

Лабораторна робота № 1

СИНТЕЗ КЕРУЮЧИХ АВТОМАТІВ ПО ГРАФУ МІКРОПРОГРАМИ

Мета роботи: закріплення теоретичних знань по синтезу керуючих автоматів (КА) із жорсткою логікою.

Підготовка до роботи

1. Одержати варіант домашнього завдання.
2. Вивчити теоретичну частину роботи.
3. Представити результати в протоколі виконання домашнього завдання.

Домашнє завдання

1. Представити в протоколі мету роботи й основні положення теоретичної частини.
2. Визначити закони функціонування КА, що реалізує задану мікропрограму (операцію).
3. Зробити оптимальне кодування станів КА.
4. Визначити перемикальні функції для сигналів збудження й вихідних сигналів автомата.
5. Підготувати таблицю даних для програми РІСТПІК.
6. Підготувати відповіді на контрольні питання.

Порядок виконання роботи

1. Представити викладачеві результати виконання домашнього завдання і одержати допуск до роботи.
2. Завантажити програму РІСТПІК.
3. Ввести із клавіатури таблицю даних і одержати результати рахунку (структурну таблицю переходів, вектор розмітки й коди станів автомату).
4. Порівняти результати машинного й ручного обчислень.

Зміст протоколу

1. Пункти 1-6 "Домашнього завдання".
2. Протоколи таблиць даних і результатів проектування КА на комп'ютері.
3. Висновки.

Контрольні питання

1. Яке призначення КА?
2. У чому складається відмінність автоматів Мілі й Мура?
3. У чому полягає канонічний метод синтезу КА?
4. Назвіть основні етапи структурного синтезу КА по ГМП.
5. Намалюйте структурну схему КА.
6. Для чого необхідно оптимальне кодування станів автомата?
7. Як виконується розмітка графа мікропрограми (ГМП) станами автоматів Мілі, Мура?

Стислі теоретичні відомості

1. Математичною моделлю КА називається абстрактний автомат, який визначається вектором

$$S = (A, X, Y, \delta, \lambda, a_1),$$

де $A = (a_1, \dots, a_n)$ - множина станів,

$X = (x_1, \dots, x_F)$ - множина вхідних сигналів,

$Y = (y_1, \dots, y_G)$ - множина вихідних сигналів,

δ і λ - відповідно функції переходів і виходів,

a_1 - початковий стан автомата.

Найбільше поширення на практиці одержали автомати Мілі й Мура, закон функціонування яких заданий відповідно рівняннями (1) і (2):

$$\begin{aligned} a(t+1) &= \delta(a(t), x(t)), \\ y(t) &= \lambda(a(t), x(t)), \end{aligned} \quad (1)$$

$$\begin{aligned} a(t+1) &= \delta(a(t), x(t)), \\ y(t) &= \lambda(a(t)). \end{aligned} \quad (2)$$

Розрізняють абстрактний і структурний синтез автомата.

На етапі абстрактного синтезу визначається закон функціонування й мінімізуються стани автомата.

Метою структурного синтезу є побудова схеми автомата з елементів заданого типу. Структурний синтез базується на застосуванні канонічного методу, що припускає використання елементарних автоматів спеціального виду: автомати з пам'яттю, що мають більше одного стану, й автомати без пам'яті – з одним станом. Результатом синтезу є система перемикальних рівнянь (5), що виражають залежність вихідних сигналів автомата й сигналів, що подаються на входи запам'ятовуючих елементів, від сигналів, що приходять на вхід усього автомата в цілому, і сигналів, що знімають із виходу елементів пам'яті:

$$y_1 = \lambda(u_1, \dots, u_k, x_1, \dots, x_L), \quad (3)$$

.....

$$y_G = \lambda(u_1, \dots, u_k, x_1, \dots, x_L),$$

$$V_1 = \delta(u_1, \dots, u_k, x_1, \dots, x_L),$$

.....

$$V_k = \delta(u_1, \dots, u_k, x_1, \dots, x_L).$$

Канонічний метод структурного синтезу припускає представлення структурної схеми автомата у вигляді двох частин: запам'ятовуючої (ЗЧ) і комбінаційної (КЧ) (рис. 1).

ЗЧ складається з попередньо вибраних елементарних автоматів пам'яті Π_1, \dots, Π_k . Таким чином, синтез структурного автомата зводиться до вибору

елементів пам'яті, кодування станів і синтезу КЧ, що реалізує систему рівнянь (4).

При синтезі КА по ГМП закон функціонування визначається так. Спочатку виконується відмітка ГМП станами автомата, потім відшукуються всілякі переходи в ГМП між сусідніми станами a_1, a_2, \dots .

Для автомата Мілі символом (станом) a_1 відзначаються вихід початкової й вхід кінцевої вершин. Виходи всіх операторних вершин послідовно відзначаються станами a_2, a_3 і т.д. Приклад відмітки ГМП станами автомата Мілі наведений на рис. 2, а.

У випадку автомата Мура кожній операторній вершині ГМП ставиться у відповідність деякий стан автомата. На ГМП (рис. 2, а) стани позначені поруч із операторними вершинами.

Граф автомата по відзначеному ГМП визначається виявленням усіляких шляхів переходу між сусідніми станами a_1, a_2, \dots . На кожному переході зі стану a_m у стан $a_s(a_m, a_s)$ повинна зустрічатися тільки одна операторна вершина. На переході виду (a_m, a_1) операторна вершина може бути відсутня. ГМП (рис. 2, а) відповідає графу автоматів Мілі (рис. 2, б) і Мура (рис. 2, в).

Автомат Мура дає рівну або більшу кількість станів, чим автомат Мілі.

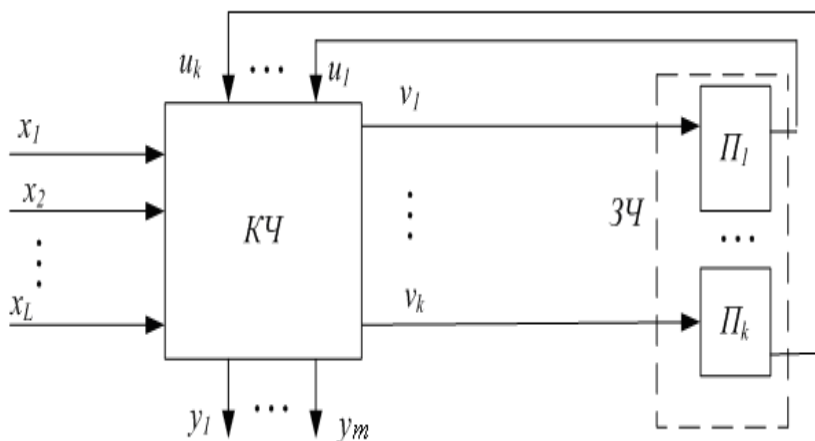
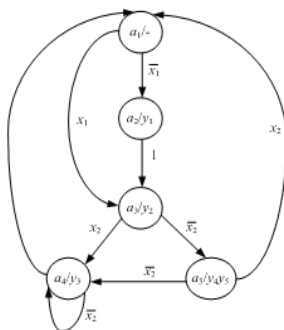
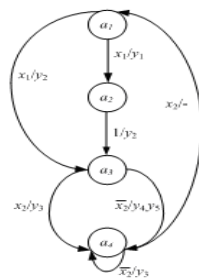
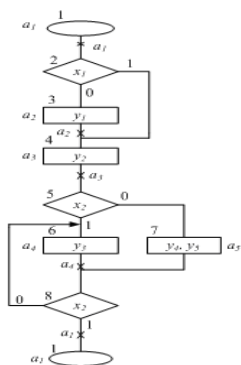


Рис. 1. Структурна схема КА



6

№ вершини ГМП	М		Зміст
	1	2	
1	0	2	-
2	3	4	x_1
3	0	4	y_1
4	0	5	y_2
5	7	6	x_2
6	0	8	y_3
7	0	8	y_4, y_5
8	6	1	x_7

2

Умовні вершини	
1	2
2	1
5	2
8	2

 ∂

Рис. 2

При великій кількості станів закон функціонування автомата зручніше представляти у вигляді структурної таблиці переходів (табл. 1).

Таблиця 1

Загальний вид структурної таблиці переходів автомата Мілі

Початковий стан	Код початкового стану	Стан переходу	Код стану переходу	Вхідний сигнал	Вихідний сигнал	Сигнал збудження
a_m	$K(a_m)$	a_s	$K(a_s)$	X	Y	V

Таблиця переходів автомата Мура має на один стовбець менше, тому що вихідний сигнал у ній записується поруч із початковим станом a_m .

2. Складність побудови КЧ автомата в основному визначається видом виразів функцій збудження пам'яті й функцій виходів, від яких, у свою чергу, залежить від способу кодування станів автомата. Скорочення апаратних витрат у КЧ може бути досягнуто за рахунок застосування сусіднього кодування, тому що для кожного переходу необхідний тільки один логічний елемент.

Розглянемо евристичний алгоритм кодування станів, який мінімізує сумарне число змін елементів пам'яті на всіх переходах. При такому критерії мінімізується також КЧ автомата. Для оцінки складності комбінаційної частини автомата вводиться вагова функція:

$$W = \sum t_{ms},$$

де t_{ms} - відстань між кодами станів a_m і a_s .

Алгоритм кодування складається з наступних кроків:

1. По структурній таблиці переходів (графу автомата) будується матриця переходів A :

$$A = \begin{bmatrix} \alpha_1 & \beta_1 \\ \dots\dots\dots \\ \alpha_r & \beta_r \end{bmatrix},$$

де α_r, β_r - номери станів, що відповідають наявності в автоматі переходів з a_{α_r} в a_{β_r} .

При побудові матриці A необхідно, щоб один з елементів r -го рядка вже містився в одному з попередніх рядків.

2. Стани з першого рядка матриці A кодуються так:

$$K_{\alpha_1} = 00\dots00, K_{\beta_1} = 00\dots01.$$

Довжина коду стану буде $I = \lceil \log_2 M \rceil$.

3. Будується матриця $A1$ шляхом видалення з A першого рядка із закодованими станами a_{α_1} та a_{β_1} .

4. Незакодований елемент у першому рядку матриці A_1 позначимо через γ . Вибравши з матриці A_1 рядки, що містять елементи γ , будується матриця A_γ .

5. У матриці A_γ виділяється множина елементів B_γ , які вже закодовані. Їхні коди позначаються через $K_{\gamma_1}, \dots, K_{\gamma_F}$.

6. Для кожного закодованого елемента множини B_γ знаходиться множина кодів $C_{\gamma_f}^1$.

7. Будується множина $D_\gamma^1 = \bigcup_{f=1}^F C_{\gamma_f}^1$. Якщо $D_\gamma^1 = \emptyset$, то будується нова множина $D_\gamma^2 = \bigcup_{f=1}^F C_{\gamma_f}^2$, де $C_{\gamma_f}^2$ - множина кодів, у яких кодова відстань з кодом

елемента множини B_γ дорівнює 2. Якщо $D_\gamma^2 = \emptyset$, то будуюмо аналогічно $D_\gamma^3, \dots, D_\gamma^n$ доти, поки не знайдеться $D_\gamma^n \neq \emptyset$. Нехай $D_\gamma^n = (K_{\delta_1}, \dots, K_{\delta_f})$.

8. Для кожного коду K_{δ_g} знаходиться кодова відстань ω_{k_g} між всіма використаними кодами K_{γ_f} .

$$\omega_{k_g} = |K_{\delta_g} \oplus K_{\delta_f}|.$$

9. Визначаємо $W_{k_g} = \sum_{f=1}^F \omega_{k_g}$. Якщо в графі автомата є переходи із a_{γ_f} в a_γ та із a_γ в a_{γ_f} , то ω_{k_g} в W_{k_g} входить двічі.

10. З множини D_γ^n вибирається K_γ , у якого W_{k_g} мінімально. Елемент γ (стан a_γ) кодується кодом $K_{\tilde{u}}$.

11. З матриці A_1 викреслюються рядки, у яких обидва елементи закодовані. Якщо в новій матриці A_1 не більше рядків переходимо до п.12, інакше – до п.4.

12. Обчислюємо функцію $W = \sum t_{ms}$. Оцінкою якості кодування служить число $K = W/P$, де P - число переходів у графі автомата.

Чим менше значення " K ", тим ближче кодування до сусіднього ($K=1$).

Приклад. Закодувати стан КА, граф якого зображений на рис.2, в.

1. Складемо матрицю переходів A

$$A = \begin{bmatrix} 1 & 2 \\ 1 & 3 \\ 2 & 3 \\ 3 & 4 \\ 3 & 5 \\ 4 & 1 \\ 5 & 1 \\ 5 & 4 \end{bmatrix}$$

і визначимо довжину коду стану $I = \lceil \log_2 5 \rceil = 3$.

2. Закодуємо стан 1-го рядка $K(a_1) = 000$, $K(a_2) = 001$.

3. Викресливши 1-й рядок у матриці A , отримаємо матрицю $A1$.

$$A1 = \begin{bmatrix} 1 & 3 \\ 2 & 3 \\ 3 & 4 \\ 3 & 5 \\ 4 & 1 \\ 5 & 1 \\ 5 & 4 \end{bmatrix}.$$

4. Незакодований елемент в 1-му рядку матриці $A1$ буде $\gamma=3$.

5. Будуємо матрицю $A\gamma = A3$ та множину $B\gamma = B3$

$$A3 = \begin{bmatrix} 1 & 3 \\ 2 & 3 \\ 3 & 4 \\ 3 & 5 \end{bmatrix}, \quad B3 = (1, 2).$$

6. За допомогою карти Карно визначаємо множину кодів C_1^1 та C_2^1

	00	01	11	10
$K(a_1)$			*	*
*	*	*		

$$C_1^1 = (010, 100),$$

$$C_2^1 = (011, 101).$$

7. Будуємо множину $D_3^1 = C_1^1 \cup C_2^1 = (010, 011, 100, 101)$.

8. Для кожного коду у множині D_3^1 знаходимо суму кодових відстаней між ним і всіма використаними кодами множини $B3$

$$W_{010} = |010 \oplus 000| + |010 \oplus 001| = 1 + 2 = 3;$$

$$W_{011} = |011 \oplus 000| + |011 \oplus 001| = 2 + 1 = 3;$$

$$W_{100} = |100 \oplus 000| + |100 \oplus 001| = 1 + 2 = 3;$$

$$W_{101} = |101 \oplus 000| + |101 \oplus 001| = 2 + 1 = 3.$$

Коди у множині D_3^1 мають однакову кодову відстань. Нехай $K(a_3) = 010$.

Після виключення закодованих пар з матриці $A1$ одержуємо:

$$A1 = \begin{bmatrix} 3 & 4 \\ 3 & 5 \\ 4 & 1 \\ 5 & 1 \\ 5 & 4 \end{bmatrix}, \quad \gamma = 4, \quad A4 = \begin{bmatrix} 3 & 4 \\ 4 & 1 \\ 5 & 4 \end{bmatrix}, \quad B4 = (1, 3).$$

	00	01	11	10
0	K (a ₁)	K		K
1	*			*

$$C_1^1 = (100),$$

$$C_3^1 = (011, 110),$$

$$D_4^1 = (011, 101, 110).$$

$$W_{011} = |011 \oplus 000| + |011 \oplus 010| = 2 + 1 = 3;$$

$$W_{101} = |101 \oplus 000| + |101 \oplus 010| = 2 + 3 = 5;$$

$$W_{110} = |110 \oplus 000| + |110 \oplus 010| = 2 + 1 = 3.$$

З множини D_4^1 вибираємо код $K(a_4) = 011$. Знову виключаємо з матриці $A1$ закодовані пари:

$$A1 = \begin{bmatrix} 3 & 5 \\ 5 & 1 \\ 5 & 4 \end{bmatrix}, \quad \gamma = 5, \quad A5 = \begin{bmatrix} 3 & 5 \\ 5 & 1 \\ 5 & 4 \end{bmatrix}, \quad B5 = (1, 3, 4).$$

	00	01	11	10
	K (a ₁)	K	K	K
	*		*	*

$$C_1^1 = (100),$$

$$C_3^1 = (110),$$

$$C_4^1 = (111),$$

$$D_5^1 = (100, 110, 111).$$

$$W_{100} = |100 \oplus 000| + |100 \oplus 010| + |100 \oplus 011| = 1 + 2 + 3 = 6;$$

$$W_{110} = |110 \oplus 000| + |110 \oplus 010| + |110 \oplus 011| = 2 + 1 + 2 = 5;$$

$$W_{111} = |111 \oplus 000| + |111 \oplus 010| + |111 \oplus 011| = 3 + 2 + 1 = 6.$$

Вибираємо код $K(a_5) = 110$.

У матриці $A1$ всі стани закодовані. Оцінимо якість кодування:

$$K = \frac{W}{P} = \frac{1+1+2+1+1+2+2+2}{8} = 1,5.$$

Методичні вказівки

1. У домашньому завданні визначаються закон функціонування й оптимальне кодування станів автомата, які в свою чергу порівнюються з результатами, що одержуються за допомогою прикладної програми РІСТПІК (розмітка, структурна таблиця та кодування).

2. При збігу результатів формується система булевих рівнянь (3), котра потім представляється в заданому базисі логічних елементів. При розбіжності результатів можливе помилкове кодування чи інший варіант кодування. У другому випадку вибір системи кодів (ручний або автоматизований варіант) визначається шляхом оцінки по Квайну витрат обладнання на реалізацію системи рівнянь (3).

3. Мікропрограма може задаватися у вигляді ГМП або найменуванням операції. В останньому випадку попередньо розробляється ГМП. У

домашньому завданні ГМП представляється у вигляді таблиці. Наприклад, ГМП (рис. 2, а) відповідає таблиці, що розміщена на рис. 2, з.

4. Початковою інформацією для РІСТПІК є ГМП, що вводиться в комп'ютер масивами M та UB . Масив M (рис. 2, з) описує зв'язки між вершинами ГМП (рис. 2, а), масив UB (рис. 2, д) – умовні вершини.

5. Варіанти домашнього завдання наведені в табл. 1. Таблиці, що описують ГМП, одержати у викладача.

Таблиця 1 **Варіанти домашнього завдання**

Варіант	Автомат	ГМП № (табл.)	ЕП	ЛЕ
1	1	1	<i>T</i>	I,АБО,НІ
2	2	2	<i>T</i>	I-НІ
3	1	3	<i>RS</i>	АБО-НІ
4	2	4	<i>D</i>	I-НІ
5	1	5	<i>T</i>	АБО-НІ
6	2	6	<i>RS</i>	I-НІ
7	1	7	<i>T</i>	I-НІ
8	2	8	<i>D</i>	I-НІ
9	1	9	<i>RS</i>	АБО-НІ
10	2	10	<i>D</i>	I-НІ
11	1	11	<i>RS</i>	АБО-НІ
12	2	12	<i>T</i>	I-НІ
13	1	13	<i>D</i>	АБО-НІ
14	2	14	<i>RS</i>	I-НІ
15	1	15	<i>T</i>	АБО-НІ
16	2	16	<i>D</i>	I-НІ
17	1	17	<i>RS</i>	АБО-НІ
18	2	18	<i>T</i>	I-НІ
19	1	19	<i>D</i>	АБО-НІ
20	2	20	<i>RS</i>	I-НІ
21	1	21	<i>T</i>	АБО-НІ
22	2	22	<i>D</i>	I-НІ
23	1	23	<i>RS</i>	АБО-НІ
24	2	24	<i>T</i>	I-НІ
25	1	25	<i>D</i>	АБО-НІ
26	2	26	<i>RS</i>	I-НІ
27	1	27	<i>T</i>	АБО-НІ
28	2	1	<i>D</i>	I-НІ
29	1	2	<i>T</i>	I,АБО,НІ
30	2	3	<i>D</i>	I-НІ
31	1	4	<i>T</i>	АБО-НІ
32	2	5	<i>RS</i>	I-НІ

Продовження табл. 1

Варіант	Автомат	ГМП № (табл.)	ЕП	ЛЕ
33	1	6	<i>D</i>	АБО-НІ
34	2	7	<i>RS</i>	І-НІ
35	1	8	<i>T</i>	АБО-НІ
36	2	9	<i>D</i>	І-НІ
37	1	10	<i>RS</i>	АБО-НІ
38	2	11	<i>T</i>	І-НІ
39	1	12	<i>D</i>	АБО-НІ
40	2	13	<i>RS</i>	І-НІ
41	1	14	<i>T</i>	АБО-НІ
42	2	15	<i>D</i>	І-НІ
43	1	16	<i>RS</i>	АБО-НІ
44	2	17	<i>T</i>	І-НІ
45	1	18	<i>D</i>	АБО-НІ
46	2	19	<i>RS</i>	І-НІ
47	1	20	<i>T</i>	АБО-НІ
48	2	21	<i>D</i>	І-НІ
49	1	22	<i>RS</i>	АБО-НІ
50	2	23	<i>T</i>	І-НІ
51	1	24	<i>RS</i>	АБО-НІ
52	2	25	<i>T</i>	І-НІ
53	1	26	<i>D</i>	АБО-НІ
54	2	27	<i>T</i>	І-НІ

Автомат = $\begin{cases} 1 - \text{Мілі,} \\ 2 - \text{Мура.} \end{cases}$
 ЕП – елемент пам’яті (тригер),
 ЛЕ – логічний елемент

