

Практична робота 3

Розробка функціональної схеми кінцевого автомата

Розробити функціональну схему кінцевого автомата, який виконував би заданий алгоритм роботи.

Кінцевий автомат – це пристрій, вихідний сигнал якого залежить не тільки від вхідного набору в даний момент часу, але й від вхідних наборів в попередні моменти часу. Для того, щоб пам'ятати, що було на вході раніше, автомат повинен мати елементи пам'яті. Як елементи пам'яті можуть застосовуватися зворотні зв'язки з затримкою або елементи з двома стійкими станами – тригери.

Комбінація виходів елементів пам'яті називається станом автомата або просто станом. Стани можуть бути стійкими та нестійкими. Стійкі стани можуть змінюватися лише при зміні вхід-

них наборів. Нестійкі стани змінюються під дією перехідних процесів в елементах автомата. Перехід від одного стійкого стану в інший стійкий здійснюється лише через нестійкий стан. Ці положення є ключовими для розуміння роботи автомата і розробки його структури.

Для розробки функціональної схеми автомата алгоритм його роботи, заданий в словесній формі, необхідно перевести в математичну. Функціонування автомата описується двома системами рівнянь.

1) системою рівнянь переходу:

$$Q_{i(t+1)} = f(X_i, Q_{it})$$

де $Q_{i(t+1)}$ – наступний стан автомата;

X_i – вхідний набір в даний момент часу;

Q_{it} – попередній стан автомата.

Ці рівняння показують, що наступний стан автомата визначається попереднім станом та вхідним набором в даний момент часу.

Кількість таких рівнянь дорівнює кількості елементів пам'яті.

2) системою рівнянь виходів:

$$Y = F(X_i, Q_i) \text{ – автомат Мілі}$$

або

$$Y = f(Q_i) \text{ – автомат Мура.}$$

Кількість таких рівнянь визначається кількістю виходів автомата.

Завдання розробника полягає в тому, щоб по словесному опису алгоритму роботи автомата знайти конкретні рівняння пере-

ходів та рівняння виходів, по яким можна розробити функціональну схему автомата.

Якщо вхідні змінні подаються у вигляді паралельного коду, то розробка схеми автомата здійснюється в такій послідовності:

1. Розробка таблиці переходів абстрактного автомата (таблиці Хаффмена).
2. Мінімізація станів автомата.
3. Кодування станів автомата.
4. Отримання таблиці переходів структурного автомата і рівнянь переходів.
5. Отримання таблиці виходів структурного автомата і рівнянь виходів.
6. Розробка функціональної схеми автомата.

Якщо в якості елементів пам'яті використовуються тригери, то по таблиці переходів структурного автомата отримують таблицю збудження, а з неї рівняння, які управляють тригерами.

Найбільш складною задачею є розробка таблиці Хаффмена, тому що процес її отримання мало формалізований.

Кількість стовпців таблиці Хаффмена дорівнює 2^n , де n – кількість вхідних змінних. До них треба додати ще стовпці для кожного виходу автомата.

Кількість рядків таблиці Хаффмена наперед визначити неможливо, але їх буде не менше кількості стовпців.

Для кожного вхідного набору необхідно визначити (по словесному опису роботи автомата) кількість можливих комбінацій виходів автомата або кількість попередніх вхідних наборів, які необхідно запам'ятати при даному вхідному. Кожному варіанту відводиться в таблиці свій стійкий стан автомата, який позначають цифрами або літерами в дужках для того, щоб відрізнити їх від нестійких станів. Кожний стійкий стан повинен розташовуватися в окремому рядку.

Решта клітин таблиці заповнюється нестійкими станами або рисками, якщо даний перехід неможливий.

Заповнення таблиці нестійкими станами проводиться таким чином: беремо за початковий який-небудь стійкий стан і змінюємо вхідний набір, тобто переміщуємося по таблиці в горизонтальному напрямку. При цьому вхідному наборі автомат повинен мати певне значення виходу, якому відповідає певний стійкий стан в цьому стовпчику. Тому в даній клітині таблиці ставимо такий же стан, але нестійкий. Якщо перехід по горизонталі з певних причин неможливий, в даній клітині ставимо ризик. Повертаємося в початковий стан і повторимо цю процедуру, але для іншого вхідного набору, доки не будуть заповнені всі клітини даного рядка.

Кількість рядків таблиці переходів визначає кількість елементів пам'яті, які необхідні для заданої роботи автомата. Співвідношення між ними визначається формулою:

$$2^Q \geq N$$

де Q – мінімальна кількість елементів пам'яті, яка задовольняє цю нерівність;

N – кількість рядків таблиці переходів.

Кількість елементів пам'яті визначає складність автомата, тому що для управління цими елементами необхідні відповідні схеми. Тому необхідно визначити можливість розробки автомата з меншою кількістю елементів пам'яті. Це можливо, якщо автомат має еквівалентні стани. Еквівалентними називаються стани, при яких автомат однаково реагує на вхідні сигнали. Два стани – два рядки таблиці переходів – будуть еквівалентними, якщо в кожній їх клітині однакові елементи, причому стійкі та нестійкі стани з однаковими номерами (з однаковими позначеннями) вважаються еквівалентними. Клітину з ризикою можна віднести до будь-якого стану.

Еквівалентні рядки об'єднуються за таким правилом: стійкий стан з нестійким або з ризикою утворює стійкий стан; нестійкий стан з нестійким або з ризикою утворює нестійкий стан; ризика з ризикою утворює ризик.

В отриманій таблиці з мінімальною кількістю рядків необхідно закодувати стани. Кількість розрядів коду дорівнює кількості елементів пам'яті. Складність автомата залежить від способу кодування. Найменш складним буде автомат, у якого мінімальна кількість переключень елементів пам'яті. Оптимальне кодування рядків таблиці виконується за таким алгоритмом:

- 1) рядок з максимальною кількістю станів кодується нульовою комбінацією. Якщо таких рядків декілька, то вибір з них довільний;
- 2) визначимо, в який рядок переходить автомат з закодованого. Код цього рядка повинен бути таким, щоб при переході до цього рядка переключався лише один елемент пам'яті;
- 3) це правило застосовують до всіх наступних рядків.

Еквівалентні рядки об'єднуються за таким правилом: стійкий стан з нестійким або з ризикою утворює стійкий стан; нестійкий стан з нестійким або з ризикою утворює нестійкий стан; риска з ризикою утворює ризик.

В отриманій таблиці з мінімальною кількістю рядків необхідно закодувати стани. Кількість розрядів коду дорівнює кількості елементів пам'яті. Складність автомата залежить від способу кодування. Найменш складним буде автомат, у якого мінімальна кількість переключень елементів пам'яті. Оптимальне кодування рядків таблиці виконується за таким алгоритмом:

- 1) рядок з максимальною кількістю станів кодується нульовою комбінацією. Якщо таких рядків декілька, то вибір з них довільний;
- 2) визначимо, в який рядок переходить автомат з закодованого. Код цього рядка повинен бути таким, щоб при переході до цього рядка переключався лише один елемент пам'яті;
- 3) це правило застосовують до всіх наступних рядків.

Після кодування можна перейти до побудови таблиці переходів та таблиці виходів структурного автомата.

Рядки таблиці переходів структурного автомата відповідають станам автомата в даний момент часу, а стовпці – вхідним наборам. Елементами таблиці переходів структурного автомата будуть стани в наступний момент часу. Тому таблицю отримують за таким формальним правилом: на місце стійких станів ставлять код цього рядка, а на місце нестійких станів – код рядка, який відповідає такому ж стійкому стану.

Рядки таблиці виходів структурного автомата відповідають станам автомата в даний момент часу, а стовпці – вхідним наборам. Елементами цієї таблиці будуть значення виходів автомата. Таку таблицю отримують за таким формальним правилом: на міс-

це стійких станів в таблиці переходів ставимо значення виходу автомата, а на місце нестійких – риску.

Таблиця виходів структурного автомата не залежить від типу елементів пам'яті. Її можна розглядати як карту Карно функції виходу, яку і отримують з цієї карти.

Якщо автомат реалізується на зворотних зв'язках, то таблицю переходів розглядають як карту Карно функцій переходів, які і отримують з цієї таблиці.

Якщо автомат реалізують на тригерах, то необхідно побудувати таблицю збудження цих тригерів. Цю таблицю будують по таблиці переходів структурного автомата і по таблиці, яка описує закон функціонування вибраного тригера. Остання показує, які сигнали необхідно подати на вхід (входи) тригера, щоб він зі стану Q_t перейшов в стан Q_{t+1} [1]. Коди рядків таблиці переходів є значеннями Q_t тригерів, тоді як елементи цієї таблиці є значеннями Q_{t+1} тригерів. Тому в таблиці переходів треба замість елементів Q_{t+1} поставити відповідні значення входів тригерів. При застосуванні D – тригерів таблиця переходів і таблиця збудження співпадають.

З таблиці збудження отримують рівняння управління вибраними тригерами.

Функціональну схему автомата будують починаючи з рівнянь переходів або з рівнянь управління тригерами.

Більш детальну інформацію по розробці схем автоматів можна отримати з [1].

1.6. Приклад вирішення задачі №3

Як приклад, вирішимо таку задачу:

Розробити автомат, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює $01 \rightarrow 11 \rightarrow 10$.

Цей алгоритм роботи автомата треба розуміти таким чином: коли вхідний набір змінюється з 01 на 11, а потім на 10, на виході

автомата повинен з'явитися одиничний сигнал. При будь-якій іншій послідовності на вході, на виході автомата буде нульовий сигнал.

З задачі видно, що автомат має дві вхідні змінні — x_1 та x_2 . Тому таблиця Хаффмена буде мати п'ять стовпців (рис. 1.9)

	x_1		x_2		
	00	01	11	10	У
[1]					0
		[2]			0
			[3]		0
			[4]		0
				[5]	0
				[6]	1

Рис. 1.9. Таблиця Хаффмена без нестійких станів

При вхідній комбінації 00 автомат не цікавить попереднє значення вхідного набору. Тому досить одного стійкого стану [1] для цього вхідного набору. Вихід автомата при цьому дорівнює нулю, тому що одиничний сигнал на виході може бути лише при вхідній комбінації 10 і при виконанні решти умов задачі. Теж саме маємо і при вхідній комбінації 01.

При вхідній комбінації 11 будемо мати два варіанта подій: коли на вході в попередній момент часу була комбінація 01 і коли на вході в попередній момент часу була якась інша комбінація. Тому при вхідному набору 11 треба передбачити два стійких стани — [3] та [4]. Один з них, наприклад [3], (вибір довільний) відповідає випадку, коли перед цим на вході була комбінація 01, інший — [4] — відповідає випадку, коли на вході була якась інша комбінація. В обох випадках вихідний сигнал автомата дорівнює нулю.

При вхідному набору 10 теж необхідно мати два стійких стани — [5] та [6]. Один з них, наприклад [6], відповідає випадку, коли перед цим на вході була комбінація 11 і при цьому вихід автомата повинен бути одиничним. Другий — [5] — відповідає випад-

ку, коли перед цим на вході була якась інша комбінація і при цьому вихід автомата повинен бути нульовим.

Таким чином, для заданого автомата маємо шість стійких станів і відповідна таблиця має шість рядків. Решта клітин таблиці необхідно заповнити нестійкими станами. Для зручності координати клітин будемо визначати номерами стану (номер рядка) та номерами вхідних наборів (номер стовпця).

Якщо в стовпчику є тільки один стійкий стан, то решта клітин цього стовпчика будуть заповнені такими ж станами, тільки нестійкими. Тому нульовий та перший стовпчики таблиці заповнені так, як це показано на рисунку 1.10

		$x_1 x_2$				
		00	01	11	10	Y
[1]	2	4	5	0		
1	[2]	3	5	0		
1	2	[3]	6	0		
1	2	[4]	5	0		
1	2	4	[5]	0		
1	2	4	[6]	1		

Рис. 1.10. Таблиця переходів абстрактного автомата

Раніше ми визначили стан [3] як стан, який відповідає випадку, коли на вході автомата в попередній момент часу була комбінація 11. Тому в клітині ([3] – 3) ставимо нестійкий стан 3. Решта вільних клітин цього стовпчика необхідно заповнити нестійким станом 4.

Раніше ми визначили стан [6] як стан, який відповідає випадку, коли на вході автомата в попередній момент часу була комбінація 11. Тому в клітині ([3] – 2) ставимо нестійкий стан 6. Решта вільних клітин цього стовпчика необхідно заповнити нестійким станом 5.

Отримана таблиця повністю відповідає заданому алгоритму роботи автомата.

Таблиця має шість рядків, а тому для реалізації автомата необхідно мати три елементи пам'яті. Визначимо можливість скорочення кількості цих елементів.

Рядки 1, 4 та 5 мають однакові числа в відповідних клітинах. Тому ці рядки (стани) еквівалентні і їх можна об'єднати в один рядок (в один стан) за правилом, яке було викладено раніше. Нова таблиця має вигляд, наведений на рисунку 1.11.

		$x_1 \ x_2$				
		00	01	11	10	Y
	00	[1]	2	[4]	[5]	0
Q_{1t}	01	1	[2]	3	5	0
Q_{2t}	11	1	2	[3]	6	0
	10	1	2	4	[6]	1

Рис. 1.11. Мінімізована таблиця переходів автомата

З цієї таблиці видно, що для реалізації автомата досить двох елементів пам'яті.

Закодуємо рядки таблиці. Згідно з описаним вище правилом першому рядку припишемо нульову комбінацію. З першого рядка автомат може перейти тільки в другий через нестійкий стан 2 при зміні вхідного нору з 00 на 01. Тому другий рядок буде мати код 01, який говорить про те, що при такому переході змінюється стан лише одного елемента пам'яті.

З другого рядка автомат може перейти в третій (він може перейти і в перший, але ми вже там були) через нестійкий стан 3. Тому третій рядок отримає комбінацію 11. Аналогічно четвертий рядок отримає комбінацію 10.

Таке кодування гарантує відсутність критичних гонок і мінімальну складність автомата.

Побудуємо таблицю виходів структурного автомата за правилами, які були викладені раніше (рис. 1.12).

З цієї таблиці можна отримати рівняння виходу, яке не залежить від типу елементів пам'яті.

$$Y = Q_{1t} \bar{Q}_{2t}$$

		$x_1 x_2$			
		00	01	11	10
Q_{1t}	00	0	-	0	0
	01	-	0	-	-
Q_{2t}	11	-	-	0	-
	10	-	-	-	1

y

Рис. 1.12. Таблиця виходів структурного автомата

Побудуємо таблицю переходів структурного автомата (рис. 1.13)

		$x_1 x_2$			
		00	01	11	10
Q_{1t}	00	00	01	00	00
	01	00	01	11	00
Q_{2t}	11	00	01	11	10
	10	00	01	00	10

$Q_{1(t+1)}, Q_{2(t+1)}$

Рис. 1.13. Таблиця переходів структурного автомата

Якщо автомат реалізується на зворотних зв'язках, то з цієї таблиці можна отримати рівняння переходів в мінімальній формі. Перші числа в цій таблиці відносяться до першого елемента пам'яті, другі – до другого.

$$Q_{1(t+1)} = x_1 x_2 Q_{2t} + x_1 \bar{x}_2 Q_{1t} = x_1 (x_2 Q_{2t} + \bar{x}_2 Q_{1t})$$

$$Q_{2(t+1)} = \bar{x}_1 x_2 + x_2 Q_{2t} = x_2 (\bar{x}_1 + Q_{2t}) = x_2 \overline{x_1 Q_{2t}}$$

По отриманих рівняннях можна отримати функціональну схему автомата. Розробку схеми починаємо з побудови схем переходів.

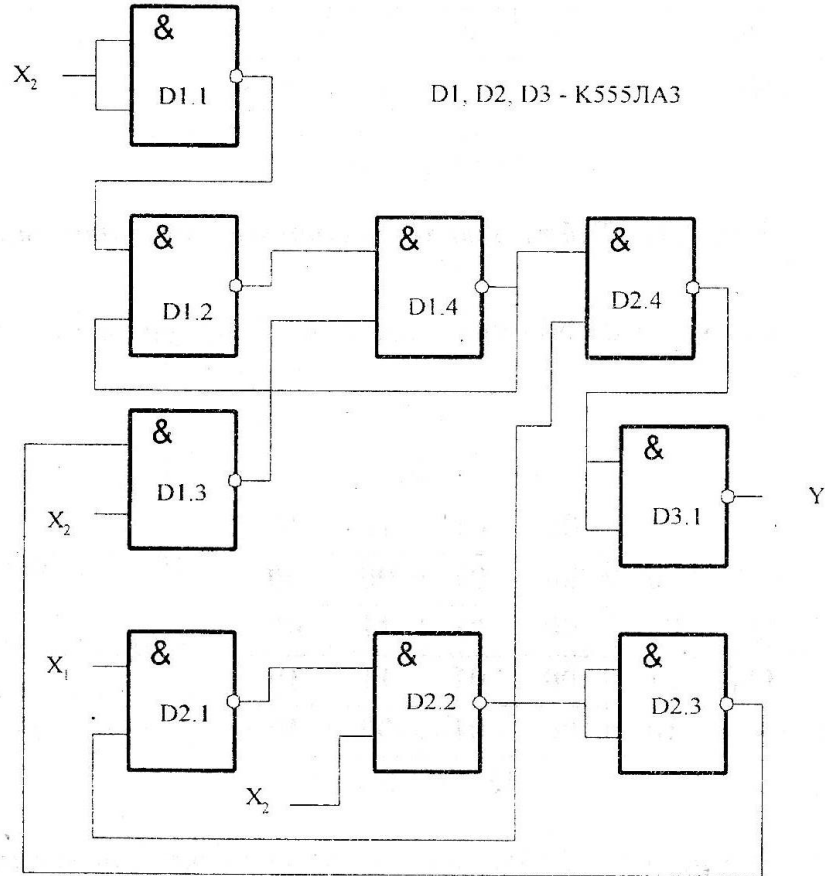


Рис. 1.14. Функціональна схема автомата на зворотних зв'язках

Вирішимо цю задачу, застосувавши в якості елементів пам'яті RS – тригер, закон функціонування якого зображений на рис.1.15

$Q_t \rightarrow Q_{t+1}$	R	S
0 → 0	*	0
0 → 1	0	1
1 → 0	1	0
1 → 1	0	*

Рис. 1.15. Закон функціонування RS – тригера

Знак “*” означає, що на даний вхід можна подати як нуль, так і одиницю.

По цій таблиці, а також по таблиці переходів абстрактного автомата побудуємо таблицю збудження першого тригера (рис. 1.16)

		x ₁ x ₂			
		00	01	11	10
00		*0	*0	*0	*0
Q _{1t}	01	*0	*0	01	*0
Q _{2t}	11	10	10	0*	0*
	10	10	10	10	0*

R₁ S₁

Рис. 1.16. Таблиця збудження першого тригера

З цієї таблиці маємо рівняння управління першим тригером:

$$R = \bar{x}_1 + x_2 \bar{Q}_{1t}$$

$$S_1 = x_1 x_2 Q_{2t}$$

Таблиця збудження другого тригера зображена на рис. 1.17

		x ₁ x ₂			
		00	01	11	10
00		*0	01	*0	*0
Q _{1t}	01	10	0*	0*	10
Q _{2t}	11	10	0*	0*	10
	10	*0	01	*0	*0

R₂ S₂

Рис. 1.17. Таблиця збудження другого тригера

З цієї таблиці маємо рівняння управління другим тригером:

$$R_2 = \bar{x}_2$$

$$S_2 = \bar{x}_1 x_2$$

Функціональна схема автомата зображена на рис. 1 18

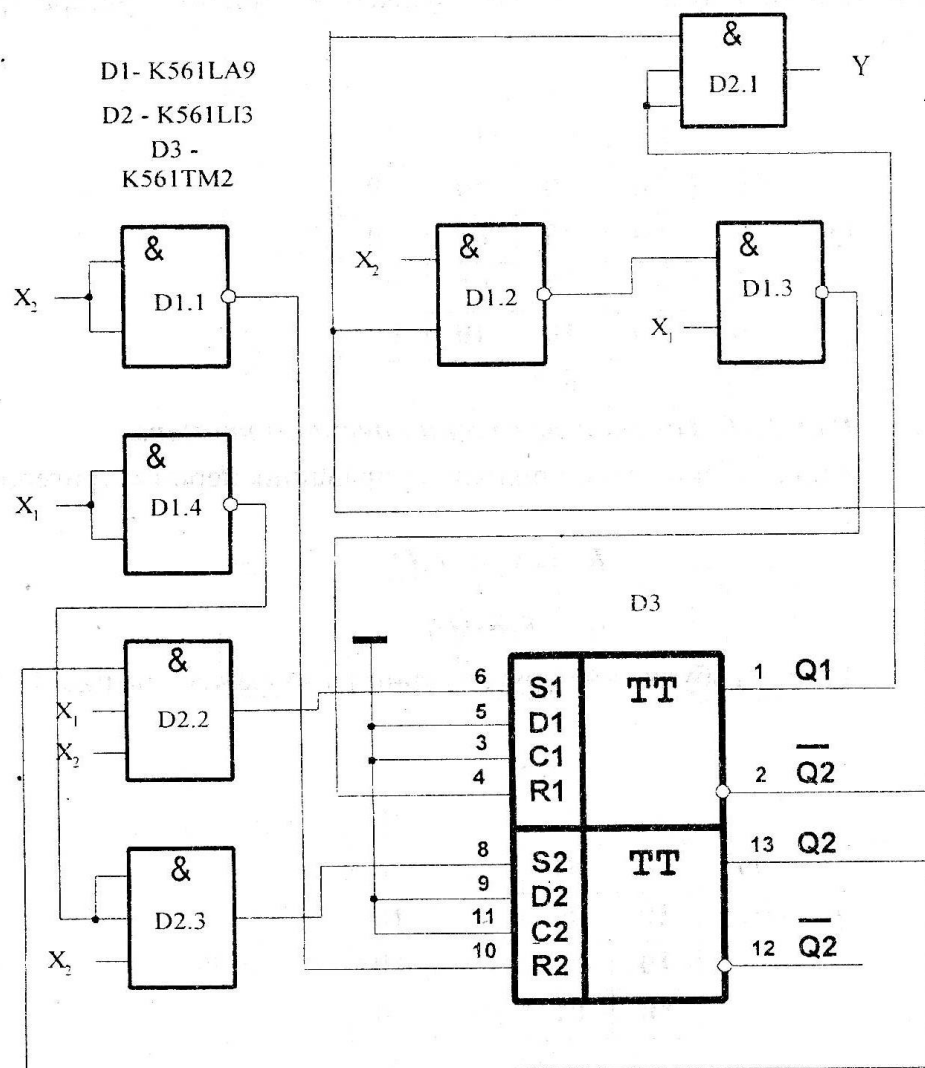


Рис.1.18. Функціональна схема автомата на тригерах

Варіант 1

Задача 3. Розробити автомат на зворотних зв'язках, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

00 → 01 → 11

Варіант 2

Задача 3. Розробити автомат на D-тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

00 → 01 → 11

Варіант 3

Задача 3. Розробити автомат на RS-тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

00 → 01 → 11

Варіант 4

Задача 3. Розробити автомат на $\overline{R}\overline{S}$ – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

00 → 01 → 11

Варіант 5

Задача 3. Розробити автомат на JK – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

00 → 01 → 11

Варіант 6

Задача 3. Розробити автомат на зворотних зв'язках, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 01$$

Варіант 7

Задача 3. Розробити автомат на D – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 01$$

Варіант 8

Задача 3. Розробити автомат на RS – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 01$$

Варіант 9

Задача 3. Розробити автомат на \overline{RS} – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 01$$

Варіант 10

Задача 3. Розробити автомат на JK – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 01$$

Варіант 11

Задача 3. Розробити автомат на зворотних зв'язках, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 110$$

Варіант 12

Задача 3. Розробити автомат на D – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 10$$

Варіант 13

Задача 3. Розробити автомат на RS – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 10$$

Варіант 14

Задача 3. Розробити автомат на \overline{RS} – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 10$$

Варіант 15

Задача 3. Розробити автомат на JK – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 11 \rightarrow 10$$

Варіант 16

Задача 3. Розробити автомат на зворотних зв'язках, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 10 \rightarrow 11$$

Варіант 17

Задача 3. Розробити автомат на D – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 10 \rightarrow 11$$

Варіант 18

Задача 3. Розробити автомат на RS – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 10 \rightarrow 11$$

Варіант 19

Задача 3. Розробити автомат на \overline{RS} – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 10 \rightarrow 11$$

Варіант 20

Задача 3. Розробити автомат на JK – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$$00 \rightarrow 10 \rightarrow 11$$

Варіант 21

Задача 3. Розробити автомат на зворотних зв'язках на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$00 \rightarrow 10 \rightarrow 01$

Варіант 22

Задача 3. Розробити автомат на D – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$00 \rightarrow 10 \rightarrow 01$

Варіант 23

Задача 3. Розробити автомат на RS – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$00 \rightarrow 10 \rightarrow 01$

Варіант 24

Задача 3. Розробити автомат на $\overline{R}\overline{S}$ – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$00 \rightarrow 10 \rightarrow 01$

Варіант 25

Задача 3. Розробити автомат на JK – тригерах, на виході якого буде одиничний сигнал, якщо вхідна послідовність дорівнює:

$00 \rightarrow 10 \rightarrow 01$