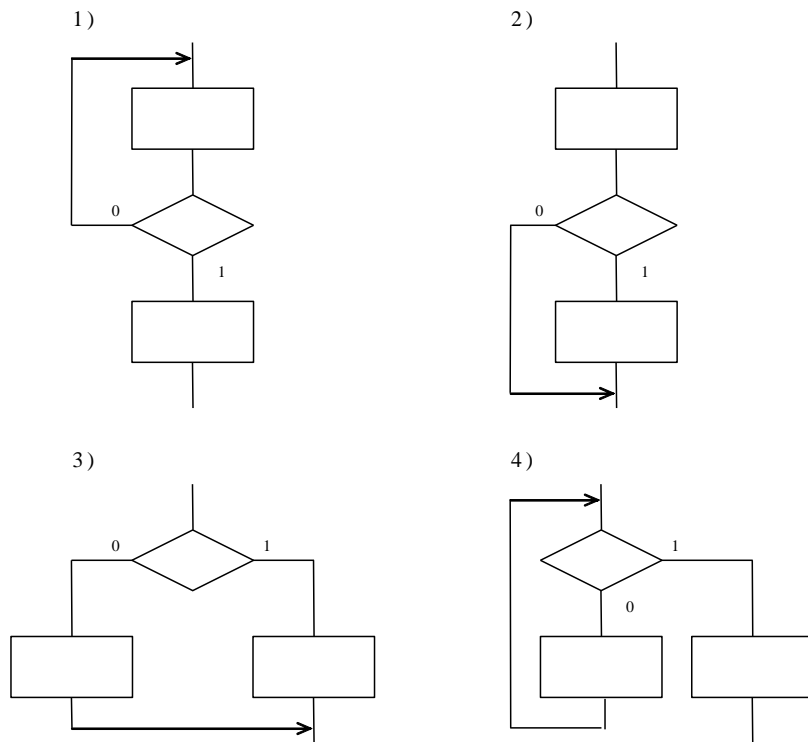


Рис.5.9. Фрагменти графічної схеми мікроалгоритму

Табл 5.4
Таблиця варіантів

<i>h8 h4 h2</i>	<i>Порядок з'єднання фрагментів</i>
<i>0 0 0</i>	<i>1, 2, 3</i>
<i>0 0 1</i>	<i>1, 4, 2</i>
<i>0 1 0</i>	<i>2, 3, 1</i>
<i>0 1 1</i>	<i>2, 1, 3</i>
<i>1 0 0</i>	<i>3, 1, 4</i>
<i>1 0 1</i>	<i>3, 4, 2</i>
<i>1 1 0</i>	<i>4, 2, 1</i>
<i>1 1 1</i>	<i>4, 3, 2</i>

<i>h8 h7 h3</i>	<i>Послідовність логічних умов</i>
<i>0 0 0</i>	<i>x_1, x_2, x_1</i>
<i>0 0 1</i>	<i>x_1, x_2, \bar{x}_1</i>
<i>0 1 0</i>	<i>$x_1, \bar{x}_2, x_2,$</i>
<i>0 1 1</i>	<i>$x_1, \bar{x}_2, \bar{x}_2,$</i>
<i>1 0 0</i>	<i>\bar{x}_1, x_2, x_2</i>
<i>1 0 1</i>	<i>$\bar{x}_1, x_2, \bar{x}_2$</i>
<i>1 1 0</i>	<i>$\bar{x}_1, \bar{x}_2, x_2$</i>
<i>1 1 1</i>	<i>$\bar{x}_1, \bar{x}_2, \bar{x}_1$</i>
<i>h9 h4 h1</i>	<i>Послідовність вихідних сигналів</i>
<i>0 0 0</i>	<i>$y_1, y_2, y_3, y_4, y_1, y_2$</i>
<i>0 0 1</i>	<i>$y_1, y_2, y_3, y_4, y_2, y_1$</i>
<i>0 1 0</i>	<i>$y_1, y_2, y_4, y_3, y_1, y_2$</i>
<i>0 1 1</i>	<i>$y_1, y_2, y_4, y_3, y_2, y_1$</i>
<i>1 0 0</i>	<i>$y_1, y_3, y_2, y_4, y_1, y_2$</i>
<i>1 0 1</i>	<i>$y_1, y_3, y_2, y_4, y_2, y_1$</i>
<i>1 1 0</i>	<i>$y_2, y_2, y_3, y_4, y_1, y_2$</i>
<i>1 1 1</i>	<i>$y_3, y_2, y_3, y_4, y_1, y_2$</i>
<i>h9 h4</i>	<i>Тип тригерів</i>
<i>0 0</i>	<i>RS</i>
<i>0 1</i>	<i>D</i>
<i>1 0</i>	<i>JK</i>
<i>1 1</i>	<i>T</i>
<i>h1</i>	<i>Тип автомата</i>
<i>0</i>	<i>Мілі</i>
<i>1</i>	<i>Мура</i>

Порядок виконання роботи

1. Побудувати та налагодити схему автомата, за допомогою програмного комплексу ПРОГМОЛС 2.0 (AFDK).
2. Дослідити часові параметри схеми.

Контрольні питання

1. Подати узагальнену структурну схему керуючого автомата.
2. Написати вирази, що визначають закон функціонування автоматів Мілі і Мура.
3. В чому відмінність автоматів Мілі і Мура?
4. Як побудувати граф автомата?
5. Як здійснюється розмітка станів автомата?
6. Від чого залежить кількість тригерів, необхідна для побудови автомата?
7. В чому сутність “протигоночного” кодування станів автомата?
8. Як скласти структурну таблицю автомата?
9. Як побудувати часову діаграму роботи автомата?
10. Скласти таблицю переходів для JK-, RS-, T- і D-тригерів.
11. Коли можливе виникнення помилкових керуючих сигналів (не передбачених графом автомата) і чим визначається їх тривалість?
12. Як визначити час переходу автомата з одного стану в інший?

Зміст звіту

Звіт повинний включати короткі теоретичні відомості, необхідні для виконання лабораторної роботи, всі схеми, формули, таблиці і графіки, отримані при виконанні теоретичного завдання і в процесі моделювання схем, а також висновки по роботі.

Література

1. Жабін В.І. Прикладана теорія цифрових автоматів: Навчальний посібник / В.І.Жабін, І.А.Жуков, І.А.Клименко, В.В.Ткаченко. – К.: Книжкове видавництво НАУ, 2009. – 364 с.

2. Самофалов К.Г. Прикладная теория цифровых автоматов / К.Г.Самофалов, А.М.Романкевич, В.Н.Валуйский и др. – К.: Вища школа, 1987. – 375 с.

3. Поспелов Д.А. Логические методы анализа и синтеза схем / Д.А. Поспелов. – М.: Энергия, 1974. – 367 с.

4. Зиссос Д. Проектирование систем на микропроцессорах/ Пер. с англ. под ред. А.И.Петренко. – К.: Техніка, 1982. – 176 с.1. Савельев А.Я. Арифметические и логические основы цифровых автоматов: Учебник. – М.: Высшая школа, 1980. – 255 с.

5. Майоров С.А. Принципы организации цифровых машин / С.А.Майоров, Г.И.Новиков. - Л.: Машиностроение, 1977. – 432 с.

6. Проектирование цифровых вычислительных машин / Под ред. С.А. Майорова. Учебное пособие для студентов вузов. – М.: Высшая школа, 1972. – 344 с.

6. ЛАБОРАТОРНА РОБОТА №6
СИНТЕЗ АВТОМАТІВ З ВИКОРИСТАННЯМ АПАРАТА
ЧАСОВИХ ФУНКЦІЙ

Ціль роботи

Вивчити метод структурного синтезу синхронних керуючих автоматів з використанням апарата часових функцій, одержати навички в їх налагодженні та експериментальному дослідженні.

Теоретичні відомості

Крім методу синтезу автомата у вигляді декомпозиції тригерів (лабораторна робота №5), існує метод прямого синтезу послідовнісних логічних схем з використанням часових функцій. Застосування такого методу дозволяє побудувати автомат у будь-якому функціонально повному базисі логічних елементів.

Часові перемикальні функції, на відміну від звичайних перемикальних функцій, в якості аргументу можуть використовувати власні значення, що обумовлено наявністю петлі з виходу елемента на його вхід.

Узагальнена структура автомата показана на рис. 6.1.

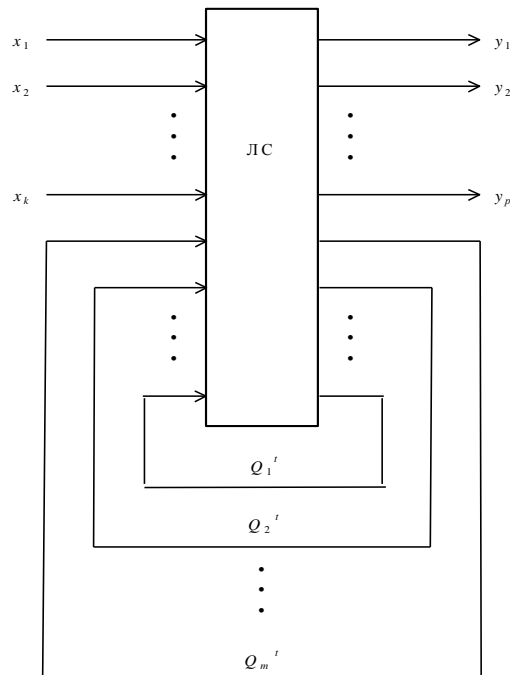


Рис. 6.1. Структура автомата

Автомат містить логічну схему (ЛС), що складається з логічних елементів. Входами ЛС є зовнішні входні сигнали (логічні умови) x_1, \dots, x_k , а також деякі виходи цієї ЛС, сигнали на яких розглядаються як часові функції Q_1^t, \dots, Q_m^t (верхній індекс визначає момент автоматного часу). ЛС виробляє зовнішні керуючі сигнали y_1, \dots, y_p для операційного пристрою (на рис. 6.1 операційний пристрій не показаний).

Стани автомата кодуються значеннями Q_i^t . У зв'язку з цим справедливо співвідношення $m \geq \lceil \log_2 s \rceil$. У загальному виді значення i -ї часової функції можна записати як

$$Q_i^{t+1} = f(x_1, \dots, x_k, Q_1^t, \dots, Q_m^t).$$

У процесі функціонування автомат переходить з одного стану в інший під дією зовнішніх керуючих сигналів. Кожному з станів $\{a_1, \dots, a_s\}$ відповідає визначений набір значень часових функцій Q_i^t .

Метод, що розглядається, може бути використаний для синтезу автоматів Мілі і Мура. Автомати можуть бути як синхронними, так і асинхронними. В даній лабораторній роботі розглядається синтез синхронних автоматів.

Керуючий автомат забезпечує формування керуючих сигналів для певного операційного пристрою.

Синтез автомата включає наступні етапи:

- 1) складання для операційного пристрою списку керуючих сигналів, що забезпечують виконання кожної мікрооперації;
- 2) визначення тривалості керуючих сигналів (в числі тактів) і періоду тактуючих сигналів автомата;
- 3) одержання закодованого мікроалгоритму;
- 4) розмітка станів автомата;
- 5) побудова графа автомата;
- 6) протигоночне кодування станів автомата значеннями Q_i^t ;
- 7) знаходження логічних формул для часових функцій;
- 8) знаходження МДНФ керуючих сигналів;
- 9) представлення керуючих сигналів і часових функцій в операторній формі з урахуванням заданого елементного базису;
- 10) побудова і оптимізація схеми керуючого автомата.

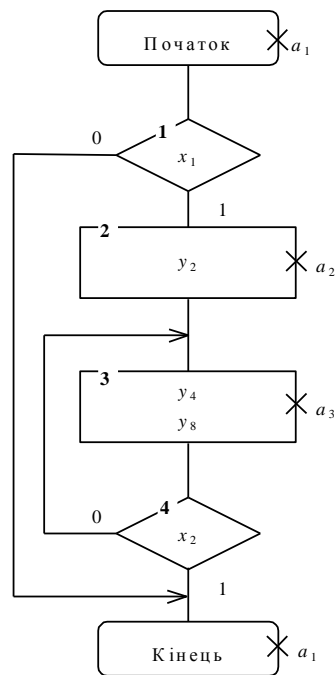


Рис. 6.2. Графічна схема закодованого мікроалгоритма

Будемо вважати, що після виконання пунктів 1, 2 і 3 одержано закодований структурний мікроалгоритм (рис. 6.2), де y_i - структурні керуючі сигнали, x_i - логичні умови (вхідні сигнали). Виконаємо синтез автомата Мура при синхронному режимі виконання мікрооперацій (всі сигнали мають однакову тривалість).

Розмітка станів циклічного автомата Мура виконується наступним чином. Символом a_i позначаються вершини „Початок” та „Кінець”. Всі операторні вершини позначаються різними символами a_i .

Після розмітки станів закодований мікроалгоритм автомата Мура приймає вигляд, що показано на рис. 6.2.

На підставі мікроалгоритму будемо граф автомата (рис. 6.3).

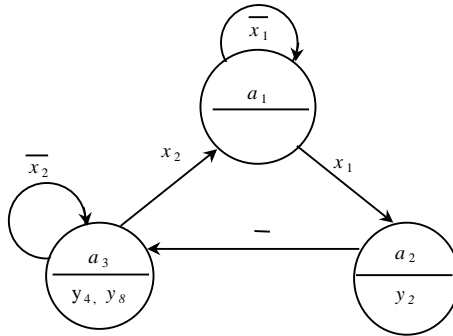


Рис. 6.3. Граф автомата

Для даного методу синтезу автоматів необхідною умовою є протигоничне (сусіднє) кодування станів автомата, з'єднаних дугами. Зв'язані дугами вершини повинні мати коди, що відрізняються значеннями тільки одного розряду. Граф на рис. 6.3 вимагає доробки, оскільки три зв'язані між собою вершини не можна забезпечити сусідніми кодами.

У загальному випадку для кодування станів автомата можна використовувати шаблони, показані на рис. 6.4. Граф накладається на шаблон з урахуванням дуг, що з'єднують вершини. Якщо при накладенні графа на шаблон немає між вершинами відповідних зв'язків, то вводяться додаткові вершини. Це дозволяє перемістити розглянуту вершину на позицію, що забезпечує необхідні зв'язки даної вершини з іншими вершинами. Необхідність введення додаткових вершин може бути також обумовлена необхідністю забезпечення перепадів керуючих сигналів.

Скориставшись шаблоном для дворозрядних кодів станів, одержимо граф на рис. 6.4, закодований сусідніми кодами. Введена додаткова вершина β .

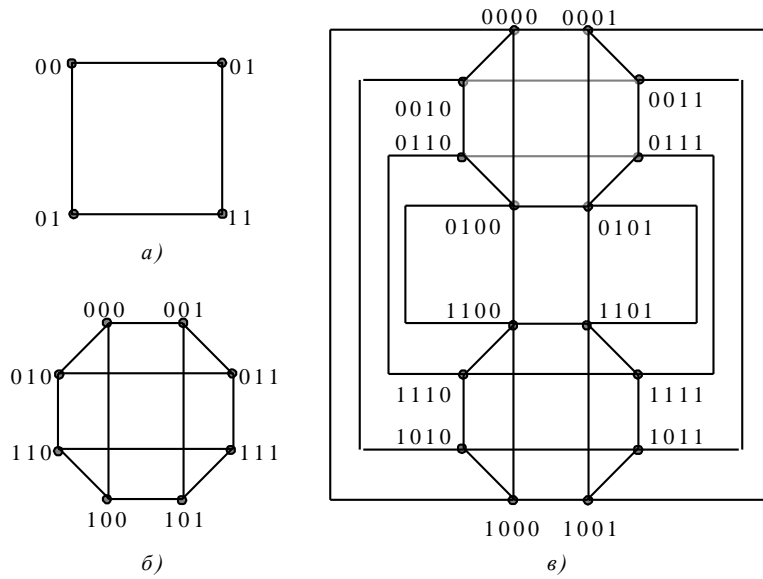


Рис. 6.4. Шаблони для кодування станів автомата: а – для 2-розрядних, б – для 3- розрядних и в – для 4- розрядних кодів станів

Коди станів зведені в табл. 6.1. Дуги, що замикаються на власні вершини, на графі для автомата Мура не відображаються, тому що значення часових перемінних визначаються тільки дугами між різними вершинами графа, а вихідні сигнали приписані вершинам графа.

Табл. 6.1

Таблиця кодування станів

Стан	Код стану
	$Q_1^t Q_2^t$
a_1	0 0
a_2	0 1
a_3	1 0
β	1 1

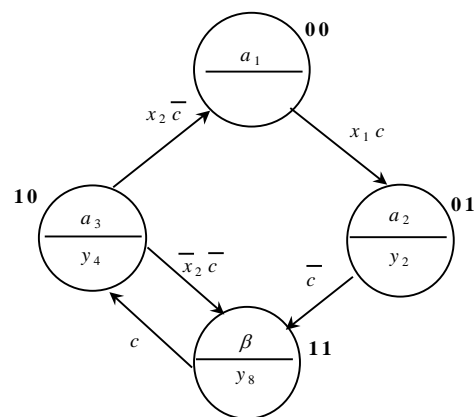


Рис. 6.5. Граф автомата із сусіднім кодуванням станів

Для синхронного режиму роботи кожній дузі графа приписується синхросигнал c або його заперечення. Один такт відповідає напівперіодові синхросигналу. Для кожної вершини вхідні і вихідні дуги повинні бути відзначені різними значеннями синхросигналів. Наприклад, вхідні дуги відзначені c , а вихідні - \bar{c} (або навпаки).

Для знаходження часових перемінних використовують формулу

$$Q_i^{t+1} = F_i \vee Q_i^t \bar{G}_i \quad (6.1)$$

або

$$Q_i^{t+1} = (F_i \vee Q_i^t) \bar{G}_i, \quad (6.2)$$

де F_i – функція встановлення часової перемінної в одиничний стан; G_i – функція скидання часової перемінної в нульовий стан.

При побудові схем у булевому базисі можна використовувати кожен з приведених формул. Для елементного базису Шеффера використовується формула (6.1), а для базису Пірса – формула (6.2). У цьому випадку відповідне операторне представлення часової перемінної можна одержати застосуванням правила де Моргана.

Для одержання функцій F_i розглядають всі дуги графа і записують через логічну операцію АБО всі умови переключення Q_i^t з 0 в 1. Наприклад, для перемінної Q_1^t на графі рис. 6.5 маються два переходи, на яких перемінна переключається з 0 в 1 ($a_1 \rightarrow a_2$ та $a_3 \rightarrow \beta$). Для функцій G_i аналогічно записують умови переключення перемінної з 1 в 0. Мінімізація функцій F_i і G_i виконується незалежно одна від одної. Якщо для кодування станів автомата використовуються не всі передбачені шаблоном коди, то невикористовуванні коди можна розглядати як набори, на яких часові функції не визначені.

Для розглянутого приклада безпосередньо з графа на рис. 6.5 одержуємо систему функцій

$$\begin{cases} F_i = a_1 x_1 c \vee a_3 \overline{x_2} c = \overline{Q_2} \overline{Q_1} x_1 c \vee Q_2 \overline{Q_1} \overline{x_2} c; \\ G_1 = \beta c = Q_2 Q_1 c; \\ F_2 = a_2 c = \overline{Q_2} Q_1 c; \\ G_2 = a_3 x_2 c = Q_2 \overline{Q_1} x_2 c. \end{cases} \quad (6.3)$$

Нехай необхідно реалізувати часові перемінні на елементах І-НІ, а вихідні сигнали на елементах І. Відповідно до (6.1) і системи (6.3), застосовуючи правило де Моргана, одержимо операторну форму часових перемінних

$$Q_1^{t+1} = \overline{\overline{Q_2 Q_1 x_1 c} \cdot \overline{Q_2 Q_1 x_2 c} \cdot R \cdot Q_1^t R \cdot Q_2 Q_1 c}, \quad (6.4)$$

$$Q_2^{t+1} = \overline{\overline{Q_2 Q_1 c} R \cdot Q_2^t R \cdot Q_2 Q_1 x_2 c}. \quad (6.5)$$

Тут R – сигнал встановлення часових перемінних у нульовий стан.

Вихідні сигнали в автоматі Мура цілком визначаються кодом стану. Для графа на рис. 6.3 одержимо $y_2 = \overline{Q_2} Q_1$, $y_4 = Q_2 \overline{Q_1}$, $y_8 = Q_2 Q_1$.

Схема, що отримана за операторними формами (6.4) і (6.5), показана в Додатку А у складі пристрою перетворення даних. У такому вигляді схема відображається на екрані дисплея при використанні моделюючої програми ПРОГМОЛС 2.0. Усі сигнали позначені заголовними буквами (наприклад, $x_2 = X2$). Заперечення сигналів починаються з букви N (наприклад, $\overline{Q_1} = NQ1$).

При синтезі автомата Мілі часові функції знаходяться аналогічно. Відмінність від автомата Мура полягає в способі розмітки станів, вигляді

графа автомата і врахуванні вхідних сигналів в якості аргументів функцій виходів. Ці відмінності розглянуті в лабораторній роботі № 5.

Підготовка до роботи

1. Для визначення варіанту завдання перевести в двійкову систему числення десятковий номер залікової книжки студента і виділити дев'ять молодших розрядів одержаного числа $h_9, h_8, h_7, \dots, h_1$.

Згідно з рис. 5.9 і табл. 6.2 побудувати структурний мікроалгоритм за своїм варіантом. Для цього необхідно з'єднати послідовно зверху вниз фрагменти блок-схеми алгоритму (рис. 5.9) в порядку, зазначеному в таблиці варіантів (табл. 6.2). У відповідності з таблицею варіантів в кожну логічну вершину отриманої графічної схеми мікроалгоритму, починаючи з верхньої, переписати в зазначеному порядку по одному вхідному структурному сигналу. Потім в зазначеному порядку зверху вниз і зліва направо записати в операторні вершини вихідні структурні керуючі сигнали.

2. Виконати синтез і побудувати функціональну схему керуючого автомата в заданому елементному базисі. Набір логічних елементів, які можна використовувати для побудови автомата, а також тип автомата зазначені в табл. 6.2.

3. Побудувати часову діаграму роботи автомата в умовних одиницях часу для кожної комбінації значень вхідних сигналів. Затримка сигналів логічними елементами складає 1, а тригерами – 6 умовних одиниць часу τ . Визначити тривалість такту t .

Порядок виконання роботи

1. Побудувати та налагодити схему автомата, за допомогою програмного комплексу ПРОГМОЛС 2.0 (AFDK).
2. Дослідити часові параметри комбінаційної схеми.

Табл. 6.2
Таблиця варіантів

h_8 h_4 h_2	Порядок з'єднання фрагментів
0 0 0	1, 2
0 0 1	1, 4
0 1 0	2, 3
0 1 1	2, 1
1 0 0	3, 1
1 0 1	3, 4
1 1 0	4, 2
1 1 1	4, 3
h_7 h_3	Послідовність логічних умов
0 0	$\overline{x_2}, x_1$
0 1	x_2, x_1
1 0	$x_1, \overline{x_2}$
1 1	x_1, x_2
h_9 h_4 h_1	Послідовність вихідних сигналів
0 0 0	y_1, y_2, y_3, y_4

0 0 1	y_1, y_2, y_4, y_3
0 1 0	y_1, y_3, y_2, y_4
0 1 1	y_1, y_4, y_3, y_2
1 0 0	y_2, y_1, y_4, y_3
1 0 1	y_2, y_1, y_3, y_4
1 1 0	y_2, y_4, y_1, y_3
1 1 1	y_2, y_3, y_4, y_1
h_1	Тип автомата
1	Мілі
0	Мура
$h_3 h_2$	Логічні елементи
0 0	I, АБО, НЕ
0 1	I, I-НЕ
1 0	I-НЕ
1 1	АБО-НЕ

Контрольні питання

1. Подати узагальнену структурну схему керуючого автомата.
2. Написати вирази, що визначають закон функціонування автоматів Мілі і Мура.
3. В чому відмінність автоматів Мілі і Мура?
4. Охарактеризувати основні етапи проектування автомата.
5. Як побудувати граф автомата?
6. Як здійснюється розмітка станів автомата?
7. Від чого залежить кількість тригерів, необхідна для побудови автомата?
8. В чому сутність “протигоночного” кодування станів автомата?

9. Як скласти структурну таблицю автомата?
10. Як побудувати часову діаграму роботи автомата?
11. Скласти таблицю переходів для JK-, RS-, T- і D-тригерів.

Зміст звіту

Звіт повинний включати короткі теоретичні відомості, необхідні для виконання лабораторної роботи, усі схеми, формули, таблиці і графіки, отримані при виконанні теоретичного завдання та у процесі моделювання схем, а також висновки по роботі.

Література

1. Жабін В.І. Прикладна теорія цифрових автоматів: Навчальний посібник / В.І.Жабін, І.А.Жуков, І.А.Клименко, В.В.Ткаченко. – К.: Книжкове видавництво НАУ, 2009. – 364 с.
2. Самофалов К.Г. Прикладная теория цифровых автоматов / К.Г.Самофалов, А.М.Романкевич, В.Н.Валуийский и др. – К.: Вища школа, 1987. – 375 с.
3. Поспелов Д.А. Логические методы анализа и синтеза схем / Д.А. Поспелов. – М.: Энергия, 1974. – 367 с.
4. Зиссос Д. Проектирование систем на микропроцессорах/ Пер. с англ. под ред. А.И.Петренко. – К.: Техніка, 1982. – 176 с.1. Савельев А.Я. Арифметические и логические основы цифровых автоматов: Учебник. – М.: Высшая школа, 1980. – 255 с.
5. Майоров С.А. Принципы организации цифровых машин / С.А.Майоров., Г.И.Новиков. - Л.: Машиностроение, 1977. – 432 с.

6. Проектирование цифровых вычислительных машин / Под ред. С.А. Майорова. Учебное пособие для студентов вузов. – М.: Высшая школа, 1972. – 344 с.

7. КУРСОВА РОБОТА

7.1. Завдання на курсову роботу

Завдання на курсову роботу включає наступне.

1. Виконати синтез і побудувати функціональну схему керуючого автомата по заданому алгоритму.

2. Виконати перетворення форм заданих перемикальних функцій і побудувати комбінаційні схеми на базі програмувальних логічних схем.

Варіант завдання визначається дев'ятьма молодшими розрядами номера залікової книжки студента, представленого в двійковій системі числення ($h_9, h_8, h_7, \dots, h_1$).

Для одержання вихідного структурного алгоритму керування необхідно з'єднати послідовно зверху вниз фрагменти графічної схеми алгоритму (рис. 7.1) в порядку, зазначеному в табл. 7.1. У кожному логічному вершині отриманої блок-схеми, починаючи з верхньої, переписати з табл. 7.2 в зазначеному порядку по одній логічній умові. Потім відповідно до табл. 7.3 в порядку зверху вниз і зліва направо записати в операторні вершини керуючі сигнали. Сигнали, зазначені в дужках, записуються в одну вершину. Отримана графічна схема алгоритму корегується з урахуванням подвоєної тривалості сигналу, зазначеного в табл. 7.3 (інші сигнали мають тривалість t).

Тип тригерів і набір логічних елементів, які можна використовувати для побудови автомата, зазначені в табл. 7.4, а тип автомата визначений в табл. 7.5.

Система з чотирьох перемикальних функцій задана табл. 7.6.

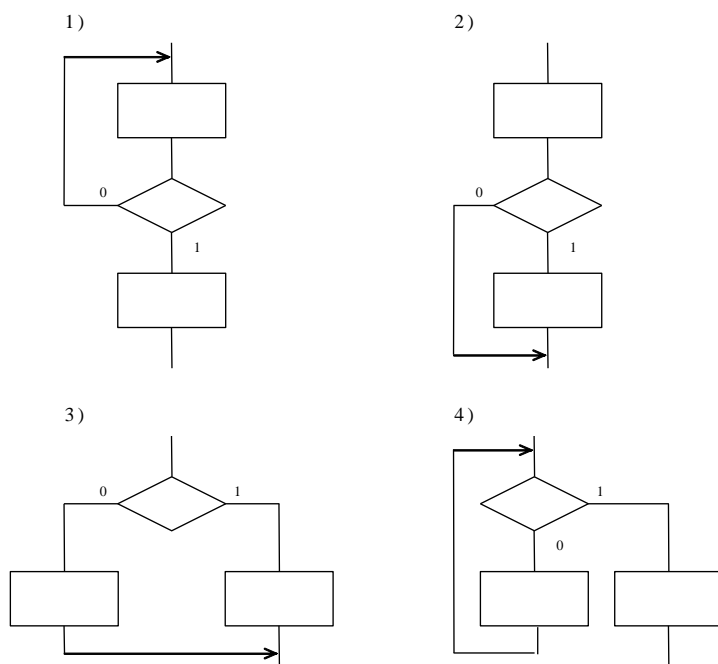


Рис. 7.1. Фрагменти графічної схеми алгоритму

Табл. 7.1

Варіанти структури алгоритму

h_8	h_4	h_2	Порядок з'єднання фрагментів
0	0	0	1, 2, 3
0	0	1	1, 4, 2
0	1	0	2, 3, 1
0	1	1	2, 1, 3
1	0	0	3, 1, 4
1	0	1	3, 4, 2
1	1	0	4, 2, 1
1	1	1	4, 3, 2

Табл. 7.2

Набори логічних умов

h_8	h_7	h_3	Логічні умови
0	0	0	x_1, x_2, x_1
0	0	1	$x_1, x_2, \overline{x_1}$
0	1	0	$\overline{x_1}, x_2, x_2$
0	1	1	$\overline{x_1}, x_2, \overline{x_2}$
1	0	0	$\overline{x_1}, x_2, x_2$
1	0	1	$\overline{x_1}, x_2, \overline{x_2}$
1	1	0	$\overline{x_1}, \overline{x_2}, x_1$
1	1	1	$\overline{x_1}, \overline{x_2}, \overline{x_1}$

Табл. 7.3

Варіанти комбінацій вихідних сигналів

h_9	h_4	h_1	Послідовність керуючих сигналів
0	0	0	$y_1, y_2, y_3, y_4, y_1, y_2$
0	0	1	$y_1, y_2, y_3, y_4, y_2, y_1$
0	1	0	$y_1, y_2, y_4, y_3, y_1, y_2$
0	1	1	$y_1, y_2, y_4, y_3, y_2, y_1$
1	0	0	$y_1, y_3, y_2, y_4, y_1, y_2$
1	0	1	$y_1, y_3, y_2, y_4, y_2, y_1$
1	1	0	$y_2, y_2, y_3, y_4, y_1, y_2$
1	1	1	$y_3, y_2, y_3, y_4, y_1, y_2$
h_6	h_2	Сигнал, тривалістю $2t$	
0	0	y_1	
0	1	y_2	
1	0	y_3	
1	1	y_4	

Табл. 7.4

Вибір типу автомата

h_1	Тип автомата
0	Мілі
1	Мура

Табл. 7.5

Варианти елементної бази

h_3	h_2	h_1	Логічні елементи
0	0	0	3І-НЕ, 2І
0	0	1	3І, 4І-НЕ
0	1	0	3АБО, 4І, НЕ
0	1	1	3І, 2АБО, НЕ
1	0	0	2АБО-НЕ, 4І
1	0	1	2І-НЕ, 4АБО
1	1	0	3АБО-НЕ, 3І
1	1	1	3І-НЕ, 3АБО-НЕ

h_3	h_2	h_1	Логічні елементи
0	0	0	3І-НЕ, 2І
0	0	1	3І, 4І-НЕ
0	1	0	3АБО, 4І, НЕ
0	1	1	3І, 2АБО, НЕ
1	0	0	2АБО-НЕ, 4І
1	0	1	2І-НЕ, 4АБО
1	1	0	3АБО-НЕ, 3І
1	1	1	3І-НЕ, 3АБО-НЕ

h_3	h_2	h_1	Логічні елементи
0	0	0	3І-НЕ, 2І
0	0	1	3І, 4І-НЕ
0	1	0	3АБО, 4І, НЕ
0	1	1	3І, 2АБО, НЕ
1	0	0	2АБО-НЕ, 4І
1	0	1	2І-НЕ, 4АБО
1	1	0	3АБО-НЕ, 3І
1	1	1	3І-НЕ, 3АБО-НЕ

h_3	h_2	h_1	Логічні елементи
0	0	0	3І-НЕ, 2І
0	0	1	3І, 4І-НЕ
0	1	0	3АБО, 4І, НЕ
0	1	1	3І, 2АБО, НЕ
1	0	0	2АБО-НЕ, 4І
1	0	1	2І-НЕ, 4АБО
1	1	0	3АБО-НЕ, 3І
1	1	1	3І-НЕ, 3АБО-НЕ

h_3	h_2	h_1	Логічні елементи
0	0	0	3І-НЕ, 2І
0	0	1	3І, 4І-НЕ
0	1	0	3АБО, 4І, НЕ
0	1	1	3І, 2АБО, НЕ
1	0	0	2АБО-НЕ, 4І
1	0	1	2І-НЕ, 4АБО
1	1	0	3АБО-НЕ, 3І
1	1	1	3І-НЕ, 3АБО-НЕ

Табл. 7.6

Таблиця істинності

x_4	x_3	x_2	x_1	f_1	f_2	f_3	f_4
0	0	0	0	1	1	1	0
0	0	0	1	1	1	0	1
0	0	1	0	1	1	1	h_3
0	0	1	1	0	0	0	h_4
0	1	0	0	-	0	1	0
0	1	0	1	0	0	0	h_5
0	1	1	0	1	-	-	0
0	1	1	1	-	-	1	h_6
1	0	0	0	1	h_4	h_7	h_7
1	0	0	1	0	0	h_8	1
1	0	1	0	0	0	h_9	h_8
1	0	1	1	h_1	0	0	h_2
1	1	0	0	1	-	1	1
1	1	0	1	h_2	h_5	0	h_9
1	1	1	0	h_3	h_6	0	h_1
1	1	1	1	1	1	1	1

1.2. Зміст курсової роботи

Пояснювальна записка повинна містити наступні розділи.

1. Вступ.
2. Синтез автомата.
3. Синтез комбінаційних схем.
4. Висновок.
5. Список літератури.

У вступі вказується, на підставі яких документів (вихідних даних) здійснюється розробка.

В розділі 2 необхідно представити закодовану графічну схему алгоритму (ГСА) в структурних термінах (маленькі x_i, y_i), виконати розмітку станів автомата, зробити структурний синтез автомата.

Виконати спільну мінімізацію функцій збудження тригерів і вихідних сигналів автомата. Одержати операторні представлення функцій у заданому елементному базисі. Функціональна схема автомата представляється на окремому листі (формат А1 або А2) за правилами виконання електричних схем Е2.

В третьому розділі здійснюється синтез комбінаційних схем.

Функцію f_4 необхідно представити в канонічних формах алгебри Буля, Жегалкіна, Пірса і Шеффера. Визначити приналежність даної функції до п'яти передповних класів. Виконати мінімізацію функції f_4 методами:

- невизначених коефіцієнтів;
- Квайна (Квайна – Мак-Класкі);
- діаграм Вейча.
- Виконати спільну мінімізацію функцій $f_1, f_2, i f_3$ аналітичним методом Квайна або Квайна – Мак-Класкі. Перевірити результат мінімізації діаграмами Вейча.

Одержати операторні представлення та побудувати функціональні схеми реалізації системи функцій в формі І/АБО та І/АБО-НЕ спочатку на логічних елементах з будь-якою кількістю входів, а потім на програмувальних логічних матрицях (ПЛМ). Повинні бути отримані мнемонічні схеми, карти програмування та визначені мінімальні параметри ПЛМ.

У висновку узагальнюються результати роботи.

1.3. Вимоги до оформлення курсової роботи

Курсова робота повинна містити наступні документи (в порядку їх комплектування):

- титульний лист (додаток 1);
- опис альбому;
- сторінка з написом у середині листа "Технічне завдання".
- технічне завдання;
- сторінка з написом у середині листа "Керуючий автомат. Схема електрична функціональна";
- керуючий автомат, схема електрична функціональна;
- сторінка з написом у середині листа "Пояснювальна записка";
- пояснювальна записка.

Усі скомплектовані документи приводяться до формату А4 і скріплюються в одну папку або альбом.

Технічне завдання розробляється студентом на підставі вихідних даних відповідно до діючих стандартів. У технічному завданні повинні бути наступні розділи:

- *призначення* розроблювального об'єкта, в якому розкриваються його області застосування;
- *вихідні дані* для розробки;
- *склад* пристроїв, в якому приводиться перелік основних складових частин проектного пристрою;
- *етапи* проектування і терміни їх виконання;
- *перелік* текстової і графічної документації.

Технічне завдання повинне бути підписане виконавцем і керівником проекту.

Порядок побудови розділів і підрозділів пояснювальної записки, правила подання тексту повинні цілком відповідати вимогам діючих стандартів. Текст роздруковується на принтері.

Перед комплектацією всі документи курсової роботи повинні бути підписані виконавцем і керівником на титульному листі і в основних написах документів і креслень.

Функціональна схема визначає основні функціональні частини виробу, їх призначення і взаємозв'язок, роз'ясняє визначені процеси, що протікають в окремих функціональних частинах або у виробі в цілому. Функціональні частини таких схем зображуються, як правило, у виді прямокутників.

У функціональних схемах використовуються умовні графічні позначення (УГП) функціональних частин. УГП елемента має форму прямокутника, до якого підводять лінії виводів. УГП елемента в загальному випадку може містити три поля: основне і два додаткових, котрі розташовують ліворуч і праворуч від основного (рис. 7.2).

В основному полі УГП поміщають позначення функції, яка реалізується елементом (табл. 7.7). В додаткових полях поміщають інформацію про призначення виводів (мітки виводів, вказівники).

Крім виду, зазначеного на рис 7.2, УГП може також складатися:

- тільки з основного поля;
- з основного поля і одного додаткового (праворуч або ліворуч від основного);

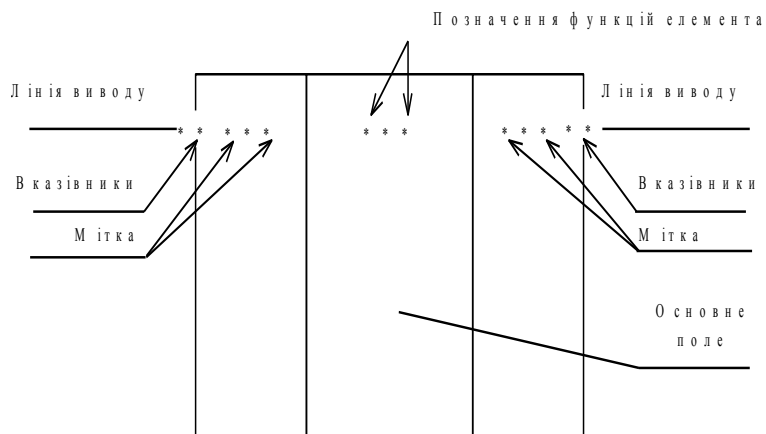


Рис. 7.2. Умовне графічне позначення елемента

Допускаються додаткові поля розділяти на зони, відокремлені горизонтальною рисою. Основне і додаткове поля можуть бути не відділені лінією. При цьому відстань між буквеними, цифровими і буквено-цифровими позначеннями, поміщеними в основне і додаткові поля, визначається однозначністю розуміння кожного позначення.

Входи елемента зображують з лівої сторони УГП, виходи – з правої сторони УГП. Двонаправлені виводи і виводи, що не несуть логічної інформації, зображують з правої або лівої сторони УГП.

При підведенні ліній виводів до контуру УГП не допускається:

- проводити їх на рівні сторін прямокутника;
- проставляти на них у стрілки, що вказують напрямок передачі інформації.


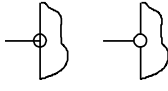
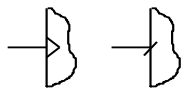
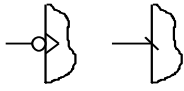
Таблиця 7.7

Позначення елементів

Найменування	Позначення
Тригер	T
Двоступінчастий тригер	TT
Логічне “І”	&
Логічне “АБО”	≥ 1 або 1
Виключне “АБО”	$= 1$
Генератор імпульсів	G

Таблиця 7.8

Позначення входів елементів

Найменування	Позначення
Прямий статичний вивід	
Інверсний статичний висновок	
Прямий динамічний вхід	
Інверсний динамічний вхід	

Таблиця 7.9
Приклади УГП

Найменування	Позначення
Елемент 3І-НЕ	
Елемент 2АБО	
D-тригер	

Розміри сторін УГП і відстані між выводами повинні бути кратні М. Діаметр вказівника інверсного виводу повинний дорівнювати М. При виконанні курсової роботи прийняти М=2,5 мм.

Позначення функцій, виконуваних елементом, утворюють із прописних букв латинського алфавіту, арабських цифр і спеціальних знаків, записаних без пробілів. Кількість знаків у позначенні функції не обмежена, однак варто прагнути до їх мінімального числа при збереженні однозначності розуміння кожного позначення.

В таблиці 7.7 приведені деякі стандартні позначення функцій, яких варто дотримуватись при виконанні курсової роботи.

Виводи елементів, які несуть логічну інформацію, підрозділяють на статичні і динамічні, а також на прям і інверсні. Властивості виводів позначають за допомогою міток відповідно до табл. 7.8.

Приклади деяких УГП приведені в табл. 7.9.

Захист курсових робіт є особливою формою перевірки якості виконання роботи і знань студента в даній області. Захист складається в

короткій доповіді (5-7 хв.) студента по виконаній роботі і відповідях на питання.

Критерії оцінювання курсової роботи (КР):

оформлення документів згідно стандартам: 0 – 20 балів,

розробка комбінаційних схем 0-30 балів;

розробка схем автоматів з пам'яттю 0-30 балів;

якість оформлення пояснювальної записки: 0-10 бали,

відповіді на теоретичні запитання викладача: 0-10 бали.

Якщо в конструкторській документації проекту будуть виявлені грубі порушення стандартів або виявиться, що спроектований виріб непрацездатний, робота повертається на доробку.

7.4. Література

1. Жабін В.І. Прикладна теорія цифрових автоматів: Навчальний посібник / В.І.Жабін, І.А.Жуков, І.А.Клименко, В.В.Ткаченко. – К.: Книжкове видавництво НАУ, 2009. – 364 с.

2. Зиссос Д. Проектирование систем на микропроцессорах / Д. Зиссос. Пер. с англ. под ред. А.И.Петренко. – К.: Техніка, 1982. – 176 с.1.

3. Майоров С.А. Принципы организации цифровых машин / С.А.Майоров., Г.И.Новиков. - Л.: Машиностроение, 1977. – 432 с.

4. Поспелов Д.А. Логические методы анализа и синтеза схем / Д.А. Поспелов. – М.: Энергия, 1974. – 367 с.3. Савельев А.Я. Арифметические и логические основы цифровых автоматов: Учебник. / А.Я.Савельев. – М.: Высшая школа, 1980. – 255 с

5. Проектирование цифровых вычислительных машин / Под ред. С.А. Майорова. Учебное пособие для студентов вузов. – М.: Высшая школа, 1972. – 344 с.

6. Савельев А.Я. Арифметические и логические основы цифровых автоматов: Учебник. – М.: Высшая школа, 1980. – 255 с.

7. Самофалов К.Г. Прикладная теория цифровых автоматов / К.Г.Самофалов, А.М.Романкевич, В.Н.Валуйский и др. – К.: Вища школа, 1987. – 375 с.

8. Складов В.А. Синтез автоматов на матричных БИС/ В.А.Складов. Под ред. С.И.Баранова. – Минск: Наука и техника, 1984. – 287 с.

8. Програмний комплекс для моделювання логічних схем

Програмний комплекс ПРОГМОЛС 2.0 (ПРОГРама МОделювання Логічних Схем) призначений для моделювання процесів у комбінаційних і послідовнісних схемах. Він дозволяє створювати і редагувати логічні схеми, здійснювати моделювання у синхронному (без урахування затримок сигналів в елементах схеми) і в асинхронному (з урахуванням затримок) режимах, а також зберігати отримані моделі.

Вигляд моделі проілюстровано на рис. 1.1.

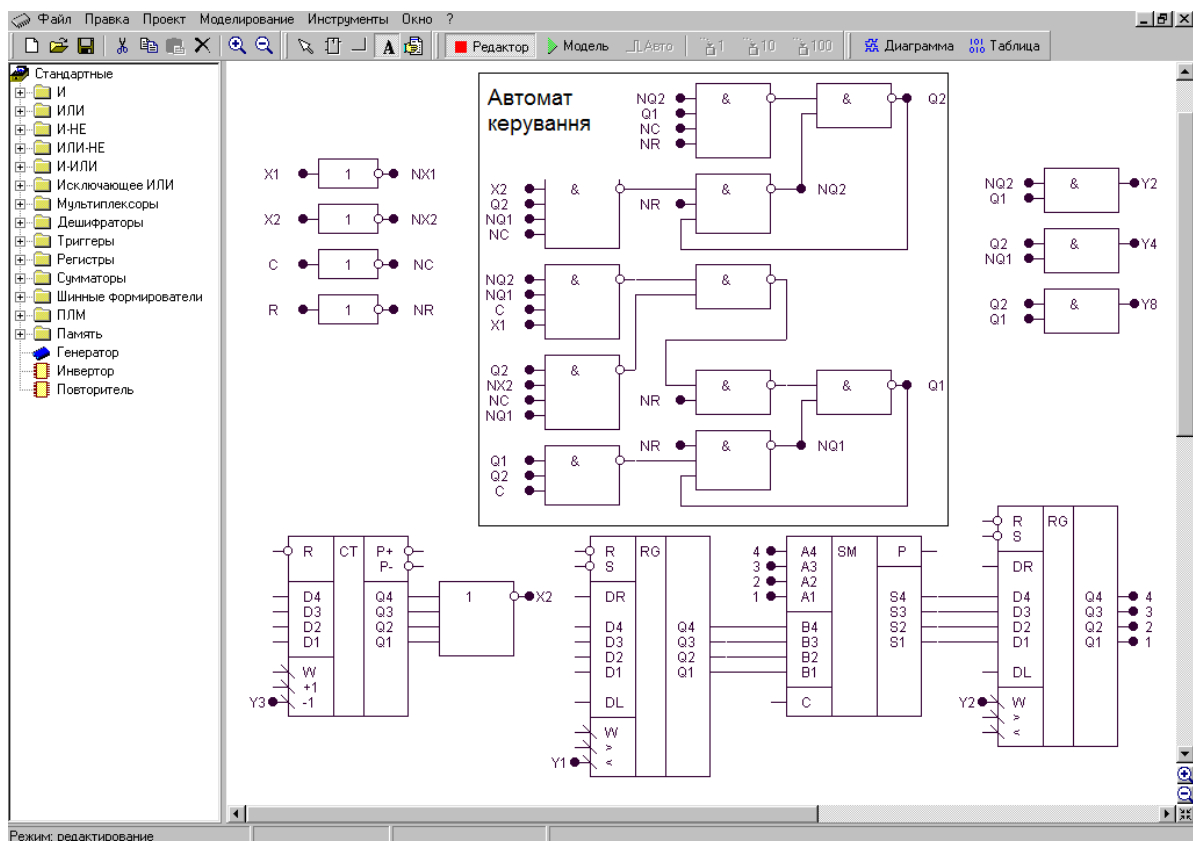


РИС. 1.1. ЗОБРАЖЕННЯ МОДЕЛІ ОПЕРАЦІЙНОГО ПРИСТРОЮ З АВТОМАТОМ КЕРУВАННЯ

Для роботи з програмою використовують систему ієрархічних меню, що містить наступні розділи:

- *файл;*
- *виправлення;*
- *проект;*
- *моделювання;*
- *інструменти;*
- *вікно;*
- *допомога.*

Комплекс включає систему підказок, що полегшує роботу в різних режимах моделювання. Побудова та елементи керування для кожного розділу пояснюються нижче.

Файл



Створити (сполучення клавіш **Ctrl+N**). Створює новий файл проекту.



Відкрити (**Ctrl+O**). Викликає діалогове вікно відкриття проекту.



Зберегти (**Ctrl+S**). Зберігає файл проекту на диск під поточним ім'ям.



Зберегти як (**Ctrl+A**). Зберігає файл проекту на диск і запитує ім'я та шлях файлу.



Закрити (**Ctrl+F4**). Закриває поточний файл проекту.



Вихід (**Alt+F4**). Закриває програму і всі вікна.

Виправлення



Вирізати (**Ctrl+X**). Вирізає фрагмент схеми у внутрішній буфер обміну програми.



Копіювати (**Ctrl+C**). Копіює фрагмент схеми у внутрішній буфер обміну програми.



Вставити (**Ctrl+V**). Вставляє фрагмент схеми з внутрішнього буфера обміну програми.



Видалити (**Delete**). Видаляє фрагмент схеми.

Проект



Корпус мікросхеми. Показати/сховати корпус мікросхеми поточного проекту.



Редактор корпусу. Викликає діалогове вікно редактора корпусу мікросхеми поточного проекту.



Компіляція (**Alt+C**). Компілює поточний проект. Дана команда використовується для відновлення списку змінних у діаграмі і

таблиці проекту після зміни схеми без включення режиму моделювання. При включенні режиму моделювання компіляція робиться автоматично.



Додати в бібліотеку. Копіює поточний проект у буфер обміну редактора бібліотек і викликає редактор бібліотек. Для вставки компонента в бібліотеку необхідно викликати команду **Вставити** редактора бібліотек.



Настроювання (Ctrl+F4). Викликає діалогове вікно настроювання проекту.

Моделювання



Відробити інтервал. Відпрацьовує заданий користувачем інтервал модельного часу.

Відробити до. Відпрацьовує до моменту модельного часу, заданого користувачем.

Генератор (G). У синхронному режимі викликає чергову зміну стану генераторів і відпрацьовує схему до закінчення перехідних процесів. В асинхронному режимі відпрацьовує 1 такт модельного часу.

Інструменти



Редактор бібліотек (Alt+L). Активізує редактор бібліотек.



Настроювання. Викликає діалогове вікно налаштувань програми.



Діаграма (Alt+D). Виводить на екран діаграму станів змінних поточного проекту.



Таблиця (Alt+T). Виводить на екран таблицю станів змінних поточного проекту.

Вікно



Каскадом. Розташовує вікна відкритих проектів каскадом.



Розділити по вертикалі. Розташовує вікна відкритих проектів по вертикалі таким чином, щоб вони не перекривалися.



Розділити по горизонталі. Розташовує вікна відкритих проектів по горизонталі таким чином, щоб вони не перекривалися.



Збільшити. Збільшує масштаб у редакторі схеми поточного проекту.



Зменшити. Зменшує масштаб у редакторі схеми поточного проекту.

Допомога

Про програму. Виводить відомості про програму і розроблювачів.



Допомога. Викликає файл довідки.

Вказані нижче елементи керування винесені на окремі панелі.



Створити (Ctrl+N). Створює новий файл проекту.



Відкрити (Ctrl+O). Викликає діалогове вікно відкриття проекту.



Зберегти (Ctrl+S). Зберігає файл проекту на диск під поточним ім'ям.



Вирізати (Ctrl+X). Вирізає фрагмент схеми у внутрішній буфер обміну програми.



Копіювати (Ctrl+C). Копіює фрагмент схеми у внутрішній буфер обміну програми.



Вставити (Ctrl+V). Вставляє фрагмент схеми з внутрішнього буфера обміну програми.



Видалити (Delete). Видаляє фрагмент схеми.



Збільшити. Збільшує масштаб у редакторі схеми поточного проекту.



Зменшити. Зменшує масштаб у редакторі схеми поточного проекту.



Виділити. Дозволяє виділити і перетягнути лівою кнопкою миші фрагмент схеми, а також інвертувати виходи та входи елементу подвійним натисканням.



Елемент. Дозволяє вставити новий елемент у схему.



Лінія. Дозволяє провести зв'язок у схемі.



Перемінна. Дозволяє визначити змінну в схемі.



Захоплення. Дозволяє перетаскувати лівою кнопкою миші весь зміст вікна редактора логічних схем.



Редактор. Переводить редактор логічних схем у режим редагування.



Модель. Переводить редактор логічних схем у режим моделювання.



Авто. У натиснутому стані включає режим автоматичного моделювання, а у віджатому – режим покрокового моделювання.



1 такт. Відпрацьовує 1 такт модельного часу.



10 тактів. Відпрацьовує 10 тактів модельного часу.



100 тактів. Відпрацьовує 100 тактів модельного часу.

Панель компонентів

Панель компонентів розташована в лівій частині робочого стола моделі і являє собою ієрархічний список, у якому відображені підключені до програми бібліотеки і їх зміст. Кожна бібліотека містить ієрархічну структуру компонентів, подібну до файлової системи. Панель компонентів призначена для вибору компонента для наступної вставки його в схему.

Для дослідження схем у загальному випадку необхідно виконати послідовність дій.

1. Створити за допомогою редактора логічну схему на екрані дисплея.
2. Позначити на схемі вхідні і вихідні змінні.
3. Створити заголовок (перелічити вхідні і вихідні змінні) і сформулювати послідовність вхідних наборів в таблиці істинності.
4. Задати необхідні величини затримок сигналів для елементів схеми.
5. Встановити початковий стан схеми.
6. Перейти до режиму моделювання схеми.

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”
ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ
Кафедра обчислювальної техніки

КУРСОВА РОБОТА
з дисципліни "Комп'ютерна логіка"

Виконав

(прізвище, ім'я, по-батькові)

Факультет ІОТ,

Група _____

Залікова книжка № _____

Допущений до захисту _____

(підпис керівника)

Київ – 201_ р.

