

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МІСЬКОГО ГОСПОДАРСТВА імені О. М. БЕКЕТОВА**

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

для самостійної і виконання
розрахунково-графічної робіт
із навчальної дисципліни

«АРХІТЕКТУРА КОМП'ЮТЕРНИХ СИСТЕМ»

*(для студентів 2 курсу денної і заочної форм навчання
першого (бакалаврського) рівня вищої освіти
спеціальностей 122 – Комп'ютерні науки,
126 – Інформаційні системи та технології,
151 – Автоматизація та комп'ютерно-інтегровані технології*

**Харків
ХНУМГ ім. О. М. Бекетова
2021**

Методичні рекомендації для самостійної і виконання розрахунково-графічної робіт із навчальної дисципліни «Архітектура комп'ютерних систем» (для студентів 2 курсу денної і заочної форм навчання першого (бакалаврського) рівня вищої освіти спеціальностей 122 – Комп'ютерні науки, 126 – Інформаційні системи та технології, 151 – Автоматизація та комп'ютерно-інтегровані технології / Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова ; уклад. А. Л. Литвинов. – Харків : ХНУМГ ім. О. М. Бекетова, 2021. – 38 с.

Укладач проф. А. Л. Литвинов

Рецензент

О. П. Нечуйвітер, доктор фізико-математичних наук, професор, завідувач кафедри інформаційних комп'ютерних технологій і математики Української інженерно-педагогічної академії

Рекомендовано кафедрою комп'ютерних наук та інформаційних технологій, протокол № 15 від 26.04.2021.

ЗМІСТ

Вступ	4
1 Методичні рекомендації щодо організації самостійної роботи	5
2 Основні поняття стосовно цифрового автомата й методики його проектування	10
3 Завдання для розрахунково-графічної роботи.	21
4 Технологія виконання розрахунково-графічної роботи.	23
4.1 Програмна реалізація автомата	23
4.2 Апаратна реалізація автомата	27
Список рекомендованих джерел	35
Додаток А Зразок оформлення титульного аркуша розрахунково-графічної роботи.	36
Додаток Б Зразок оформлення змісту розрахунково-графічної роботи.	37

ВСТУП

Дисципліна «Архітектура комп'ютерних систем» є однією з головних в освітніх програмах галузі знань «Інформаційні технології» і дозволяє вивчити базові теоретичні і схемотехнічні принципи побудови та функціонування комп'ютерних систем, сформувати практичні навички дослідження і проєктування вузлів комп'ютерних систем. Комп'ютерна система є основою побудови інформаційних систем і технологій у різних сферах людської діяльності. Комп'ютерна техніка забезпечує ефективну роботу з базами даних, моделювання найскладніших процесів, управління різноманітними технологічними процесами та автоматизоване проєктування.

Мета викладання навчальної дисципліни «Архітектура комп'ютерних систем» – засвоєння необхідних знань з основ теорії побудови та функціонування архітектури сучасної комп'ютерної техніки, необхідних для подальшого навчання, а також для практичної діяльності на виробництві. Основні завдання вивчення дисципліни – підготовка висококваліфікованих спеціалістів, які вміють раціонально використовувати сучасні типи комп'ютерів під час розроблення інформаційних систем різного призначення.

У результаті вивчення навчальної дисципліни студент повинен:

знати: порядок аналізу та синтезу систем із заданими властивостями; основи комп'ютерної інженерії (комп'ютерну схемотехніку, архітектуру комп'ютерів, мікропроцесорні системи); роль та місце комп'ютерної схемотехніки в завданнях проєктування складних систем;

уміти: оцінювати характеристики елементів та вузлів, виявляти та усувати несправності в елементах та схемах складної техніки; розробляти основні типи цифрових електронних систем, розраховувати їхню роботу, параметри та характеристики.

Вивчення дисципліни «Архітектура комп'ютерних систем» передбачає засвоєння значного обсягу навчального матеріалу у вільний час у формі самостійної навчальної роботи та виконання розрахунково-графічної роботи (РГР). Згідно з робочою програмою з дисципліни, на цей вид діяльності припадає до 40 % кількості годин, відведених на вивчення дисципліни. Це уможливорює формування спеціаліста, який уміє самостійно проводити роботи з проєктуванні комп'ютерних систем, аргументовано обирати програмні й технічні засоби для створення інформаційних систем і технологій на основі аналізу їхніх властивостей.

1 МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ЩОДО ОРГАНІЗАЦІЇ САМОСТІЙНОЇ РОБОТИ

Опанування дисципліни «Архітектура комп'ютерних систем» передбачає засвоєння значного обсягу навчального матеріалу у вільний час у формі самостійної навчальної роботи, покликаній сформувати у студентів практичні навички опрацювання спеціальних джерел, зорієнтувати їх на інтенсивну роботу, критичне осмислення здобутих знань та глибоке вивчення теоретичних і практичних аспектів комп'ютерних систем. Як результат, це формує спроможність студента давати відповіді, пояснювати, розуміти пояснення, дискутувати, звітувати державною мовою на достатньому для професійної діяльності рівні, здатність спілкуватися іноземною мовою, застосовувати сучасні інформаційні й комунікаційні технології для розв'язання практичних завдань щодо розроблення комп'ютерних систем і проєктування інформаційних технологій.

Самостійна навчальна робота зорієнтована на набуття студентами здатності генерувати нові ідеї (креативність), працювати автономно та в команді, набувати знання та розуміти предметну сферу, суть професійної діяльності.

Самостійна робота над засвоєнням навчального матеріалу з дисципліни може виконуватися в бібліотеці, навчальних кабінетах та лабораторіях кафедри «Комп'ютерних наук та інформаційних технологій», на базі платформи для дистанційного навчання Moodle, а саме дистанційного курсу «Архітектура комп'ютерних систем» (<https://dl.kname.edu.ua/course/view.php?id=827>). Раціональна організація самостійної роботи потребує умілого розподілу часу між аудиторною та позааудиторною роботою.

Значна частка самостійної роботи студента в курсі «Архітектура комп'ютерних систем» припадає на розрахунково-графічну роботу (РГР). РГР – це самостійно виконана робота студента. Виконуючи РГР, студент удосконалює знання та вміння, отримані в процесі вивчення дисципліни «Архітектура комп'ютерних систем», а саме: визначати мету розробки, відокремлювати завдання, формулювати проблеми та знаходити способи їхнього розв'язання.

Мета виконання РГР [1]:

- Систематизувати, закріпити та розширити теоретичні знання і практичні вміння студента;

- закріпити теоретичні знання та набуті практичних навичок щодо проектуванні окремих вузлів комп'ютерної системи комплексу технічних засобів інформаційних систем;
- набуті досвіду роботи з літературою та іншими джерелами інформації, умінь узагальнювати та аналізувати наукову інформацію, формувати власне ставлення до проблеми;
- набуті вмінь застосовувати інформаційні та комп'ютерні технології для розв'язання прикладних інформаційних задач;
- розвивати навички оволодіння спеціалізованим програмним забезпеченням;
- обґрунтовано аналізувати результати власних досліджень і формувати змістовні висновки стосовно якості отриманих результатів.

Виконання завдання із РГР обов'язкове для кожного студента.

Види організації самостійної роботи можуть бути найрізноматнішими, однак завжди варто керуватися принципом активізації розумової діяльності, стимулюючи потребу поглиблювати одержані знання шляхом використання різних видів самостійної роботи. Дуже важливо в кожному конкретному випадку застосовувати ті види самостійної роботи, які активно сприяють формуванню відповідних умінь. За різновидом організації всі види самостійної роботи можна розподілити на дві групи: види самостійної роботи у процесі проведення очних занять і види самостійної роботи, що проводяться в позаурочний час.

Однією із форм організації самостійної роботи, що застосовується безпосередньо на лекції, є конспектування. Будь-який конспект повинен орієнтуватися тільки на головні думки, відображаючи при цьому суть теми. Якщо конспект пишеться від руки, то у вас обов'язково повинні бути при собі зошит і комплект різнокольорових ручок (не буде зайвим застосувати і різнокольоровими маркерами). Конспектування доцільно розпочинати з дати і назви (якщо в зошиті є якийсь більш ранній матеріал, від нього необхідно відступати кілька рядків, щоб масиви даних не зливалися), найважливіше й істотніше бажано виділяти кольором (наприклад писати ручкою іншого кольору або застосовувати маркер). Важливо визначати у виступі лектора найістотніші моменти, які й потрібно заносити в конспект. Не рекомендується писати все в один рядок. Використовуйте цитатну техніку написання. Це означає, що кожен нову ідею потрібно фіксувати з нового рядка. До того ж між усіма рядками бажано робити відступи в один рядок. Це значно спрощує сприйняття і дозволяє доповнювати записи. Те, що лектор або викладач

зображує на дошці, потрібно фіксувати й у себе в зошиті, адже якщо він це робить, це особливо важливо. Записувати ж це можна або графічно, або тезисно. Не треба нехтувати аббревіатурами та скороченнями. Цифри завжди пишіть знаками. Це багаторазово зменшує обсяг матеріалу, але значення не втрачається. Включайте в конспект не тільки текст, скорочення, цифри й розділові знаки, а й таблиці, схеми, діаграми, формули, графіки та малюнки. Ці графічні підказки економлять час на фіксування інформації, а також легко запам'ятовуються.

Важливу роль під час написання конспекту відіграють скорочення. Можна скорочувати ключові слова до великої літери і використовувати ці скорочення в тексті. Замість тексту «Архітектура комп'ютерних систем» можна використовувати «АКС », «магнітний резонанс» можна скоротити до «МР» тощо. Якщо якісь літери співпадають, відокремити одні від інших можна за допомогою спеціальних позначень, наприклад обвівши букву в коло чи квадрат тощо. Коли зустрічаєтеся з поширеними довгими словами, оперуйте загальноприйнятими скороченнями, наприклад: функція – ф-я; формула – ф-ла; керівництво – кер-во; література – літ-ра тощо.

У сучасних смартфонах є функція високоякісного диктофона, тому особливо важливі лекції можна просто записувати, супроводжуючи їх конспектуванням формул і малюнків. Можна навіть знімати (якщо лектор не заперечує) відеофільм. З розвитком планшетних комп'ютерів, обладнаних сенсорним екраном (тачскрин – touch screen), з'явилася можливість конспектування на екрані. Конспект можна зберегти в pdf-форматі і переслати в хмару. Як додаток можна використовувати OneNote. Як інструмент запису необхідно підібрати відповідний стилус.

До самостійної роботи, що виконуються студентами в позаурочний час належать: робота з навчальною літературою, рекомендованою викладачем; робота з конспектами; виконання домашніх завдань, розрахунково-графічних робіт; написання рефератів; виконання кваліфікаційних робіт, курсових проєктів та інших робіт. Інформацію для них можна знайти в Інтернеті за допомогою пошукових систем, наприклад Google. Доцільно розпочинати пошук з простіших запитів, наприклад «зовнішня пам'ять». У пошукових системах використовуються спеціальні символи, які дозволяють отримати точніший результат. Наприклад, для пошуку чіткої фрази, її потрібно розмістити в лапках, для пошуку всередині діапазону потрібно ввести .. . Приклад: *комп'ютер підручник 2000..2019*.

Під час виконання кваліфікаційних робіт, курсових проєктів та інших робіт важливо підбирати цитовану літератури та працювати з нею. Для цього рекомендується використовувати багатофункціональні системи пошуку наукових праць, такі як Google Scholar, Scopus, Web of Science. Для студентів рекомендується використовувати Google Scholar (Google Академія). Це безкоштовна пошукова система щодо повних текстів наукових публікацій всіх форматів і дисциплін. Індекс Google Scholar включає дані з більшості рецензованих онлайн-журналів найбільших наукових видавництв Європи та Америки. Система дозволяє здійснювати пошук повнотекстових журнальних статей, препринтів, дисертацій, монографій та інших документів. Користуватися базою даних Академії, як пошуковою системою, може безкоштовно будь-який користувач. Пошук статей в Google Scholar здійснюється такими самими способами, що й пошук в Google або будь-якій іншій пошуковій системі: шляхом уведенням пошукових слів (словосполучень) в рядок пошуку. Для входу в Google Академію необхідно ввести веб-адресу <https://scholar.google.com.ua/>. Викликається головне вікно системи (рис. 1.1)

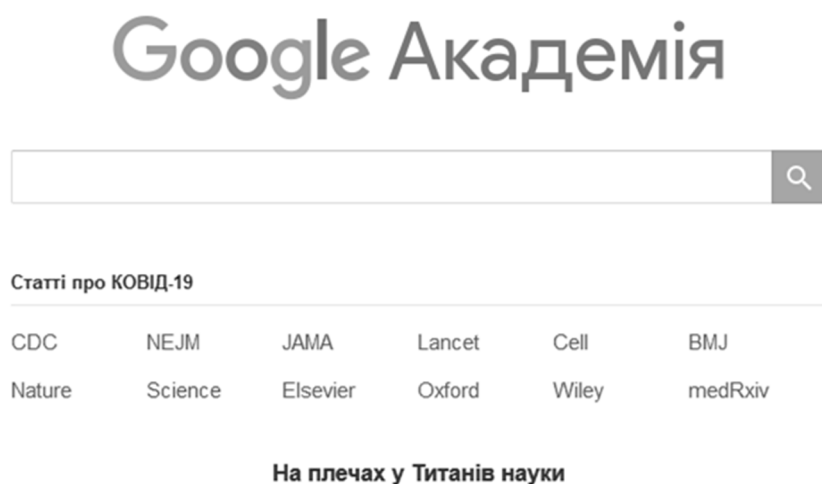


Рисунок 1.1 – Головне вікно Google Академії

Підберемо джерела по цифровим автоматам. У вікно пошуку вводимо «Цифрові автомати». Результат пошуку поданий на рисунку 1.2.

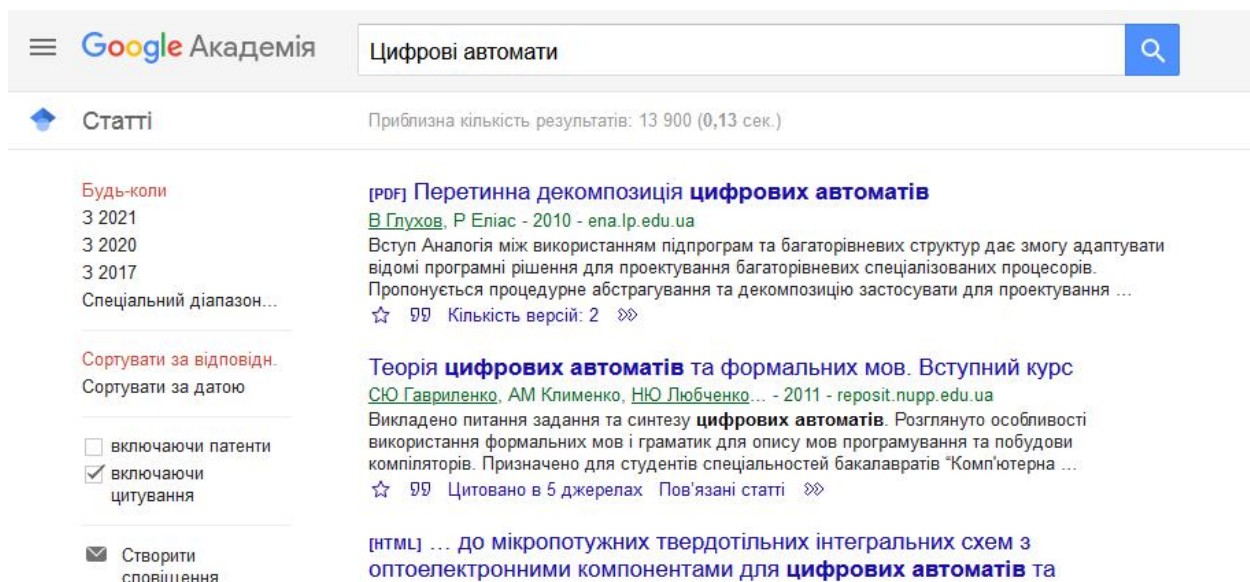


Рисунок 1.2 – Результат пошуку за запитом «Цифрові автомати»

Для сортування результатів пошуку використовують панель результатів, розташовану зліва від основного тексту. Усі посилання інтерактивні і, натиснувши на посилання, що нас цікавлять, потрапимо на веб-сторінку повнотекстової версії статті, якщо вона відкрита, або на сторінку онлайн-бібліотеки. Припустимо, нас зацікавив навчальний посібник «Прикладна теорія цифрових автоматів». Натиснувши на посилання, переходимо на опис навчального посібника (рис. 1.3).



Рисунок 1.3 – Сторінка навчального посібника

На цій сторінці можна переглянути посібник і навіть завантажити його.

У разі використання фрагментів із літературних джерел обов'язково посилатися на джерело, розмістивши його в квадратних дужках – []. Виконану роботу потрібно перевірити на плагіат за допомогою відповідного програмного забезпечення або інтернет-сервісів, наприклад UNICHECK.

2 ОСНОВНІ ПОНЯТТЯ СТОСОВНО ЦИФРОВОГО АВТОМАТА Й МЕТОДИКУ ЙОГО ПРОЄКТУВАННЯ

До складу процесора комп'ютера входять пристрої, які здійснюють перетворення дискретної інформації. Вони поділяються на два види – комбінаційні схеми та цифрові автомати. У комбінаційних схемах сукупність вихідних сигналів у будь-який момент часу однозначно визначається комбінацією вхідних сигналів, які надходять на вхід у той самий момент часу (рис. 2.1).



Рисунок 2.1 – Комбінаційна схема

Типовим прикладом комбінаційної схеми є дешифратор, який за кожної комбінації на вході схеми (а їх всього може бути 2^n), співставляє певну комбінацію унітарних сигналів на виході схеми. Приміром, якщо на вході використовується комбінація 011, то на виході може бути інформація у вигляді 00000100. Закон функціонування комбінаційної схеми визначено, якщо задано відповідність між її вхідними та вихідними сигналами. Це може бути так звана таблиця істинності, приклад якої наведено в таблиці 2.1.

Таблиця 2.1 – Приклад таблиці істинності

x_1	x_2	y_1	y_2	y_3	y_4
0	0	1	0	0	1
0	1	1	1	0	1
1	0	0	1	0	0
1	1	1	0	1	0

Закон функціонування комбінаційної схеми можна задати у вигляді набору булевих функцій: $y_i = f(x_1, x_2, \dots, x_n)$, $i = 1, 2, \dots, m$. Приміром, таблиці 2.1 буде відповідати такий набір булевих функцій:

$$y_1 = \bar{x}_1 + x_2, y_2 = \bar{x}_1 \cdot x_2 + x_1 \cdot \bar{x}_2, y_3 = x_1 \cdot x_2, y_4 = \bar{x}_1 \cdot \bar{x}_2 + \bar{x}_1 \cdot x_2.$$

Функція y_1 отримана як конститuent нуля за єдиною комбінацією, де функція набуває значення 0. Функція y_3 отримана як конститuent одиниці за єдиною комбінацією, де функція набуває значення 1. Функції y_2 та y_4 отримані як диз'юнкція конститuentів одиниці за комбінаціями, де функція набуває значення 1. Булеві функції дозволяють реалізувати комбінаційну схему у вигляді логічних схем, які є основою побудови цифрових схем.

Ще один спосіб завдання булевих функцій за кожною вихідною змінною комбінаційної схеми – це використання карт Карно, які дозволяють оптимізувати вираз для булевої функції. На рисунку 2.2 зображена карта Карно для булевої функції y_4 .

x_2	
x_1	1
	1
	0
	0

Рисунок 2.2 – Карта Карно для функції y_4

Накресливши контур, що охоплює сусідні клітки з одиницями, отримаємо оптимальний вираз для функції y_4 : $y_4 = \bar{x}_1$. Тепер можна скласти схему (рис. 2.3), що реалізує комбінаційну схему, задану таблицею істинності (табл. 2.1.).

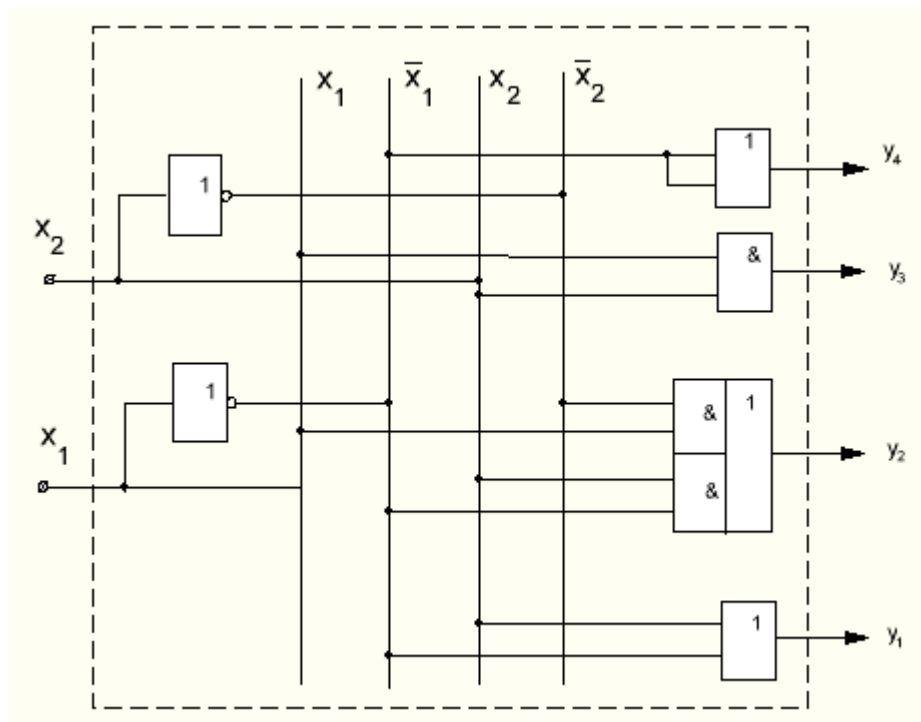


Рисунок 2.3 – Апаратна реалізація комбінаційної схеми

Інший, більш складний клас перетворювачів дискретної інформації становлять цифрові автомати [2]. Вони застосовуються в цифрових електронних пристроях для апаратної реалізації алгоритмів управління трактами передачі та обробки даних. Цифровий автомат, на відміну від комбінаційної схеми, має деяке кінцеве число різних внутрішніх станів. Під впливом вхідної інформації у вигляді вхідного двійкового слова певної розрядності цифровий автомат переходить з одного стану в інший і видає вихідну інформацію у вигляді вихідного двійкового слова певної розрядності. Вихідне слово на виході цифрового автомата в такті загалом визначається вхідним словом, яке надійшло у цей такт на вхід автомата, і внутрішнім станом автомата, який став результатом впливу на автомат вхідних слів у попередніх тактах. Комбінація вхідного слова і поточного стану автомата в цьому такті визначає не тільки вихідне слово, але й той стан, у який автомат перейде до початку наступного такту.

На рисунку 2.4 зображена типова структурна схема цифрового автомата. Цифровий автомат містить функціональний блок, що складається з двох комбінаційних схем далі – (КС), елементів затримки (DEL) і блоку пам'яті, що складається із запам'ятовувальних елементів – тригерів (Т).

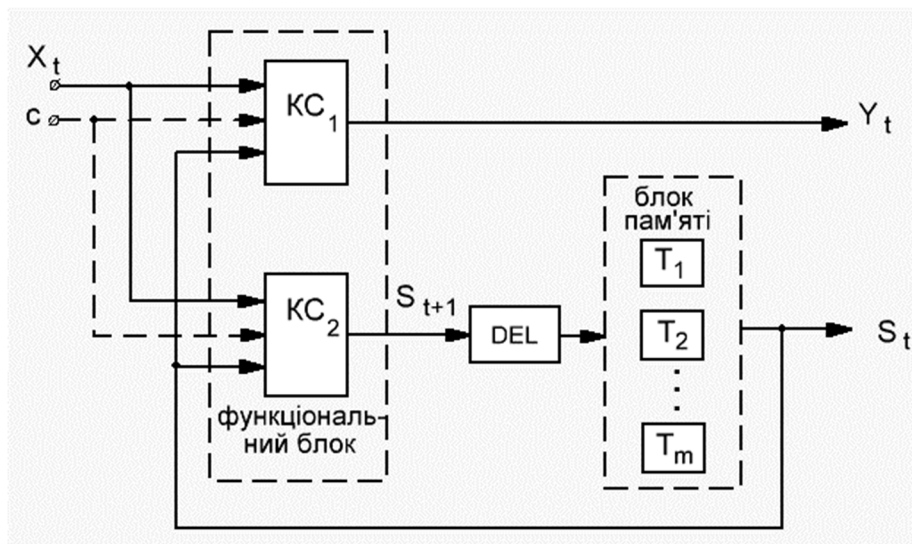


Рисунок 2.4 – Структурна схема цифрового автомата

Для завдання цифрового автомата повинні бути вказані:

- а) множина вхідних сигналів X_t ;
- б) множина вихідних сигналів Y_t ;
- в) множина станів S_t .

За наявності синхронізуючого сигналу C автомат працює в синхронному режимі і вироблення вихідного сигналу Y_t та інформації щодо нового стану автомата S_{t+1} здійснюється за наявності вхідного сигналу X_t і синхронізуючого сигналу. Після зняття сигналу C інформація про новий стан автомата S_{t+1} заноситься в блок пам'яті і автомат переходить в новий стан. Якщо синхронізуючий сигнал C не передбачений, то вироблення вихідного сигналу Y_t та інформація про новий стан автомата S_{t+1} здійснюється під час надходження вхідного сигналу X_t . Після зняття сигналу X_t автомат переходить в новий стан S_{t+1} ;

г) функції виходів $F(S, X)$ і переходів $G(S, X)$, які однозначно визначають залежність виходу автомата Y_t від стану автомата S_t і вхідного сигналу X_t в момент часу t і новий стан автомата в наступний момент часу $t+1$ від вхідного сигналу X_t та поточного стану автомата S_t . Під час практичної реалізації необхідно задати початковий стан автомата S_0 .

Використовуючи функції виходів і переходів, функціонування автомата можна описати за допомогою таких виразів:

$$Y(t) = F(S(t), X(t)), \quad S(t+1) = G(S(t), X(t)). \quad (2.1)$$

Вирази (2.1) описують так званий автомат Мілі. У комп'ютерах і пристроях автоматики широко застосовуються автомати Мура, у яких

вихідний сигнал Y_t в момент t залежить винятково від стану автомата, а стан автомата в момент часу $t+1$ залежить від стану автомата S_t і вхідного сигналу X_t в момент часу t . Надалі цей вид цифрових автоматів розглядатися не буде.

Опис цифрового автомата, тобто завдання функцій виходів і переходів, може здійснюватися по-різному. Найповніше автомат задається двома таблицями – таблицею виходів і таблицею переходів. Таблиця виходів кожному сполученню $\{x_i, s_j\}$ співставляє певне значення y_k . Приклад наведено в таблиці 2.2. Таблиця переходів кожному сполученню $\{x_i, s_j\}$ співставляє певне значення s_k , у яке автомат перейде в момент часу $t+1$. Приклад наведено в таблиці 2.3.

Таблиця 2.2 –Таблиця виходів автомата

Стани	Вхідні сигнали	
	X_1	X_2
s_1	y_3	y_1
s_2	y_2	y_4
s_3	y_3	y_3
s_4	y_4	y_2

Таблиця 2.3 – Таблиця переходів автомата

Стани	Вхідні сигнали	
	X_1	X_2
s_1	s_1	s_4
s_2	s_4	s_3
s_3	s_4	s_2
s_4	s_1	s_2

Зазвичай ці дві таблиці об'єднують (табл. 2.4).

Таблиця 2.4 – Сумісна таблиця переходів і виходів автомата

Стан	Вхідні сигнали	
	x_1	x_2
s_1	s_1/y_3	s_4/y_1
s_2	s_4/y_2	s_3/y_4
s_3	s_4/y_3	s_2/y_3
s_4	s_1/y_4	s_2/y_2

Якщо кількість станів невелика, цифровий автомат можна задавати за допомогою графа. Граф автомата – це зв'язний граф, вершини якого відповідають внутрішнім станам автомата, а дуги визначають переходи між станами. Дві вершини графа автомата s_i і s_j з'єднуються дугою, спрямованою від s_i до s_j , якщо в автоматі стан s_i переходить у стан s_j . Дузі графа

приписується вхідний сигнал x_k і вихідний сигнал y_m , як показано на рисунку 2.5.

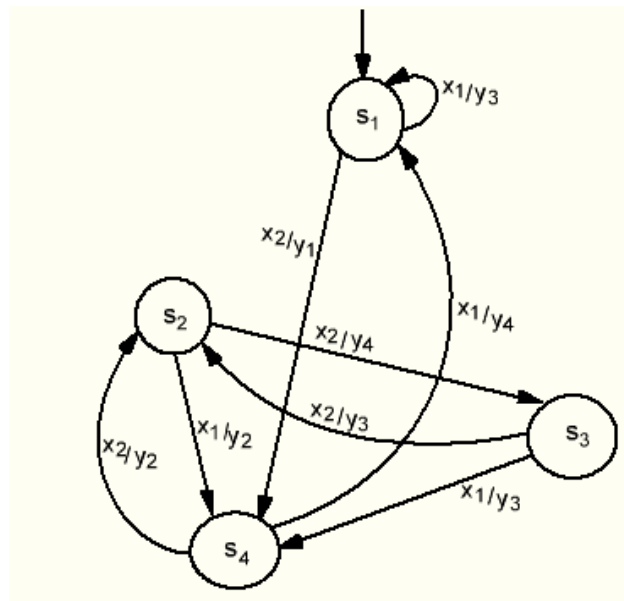


Рисунок 2.5 – Граф автомата за таблицею 2.4

Граф автомата зручно використовувати під час його програмної реалізації. Але якщо кількість станів велика він стає безмежним.

Процес розроблення автоматів можна розподілити на низку етапів.

На першому, попередньому етапі формуються, зазвичай словесно, умови роботи автомата, тобто визначаються умови його взаємодії з іншими пристроями або будь-якими об'єктами, виявляються необхідні вхідні й вихідні сигнали автомата, їхня кількість, визначається загальний закон появи вихідних сигналів залежно від сигналів на вході автомата.

На другому етапі розроблення формуються закони функціонування автомата, тобто визначаються функції переходів і виходів. Формальний опис автомата має подаватися за допомогою одного з прийнятих способів (матриця, граф). Цю етап прийнято називати етапом абстрактного синтезу, або синтезом абстрактного автомата. До уваги не береться яких конкретних числових значень можуть набувати вхідні й вихідні сигнали та елементи пам'яті. Важливо знати кількість різних можливих його внутрішніх станів, станів входів і виходів, а також закони змінювання внутрішнього стану автомата й вироблення вихідних сигналів під час надходження тієї чи іншої послідовності вхідних сигналів.

На третьому етапі розроблення автомата здійснюється кодування вхідних, вихідних сигналів і внутрішніх станів автомата. Після кодування внутрішнього стану автомата, станів входу й виходу складаються булеві

функції за кожним розрядом функцій виходу та нового стану автомата. Третій етап перебуває на границі абстрактного і структурного синтезу автомата.

На четвертому етапі розроблення цифрового автомата завершується вибір структури, будується так звана структурна схема, що складається з комбінаційної частини й пам'яті автомата, тобто відбувається структурний синтез автомата.

На п'ятому етапі здійснюється логічний синтез окремих блоків автомата за допомогою логічних елементів І, АБО, НІ та будується загальна логічна схема автомата.

Шостий етап розроблення автомата включає проведення електричного та інших розрахунків елементів схем, складання принципової схеми пристрою та моделювання роботи автомата з метою перевірки його працездатності. Далі здійснюється складання монтажних схем і технічної документації.

Перші п'ять етапів розроблення цифрового автомата прийнято називати логічним синтезом автомата (проектуванням). Шостий – технічним синтезом автомата.

Розглянемо процес розроблення цифрового автомата на прикладі синтезу послідовного двійкового суматора.

Робота однорозрядного двійкового суматора описується таблицею істинності (табл. 2.5).

Таблиця 2.5 – Таблиця істинності
для однорозрядного двійкового суматора

a_i	b_i	p_i	Σ_i	p_{i+1}
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

Однорозрядний двійковий суматор додає два відповідні розряди двох двійкових слів a_i і b_i , додаючи перенесення із молодшого розряду p_i , і формуючи часткову суму i -го розряду Σ_i і перенесення в наступний старший розряд p_{i+1} .

При паралельному додаванні двох слів суматор реалізується за допомогою комбінаційних схем [3]. У послідовному суматорі на нього подаються дві двійкові послідовності a і b молодшими бітами вперед плюс перенесення з молодшого розряду. На виході послідовно з'являються біти суми та біт перенесення в старший розряд. Часткову суму можна сформувати комбінаційною схемою, але вона залежить від перенесення з молодшого розряду, який треба зберігати окремо. Таким чином, функціонування послідовного суматора можна задати цифровим автоматом. За його стан можна прийняти значення перенесення, який має двійковий вигляд $p = \{0, 1\}$. Вхідний сигнал описується двовимірним двійковим вектором $X = (a, b)$. Вихідний сигнал – часткова сума може набувати двох значень $\Sigma = \{0, 1\}$.

Сформуємо структурну схему цифрового автомата, який реалізує двійковий послідовний суматор (рис. 2.6).

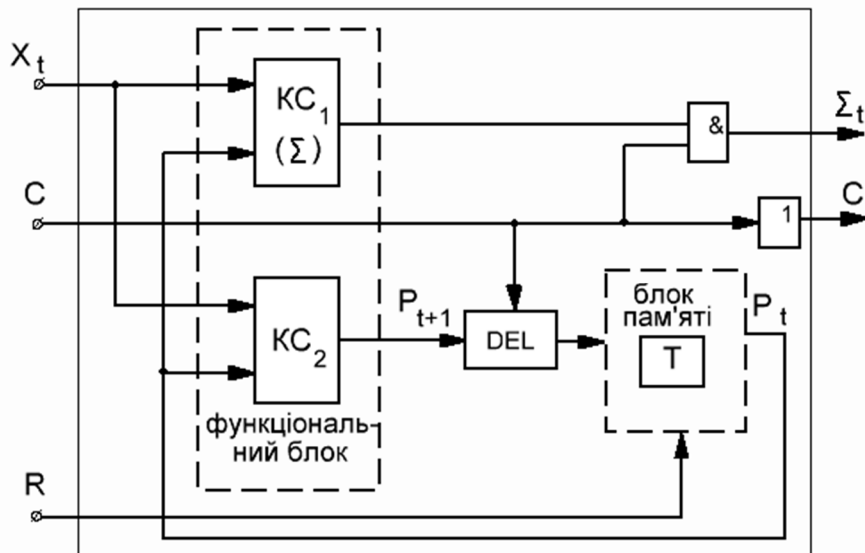


Рисунок 2.6 – Структурна схема цифрового автомата, який реалізує двійковий послідовний суматор

Робота двійкового послідовного суматора синхронізується синхроімпульсами C , які одночасно, разом із розрядами суми передаються в лінії передачі даних. Блок пам'яті можна реалізувати на двотактному D-тригері. Запис нового стану автомата в тригер здійснюється після передачі чергового розряду суми в лінії зв'язку. У цьому разі елемент затримки не потрібен. Для приведення суматора у вихідний нульовий стан використовується сигнал R .

Сформуємо таблицю переходів і виходів цифрового автомата, який реалізує двійковий послідовний суматор (табл. 2.6).

Таблиця 2.6 – Таблиця переходів і виходів автомата

Стан (P)	Вхідні сигнали (a, b)			
	0, 0	0, 1	1, 0	1, 1
0	0/0	0/1	0/1	1/0
1	0/1	1/0	1/0	1/1

Дроби інтерпретуються як p_{i+1}/Σ_i , де p_{i+1} – новий стан автомата (перенесення), Σ_i – часткова сума. Ця таблиця подібна до таблиці 2.5.

Сформуємо комбінаційні схеми КС₁, КС₂. Взагалі це можна зробити за таблицею переходів і виходів автомата (див. табл. 2.6). Використаємо досконалу диз'юнктивну нормальну форму, тобто формуємо значення функцій Σ_i і p_{t+1} за одиницям [4]:

$$\Sigma_t = \bar{a}b\bar{p}_t + a\bar{b}\bar{p}_t + \bar{a}bp_t + abp_t,$$

$$p_{t+1} = ab\bar{p}_t + \bar{a}bp_t + \bar{a}\bar{b}p_t + abp_t.$$

Використовуючи карти Карно, можна спростити ці вирази.

Карта Карно для булевої функції Σ_t виглядає так (рис. 2.7)

		p_t			
Σ_t	a	0	1	0	1
		1	0	1	0
			b		

Рисунок 2.7 – Карта Карно для функції Σ_t

Контури створити не можна, тому $\Sigma_t = \bar{a}b\bar{p}_t + a\bar{b}\bar{p}_t + \bar{a}bp_t + abp_t$.

Схема для формування функції Σ_t виглядає так (рис. 2.8).

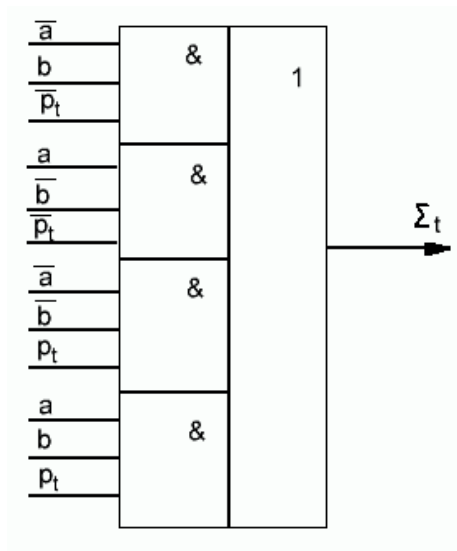


Рисунок 2.8 – Схема для формування функції Σ_t

Карта Карно для булевої функції p_{t+1} виглядає так (рис. 2.9)

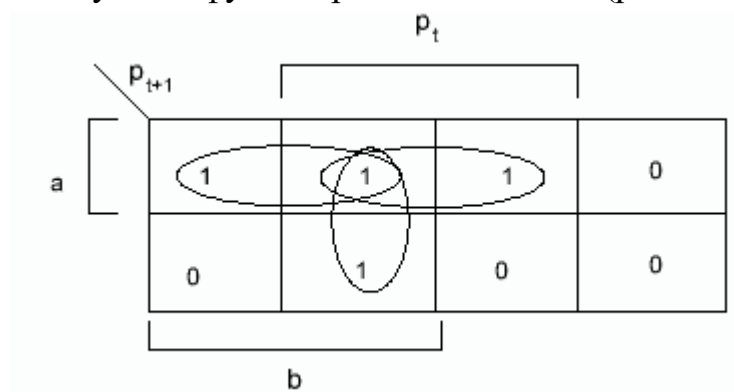


Рисунок 2.9 – Карта Карно для функції p_{t+1}

За нею можна скласти три контури для сусідніх одиниць. Оптимальний вираз для функції p_{t+1} виглядає так: $p_{t+1} = ab + ap_t + bp_t$. Схема для формування функції p_{t+1} виглядає так (рис. 2.10).

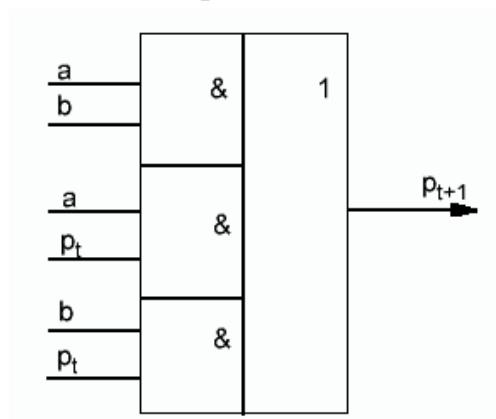


Рисунок 2.10 – Схема для формування функції p_{t+1}

Будуємо, згідно зі схемою (рис. 2.6), загальну логічну схему цифрового автомата, що реалізує послідовний двійковий суматор (рис. 2.11).

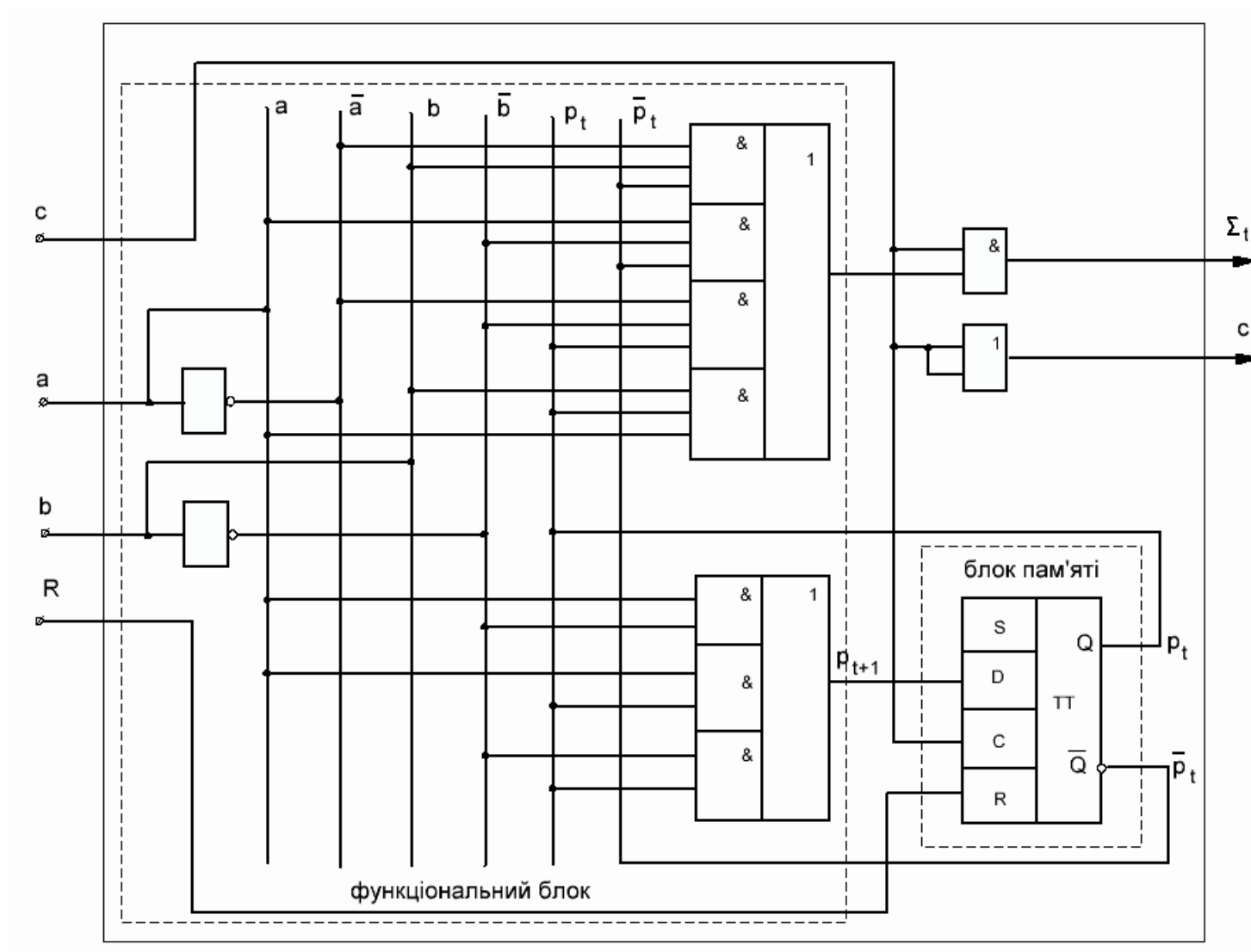


Рисунок 2.11 – Логічна схема послідовного двійкового суматора

Вхідний сигнал автомата X_t реалізується у вигляді відповідних розрядів доданків a і b , що надходять синхронно і послідовно на вхід суматора.

3 ЗАВДАННЯ ДЛЯ РОЗРАХУНКОВО-ГРАФІЧНОЇ РОБОТИ

На вхід цифрового автомата, який може знаходитися в одному з чотирьох станів $S1 - S4$, надходить сигнал X , який приймає значення 0 і 1. Під його впливом автомат переходить з одного стану в інший, при цьому виробляючи вихідний сигнал $Y1 - Y2$. Задана сумісна таблиця переходів и виходів автомата (табл. 3.1). Потрібно:

1. За таблицею 3.1 побудувати граф автомата Мілі.
2. Реалізувати автомат Мілі програмним способом.
3. Реалізувати автомат Мілі апаратним способом на рівні логічних елементів.

Таблиця 3.1 – Варіанти розрахунково-графічної роботи

Вар.1	Вхідні сигнали		Вар.2	Вхідні сигнали		Вар.3	Вхідні сигнали		Вар.4	Вхідні сигнали	
	x_1	x_2		x_1	x_2		x_1	x_2		x_1	x_2
Стан	0	1	Стан	0	1	Стан	0	1	Стан	0	1
	Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали	
S1	S1/Y3	S4/Y1	S1	S1/Y4	S2/Y1	S1	S4/Y1	S2/Y4	S1	S4/Y2	S2/Y
S2	S4/Y2	S3/Y4	S2	S4/Y2	S1/Y4	S2	S3/Y4	S2/Y2	S2	S1/Y3	S3/Y
S3	S4/Y3	S2/Y3	S3	S4/Y3	S2/Y4	S3	S4/Y1	S2/Y2	S3	S4/Y3	S1/Y
S4	S1/Y4	S2/Y2	S4	S3/Y3	S1/Y3	S4	S4/Y3	S1/Y2	S4	S3/Y2	S2/Y

Вар.5	Вхідні сигнали		Вар.6	Вхідні сигнали		Вар.7	Вхідні сигнали		Вар.8	Вхідні сигнали	
	x_1	x_2		x_1	x_2		x_1	x_2		x_1	x_2
Стан	0	1	Стан	0	1	Стан	0	1	Стан	0	1
	Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали	
S1	S4/Y4	S1/Y2	S1	S3/Y2	S4/Y2	S1	S4/Y2	S2/Y4	S1	S2/Y2	S2/Y1
S2	S4/Y1	S1/Y4	S2	S4/Y3	S1/Y4	S2	S1/Y3	S4/Y1	S2	S1/Y1	S3/Y4
S3	S1/Y2	S4/Y3	S3	S4/Y2	S2/Y3	S3	S4/Y3	S2/Y3	S3	S2/Y2	S4/Y1
S4	S3/Y3	S2/Y2	S4	S1/Y3	S2/Y2	S4	S3/Y2	S2/Y1	S4	S1/Y2	S4/Y3

Вар.9	Вхідні сигнали		Вар.10	Вхідні сигнали		Вар.11	Вхідні сигнали		Вар.12	Вхідні сигнали	
	x_1	x_2		x_1	x_2		x_1	x_2		x_1	x_2
Стан	0	1	Стан	0	1	Стан	0	1	Стан	1	1
	Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали	
S1	S2/Y2	S3/Y1	S1	S4/Y2	S2/Y4	S1	S2/Y2	S1/Y3	S1/Y3	S4/Y2	S1/Y4
S2	S1/Y4	S3/Y4	S2	S1/Y1	S4/Y3	S2	S1/Y2	S3/Y1	S4/Y2	S3/Y2	S1/Y4
S3	S1/Y2	S4/Y3	S3	S4/Y1	S1/Y2	S3	S1/Y1	S4/Y4	S4/Y3	S1/Y4	S4/Y2

S4	S3/Y1	S1/Y2	S4	S3/Y1	S1/Y3	S4	S2/Y2	S3/Y1	S1/Y4	S1/Y3	S2/Y1
----	-------	-------	----	-------	-------	----	-------	-------	-------	-------	-------

Продовження таблиці 3.1

Вар.13	Вхідні сигнали		Вар.14	Вхідні сигнали		Вар.15	Вхідні сигнали		Вар.16	Вхідні сигнали	
	x_1	x_2		x_1	x_2		x_1	x_2		x_1	x_2
Стан	0	1	Стан	0	1	Стан	0	1	Стан	0	1
	Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали	
S1	S2/y3	S1/y4	S1	S3/y3	S4/y1	S1	S3/y4	S2/y1	S1	S4/y4	S2/y1
S2	S1/y2	S4/y1	S2	S4/y2	S3/y1	S2	S4/y2	S1/y4	S2	S3/y2	S2/y4
S3	S2/y4	S1/y2	S3	S3/y4	S2/y3	S3	S3/y3	S2/y4	S3	S4/y1	S1/y3
S4	S3/y1	S1/y3	S4	S1/y4	S2/y2	S4	S3/y3	S1/y2	S4	S4/y3	S1/y2

Вар.17	Вхідні сигнали		Вар.18	Вхідні сигнали		Вар.19	Вхідні сигнали		Вар.20	Вхідні сигнали	
	x_1	x_2		x_1	x_2		x_1	x_2		x_1	x_2
Стан	0	1	Стан	0	1	Стан	0	1	Стан	0	1
	Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали	
S1	S4/y2	S2/y1	S1	S4/y2	S2/y1	S1	S4/y2	S3/y4	S1	S2/y2	S3/y1
S2	S1/y3	S3/y4	S2	S1/y3	S4/y4	S2	S2/y3	S1/y4	S2	S1/y1	S3/y4
S3	S4/y1	S1/y3	S3	S1/y1	S3/y3	S3	S4/y2	S2/y1	S3	S2/y2	S4/y1
S4	S3/y4	S2/y2	S4	S2/y4	S3/y2	S4	S1/y3	S3/y1	S4	S1/y2	S3/y1

Вар.21	Вхідні сигнали		Вар.22	Вхідні сигнали		Вар.23	Вхідні сигнали		Вар.24	Вхідні сигнали	
	x_1	x_2		x_1	x_2		x_1	x_2		x_1	x_2
Стан	0	1	Стан	0	1	Стан	0	1	Стан	1	1
	Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали	
S1	S4/Y2	S2/Y1	S1	S2/y2	S3/y3	S1	S4/y2	S2/y3	S1/Y ₂	S2/y2	S1/y
S2	S1/Y3	S4/Y4	S2	S1/y4	S4/y1	S2	S1/y1	S4/y4	S4/Y ₂	S1/y1	S3/y
S3	S4/Y2	S1/Y3	S3	S1/y2	S2/y3	S3	S4/y1	S2/y2	S4/Y ₂	S4/y1	S2/y
S4	S2/Y2	S3/Y1	S4	S3/y1	S1/y4	S4	S3/y1	S1/y4	S1/Y ₄	S2/y1	S3/y

Вар.25	Вхідні сигнали		Вар.26	Вхідні сигнали		Вар.27	Вхідні сигнали		Вар.28	Вхідні сигнали	
	x_1	x_2		x_1	x_2		x_1	x_2		x_1	x_2
Стан	0	1	Стан	0	1	Стан	0	1	Стан	0	1
	Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали			Нові стани автомата та вихідні сигнали	
S1	S2/y3	S1/y4	S1	S1/y2	S4/y1	S1	S3/y4	S2/y1	S1	S2/y1	S3/y4
S2	S3/y2	S4/y1	S2	S4/y2	S3/y1	S2	S4/y2	S1/y4	S2	S1/y2	S4/y1
S3	S2/y4	S1/y2	S3	S3/y4	S2/y3	S3	S3/3	S2/y4	S3	S2/y4	S1/y2
S4	S3/y1	S1/y3	S4	S1/y4	S2/y2	S4	S3/y3	S1/y2	S4	S3/y1	S1/y3

4 ТЕХНОЛОГІЯ ВИКОНАННЯ РОЗРАХУНКОВО-ГРАФІЧНОЇ РОБОТИ

Технологію виконання розрахунково-графічної роботи продемонструємо на такому прикладі. Задано таблицю переходів і виходів цифрового автомата (табл. 4.1).

Таблиця 4.1 – Таблицям переходів і виходів до прикладу

Вар. **	Вхідні сигнали	
	x_1	x_2
	0	1
Стан	Нові стани автомата та вихідні сигнали	
S1	S4/y4	S1/y1
S2	S4/y1	S1/y4
S3	S1/y2	S2/y3
S4	S3/y3	S2/y2

За таблицею переходів і виходів будуємо граф функціонування автомата (рис. 4.1).

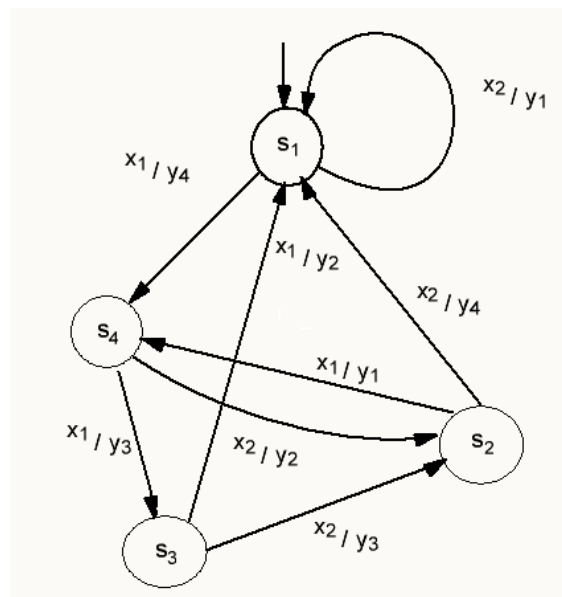


Рисунок 4.1 – Граф функціонування автомата

Будуємо блок-схему програми, що реалізує функціонування автомата. На рисунку 4.2 зображено блок-схему програми, що реалізує поведінку автомата. Топологія блок-схеми програми повторює топологію графа станів і

переходів кінцевого автомата. Натискання клавіші «1» імітує подачу сигналу «0» на вхід автомата. Натискання клавіші «2» імітує подачу сигналу «1» на вхід автомата. Натискання клавіші «3» приводить до виходу з програми. Для відображення стану автомата використовуються два вектори: $S_t[4]$ – поточний стан і $S_{t+1}[4]$ – наступний. S_i і S_{i_n} – компоненти цих векторів. Залежно від того, який сигнал надійшов на вхід, виконується та чи інша функція $y_1 - y_4$, і відбувається перехід до нового стану.

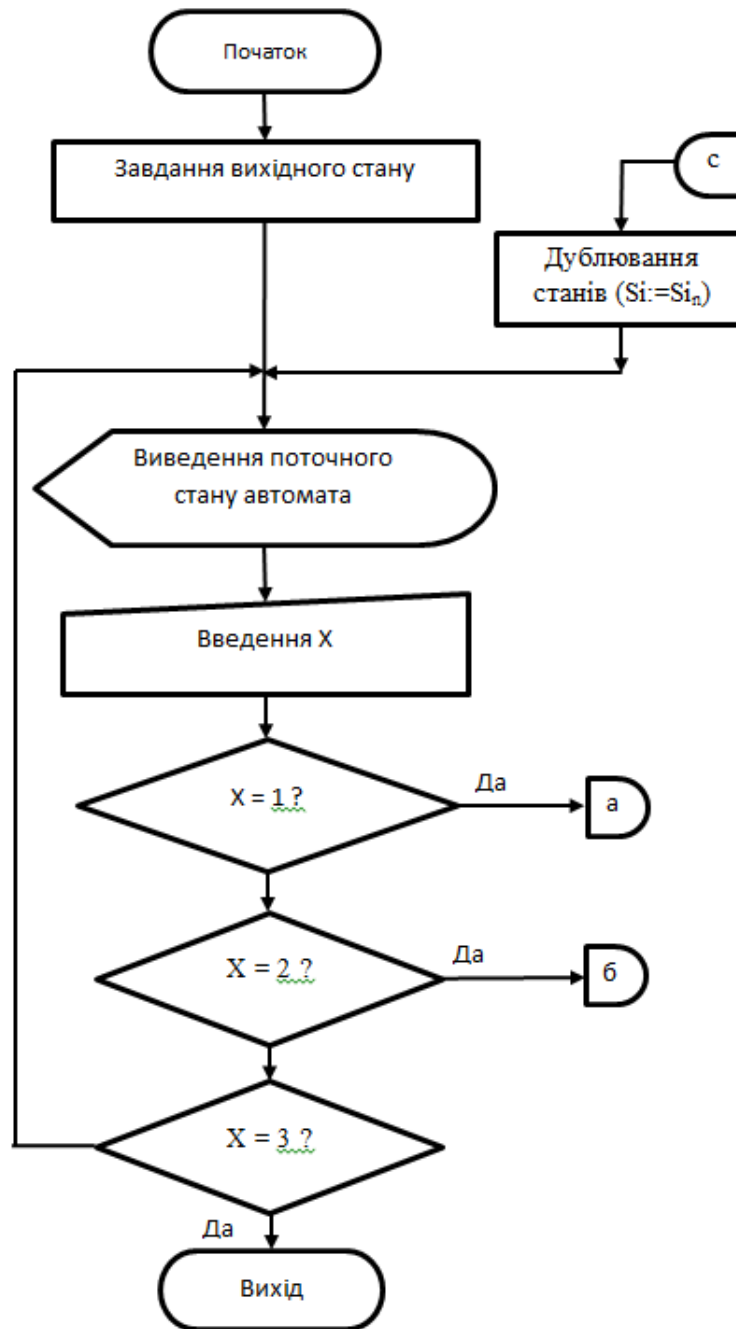
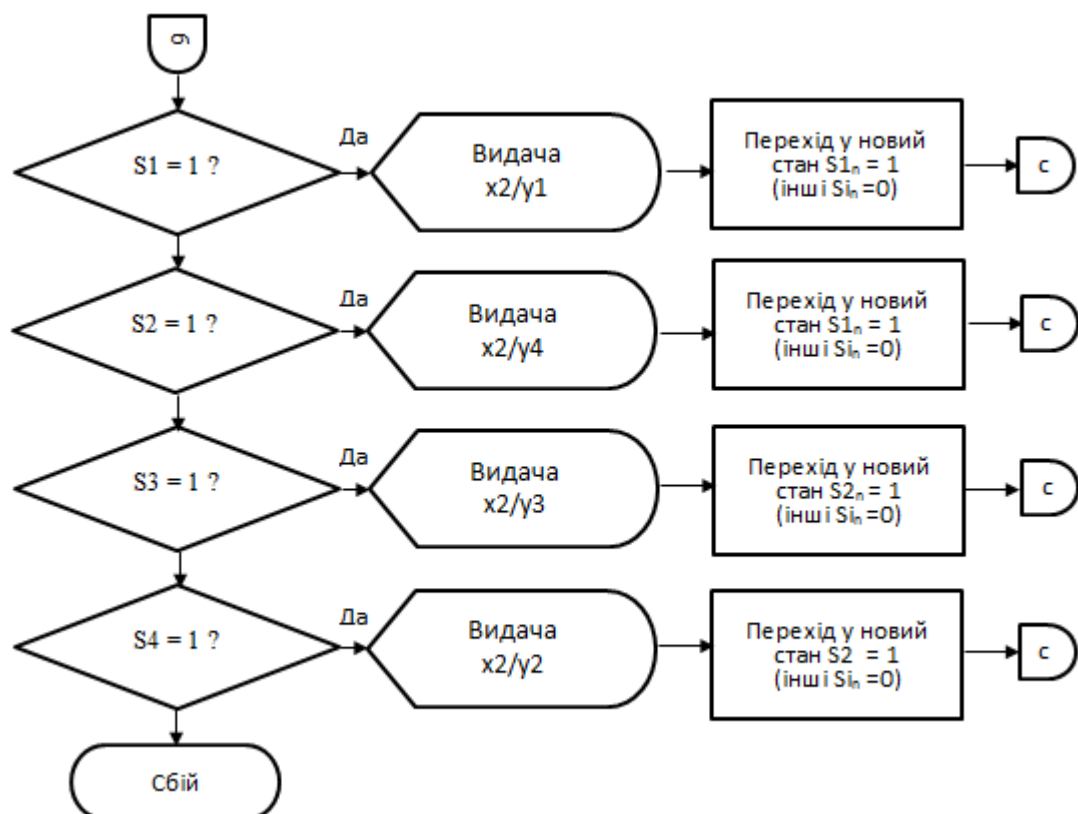
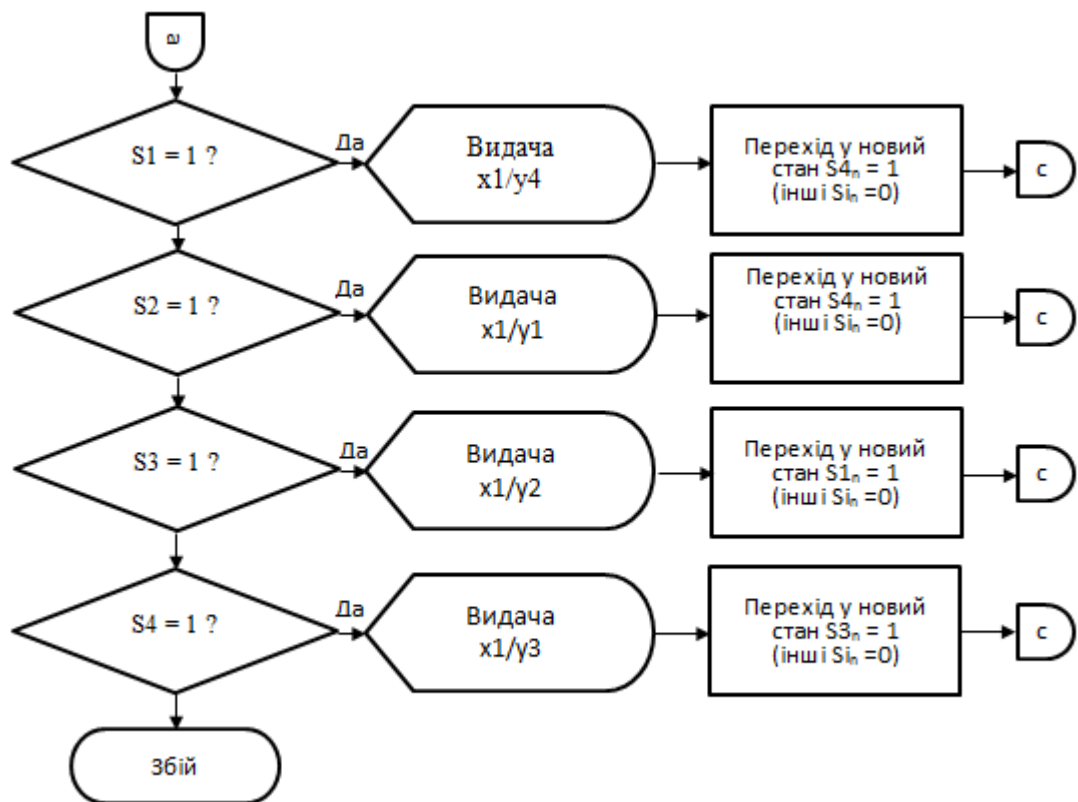


Рисунок 4.2 – Блок-схема алгоритму програмної реалізації автомата



Продовження рисунка 4.2

Для реалізації рекомендується обрати мову програмування Cі ++. Копія екрану функціонування відповідної програми може бути такою

(рис. 4.3). Для перевірки стану автомата можна використовувати функцію IF.

Динамічне відпрацювання автомата Міля			
Автомат у стані S1			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	1
Управління x1/y4 новий стан S4			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	1
Управління x1/y3 новий стан S3			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	1
Управління x1/y2 новий стан S1			
Автомат у стані S1			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	1
Управління x1/y4 новий стан S4			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	1
Управління x1/y3 новий стан S3			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	2
Управління x2/y3 новий стан S2			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	1
Управління x1/y1 новий стан S4			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	2
Управління x2/y2 новий стан S2			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	2
Управління x2/y4 новий стан S1			
Автомат у стані S1			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	2
Управління x2/y1 новий стан S1			
Автомат у стані S1			
Введіть значення	X	(x1 – 1, x2 – 2 або вихід – 3)	3

Рисунок 4.3 – Скриншот діалогу функціонування автомата

Важливий етап у розробці програми – її тестування. Потрібно переконатися, що програма однозначно імітує процес функціонування автомата і процес переходів зі стану в стан із видачею вихідних сигналів під

час подачі вхідних сигналів повністю співпадає з таблицею переходів і виходів.

Кодуємо вхідні, вихідні сигнали та внутрішні стани автомата двійковими векторами (числами) певної розрядності.

Вхід X : $x_1 \leftrightarrow 0$, $x_2 \leftrightarrow 1$ (однорозрядне кодування)

Поточний стан автомата кодуємо двовимірним булевим вектором $S_t(q_1, q_0)$. Його стани: $S1 \leftrightarrow 00$, тобто $q_1 = 0, q_0 = 0$, $S2 \leftrightarrow 01$, $S3 \leftrightarrow 10$, $S4 \leftrightarrow 11$. Значення q_1 і q_0 будуть передаватися під час апаратурної реалізації по окремим лініям. При подачі управляючого сигналу X автомат переходить у новий стан $S_{t+1}(Q1, Q0)$ згідно з функцією $S_{t+1}(Q1, Q0) = F1(X, S_t(q_1, q_0))$, яка залежить від значення вхідного сигналу X та попереднього стану автомата $S_t(q_1, q_0)$. Його стани: $S1_{t+1} \leftrightarrow 00$, $S2_{t+1} \leftrightarrow 01$, тобто $Q_1 = 0, Q_0 = 0$, $S3_{t+1} \leftrightarrow 10$, $S4_{t+1} \leftrightarrow 11$. Значення Q_1 і Q_0 також будуть передаватися при апаратурній реалізації по окремим лініям.

Вихідні сигнали автомата кодується двовимірним булевим вектором $Y(Y_1, Y_0)$, формування якого здійснюється згідно з функцією $Y(Y_1, Y_0) = F2(X, S_t(q_1, q_0))$, яка залежить від значення вхідного сигналу X та попереднього стану автомату $S_t(q_1, q_0)$. Стани вектора $Y(Y_1, Y_0)$: $y1 \leftrightarrow 00$, тобто $Y_1 = 0, Y_0 = 0$, $y2 \leftrightarrow 01$, $y3 \leftrightarrow 10$, $y4 \leftrightarrow 11$. Для перетворення кодових сигналів в унітарні використовуємо дешифратори. Структурна схема розроблюваного автомата зображена на рисунку 4.4.

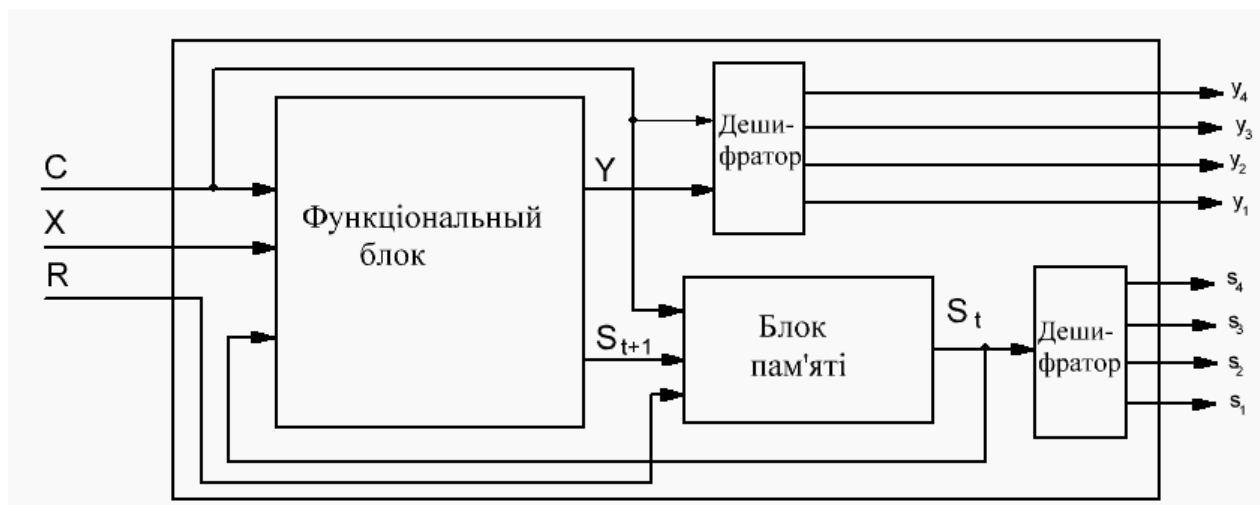


Рисунок 4.4 – Структурна схема автомата

Значення сигналів та модулів на рисунку 4.4 такі: X – вхідний сигнал, приймає значення або «0», або «1»; C – синхросигнал у вигляді логічної одиниці. Усі події в автоматі відбуваються тільки під дією цього сигналу. R (reset) – сигнал у вигляді логічної одиниці. Призначений для приведення автомата до початкового стану $S_1 - S_t(0,0)$. Блок пам'яті, реалізований на тригерах, зберігає поточний стан автомата – $S_t(q_1, q_0)$. Функціональний блок призначений для вироблення вихідного сигналу Y в кодованому вигляді і код нового стану автомата – S_{t+1} . Дешифратори призначені для перетворення кодових сигналів на унітарні.

Автомат працює так чином. Подача логічної одиниці на вхід R призводить до встановлення блока пам'яті автомата у початковий стан $S_t(0,0)$. Надалі блок пам'яті зберігає поточний стан автомата $S_t(q_1, q_0)$. З виходу блока пам'яті код стану автомата (q_1, q_0) надходить на вхід дешифратора, який виробляє унітарний сигнал S_i відповідно до коду, який надходить на його вхід. Водночас поточний код стану автомата надходить на вхід функціонального блока. При подачі вхідного сигналу X та синхросигналу C на вхід автомата функціональний блок залежно від поточного стану автомата виробляє код вихідного сигналу Y і код нового стану автомата S_{t+1} . Код вихідного сигналу Y надходить на вхід дешифратора, який виробляє унітарний сигнал Y_i відповідно до коду (Y_1, Y_0) , який надходить на його вхід. Після зняття синхросигналу C код нового стану автомата записується в пам'ять автомата.

Будуємо кодовану таблицю переходів і виходів автомата Міля відповідно до таблиці 4.1 (табл. 4.2).

Таблиця 4.2 – Таблицю переходів і виходів автомата Міля

X		Поточний внутрішній стан			Наступний внутрішній стан			Вихідні сигнали		
		S_t	q_1	q_0	S_{t+1}	Q1	Q0	Y	Y1	Y0
x_1	0	s1	0	0	s4	1	1	y4	1	1
x_1	0	s2	0	1	s4	1	1	y1	0	0
x_1	0	s3	1	0	s1	0	0	y2	0	1
x_1	0	s4	1	1	s3	1	0	y3	1	0
x_2	1	s1	0	0	s1	0	0	y1	0	0
x_2	1	s2	0	1	s1	0	0	y4	1	1
x_2	1	s3	1	0	s2	0	1	y3	1	0
x_2	1	s4	1	1	s2	0	1	y2	0	1

Будуємо аналітичні вирази для булевих функцій Q0, Q1, Y0, Y1 щодо вхідних змінних X, q_1, q_0 . Для цього використовуємо карти Карно

Карта Карно для булевої функції Q0 виглядає так (рис. 4.5).

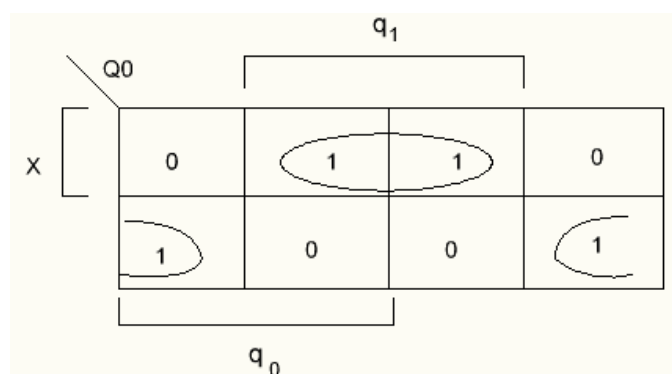


Рисунок 4.5 – Карта Карно для функції Q0

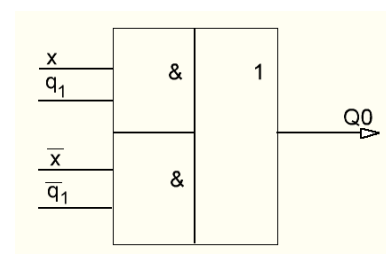


Рисунок 4.6 – Схема для формування функції Q0

Технологія її побудови наступна. Q0 залежить від трьох змінних: X, q_1, q_0 і тому містить $2^3 = 8$ клітинок. Кожна змінна поділяє клітинки таблиці на дві рівні частини: в одній вона набуває значення 0, а в другій – 1. Клітинки, у яких будь яка змінна набуває значення 1 позначені збоку квадратною дужкою. Кожній клітині таблиці відповідає відповідна комбінація змінних X, q_1, q_0 і в ній записано значення Q0 відповідно до таблиці 4.2. За сусідніми клітинками, які містять одиниці, будуємо контури [4]. Сусідніми називаються клітинки, які відрізняються одна від одної значенням однієї

змінної. Кожен контур містить дві або чотири одиниці (у нас – дві). Ті вхідні змінні, які входять у координати цього контуру без інверсій, включаються в підсумкову кон'юнкцію безпосередньо, а ті змінні, які входять в координати цього контуру спільно зі своїми інверсіями, виключаються з підсумкової кон'юнкції. Вираз для функції Q_0 запишеться, як диз'юнкція отриманих кон'юнкцій так: $Q_0 = x \& q_1 \cup \bar{x} \& \bar{q}_1$. $\&$ – знак операції кон'юнкції (булева операція І), \cup – знак операції диз'юнкції (булева операція АБО). Апаратна реалізація для функції Q_0 шляхом використання схеми 2І-АБО зображена на рисунку 4.6.

Аналогічно будемо карту Карно для булевої функції Q_1 (рис. 4.7).

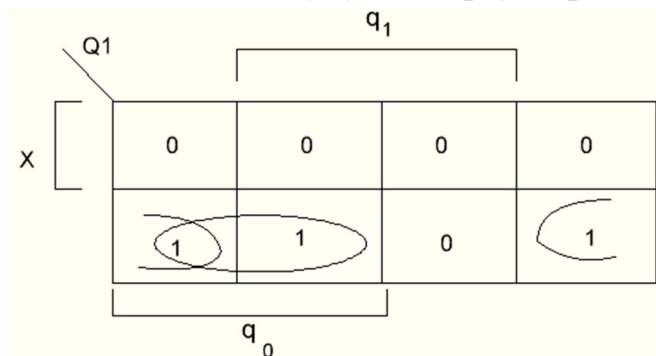


Рисунок 4.7 – Карта Карно для функції Q_1

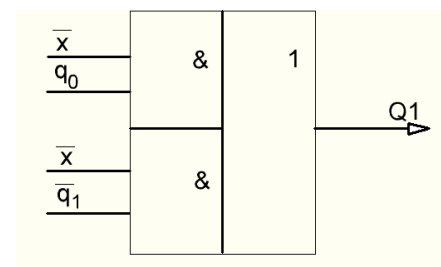


Рисунок 4.8 – Схема для формування функції Q_1

Вираз для функції Q_1 запишеться так: $Q_1 = \bar{x} \& \bar{q}_1 \cup \bar{x} \& q_0$. Апаратна реалізація для функції Q_1 шляхом використання схеми 2І-АБО буде така (рис. 4.8)

Будемо карту Карно для булевої функції Y_0 (рис. 4.9).

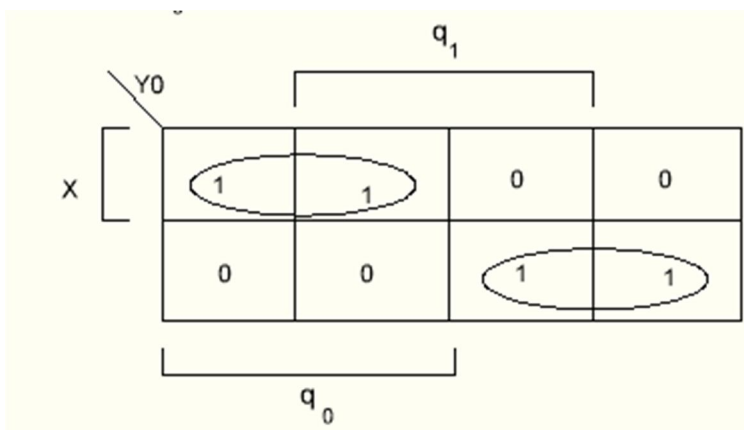


Рисунок 4.9 – Карта Карно для функції Y_0

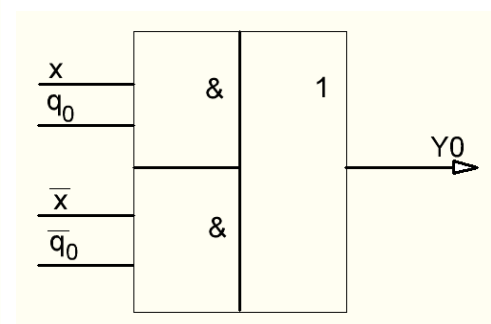


Рисунок 4.10 – Схема для формування функції Y_0

Вираз для функції Y_0 запишеться так: $Y_0 = x \& q_0 \cup \bar{x} \& \bar{q}_0$. Апаратна реалізація для функції Y_0 шляхом використання схеми 2І-АБО зображена на рисунку 4.10.

Будуємо карту Карно для булевої функції Y_1 (рис. 4.11).

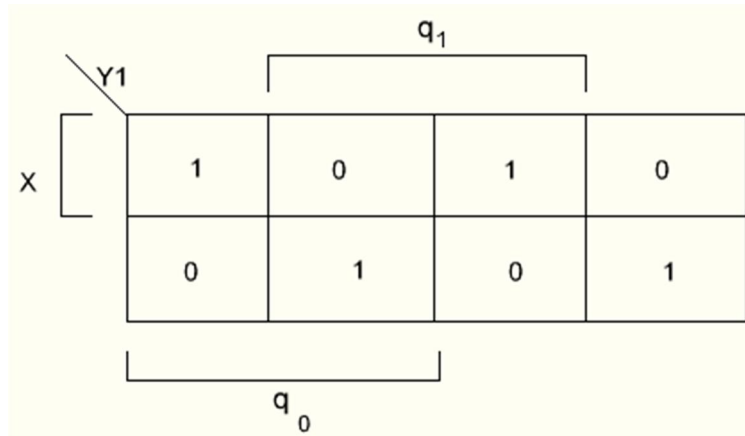


Рисунок 4.11 – Карта Карно для функції Y_1

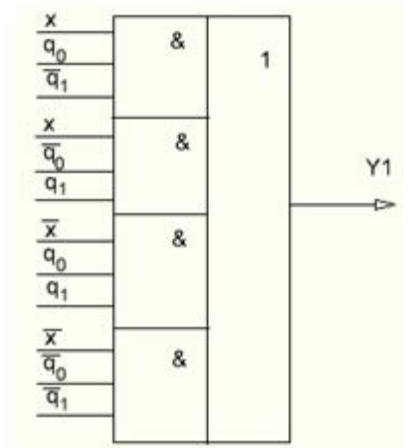


Рисунок 4.12 – Схема для формування функції Y_1

Із аналізу карти Карно для функції Y_1 випливає, що не можна побудувати жодного контуру, який об'єднував клітинки з одиницями або нулями. Тому вираз для булевої функції Y_1 сформується як диз'юнкція конститuentів одиниці за кожною клітинкою, яка містить одиницю, тобто $Y_1 = x \& q_0 \& \bar{q}_1 \cup x \& \bar{q}_0 \& q_1 \cup \bar{x} \& q_0 \& q_1 \cup \bar{x} \& \bar{q}_0 \& \bar{q}_1$.

Апаратна реалізація для функції Y_1 шляхом використання схеми 4І-АБО зображена на рисунку 4.12.

Блок пам'яті сформуємо з двох тригерів, які забезпечують зберігання чотирьох станів автомата. Як тригери застосовуємо двотактні синхронні D-тригери. Умовне графічне позначення такого тригера приведено на рисунку 4.13.

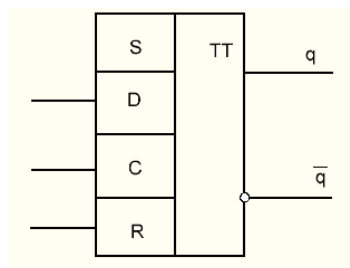


Рисунок 4.13 – Умовне графічне позначення двотактного D-тригера

Інформація за входом D приймається у D-тригер під час дії синхросигнала C і з'являється на його виходах у протифазі після зняття

сигналу С. Вхід R використовується для приведення тригера до початкового стану – нульового.

Спроекуємо дешифратор для перетворення кодованого вихідного сигналу автомата в унітарний двійковий код. Для цього складемо об'єднану таблицю істинності для всіх виходів $y_1 - y_4$ (табл. 4.3).

Таблиця 4.3 – Таблиця істинності для формування вихідних сигналів

Y1	Y0	En	y_1	y_2	y_3	y_4
0	0	0	0	0	0	0
0	1	0	0	0	0	0
1	0	0	0	0	0	0
1	1	0	0	0	0	0
0	0	1	1	0	0	0
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	1

Сигнал En (enable) дозволяє сформувати вихідні сигнали. За таблицею 4.3 формуємо вирази для функцій $y_1 - y_4$. Запишемо їх як конституенти одиниці:

$$y_1 = \bar{Y}_1 \& \bar{Y}_0 \& En, \quad y_2 = \bar{Y}_1 \& Y_0 \& En, \quad y_3 = Y_1 \& \bar{Y}_0 \& En, \quad y_4 = Y_1 \& Y_0 \& En.$$

Таким чином, для формування дешифратора потрібно два елемента НІ і чотири тривходові елементи І. За отриманими виразами для $y_1 - y_4$ складаємо схему дешифратора (рис. 4.14).

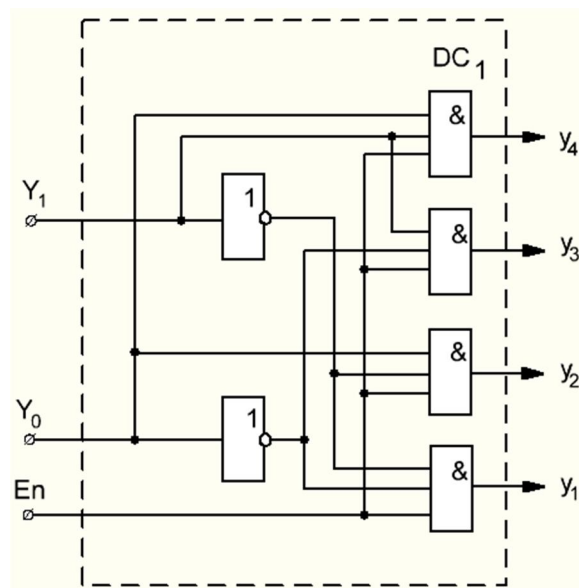


Рисунок 4.14 – Схема дешифратора для формування сигналів $y_1 - y_4$

Дешифратор для перетворення кодованого стану автомата S в унітарний двійковий код буде відрізнятися від дешифратора DC₁ відсутністю

дозвільного сигналу En . Таким чином, у ньому замість тривходових елементів І будуть використовуватися двовходові елементи І. Його схема наведена на рисунку 4.15.

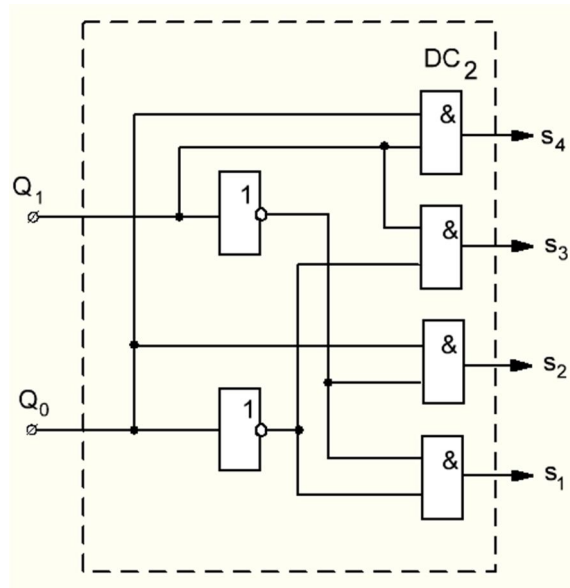


Рисунок 4.15 – Схема дешифратора для формування сигналів $s_1 - s_4$

Об'єднуємо окремі модулі в загальний пристрій згідно структурної схемою автомата (рис. 4.16).

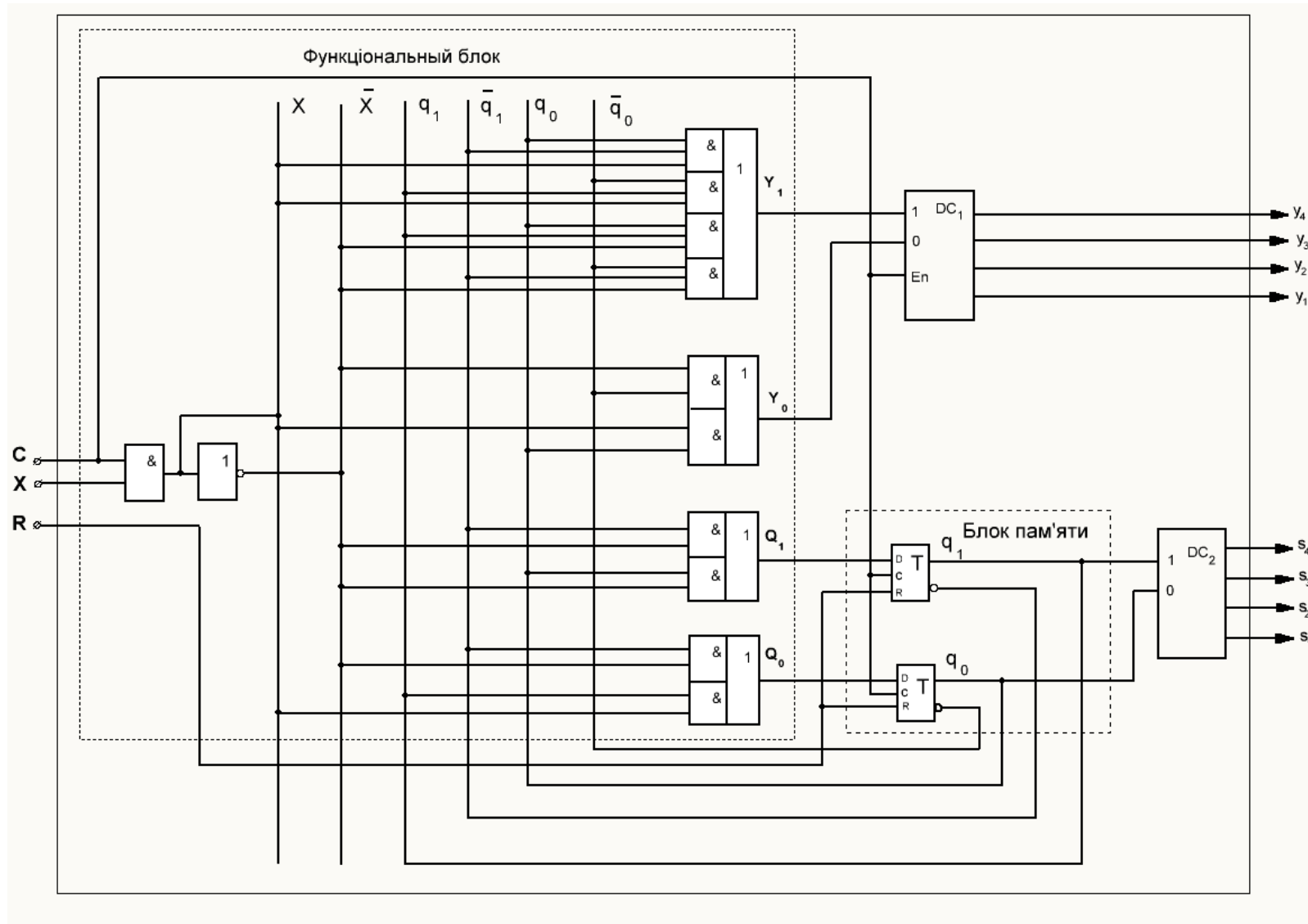


Рисунок 4.16 – Загальна схема цифрового автомата

СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

1. Розрахунково-графічна робота з дисципліни «Медична інформатика». URL: <https://studfile.net/preview/2284922/>.
2. Каган Б. М. Электронные вычислительные машины и системы : учеб. пособие для вузов / Б. М. Каган. – М.: Энергоатомиздат, 1991. – 592 с.
3. Таненбаум Э. Архитектура компьютера / Э. Таненбаум, Т. Остин. – СПб. : Питер, 2013. – 816 с.
4. Практикум з архітектури комп'ютерних систем : навч. посібник / А. Л. Литвинов ; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2020. – 68 с.

ДОДАТОК А

Зразок оформлення титульного аркуша розрахунково-графічної роботи

**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МІСЬКОГО ГОСПОДАРСТВА імені О. М. БЕКЕТОВА**

Навчально-науковий інститут енергетичної, інформаційної та транспортної
інфраструктури

Кафедра комп'ютерних наук та інформаційних технологій

**РОЗРОБКА І ОПТИМІЗАЦІЯ МОДУЛЯ КОМП'ЮТЕРНОЇ СИСТЕМИ
НА БАЗІ ЦИФРОВОГО АВТОМАТА**

розрахунково-графічна робота

з дисципліни «Архітектура комп'ютерних систем»

Варіант 17

Виконав:

студент 2 курсу групи КН 2019-1

Кушнар'єв А. І.

Керівник:

проф. Литвинов А. Л.

РГР захищена

2021 р.

м. Харків – 2021 рік

ДОДАТОК Б

Зразок оформлення змісту розрахунково-графічної роботи

ЗМІСТ

Вступ	3
Завдання для РГР	4
1 Структурний синтез автомата	5
2 Програмна реалізація автомата.....	8
3 Апаратна реалізація автомата	12
Висновки	14
Список літератури	15

Виробничо-практичне видання

Методичні рекомендації

для самостійної і виконання розрахунково-графічної робіт

із навчальної дисципліни

«АРХІТЕКТУРА КОМП'ЮТЕРНИХ СИСТЕМ»

(для студентів 2 курсу денної і заочної форм навчання

першого (бакалаврського) рівня вищої освіти

спеціальностей 122 – Комп'ютерні науки,

126 – Інформаційні системи та технології,

151 – Автоматизація та комп'ютерно-інтегровані технології

Укладач **ЛИТВИНОВ** Анатолій Леонідович

Відповідальний за випуск *М. М. Булаєнко*

Редактор *О. А. Норик*

Комп'ютерне верстання *А. Л. Литвинов*

План 2021, поз. 292М

Підп. до друку 04.06.2021. Формат 60 × 84/16.

Друк на ризографі. Ум. друк. арк. 2,2.

Тираж 60 пр. Зам. №

Видавець і виготовлювач:

Харківський національний університет
міського господарства імені О. М. Бекетова,

вул. Маршала Бажанова, 17, Харків, 61002.

Електронна адреса: rectorat@kname.edu.ua.

Свідоцтво суб'єкта видавничої справи:

ДК № 5328 від 11.04.2017.