

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕКОНОМІКИ ТА ПІДПРИЄМНИЦТВА
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ЕКОНОМІЧНОЇ КІБЕРНЕТИКИ



С. М. ЦИРУЛЬНИК

КОМП'ЮТЕРНА СХЕМОТЕХНІКА ТА АРХІТЕКТУРА

КОМП'ЮТЕРІВ

Методичні вказівки
для проведення практичних занять
та самостійної роботи
для здобувачів вищої освіти
першого (бакалаврського) рівня
галузі знань 12 «Інформаційні технології»
спеціальності 122 «Комп'ютерні науки»
денної та заочної форми навчання

ВІННИЦЯ
2021

Цирульник С. М. Комп'ютерна схемотехніка та архітектура комп'ютерів. Методичні вказівки для проведення практичних занять та самостійної роботи для здобувачів вищої освіти першого (бакалаврського) рівня галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки» денної та заочної форми навчання/ С. М. Цирульник. – Вінниця: РВВ ВНАУ, 2021. – 146 с.

Автори: Цирульник С. М., к.т.н., доцент кафедри комп'ютерних наук та економічної кібернетики

Затверджено до видання науково-методичною комісією ВНАУ (протокол №4 від 15.10.21р.) за поданням навчально-методичної комісії факультету економіки та підприємництва (протокол №3 від 12.10.21р.)

Методичні вказівки є навчально-методичним документом, який містить рекомендації для вивчення дисципліни «Комп'ютерна схемотехніка та архітектура комп'ютерів», виконання практичних робіт та самостійної роботи з метою оволодіння та закріплення знань у студентів в галузі інформаційних технологій з принципами функціонування схем та елементів комп'ютерної схемотехніки та архітектури комп'ютерів, а також методами їх розрахунку, аналізу, синтезу та організації взаємодії. Призначені для використання студентами галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки» першого (бакалаврського) освітнього рівня денної та заочної форми навчання.

Зміст

	стор.
Вступ	4
Опис навчальної дисципліни	7
Орієнтовна структура навчальної дисципліни	8
Організація самостійної роботи студента	9
Вимоги до оформлення практичних робіт	10
Практична робота № 1. Дослідження роботи електронної схеми, отриманої комбінуванням логічних елементів	12
Практична робота № 2. Мінімізація логічних функцій за допомогою карт Карно	24
Практична робота № 3. Дослідження схем тригерів	31
Практична робота № 4. Дослідження регістрів	42
Практична робота № 5. Дослідження лічильників імпульсів	49
Практична робота № 6 Дослідження основних комбінаційних пристроїв	61
Практична робота № 7. Вивчення архітектури персонального комп'ютера за допомогою програми AIDA64	76
Практична робота № 8. Дослідження роботи комп'ютера. Емулятор EMU8086	86
Практична робота № 9. Складання комп'ютера	94
Практична робота № 10. Дослідження порядку запуску комп'ютера	103
Перелік питань для самостійної роботи студента	117
Тести для самоконтролю	119
Критерії оцінки знань студентів	139
Рекомендовані джерела інформації	140
Додаток А. Приклад оформлення титульної сторінки звіту	143

ВСТУП

Успішне використання потужного комп'ютеризованого засобу неможливо без чіткого уявлення особливостей функціонування всіх його складових частин, а це, в свою чергу, вимагає твердих знань фізичних процесів, що відбуваються на рівні схемотехнічного подання структури елементів та вузлів комп'ютерів під час їхньої роботи.

Знання основ побудови комп'ютерної схемотехніки та архітектури комп'ютерів стає все далі актуальнішим, оскільки тенденції розвитку комп'ютерної техніки свідчать про те, що з одного боку складність та функціональні можливості комп'ютерної техніки постійно і швидко зростають, а з другого боку, спостерігається постійна тенденція до персоніфікації цієї складної техніки. Тобто задача підтримки персонального комп'ютера в працездатному стані, налагоджування його роботи та конфігурації, своєчасна модернізація, ремонт та обслуговування все далі стає проблемою не професіоналів-фахівців, а конкретного користувача цього персонального комп'ютера.

Сучасну комп'ютерну схемотехніку складають елементи та вузли, які за способами подання та оброблення сигналів діаметрально відрізняються один від одного. Так до елементів аналогової електроніки відносяться ті електронні засоби, які призначені для перетворення і оброблення інформації що змінюється за законом безперервної функції, а до елементів цифрової електроніки відносяться ті засоби для перетворення і оброблення інформації яка змінюється за законом дискретної функції. Сучасні інформаційні технології у повній мірі використовують як аналогові так і цифрові схемотехнічні рішення для оброблення сигналів.

Мета дисципліни «Комп'ютерна схемотехніка та архітектура комп'ютерів» - засвоєння необхідних знань з основ теорії побудови та функціонування основних пристроїв, вузлів, базових елементів та архітектури сучасної комп'ютерної техніки, що виконані на базі інтегральної технології, формування твердих практичних навичок щодо оцінювання технічного стану

комп'ютерної техніки, розрахунків параметрів аналогових та цифрових схем, аналізу умов функціонування та синтезу схем з заданими характеристиками, а також підготовка висококваліфікованих спеціалістів, які вміють раціонально вибирати та використовувати сучасні типи комп'ютерів в умовах автоматизованого проектування; аналізувати, розраховувати, синтезувати та проектувати цифрові електронні пристрої, які використовуються в комп'ютерних та мікропроцесорних системах..

Завданням дисципліни є: надати інформацію з арифметичних основ схемотехніки комп'ютера; надати інформацію з логічних основ схемотехніки комп'ютера; навчити студентів методам мінімізації логічних функцій; надати інформацію з внутрішнього устрою персонального комп'ютера; надати інформацію з устрою і принципу функціонування периферійного обладнання; навчити студентів приймати конструктивні рішення з розробки блоків арифметичного логічного пристрою.

Практичні роботи виконуються згідно з програмою дисципліни «Комп'ютерна схемотехніка та архітектура комп'ютерів» здобувачами вищої освіти галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки» першого (бакалаврського) освітнього рівня з метою напрацювання у студентів прикладних вмінь і навичок та закріплення теоретичних знань щодо побудови та функціонування основних пристроїв, вузлів, базових елементів та архітектури сучасної комп'ютерної техніки.

Загальною метою практичних робіт є поглиблення та закріплення знань з розділів дисципліни «Комп'ютерна схемотехніка та архітектура комп'ютерів».

У результаті вивчення дисципліни «Комп'ютерна схемотехніка та архітектура комп'ютерів» у студентів мають сформуватися такі програмні компетентності:

Інтегральні компетентності	Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів комп'ютерних наук, інформаційних
----------------------------	---

Загальні компетентності	<p>технологій</p> <ul style="list-style-type: none"> – здатність ідентифікувати, формулювати та розв’язувати задачі; – здатність генерувати нові ідеї (креативність); – здатність спілкуватися з нефахівцями однієї галузі; – здатність до абстрактного мислення, аналізу та синтезу
Спеціальні компетентності	<ul style="list-style-type: none"> – здатність опанувати сучасні технології математичного моделювання об’єктів, процесів і явищ, розробляти обчислювальні моделі та алгоритми чисельного розв’язання задач математичного моделювання з урахуванням похибок наближеного чисельного розв’язання професійних задач; – здатність застосовувати теоретичні та практичні основи методології та технології моделювання, реалізовувати алгоритми моделювання для дослідження характеристик і поведінки складних об’єктів і систем, проводити експерименти за програмою моделювання з обробкою й аналізом результатів

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Галузь знань, спеціальність, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
<p>Кількість кредитів – 5</p> <p>Атестацій– 2</p> <p>Загальна кількість годин – 150</p> <p>Тижневих годин для денної форми навчання: аудиторних – 2 самостійної роботи студента – 8</p>	<p>Галузь знань</p> <p>12 «Інформаційні технології»</p> <p>Спеціальність</p> <p>122 «Комп'ютерні науки»</p> <p>Перший (бакалаврський) освітній рівень</p>	Нормативна	
		Рік підготовки:	
		2-й	3, 4-й
		Семестр	
		3-й	6, 7-й
		Лекції	
		16 год.	4 год.
		Практичні, семінарські	
		14 год.	2 год.
		Лабораторні	
		Самостійна робота	
		120 год.	144 год.
		Вид контролю: залік	

ОРІЄНТОВНА СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назви атестацій і тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	У тому числі				
		Л	П	ЛР	Інд	СРС		Л	П	ЛР	Інд	СРС
1	2	3	4	5	6	7	8	9	10	11	12	13
Атестація 1. Структури мікропроцесорних систем												
Тема 1. Форми зображення інформації	5	1	-			4	8					8
Тема 2. Логічні основи побудови елементів	7	1	2			4	8	1	1			6
Тема 3. Схемотехніка комбінаційних вузлів	7	1	2			4	6					6
Тема 4. Схемотехніка цифрових елементів	7	1	2			4	8	1	1			6
Тема 5. Схемотехніка цифрових вузлів	7	1	2			4	6					6
Тема 6. Інтегровані системи елементів	7	1	2			4	6					6
Тема 7. Схемотехніка аналогових вузлів	7	1	2			4	6					6
Тема 8. Схемотехніка обслуговуючих елементів	7	1	2			4	6					6
Тема 9. Джерела живлення. Схемотехніка комбінаторних вузлів	6		-			6	6					6
Разом	60	8	14			38	60	2	2			56
Атестація 2. Цифрові комп'ютери												
Тема 10. Цифрові комп'ютери	9	1	2			6	8					8
Тема 11. Запам'ятовуючі пристрої	11	1	4			6	8					8
Тема 12. Процесори	9	1	4			4	8	1	1			6
Тема 13. Суперкомп'ютери. Паралельні обчислювальні системи	7	1	-			6	8					8
Тема 14. Універсальні мікропроцесори	9	1	4			4	8	1	1			6
Тема 15. Схеми підтримки МП на системних платах	5	1	-			4	8					8
Тема 16. Структури мікропроцесорних систем	5	1	-			4	6					6
Тема 17. RISC-процесори	5	1	-			4	6					6
Разом	60	8	14			38	60	2	2			56
Усього годин	120	16	28			76	120	4	4			112

ОРГАНІЗАЦІЯ САМОСТІЙНОЇ РОБОТИ СТУДЕНТА

Самостійна робота студента є основним засобом оволодіння навчальним матеріалом у час, вільний від обов'язкових навчальних занять. Зміст самостійної роботи над дисципліною «Комп'ютерна схемотехніка та архітектура комп'ютерів» визначається навчальною програмою дисципліни, завданнями та вказівками викладача, даними методичними вказівками.

Забезпечується самостійна робота студента системою навчально-методичних засобів, передбачених для вивчення дисципліни: підручники, навчальні та методичні посібники, практикум тощо.

Навчальний матеріал дисципліни «Комп'ютерна схемотехніка та архітектура комп'ютерів», передбачений робочим навчальним планом для засвоєння студентом в процесі самостійної роботи, виноситься на підсумковий контроль поряд з навчальним матеріалом, який опрацьовувався при проведенні навчальних занять.

Контроль за самостійною роботою студентів здійснюється у вигляді обговорень, опитувань, розв'язання задач, тестування, перевірки та оцінювання практичних робіт та їх захисту студентами.

ВИМОГИ ДО ОФОРМЛЕННЯ ПРАКТИЧНИХ РОБІТ

Виконувати практичні роботи необхідно за своїм варіантом, номер якого відповідає номеру бригади або відповідає номеру студента у списку групи журналу старости групи, або вказується викладачем. Роботи потрібно виконувати акуратно, на аркушах стандартного формату А4 з титульним аркушем, друкованим чи рукописним способом, чорнилом будь-якого кольору за винятком червоного та зеленого. Для оформлення звіту рекомендується використовувати шрифт Times New Roman, розмір шрифту

- 14, міжрядковий інтервал 1.5, поля: верхнє та нижнє – 2 см, ліве – 3 см, праве – 1 см. Звіт про виконання кожної лабораторної роботи оформлюється у вигляді пояснювальної записки і повинен містити наступні елементи:

- 1) титульний листок, зразок якого наведено у додатку;
- 2) текст звіту, друкованим чи рукописним способами акуратно на листках стандартного формату А4 з однієї сторони, чорнилами будь-якого кольору за винятком червоного та зеленого;
- 3) список використаної літератури;
- 4) додатки (за необхідністю).

Звіт про виконання кожної роботи оформлюється у вигляді пояснювальної записки і повинен містити наступні елементи:

- 1) титульний листок, зразок якого наведено у додатку;
- 2) текст звіту, друкованим чи рукописним способами акуратно на листках стандартного формату А4 з однієї сторони, чорнилами будь-якого кольору за винятком червоного та зеленого;
- 3) список використаної літератури;
- 4) додатки (за необхідністю).

Текст звіту про виконання кожної роботи обов'язково повинен мати:

- 1) номер практичної роботи;
- 2) назву теми;

- 3) мету роботи (постановка задачі)
- 4) початкові умови;
- 5) короткі теоретичні відомості;
- 6) опис схеми;
- 7) результати комп'ютерного моделювання роботи схеми;
- 8) висновки.

Оформлені таким чином роботи повинні бути захищені у визначені викладачем строки. При співбесіді студент повинен знати відповідний теоретичний матеріал і вміти моделювати пристрої, вузли, базові елементи та комп'ютерної техніки у прикладних програмах.

Практична робота № 1.

Тема: Дослідження роботи електронної схеми, отриманої комбінуванням логічних елементів.

Мета роботи: Експериментально визначити усі можливі стани електронної схеми, отриманої комбінуванням логічних елементів

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Математичною основою комп'ютерної схемотехніки є алгебра логіки (Булева алгебра). Аргументи (X) приймають тільки два значення: 0 чи 1. Функції (Y) також можуть приймати тільки одне з двох значень: 0 чи 1. Функція алгебри логіки (ФАЛ) представляється у вигляді:

$$Y = F(X_1; X_2; X_3 \dots X_N).$$

Дана форма завдання ФАЛ називається алгебраїчною. Основними логічними функціями є: логічне заперечення (інверсія) $Y = \bar{X}$; логічне додавання (диз'юнкція) $Y = X1 + X2$ ($Y = X1 \vee X2$); логічне множення (кон'юнкція) $Y = X1 \cdot X2$ ($Y = (X1 \& X2)$).

Логічний елемент – це електронний пристрій, що реалізує одну з логічних операцій. Логічні елементи являють собою електронні пристрої, у яких оброблювана інформація закодована у вигляді двійкових чисел, що представляється напругою (сигналом) високого та низького рівня. Термін «логічні» прийшов у комп'ютерну схемотехніку з алгебри логіки, що оперує зі змінними величинами та їхніми функціями, що можуть приймати тільки два значення: «true» чи «false». Кожна логічна змінна може приймати тільки одне значення: 1 або 0.

На рис. 1.1–1.7 представлені логічні елементи та таблиці станів (таблиці істинності), що описують відповідні логічні функції в двійковому коді у виді станів вхідних та вихідних змінних. Таблиця істинності є також табличним способом завдання ФАЛ.

На рис.1.1 представлений елемент «НЕ», що реалізує функцію логічного

заперечення $Y = \bar{X}$.

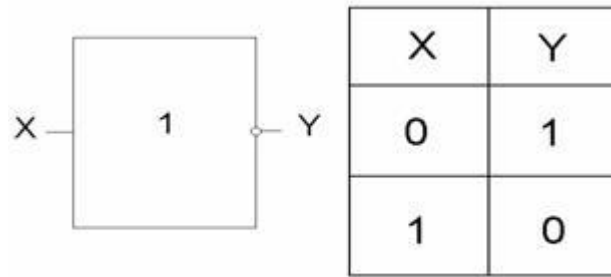


Рис. 1.1. Елемент НЕ

Елемент «АБО» (рис. 1.2) та елемент «І» (рис. 1.3) реалізують функції логічного додавання і логічного множення відповідно.

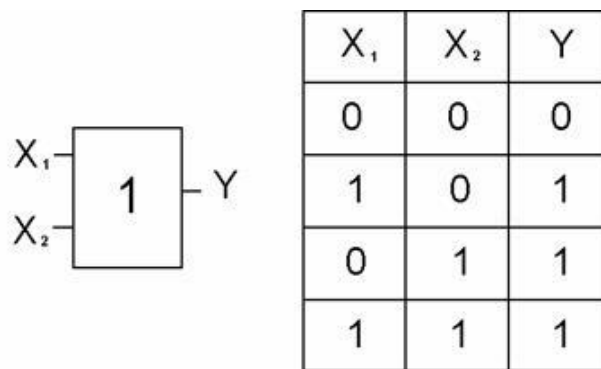


Рис. 1.2. Елемент АБО

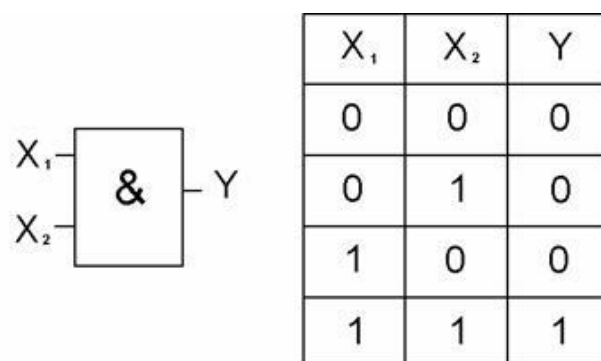


Рис. 1.3. Елемент І

Функції «стрілка Пірса» та функція «штрих Шефера» реалізуються за допомогою елементів «АБО-НЕ» та «І-НЕ», представлених на рис. 1.4 та рис. 1.5 відповідно.

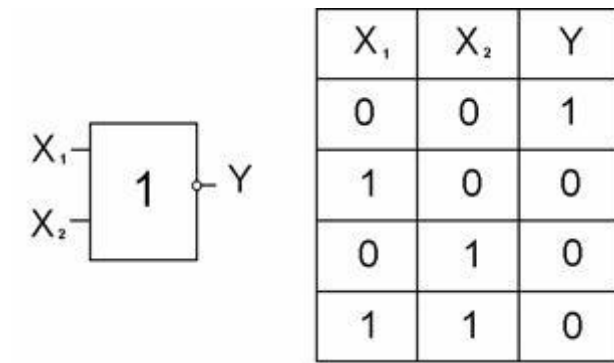


Рис. 1.4. Елемент АБО-НЕ

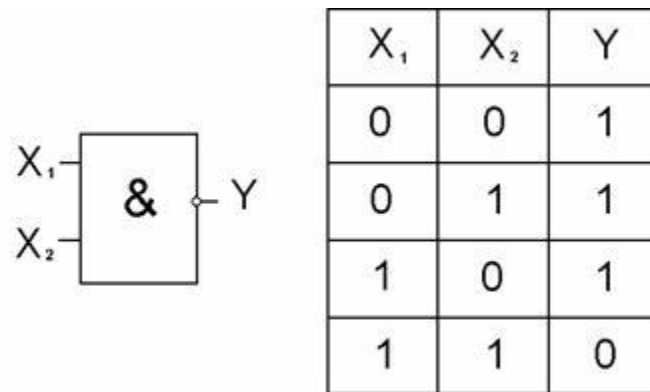


Рис. 1.5. Елемент І-НЕ

На рис. 1.6 наведений елемент «ВИКЛЮЧНЕ АБО», що реалізує функцію «Виключна диз'юнкція», на рис. 1.7 наведений елемент «ВИКЛЮЧНЕ АБО-НЕ», що реалізує функцію еквіваленція.

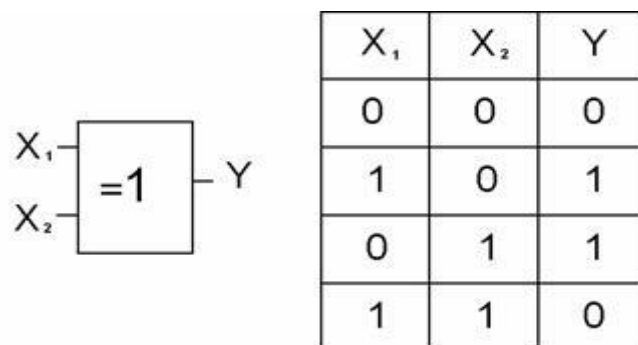


Рис. 1.6. Елемент «ВИКЛЮЧНЕ АБО»

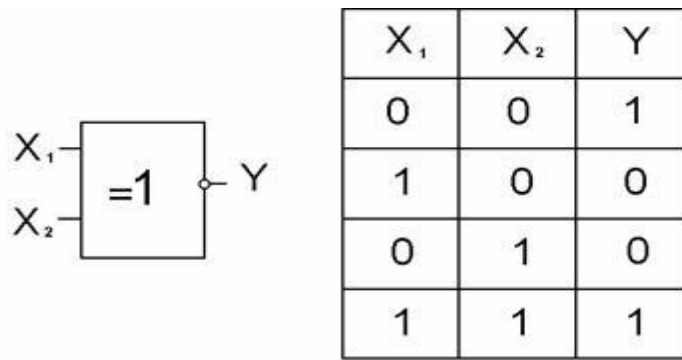


Рис. 1.7. Елемент «ВИКЛЮЧНЕ АБО-НЕ»

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

1. Запустити САПР Proteus VSM (ISIS), відкрити файл бібліотеки електронних елементів.
2. Згідно з варіантом (за номером у журналі) скласти електронну схему. Входи і вихід електронної схеми з'єднані з LogicState, LogicProbe (Debugging Tools), які показують рівень сигналу (0 чи 1).
3. Візуально перевірити правильність складання схеми.
4. Скопіювати складену електронну схему із Proteus VSM (ISIS) до звіту з виконання практичної роботи.
5. Експериментально визначити усі можливі стани електронної схеми, поступово заповнюючи таблицю істинності.

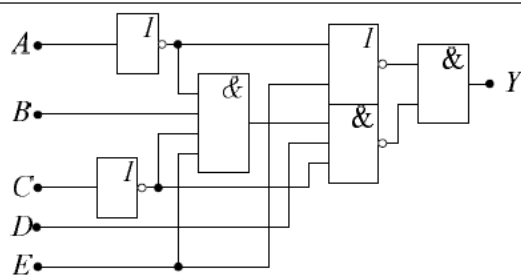
КОНТРОЛЬНІ ПИТАННЯ

1. Що таке кон'юнкція?
2. Що таке диз'юнкція?
3. Чому в комп'ютерній техніці застосовується двійкова система числення?
4. Що називається логічним елементом?

Індивідуальні завдання до лабораторної роботи 1

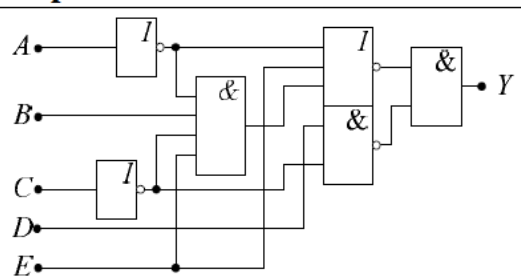
Варіант 1

N	A	B	C	D	E	Y
1	1	0	1	1	0	
2	1	0	0	0	1	
3	0	0	0	1	0	
4	1	0	0	1	0	
5	0	0	1	0	0	
6	1	0	1	0	1	
7	0	0	1	1	0	
8	1	0	1	1	1	



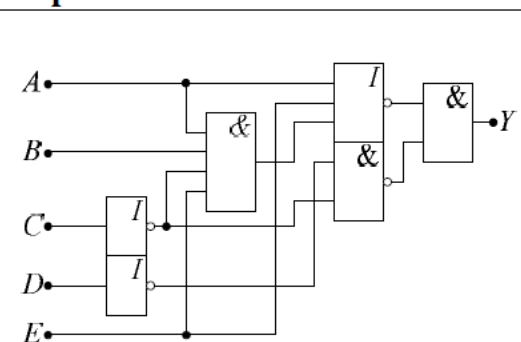
Варіант 2

N	A	B	C	D	E	Y
1	1	0	0	0	0	
2	1	1	0	0	1	
3	1	0	0	1	0	
4	1	1	0	1	1	
5	1	0	1	0	0	
6	1	1	1	0	1	
7	1	0	1	1	0	
8	1	1	1	1	1	



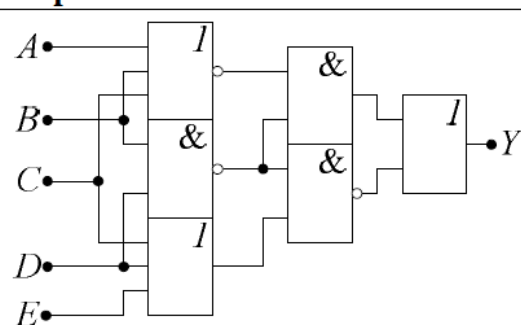
Варіант 3

N	A	B	C	D	E	Y
1	0	1	0	1	0	
2	0	1	0	1	1	
3	0	1	1	0	0	
4	0	1	1	0	1	
5	0	1	1	1	0	
6	0	1	1	1	1	
7	1	0	0	0	0	
8	1	0	0	0	1	

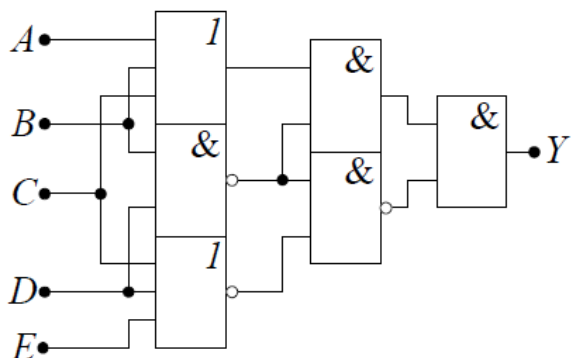


Варіант 4

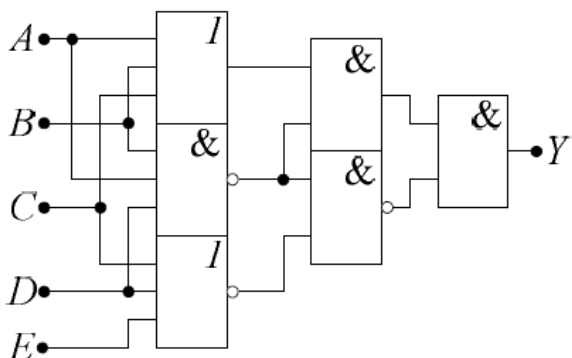
N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	



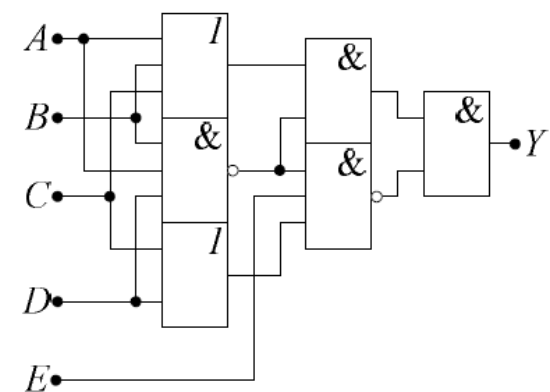
Варіант 5						
N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	



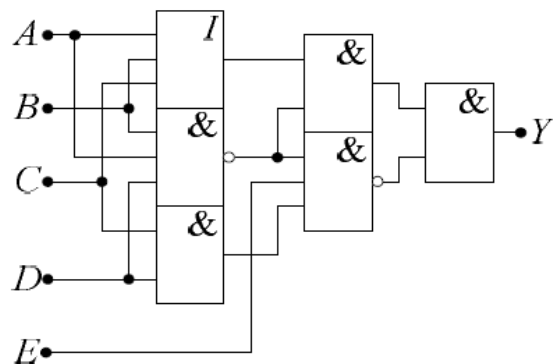
Варіант 6						
N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	0	0	1	1	0	
8	0	0	1	1	1	



Варіант 7						
N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	



Варіант 8						
N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	



Варіант 9

N	A	B	C	D	E	Y
1	0	1	0	1	0	
2	0	1	0	1	1	
3	0	1	1	0	0	
4	0	1	1	0	1	
5	0	1	1	1	0	
6	0	1	1	1	1	
7	1	0	0	0	0	
8	1	0	0	0	1	

Варіант 10

N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	

Варіант 11

N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	0	0	1	1	0	
8	0	0	1	1	1	

Варіант 12

N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	0	0	1	1	0	
8	0	0	1	1	1	

Варіант 13

N	A	B	C	D	E	Y
1	0	1	0	0	0	
2	0	1	0	0	1	
3	0	1	0	1	0	
4	0	1	0	1	1	
5	0	1	1	0	0	
6	0	1	1	0	1	
7	0	1	1	1	0	
8	0	1	1	1	1	

Варіант 14

N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	

Варіант 15

N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	0	0	1	1	0	
8	0	0	1	1	1	

Варіант 16

N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	1	0	0	0	0	
8	1	0	0	0	1	

Варіант 17

N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	

Варіант 18

N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	1	0	0	0	0	
8	1	0	0	0	1	

Варіант 19

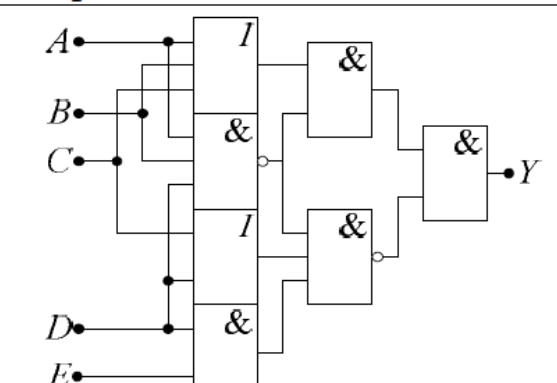
N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	

Варіант 20

N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	

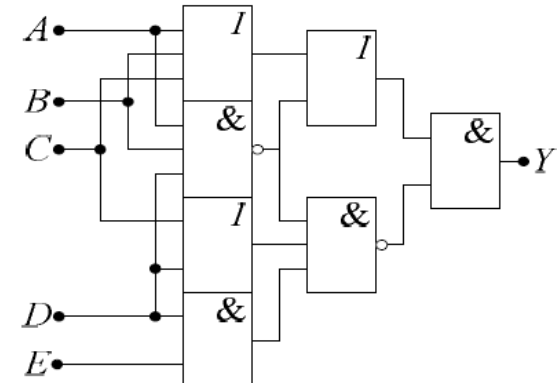
Варіант 21

N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	0	0	1	1	0	
8	0	0	1	1	1	



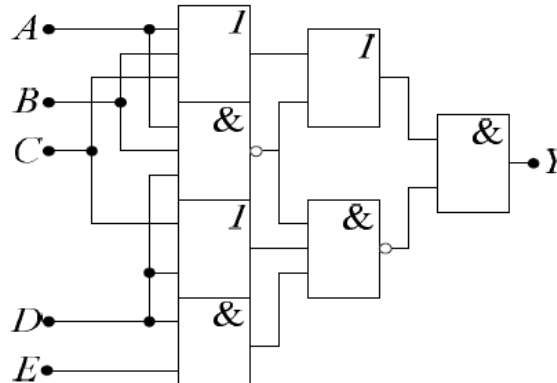
Варіант 22

N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	0	0	1	1	0	
8	0	0	1	1	1	



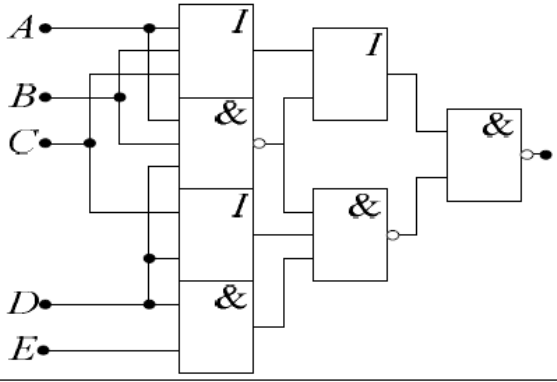
Варіант 23

N	A	B	C	D	E	Y
1	1	0	0	0	0	
2	1	0	0	0	1	
3	1	0	0	1	0	
4	1	0	0	1	1	
5	1	0	1	0	0	
6	1	0	1	0	1	
7	1	0	1	1	0	
8	1	0	1	1	1	

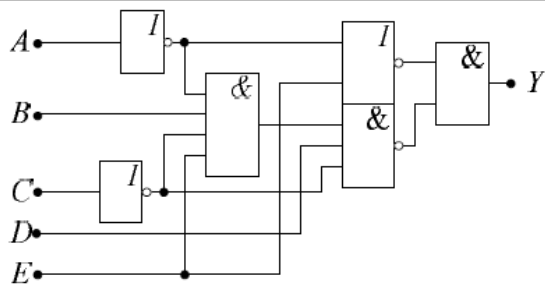


Варіант 24

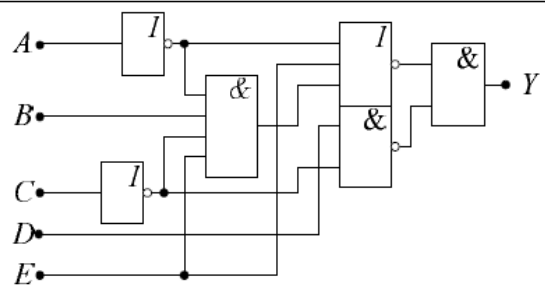
N	A	B	C	D	E	Y
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	0	0	1	1	0	
8	0	0	1	1	1	



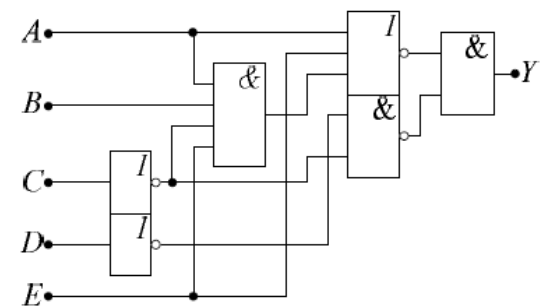
Варіант 25						
N	A	B	C	D	E	Y
1	1	0	1	1	0	
2	1	0	0	0	1	
3	0	0	0	1	0	
4	1	0	0	1	0	
5	0	0	1	0	0	
6	1	0	1	0	1	
7	0	0	1	1	0	
8	1	0	1	1	1	



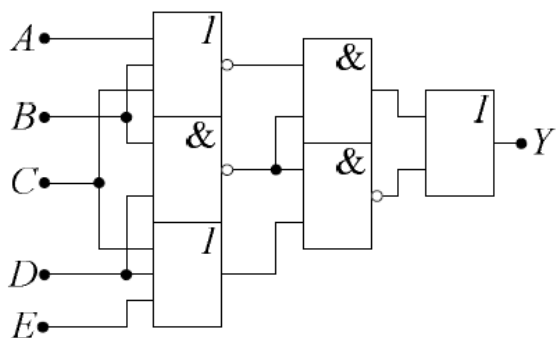
Варіант 26						
N	A	B	C	D	E	Y
1	1	0	0	0	0	
2	1	1	0	0	1	
3	1	0	0	1	0	
4	1	1	0	1	1	
5	1	0	1	0	0	
6	1	1	1	0	1	
7	1	0	1	1	0	
8	1	1	1	1	1	



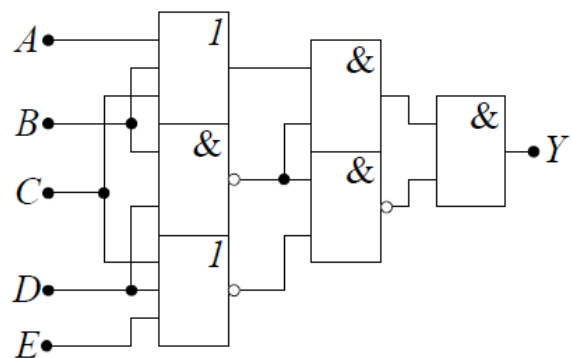
Варіант 27						
N	A	B	C	D	E	Y
1	0	1	0	1	0	
2	0	1	0	1	1	
3	0	1	1	0	0	
4	0	1	1	0	1	
5	0	1	1	1	0	
6	0	1	1	1	1	
7	1	0	0	0	0	
8	1	0	0	0	1	



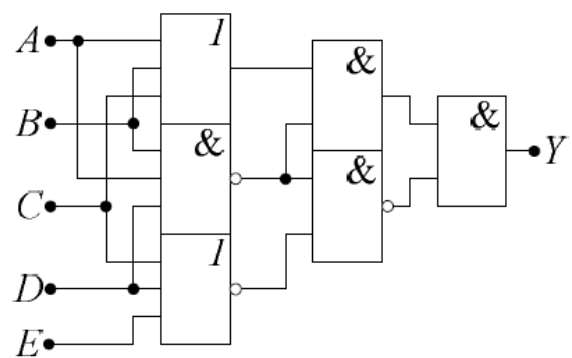
Варіант 28						
N	A	B	C	D	E	Y
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	



Варіант 29						
<i>N</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>Y</i>
1	1	1	0	0	0	
2	1	1	0	0	1	
3	1	1	0	1	0	
4	1	1	0	1	1	
5	1	1	1	0	0	
6	1	1	1	0	1	
7	1	1	1	1	0	
8	1	1	1	1	1	



Варіант 30						
<i>N</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>Y</i>
1	0	0	0	0	0	
2	0	0	0	0	1	
3	0	0	0	1	0	
4	0	0	0	1	1	
5	0	0	1	0	0	
6	0	0	1	0	1	
7	0	0	1	1	0	
8	0	0	1	1	1	



Практична робота № 2.

Тема: Мінімізація логічних функцій за допомогою карт Карно.

Мета роботи: закріплення теоретичних знань і придбання студентами практичних навиків логічного синтезу цифрових схем методом карт Карно та аналізу їх характеристик.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Карта Карно є прямокутною таблицею, розділеною горизонтальними і вертикальними лініями на комірки (клітки), загальне число яких збігається з числом мінтермів даного числа аргументів. Кількість кліток карти Карно $k = 2^n$, де n число змінних.

У кожному комірку таблиці заносяться значення одного мінтерма. Їх розміщення здійснюється так, щоб два суміжні мінтерми знаходилися в сусідніх комірках (суміжні – розрізняються формою входження не більш, ніж одного аргументу).

Карта Карно (рис. 2.1) для чотирьох змінних $A(x_0)$, $B(x_1)$, $C(x_2)$, $D(x_3)$ (число мінтермів дорівнює $k = 2^n = 2^4 = 16$):

<u>A</u>				
C	0	1	3	2
	4	5	7	6
	12	13	15	14
	8	9	11	10
<u>B</u>				

Рис. 2.1. Карта Карно для чотирьох змінних

Правила групування мінтермів.

1. Грукуються дві клітки, що стоять поруч, в стовпці, або ряду. Кожна з групованих кліток відрізняється від будь-якої сусідньої лише однією змінною, яка при цьому і виключається; число групованих кліток має бути парним; можна групувати крайні клітки між собою, оскільки карта – по суті тор.
2. Грукуються клітки, що є повними квадратами з 4, 16 кліток.

3. Грукуються клітки, що є повними горизонтальними рядами, або вертикальними стовпцями.

4. Грукуються клітки, що представляють два поруч розташованих стовпця, або рядка.

5. Клітка може входити в декілька об'єднань.

Порядок логічного синтезу.

Вихідні вимоги до пристрою, що розробляється, мають бути задані у вигляді словесного опису. Визначення структури логічного пристрою, яка задовольняє заданому опису, виконується шляхом абстрактного і структурного синтезу.

Абстрактний синтез полягає в переході від словесного опису пристрою до завдання його оператора у вигляді графів, таблиць або матриць. Незалежно від форми оператора він повинен визначати перелік вхідних та вихідних змінних і встановлювати зв'язок між ними.

Структурний синтез полягає у визначенні структурної схеми логічного пристрою за допомогою алгебри логіки, яка дозволяє перейти до завдання оператора у вигляді формул і спростити їх.

Типовий порядок логічного проектування:

- кодування вхідних та вихідних змінних і перехід від словесного завдання оператора до табличного;
- перехід від табличної форми оператора до алгебраїчного у вигляді диз'юнктивної нормальної форми (ДНФ);
- спрощення ДНФ функції (оператора) і здобуття мінімальної ДНФ;
- перехід від мінімальної ДНФ до мінімальної форми в базисі вибраного - функціонального повного набору, що реалізовується логічними елементами;
- складання структурної схеми логічного пристрою відповідно до результуючої мінімальною формою алгебри.

Приклад логічного синтезу

Потрібно синтезувати пристрій з трьох датчиків з двійковими вихідними сигналами. Використовуючи логічні елементи забезпечити індикацію станів, коли одиничні значення приймають щонайменше два з трьох сигналів.

1. Кодування вихідних і вхідних змінних:

A, B, C – вхідні змінні, F – вихідна змінна.

Значення $F = 1$ відповідають наборам A, B, C, в яких дві або три вхідні змінні дорівнюють 1; при останніх наборах $F = 0$. Складаємо таблицю істинності:

C	B	A	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

2. По таблиці здійснюємо перехід до алгебраїчної форми функції

$$F = \bar{C}BA + C\bar{B}A + CB\bar{A} + CBA$$

3. Складаємо таблицю Карно (рис. 2.2):

	<u>A</u>					<u>A</u>			
	0	1	3	2		0	0	1	0
C	4	5	7	6		0	1	1	1
	<u>B</u>					<u>B</u>			

Рис. 2.2. Карта Карно для трьох змінних

Виконаємо попарне групування і отримуємо мінімізовану функцію

$$F = AB + AC + BC$$

4. Використовуючи як функціональний набір основний функціонально

повний набір вибираємо необхідні логічні елементи:

- три двох входових елемента І;
- один трьох входовий елемент АБО.

5. Побудова структурної схеми логічного пристрою, що включає вибрані елементи (рис. 2.3).

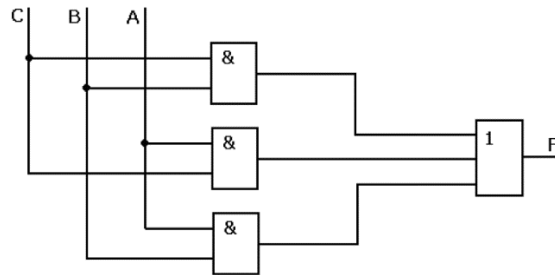


Рис. 2.3. Синтезована схема логічного пристрою

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

1. За структурною схемою (рис. 2.4) провести аналіз та встановити функціональну залежність у вигляді формул алгебри логіки та таблиці станів (істинності). Для цього скласти схему (за індивідуальним варіантом) у середовищі Proteus VSM та експериментально визначити таблицю станів роботи схеми.

2. За таблицею станів заповнити карту Карно, мінімізувати логічну функцію. За необхідністю можна скористуватись програмою [Carno Minimizer](#).

3. Синтезувати схему цифрового пристрою у базисі І-НЕ.

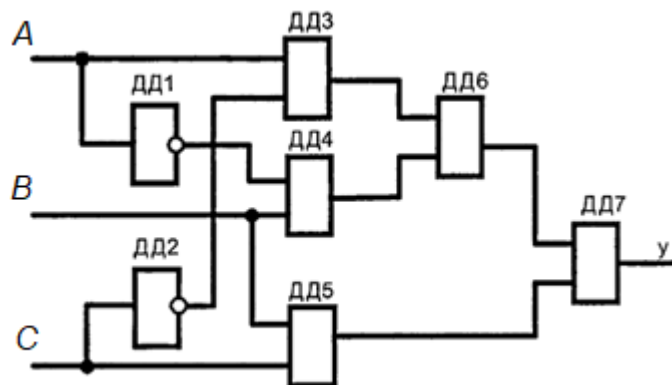


Рис. 2.4. Структурна схема виконання завдання практичної роботи

Таблиця 2.1 – Варіанти завдань

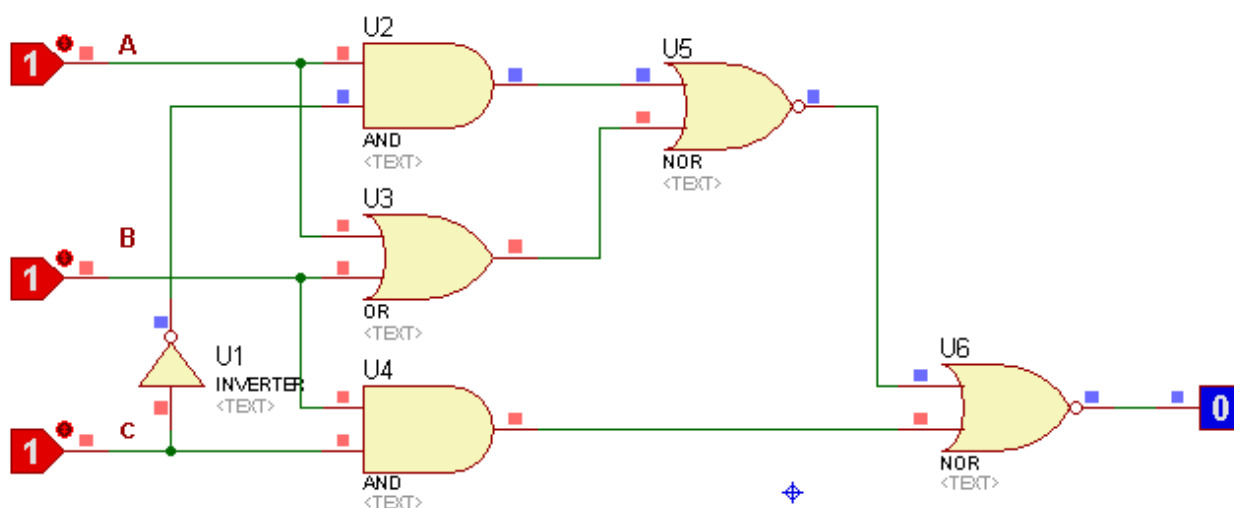
№	ДД1	ДД2	ДД3	ДД4	ДЦ5	ДД6	ДД7
1	+	-	I	АБО	I-НЕ	АБО	I
2	-	+	АБО-НЕ	I-НЕ	АБО-НЕ	I	АБО
3	-	+	I	АБО	АБО-НЕ	I	АБО-НЕ
4	-	+	I-НЕ	АБО-НЕ	I-НЕ	АБО	I-НЕ
5	+	-	АБО	I	I-НЕ	АБО	I-НЕ
6	+	-	I-НЕ	I	АБО-НЕ	I	АБО-НЕ
7	-	+	АБО-НЕ	АБО	АБО	I-НЕ	I
8	+	-	I	I-НЕ	АБО-НЕ	I	АБО-НЕ
9	-	+	I-НЕ	I	I-НЕ	АБО	I-НЕ
10	+	-	АБО	I	АБО-НЕ	I	АБО-НЕ
11	+	-	АБО-НЕ	АБО	I-НЕ	АБО	I-НЕ
12	-	+	АБО НЕ	АБО	I	АБО-НЕ	АБО
13	-	+	АБО	I-НЕ	АБО	I-НЕ	I
14	-	+	АБО-НЕ	I-НЕ	АБО-НЕ	I	АБО-НЕ
15	+	-	I-НЕ	АБО	I	АБО-НЕ	АБО-НЕ
16	+	-	АБО-НЕ	I	АБО	I-НЕ	I
17	+	-	I	АБО	I-НЕ	АБО	I-НЕ
18	+	-	I-НЕ	I	АБО-НЕ	I	АБО
19	-	+	АБО-НЕ	I-НЕ	I	I	АБО
20	-	+	АБО-НЕ	АБО	АБО	I-НЕ	I-НЕ
21	+	-	I	I-НЕ	АБО	I-НЕ	I-НЕ
22	+	-	I-НЕ	АБО	I	АБО-НЕ	АБО-НЕ
23	-	+	АБО-НЕ	АБО	АБО	I-НЕ	I
24	-	+	I-НЕ	I	I-НЕ	АБО	I-НЕ
25	+	-	АБО	I-НЕ	I	АБО-НЕ	АБО
26	+	-	АБО-НЕ	АБО	АБО	I-НЕ	I-НЕ
27	-	+	I-НЕ	I	I	АБО-НЕ	АБО
28	+	-	АБО	АБО-НЕ	АБО-НЕ	I	АБО
29	-	+	I	АБО	АБО-НЕ	I	АБО-НЕ
30	-	+	I	АБО	I	АБО-НЕ	АБО-НЕ

ДД1, ДД2 – інвертори, знак «-» означає, що елемент відсутній, знак «+» відповідає наявності інвертора в схемі.

КОНТРОЛЬНІ ПИТАННЯ

1. Назвіть основні логічні функції.
2. Приведіть тотожності булевої алгебри.
3. Розкрийте сенс основних законів булевої алгебри.
4. Що таке нормальна форма представлення двійкової функції ?
5. Приведіть послідовність мінімізації логічних функцій методом карт Карно.
6. Назвіть значення логічних рівнянь сигналів (0 і 1) для різних типів логіки.

ПРИКЛАД ВИКОНАННЯ ПРАКТИЧНОЇ РОБОТИ



№	C	B	A	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	0

Рис. 2.5. Схема та таблиця станів цифрового пристрою (варіант 30)

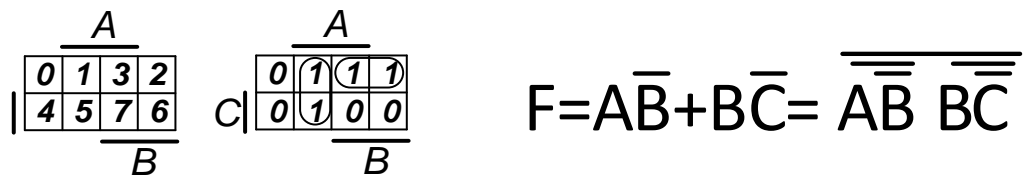


Рис. 2.6. Карти Карно та мінімізована логічна функція

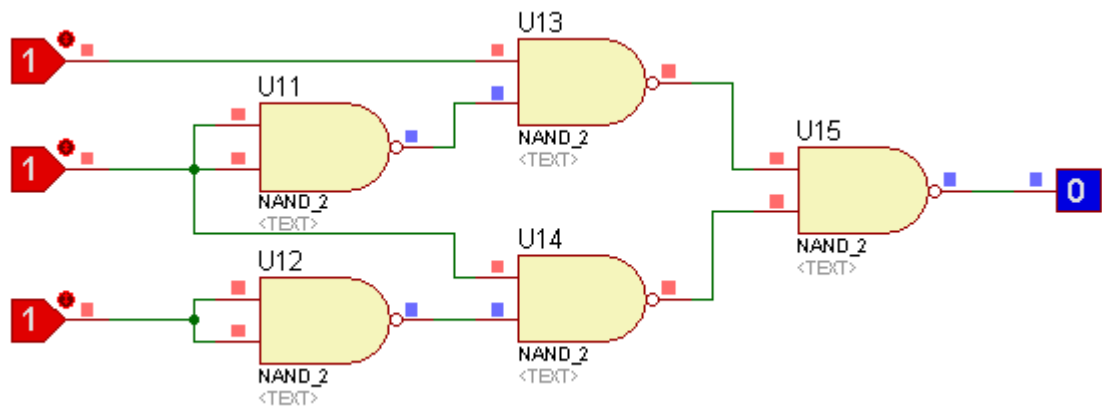


Рис. 2.7. Синтезована схема цифрового пристрою у базисі І-НЕ

Практична робота № 3

Тема: Дослідження схем тригерів.

Мета роботи: вивчення принципів роботи тригерів, виконаних у вигляді окремих мікросхем.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Тригером називається пристрій з двома стійкими станами. Тригери являють собою найпростіші послідовні пристрої і широко використовуються в електронних пристроях різного призначення як у вигляді самостійних вузлів, так і як елементи для побудови більш складних цифрових пристроїв (лічильників, регістрів, запам'ятовуючих пристроїв). Основною властивістю тригера є наявність пам'яті, під якою мається на увазі його здатність зберігати свій стан («0» або «1») і після припинення дії зовнішніх сигналів. Таким чином, тригер є елементарною коміркою пам'яті для зберігання одного двійкового розряду числа.

Узагальнена схема тригерного пристрою показана на рис. 3.1 та складається з пристрою управління УУ та тригерної комірки ТЯ.

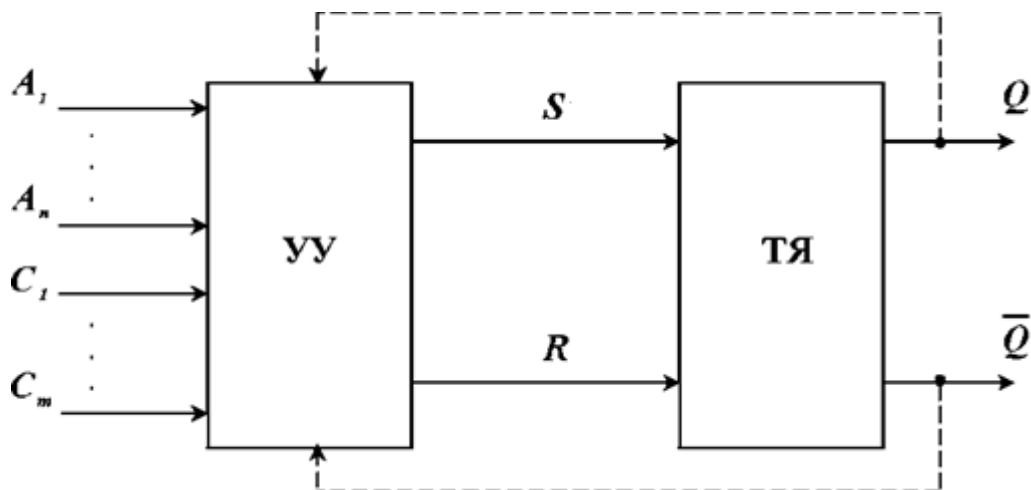


Рис. 3.1. Структурная схема триггера

На рис.3.1 сигнали мають таке призначення: $A_1...A_n$ – інформаційні сигнали; $C_1...C_m$ – тактові сигнали; S (*set*), R (*reset*) – входні сигнали; Q –

вихідні сигнали.

RS - тригери

RS-тригер має два входа керування S (set), R (reset), з допомогою яких виконується встановлення тригера в той або інший стан (рис. 3.2):

$Q = 1$ при $S=1$ та $R=0$ (встановлення тригера);

$Q=0$ при $S=0$ и $R=1$ (скидання тригера);

$Q^{n+1} = Q^n$ при $S=R=0$ (режим зберігання попереднього стану);

$S=R=1$ – заборонена комбінація сигналів керування, яка може привести до невизначеного стану тригера.

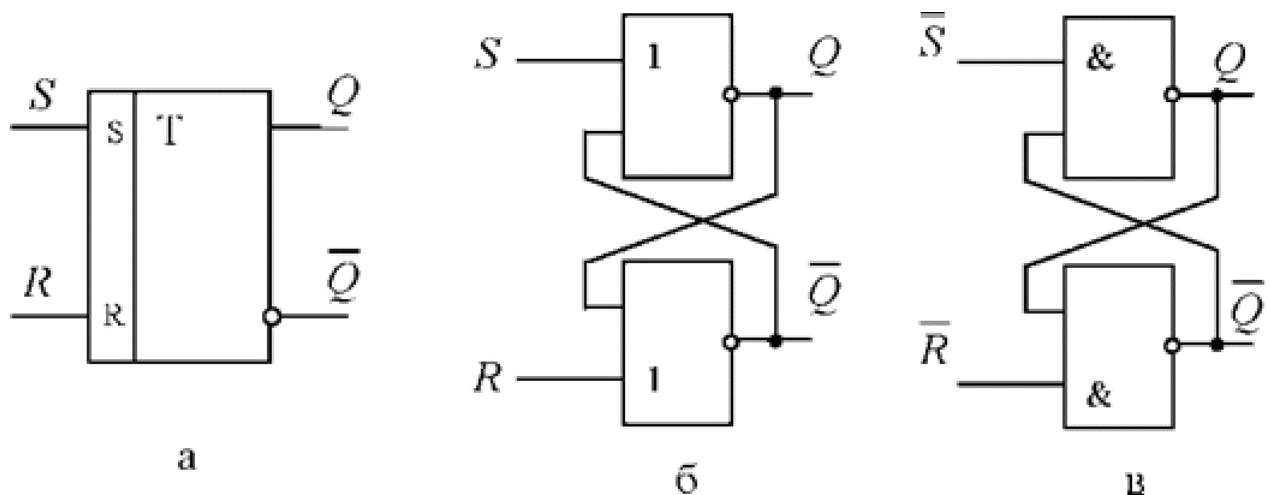


Рис. 3.2. RS-тригери

Табл. 3.1. Таблиця станів RS тригера

R	S	Q^{n+1}
0	0	Q^n
0	1	1
1	0	0
1	1	×

Розглянутий тригер є асинхронним, тому що зміна його стану відбувається безпосередньо з надходженням сигналів.

Схемотехнічно RS-тригер може бути реалізований на елементах 2АБО-НЕ (рис. 2, б) та 2 І-НЕ (рис. 3.2, в) з використанням перехресних позитивних

зворотних зв'язків. У тригері на елементах 2І-НЕ зміна стану відбувається при низьких рівнях сигналів \bar{S} і \bar{R} .

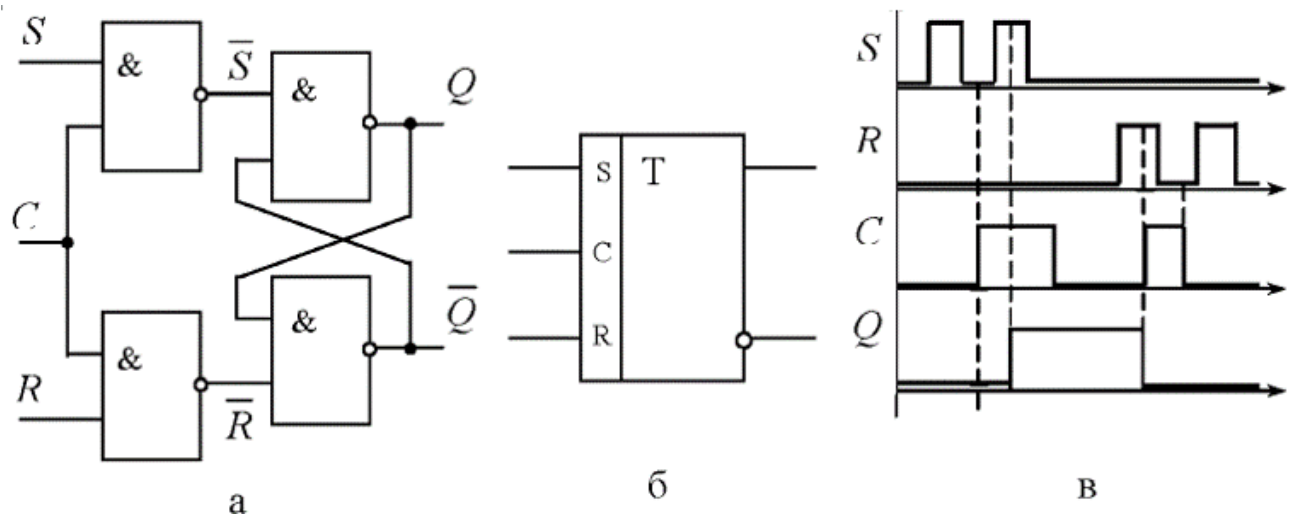


Рис. 3.3. RS-тригер зі синхронізацією за рівнем

У RS-тригері з синхронізацією по фронту зміна стану відбувається в момент зміни рівня сигналу C . При цьому можлива синхронізація як за переднім, так і за заднім фронтом (зрізу). Такі тригери будуються за двоступеневою схемою і в них процеси прийому і запису даних розділені в часі. Схема тригера з синхронізацією за заднім фронтом та його умовне позначення наведені на рис. 3.4.

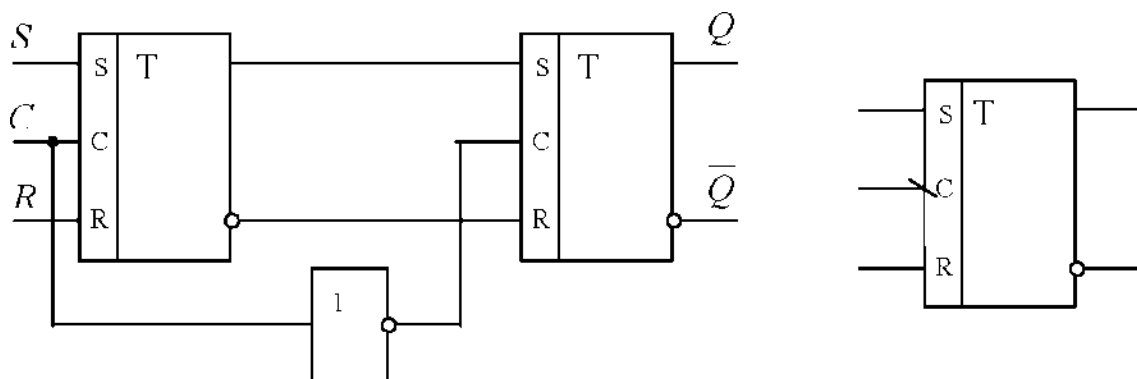


Рис. 3.4. RS-тригер з синхронізацією за заднім фронтом

У синхронних RS-тригерах можуть бути використані різні способи синхронізації. На рис. 3.3, а і б показана схемотехнічна реалізація та умовне позначення RS-тригера з синхронізацією за рівнем (високий). На рис. 3.3, в наведені діаграми роботи такого тригера. Зміна станів відбувається тільки при

високих рівнях сигналу синхронізації С.

Двоступеневі тригер будують за способом «М-S» і забезпечують поєднання двох процесів - одночасного записування нової інформації та зчитування старої. При подачі високого рівня сигналу С проводиться запис в перший тригер (прийом даних), а після закінчення сигналу С - запис у другий тригер. Двоступеневий RS-тригер застосовується для побудови інших більш складних типів тригерів і регістрів зсуву.

Під час дії синхроімпульсу С перший ступінь «М» (Master - основний) приймає нову вхідну інформацію, а друга ступінь "S" (Slave - допоміжний) в цей час передає у зовнішні схеми стару інформацію. По завершенню синхроімпульсу С інформація з першої ступені переписується до другої ступені.

При однотоктному обміні інформацією зв'язок між ступенями реалізується за допомогою інвертора (рис. 3.5, а), забороняючими зв'язками (рис. 3.5, б) або різнополярного керування (рис. 3.5, в). При двотоктному обміні зв'язок між ступенями забезпечується двома серіями синхросигналів - С1 и С2 (рис. 3.5, г).

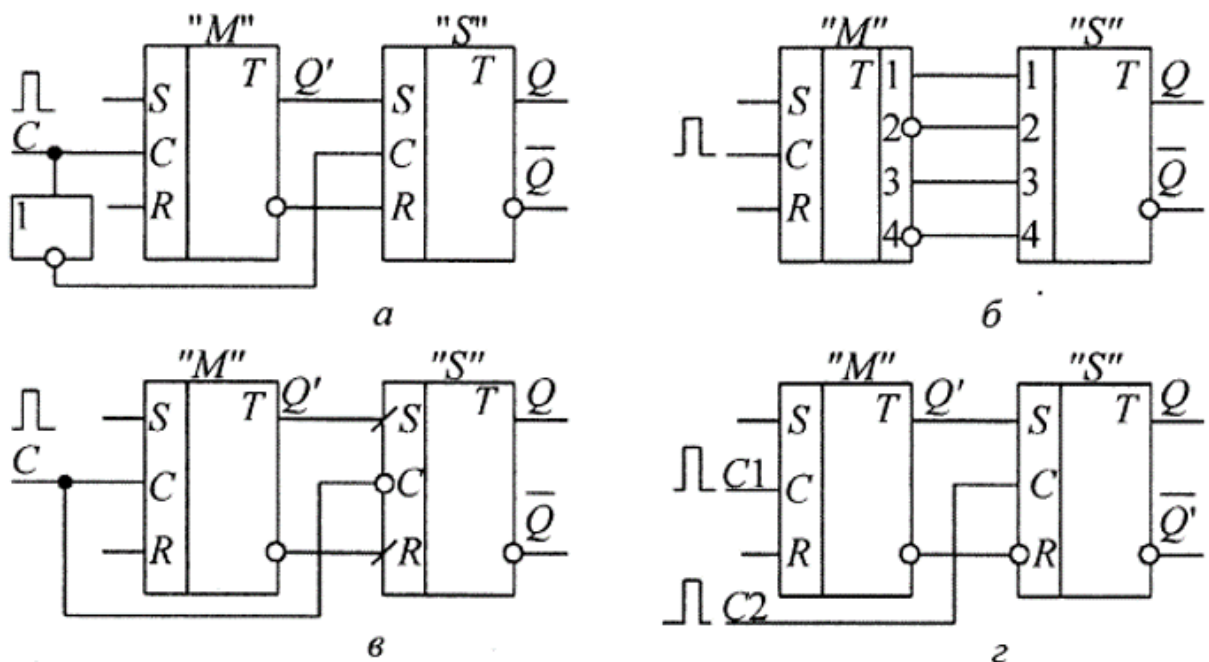


Рис. 3.5. Організація зв'язку між ступенями тригера: а) з інвертором; б) із забороняючими зв'язками; в) з різнополярним керуванням; г) з двофазним обміном

JK-тригери

JK-тригер має два керуючих входи J (jump) і K (keep) і функціонує подібно RS-тригеру, але при цьому не має заборонених комбінацій керуючих сигналів. J - вхід подібний S - входу, а K-вхід подібний R-входу. При всіх комбінаціях сигналів на вході, крім $J = K = 1$, він діє подібно RS-тригеру. При $J = K = 1$ в кожному такті відбувається «перекидання» тригера і його стан змінюється на протилежне (табл. 3.2). На рис. 3.6 показано умовне позначення JK-тригера з синхронізацією по передньому фронту.

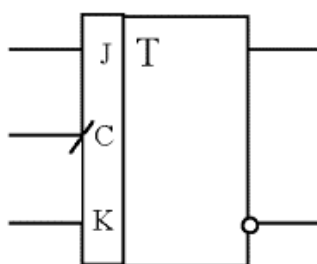


Рис. 3.6. JK-тригер

Табл. 3.2. Таблиця станів JK-тригера

J	K	Q^{n+1}
0	0	Q^n
0	1	0
1	0	1
1	1	$\overline{Q^n}$

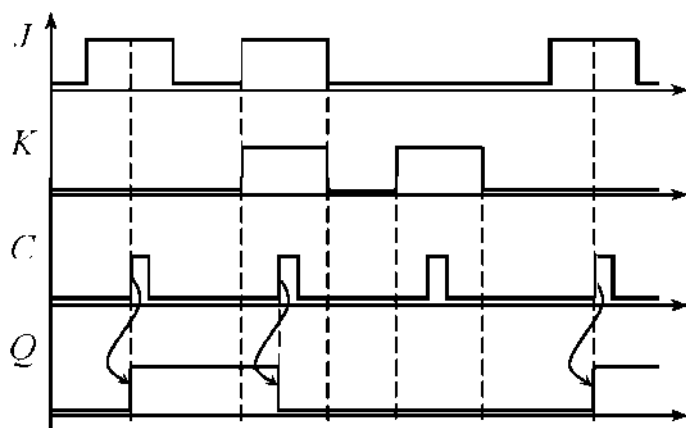


Рис. 3.7. Діаграми роботи JK-тригера

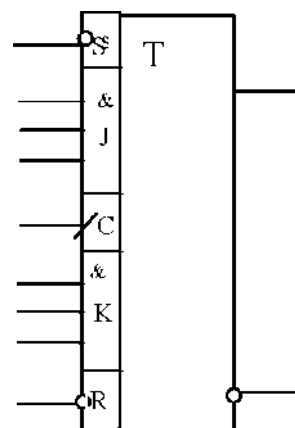


Рис. 3.8. Мікросхема 7472

Робота JK-тригера пояснюється часовими діаграмами на рис. 3.7. JK-тригери відносяться до універсальних пристроїв щодо їх застосування як для побудови інших типів тригерів, так і більш складних пристроїв послідовного принципу дії. У всіх серіях ІС випускаються JK-тригери з різними функціональними можливостями. Наприклад, мікросхема 155TB1(SN7472) (рис. 3.8) має по три входи J і K, які зв'язані логічною операцією І, що суттєво розширює можливості її застосування при реалізації різних алгоритмів управління станами тригера без застосування додаткових елементів.

Т-тригери

Т-тригери називають лічильними і застосовуються для побудови лічильників та дільників частоти. Такий тригер має один тактовий вхід і його стан змінюється кожен раз при подачі тактового імпульсу $T = 1$ і залишається незмінним при $T = 0$. Таблиця стану тригера наведена в табл. 3.3. Позначення Т-тригера і діаграми роботи наведені на рис. 3.9.

Табл. 3.3 Таблиця станів Т-тригера

T	Q^{n+1}
0	Q^n
1	$\overline{Q^n}$

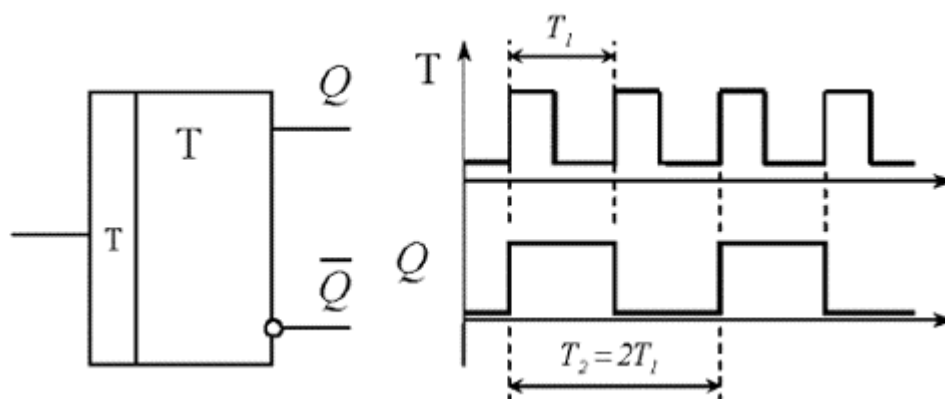


Рис. 3.9. Т-триггер

Як видно з рис. 3.9, Т-триггер ділить частоту вхідних імпульсів в 2 рази. Для отримання великих значень коефіцієнта ділення частоти застосовується

каскадне з'єднання Т-тригерів. Як самостійний виріб Т-тригер у вигляді ІС не випускається і при необхідності реалізується на базі інших типів тригерів. Наприклад, JK-тригер при $J = 1$ і $K = 1$ (рис. 3.10, а) перетворюється в Т-тригер. Приклад реалізації Т- тригера на основі RS-тригера показаний на рис. 3.10, б.

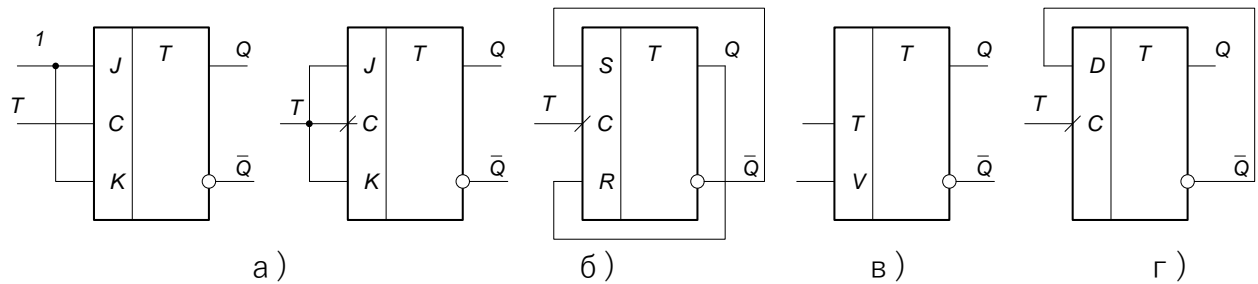


Рис. 3.10. Варіанти реалізації Т-тригера

В окремих випадках для розширення функціональних можливостей Т-тригер забезпечується R і V-входами (вхід дозволу зміни стану). Такі TV - тригери (рис. 3.10, в) застосовуються для побудови синхронних лічильників.

Д-тригери

Особливістю Д-тригера (тригера затримки) є те, що він зберігає інформацію, що надійшла на D-вхід в попередньому такті роботи до приходу синхроімпульсу, тобто його стан може змінюватися з затримкою на один такт. Синхронізація роботи проводиться за переднім або заднім фронтом. Умовне позначення Д-тригера з синхронізацією за переднім фронтом та діаграми його роботи показані на рис. 3.11.

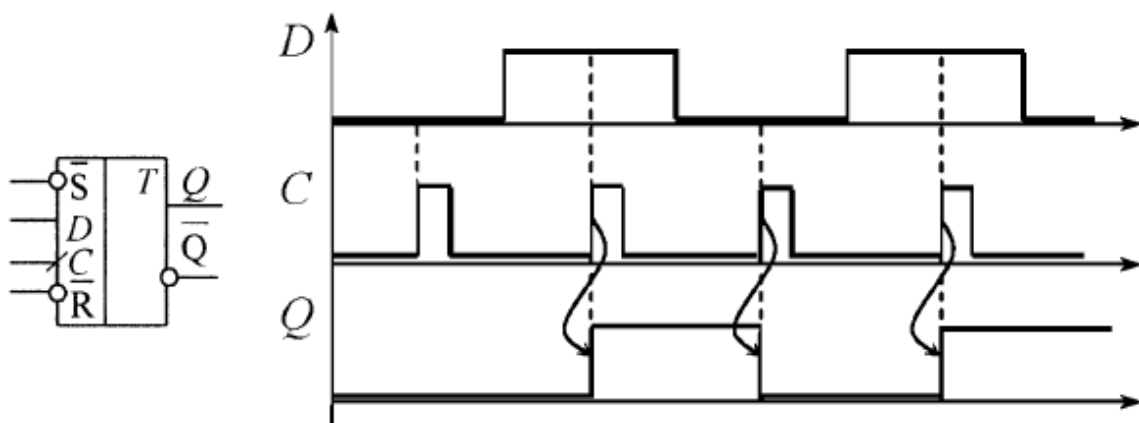


Рис. 3.11. Д-тригер

Табл. 3.4 Таблиця станів D-тригера

Входи				Виходи	
-S	-R	C	D	Q	-Q
0	1	X	X	1	0
1	0	X	X	0	1
0	0	X	X	Не визначено	
1	1	0→1	1	1	0
1	1	0→1	0	0	1
1	1	0	X	Не міняється	
1	1	1	X	Не міняється	
1	1	1→0	X	Не міняється	

У DV-тригерах є додатковий V-вхід для дозволу зміни стану. D-тригери є основою для побудови регістрів різного типу. D-тригер може бути реалізований на базі JK- тригера шляхом виключення комбінації $J = K$ (рис. 3.12, а).

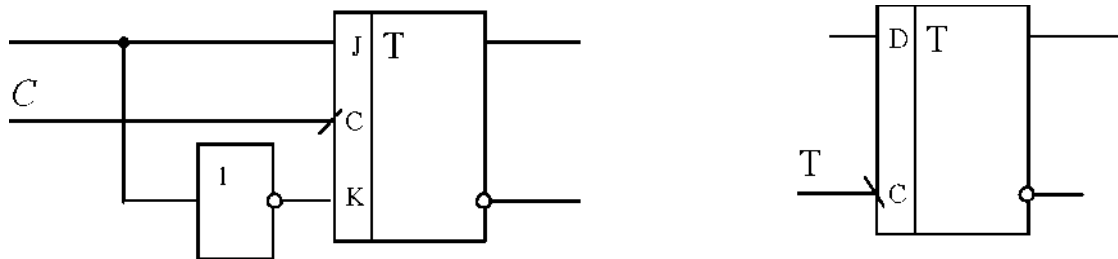


Рис. 3.12. Варіанти реалізації D-тригера

На базі D-тригера може бути реалізований T-тригер, якщо інверсний вихід з'єднати з D-входом (рис. 3.10, з).

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

1. Включити комп'ютер.
2. Запустити програму Proteus 7 Professional.
3. На робочому полі програми зібрати схему для дослідження D-тригера (рис. 3.13).

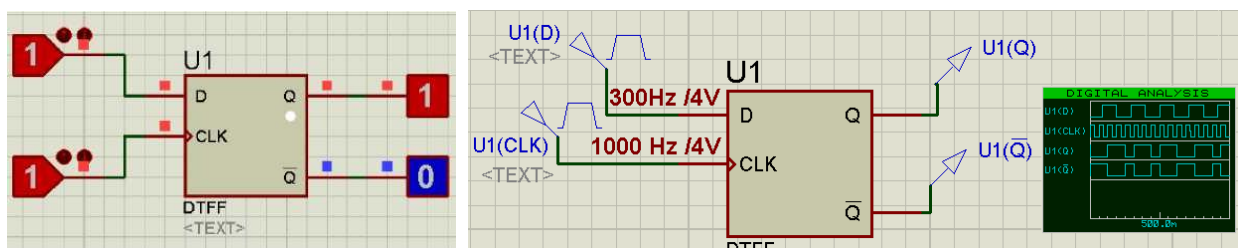
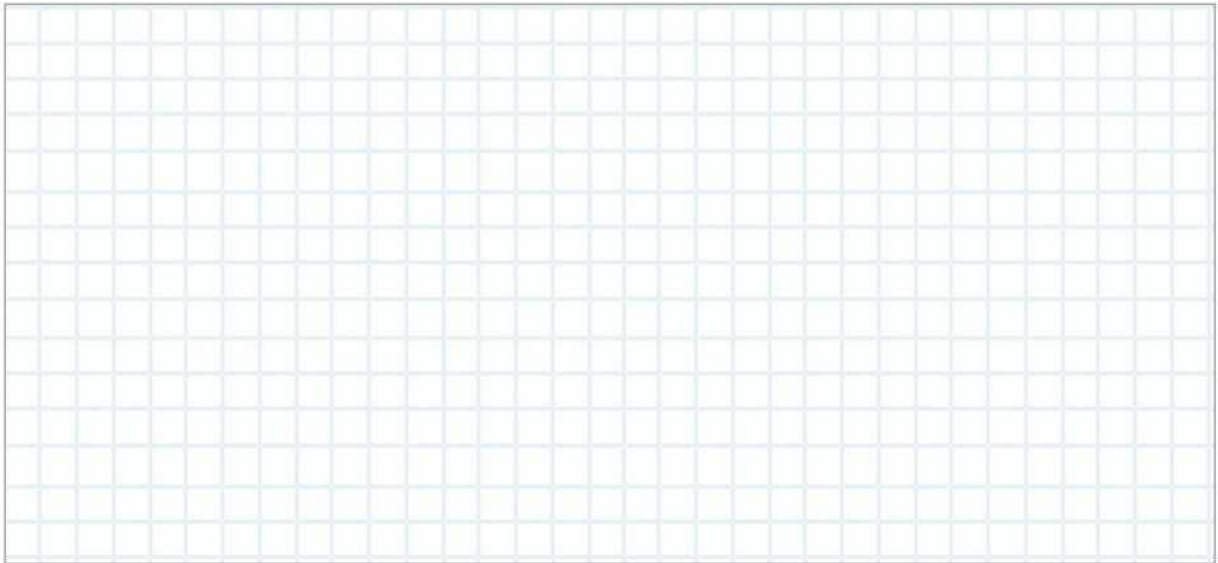


Рис. 3.13 - Схема для дослідження D-тригера

4. Привести часові діаграми роботи тактованого D - тригера:



5. На робочому полі програми зібрати схему для дослідження JK-тригера (рис. 3.14).

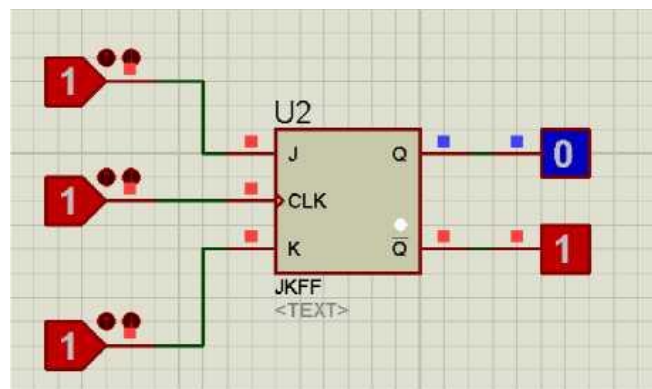


Рис. 3.14 - Схема для дослідження JK-тригера

6. Скласти таблицю станів JK-тригера:

J	K	Q

7. На робочому полі програми зібрати схему для дослідження мікросхеми 4013 (рис. 3.15).

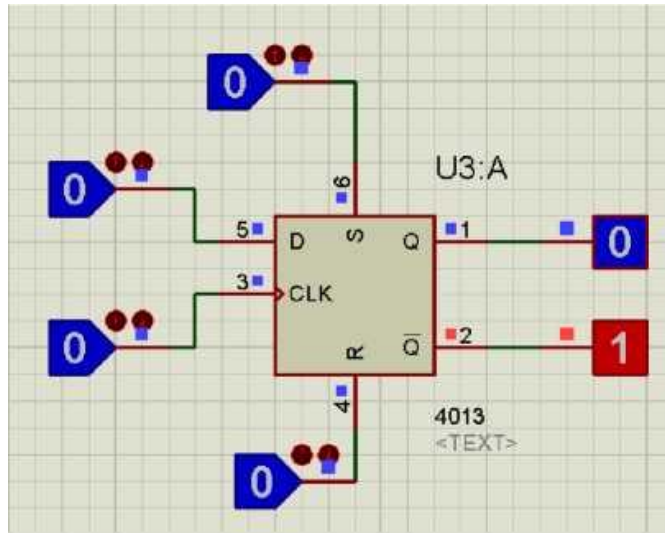


Рис. 3.15 - Схема для дослідження мікросхеми 4013

8. Описати принцип роботи мікросхеми 4013:

9. На робочому полі програми зібрати схему для дослідження мікросхеми 74107 (рис. 3.16).

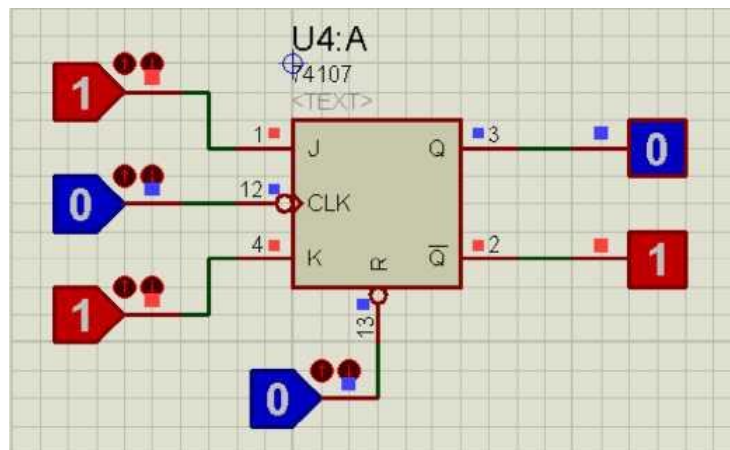


Рис. 3.16 - Схема для дослідження мікросхеми 74107

10. Описати принцип роботи мікросхеми 74107:

11. На робочому полі програми зібрати схему для дослідження мікросхеми

7474 (рис. 3.17).

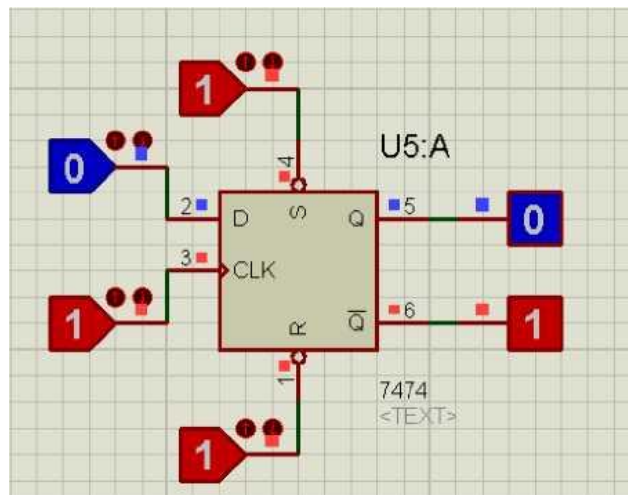


Рис. 3.17 - Схема для дослідження мікросхеми 7474

12. Описати принцип роботи мікросхеми 7474:

13. Зробити висновки по роботі. **Висновки:**

14. Надати відповіді на контрольні запитання.

КОНТРОЛЬНІ ПИТАННЯ

1. Поясніть, що таке тригер.

Тригер це _____

2. Наведіть умовне графічне позначення і проаналізуйте логіку роботи асинхронного RS-тригера в базисі І-НІ (наведіть таблицю істинності).

3. Наведіть умовне графічне позначення і проаналізуйте логіку роботи

- асинхронного RS-тригера в базисі АБО-НІ (наведіть таблицю істинності).
4. Наведіть умовне графічне позначення і проаналізуйте логіку роботи синхронного RS-тригера.
 5. Наведіть умовне графічне позначення і проаналізуйте логіку роботи D-тригера (наведіть часові діаграми роботи).
 6. Наведіть умовне графічне позначення і проаналізуйте логіку роботи T-тригера (наведіть часові діаграми роботи).
 7. Наведіть умовне графічне позначення і проаналізуйте логіку роботи JK-тригера (наведіть таблицю істинності).

Практична робота № 4

Тема: Дослідження регістрів.

Мета роботи: Вивчення принципів побудови і роботи регістрів різного типу.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Регістрами називаються ЦПП, що виконують операції запису (приймання, введення), збереження, перетворення (наприклад, шляхом зсуву) та зчитування (передавання, виведення) двійкових багатоцифрових чисел (слів). Особливістю регістрів є регулярність їх структури: кожний розряд складається з однакових тригера і додаткових елементів керування розрядом.

Залежно від способу запису і зчитування інформації регістри поділяються на два основні типи: *паралельні* і *зсуву*.

Паралельними (регістрами пам'яті, схову) є регістри, в яких запис і зчитування всіх розрядів слова відбувається одночасно і розряди слова займають фіксоване положення відносно розрядів регістра. Паралельний регістр є набір тригерів з окремими інформаційними входами і виходами в кожному розряді та зі спільним для всіх розрядів синхровходом. Найпоширенішими є регістри з *однофазним записом* інформації на D-тригерах (рис. 4.1, а). З надходженням синхроімпульсу ($G = 1$) відбувається *запис* до

регістра в паралельному коді вхідного слова D , наприклад, чотирирозрядного $Q_3 \dots Q_0 = d_3 \dots d_0$, а за його відсутності ($G = 0$) регістр перебуває в режимі зберігання записаної інформації, коли її можна зчитувати (також у паралельному коді) з виходів $Q_3 \dots Q_0$.

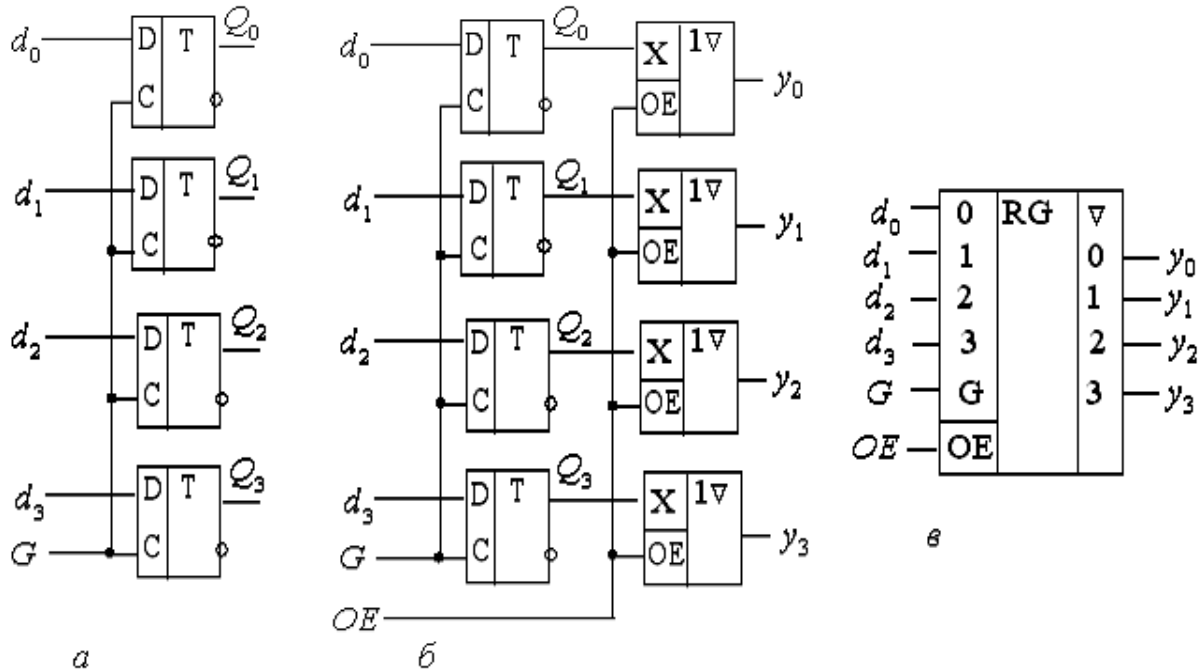


Рис. 4.1. Паралельні регістри

Регістри з трьома станами виходу (рис. 4.1, б) дозволяють здійснювати гнучкіший зв'язок з вихідною шиною. У такому регістрі так само активним рівнем синхроімпульсу $G = 1$ вхідне слово записується до тригерів, які за пасивного рівня $G = 0$ перебувають у режимі зберігання. Елементи з трьома станами виходу за відсутності сигналу дозволу ($OE = 0$) перебувають у високоімпедансному стані $y_i = Z$ і від'єднують регістр від шини, а під час зчитування, коли тригери перебувають у режимі зберігання, сигналом $OE = 1$ інформація передається в шину: $y_i = Q_i$. Такі регістри з потужними вихідними елементами називають *буферними* (рис. 4.1, в), бо забезпечують підвищену навантажувальну здатність і часову буферизацію даних.

Регістрами зсуву (послідовними) є регістри, в яких запис і зчитування розрядів слова відбувається послідовно в часі (у послідовному коді) через один вхід та один вихід відповідно, отже, розряди слова з кожним тактом

синхроімпульсів просуваються з одного розряду регістра до іншого. Регістр зсуву є набір тригерів з послідовно з'єднаними входами і виходами та зі спільним для всіх розрядів синхровходом. Напрямок з'єднань визначає і напрямок просування розрядів слова, залежно від якого розрізняють регістри прямого зсуву, зворотного зсуву та реверсивні (рис. 4.2, а,б,в відповідно).

У регістрі *прямого зсуву* (зсуву праворуч), наприклад, чотирирозрядного (див. рис. 4.2. а), розряди вхідного слова $D = d_3d_2d_1d_0$, починаючи від старшого d_3 , надходять до входу послідовного введення D_R , що є входом молодшого розряду регістра, а зчитуються з виходу послідовного виведення Q_3 , що є виходом старшого розряду регістра.

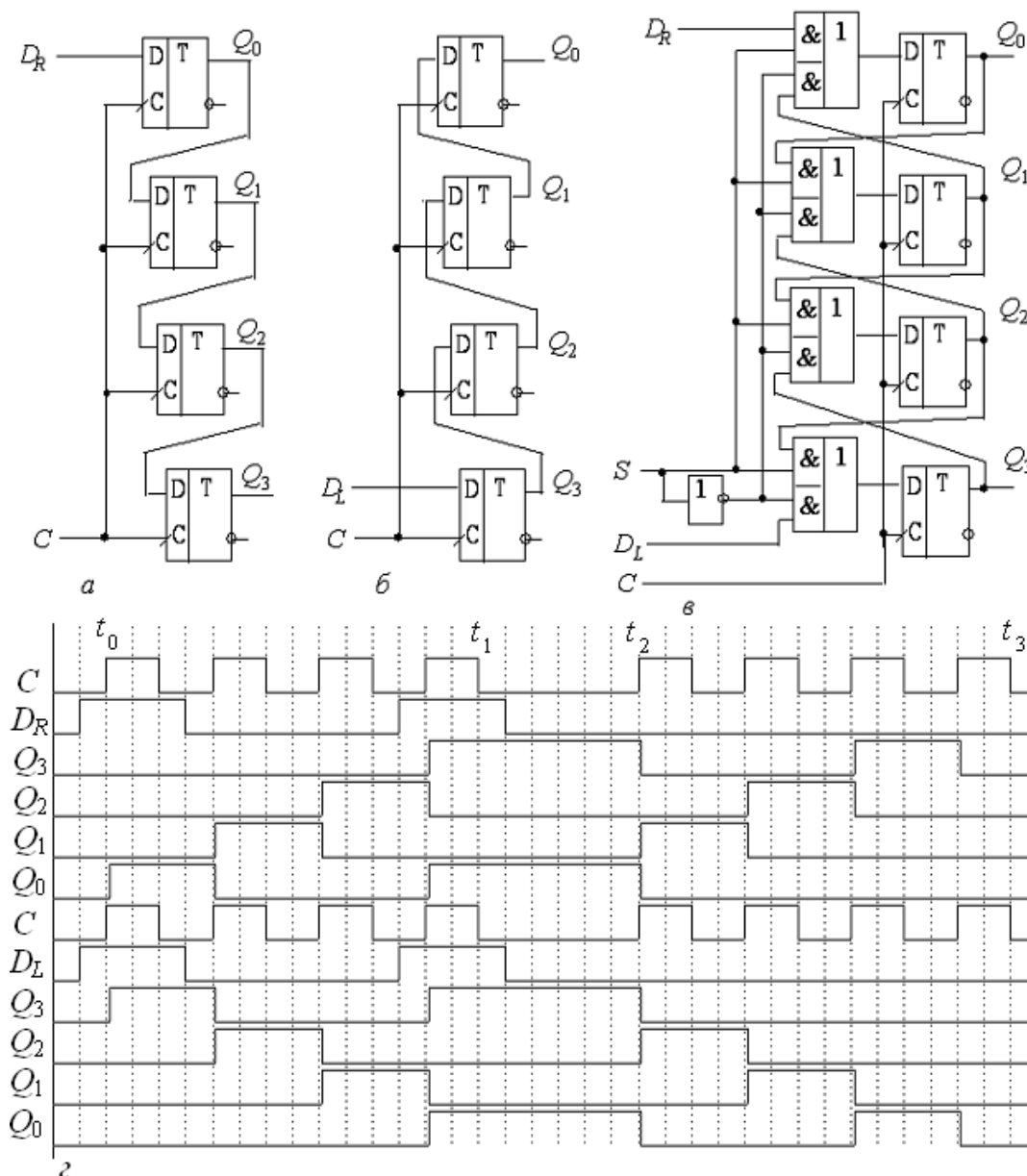


Рис. 4.2. Регістри зсуву та сигналогами їх роботи

Розглянемо для наочності *запис* до цього регістра чотирирозрядного слова $D = d_3d_2d_1d_0 = 1001_2$ (на рис. 4.2, *з* верхні шість часових діаграм). Припустимо, що попередньо регістр обнулено: $Q_3 = Q_2 = Q_1 = Q_0 = 0$. Тоді, з надходженням в момент t_0 *позитивного перепаду* першого синхроімпульсу C до тригерів одночасно (із затримкою на час перемикання одного тригера) записується інформація, присутня на їх входах D , а саме: $Q_0 = D_R = d_3 = 1$, $Q_1 = Q_0 = 0$, $Q_2 = Q_1 = 0$, $Q_3 = Q_2 = 0$. З надходженням позитивного перепаду другого синхроімпульсу C до входу послідовного введення прикладено наступний розряд слова $D_R = d_2 = 0$, а до входів інших тригерів – інформація з виходів попередніх, отже, відбувається запис: $Q_0 = D_R = d_2 = 0$, $Q_1 = Q_0 = d_3 = 1$, $Q_2 = Q_1 = 0$, $Q_3 = Q_2 = 0$. Таким чином, з кожним синхроімпульсом біти вхідного слова просуваються в бік старших розрядів і по закінченні серії з чотирьох синхроімпульсів (момент t_1) все слово виявляється записаним до регістра: $Q_0 = d_0 = 1$, $Q_1 = d_1 = 0$, $Q_2 = d_2 = 0$, $Q_3 = d_3 = 1$. Щоб протягом одного такту біти слова не могли просунутися більше, ніж на один розряд, у регістрах зсуву застосовують тригери тільки з *динамічним керуванням*.

Для *зчитування* слова з виходу Q_3 у послідовному коді необхідно подати ще одну серію з чотирьох синхроімпульсів на інтервалі (t_2, t_3) : перед позитивним перепадом кожного синхроімпульсу на виході Q_3 з'являється черговий розряд. Якщо при цьому до входу послідовного введення прикладено $D_R = 0$, регістр буде *обнулено*, а якщо надходитимуть біти нового слова, воно буде записано під час зчитування попереднього.

Внаслідок того, що всі тригери регістра перемикаються одночасно, *швидкодія* регістра зсуву, як і паралельного, визначається часом перемикання одного тригера.

У регістрі *зворотного зсуву* (зсуву ліворуч), наприклад, чотирирозрядного (див. рис. 4.2, *б*), розряди вхідного слова $D = d_3d_2d_1d_0$, починаючи від молодшого d_0 , надходять до входу послідовного введення D_L , що є входом старшого розряду регістра, а зчитуються з виходу послідовного виведення Q_0 , що є виходом молодшого розряду. Процеси запису і зчитування

(на рис. 4.2, з нижні шість часових діаграм) відбуваються так само, як і в регістрі прямого зсуву, але в протилежному напрямку. Проте напрямок зсуву є відносним: досить перенумерувати розряди для взаємного перетворення регістрів прямого і зворотного зсуву. Про це свідчать і часові діаграми (див. рис. 4.2, з): при зміні напрямку нумерації розрядів одного з регістрів (від старшого розряду до молодшого чи навпаки) діаграми обох регістрів збігаються. Тому й випускають регістри зсуву без поділу на два види.

Принципового значення напрямок зсуву набуває в *реверсивних* регістрах (див. рис. 4.2, в), в яких шляхом перемикання міжрозрядних зв'язків слово просувають у прямому або зворотному напрямку. Дійсно, при керувальному сигналі $S = 1$ пристрій перетворюється на регістр прямого зсуву (див. рис. 4.2, а), а при $S = 0$ – на регістр зворотного зсуву (див. рис. 4.2, б).

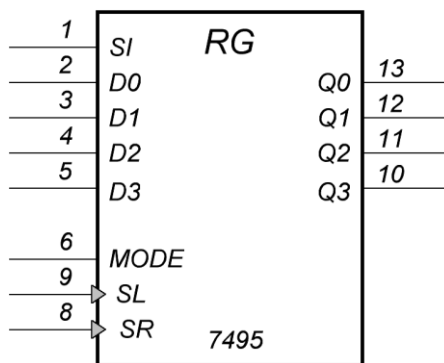


Рис. 4.3. Умовне зображення чотирьохрозрядного регістру SN7495N

Мікросхема SN7495N (K155IP1) – це чотирьохрозрядний регістр зсуву з послідовним або паралельним введенням інформації та паралельним виведенням її. Мікросхема 7495 може працювати в двох режимах: зсуву і завантаження. Мікросхема 7495 переходить в режим роботи зсуву праворуч, коли на вхід Mode (вивід 6) подається напруга низького рівня. Тоді за негативним фронтом тактового імпульсу на вході SR (зсув праворуч) дані будуть зсунуті на один розряд праворуч. У режимі зсуву у мікросхеми 7495 інформація надходить на вхід SI і при першому негативному фронті ($1 \rightarrow 0$) тактового імпульсу на вході SR – на вихід QA, з QA \rightarrow QB, з QB \rightarrow QC, з QC \rightarrow QD, інформація, що міститься на виході QD, або втрачається, або зсувається в

наступний модуль.

Зсув даних в мікросхемі 7495 вліво можливий в тому випадку, коли вихід QB зовні з'єднується з входом A, вихід QC - з входом B і вихід QD - з входом C. Дані надходять на вход D (на вхід Mode подається напруга високого рівня) і по негативному тактовому імпульсу на вході SL зсуваються на один розряд ліворуч. При цьому входи SI (послідовне введення даних) і SR не використовуються.

В режимі паралельного запису даних в мікросхемі 7495 на вхід Mode подається напруга високого рівня, і дані зі входів A, B, C і D по негативному фронту тактового імпульсу на вході SL записуються в регістр.

Таким чином, вхід SL виконує подвійне завдання: зсуває інформацію ліворуч (Mode=0) і записує паралельні дані в регістр (Mode=1).

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

1. Включити комп'ютер.
2. Запустити програму Proteus 7 Professional.
3. На робочому полі програми зібрати схему для дослідження паралельного регістра (рис. 4.4).

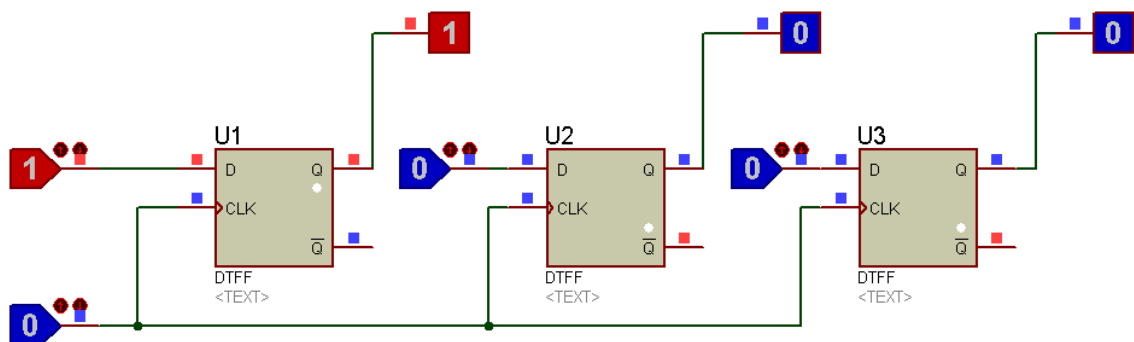


Рис. 4.4. Схема паралельного регістра на D-тригерах

4. Провести запис у регістр всього набору можливих трирозрядних двійкових чисел.
5. На робочому полі програми зібрати схему для дослідження послідовного регістра (рис. 4.5).

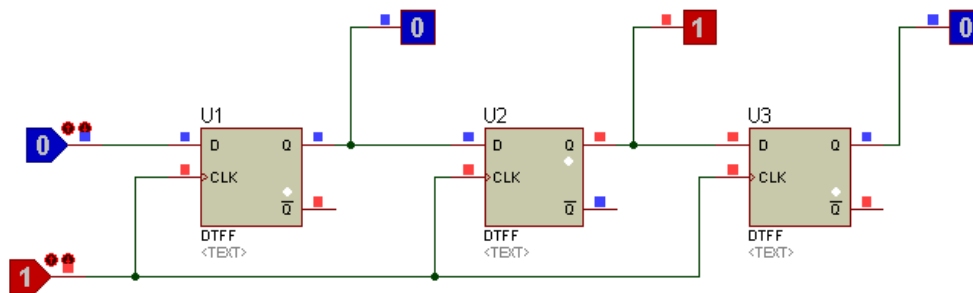


Рис. 4.5. Схема послідовного регістра на D-тригерах

6. Заповнити таблицю станів роботи послідовного регістра:

Вхід «Дані»	Q2	Q1	Q0
0			
1			
1			
0			
1			
0			
1			

7. Зібрати схему для дослідження універсального регістра (рис. 4.6).

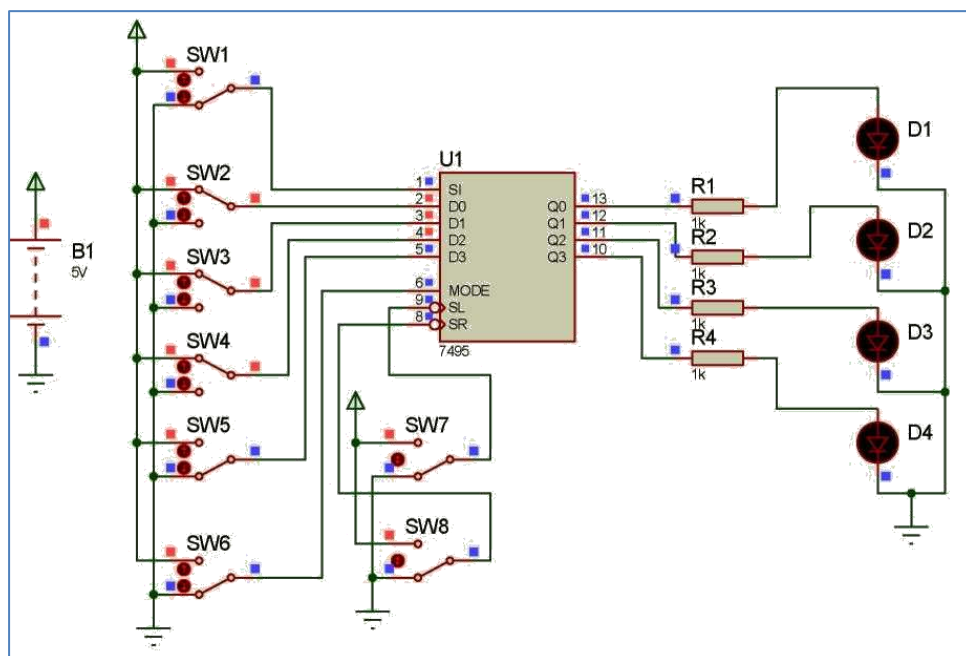


Рис. 4.6. Схема для дослідження універсального регістра

8. Провести дослідження паралельного і послідовного режимів роботи універсального регістра. Навести таблицю станів та screenshot роботи

регістра в обох режимах

9. Зробити висновки по роботі.

Висновки: _____

10. Надати письмові відповіді на контрольні запитання.

КОНТРОЛЬНІ ПИТАННЯ

1. Поясніть, що таке регістр.
2. Наведіть схему і поясніть принцип роботи паралельного регістра на D - тригерах.
3. Наведіть схему і поясніть принцип роботи послідовного регістра на D - тригерах.
4. Поясніть, що таке універсальний регістр.
5. Наведіть умовне графічне позначення і режими роботи універсального регістра (на прикладі мікросхеми SN7495).

Практична робота № 5

Тема: Дослідження лічильників імпульсів.

Мета роботи: Вивчення принципів побудови і роботи регістрів різного типу.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Лічильник імпульсів (Pulse Counter) – електронний пристрій, що виконує підрахунок числа імпульсів, що надійшли на його вхід. Лічильники імпульсів побудовані на базі тригерів.

Кількість комбінацій вихідних сигналів, що знімаються з виходів тригерів лічильника або максимальну кількість імпульсів N , яку може підрахувати

лічильник без його переповнення, для двійкових лічильників $N = 2^n$, де n -число розрядів (тригерів) лічильника, називають коефіцієнтом або модулем лічби.

У разі переповнення лічильника ($N > 2^n$) лічильник скидається, а рахунок циклічно поновлюється. Сукупність логічних одиниць і нулів на виходах тригерів (виходах лічильника) являє собою n -розрядне двійкове число, що визначає кількість імпульсів, що пройшли через лічильник.

Найпростіші лічильники імпульсів складаються з ланцюжка послідовно з'єднаних Т-тригерів, кожен з яких ділить частоту вхідного сигналу на два. Кожен з тригерів цього ланцюжка називають розрядом лічильника. Так, наприклад, чотирьох розрядні лічильники дозволяють отримати на виходах тригерів імпульси, частота яких нижче частоти вхідного сигналу в 2, 4, 8 і 16 разів або в 2^n раз, де n -номер тригера в ланцюжку. Такі лічильники отримали назву лічильники-подільники.

За принципом дії лічильники діляться на підсумовуючі (прямої лічби), віднімаючі (зворотної лічби) та реверсивні.

Підсумовуючий лічильник збільшує своє значення на одиницю при надходженні на його вхід чергового вхідного імпульсу.

Віднімаючий лічильник зменшує свій вміст на одиницю при надходженні на його вхід чергового вхідного імпульсу.

Реверсивний лічильник є поєднанням лічильників прямої та зворотної лічби. Для розпізнавання напрямку рахунку такий лічильник має додатковий вхід, який переключає режим рахунку, або має два роздільних входи для подачі на них імпульсів прямої та зворотної лічби.

Лічильники імпульсів поділяються на:

- асинхронні (послідовні);
- синхронні з асинхронним перенесенням (паралельні з послідовним перенесенням);
- синхронні (паралельні).

Асинхронні лічильники – це послідовні лічильники, які складаються з ланцюжка тригерів, які працюють в рахунковому режимі. Вихідний сигнал попереднього тригера служить вхідним сигналом для наступного.

Всі тригери включаються послідовно, отже, і виходи лічильника також перемикаються послідовно, рис. 5.1. Кожен наступний розряд перемикається з затримкою щодо попереднього, рис. 5.2. Чим більше розрядів має лічильник, тим більший час йому потрібно на повне перемикання всіх розрядів.

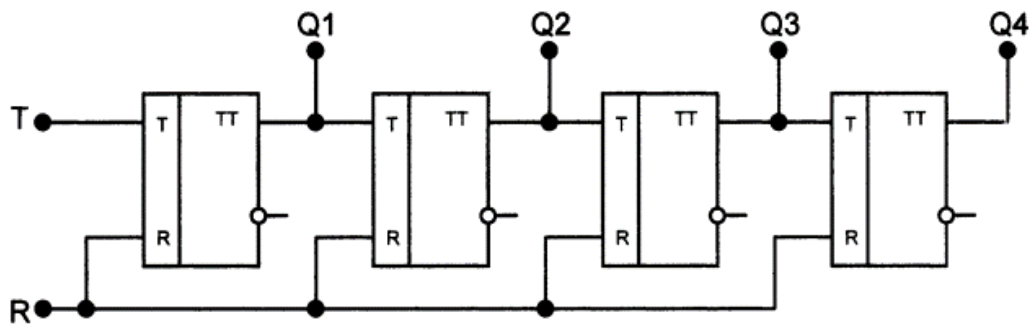


Рис. 5.1. Схема асинхронного (послідовного) підсумовуючого лічильник на Т-тригерах

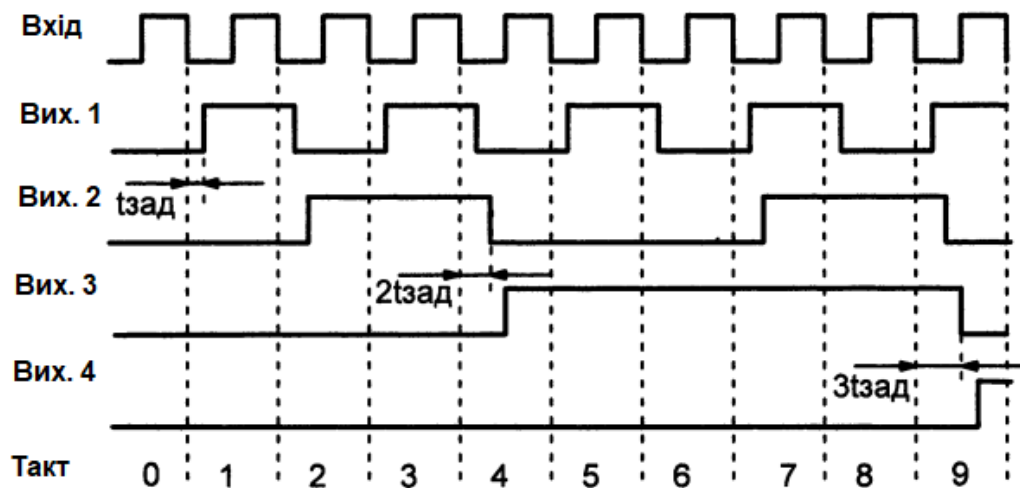


Рис. 5.2. Часова діаграма роботи чотирьох розрядного асинхронного лічильника

Затримка перемикання кожного розряду приблизно дорівнює затримці тригера. Повна затримка встановлення коду на виході лічильника дорівнює затримці одного розряду, помноженої на число розрядів лічильника.

При періоді вхідного сигналу, меншому повної затримки встановлення коду лічильника, правильний код на його виході не встигне встановитися. Це накладає обмеження на максимальну частоту вхідного сигналу.

Оскільки переключення розрядів відбувається з прогресуючою затримкою, сигнали на виходах лічильника з'являються не одночасно з вхідним сигналом, тобто асинхронно.

Мікросхеми асинхронних лічильників не дуже розповсюджені. Це, наприклад, ТТЛ-мікросхеми типу ІЕ2 (чотирьох розрядний двійково-десятковий лічильник), ІЕ5 (чотирьох розрядний двійковий лічильник) та ІЕ19 (здвоєний чотирьох розрядний лічильник), рис. 5.3.

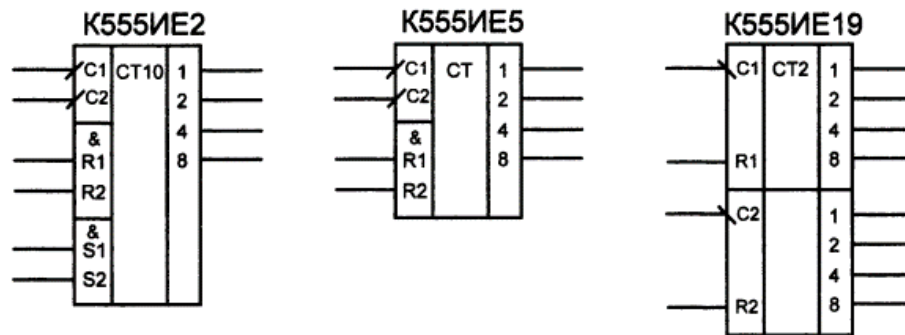


Рис. 5.3. Асинхронні лічильники серії ТТЛ

Дані типи лічильників мають входи скидання в нуль (вхід R), вхід установки в 9 (вхід S тільки у ІЕ2), рахунковий або тактовий вхід (вхід С) та виходи, які можуть позначатися як номери розрядів (0, 1, 2, 4) або як вага кожного розряду (1, 2, 4, 8). Усі асинхронні лічильники працюють за негативним фронтом вхідного сигналу С (заднім фронтом позитивного вхідного сигналу).

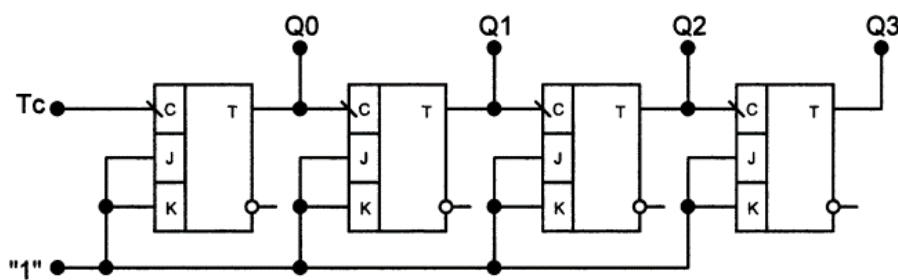


Рис. 5.4. Схема підсумовуючого лічильника на основі JK-тригерів

Підсумовуючий лічильник на основі JK-тригерів показаний на рис. 4. Таблиця станів тригерів асинхронного лічильника, що працює на додавання, представлена в табл. 5.1.

Табл. 5.1. Таблиця станів тригерів асинхронного лічильника, що працює на додавання

Число вхідних імпульсів	Стан виходів асинхронного лічильника імпульсів			
N	Q3	Q2	Q1	Q0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16	0	0	0	0

Схема двійкового лічильника імпульсів на Т-тригерах, що працює на віднімання, наведена на рис. 5.5. Інформація на Т-входи кожного наступного тригера подається з інверсією виходів попередніх тригерів. Інформація про кількість імпульсів знімається з прямих виходів тригерів.

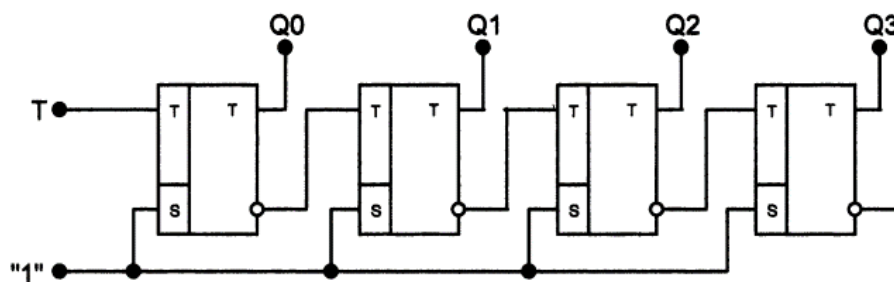


Рис. 5.5. Схема асинхронного лічильника на Т-тригерах, що працює на віднімання

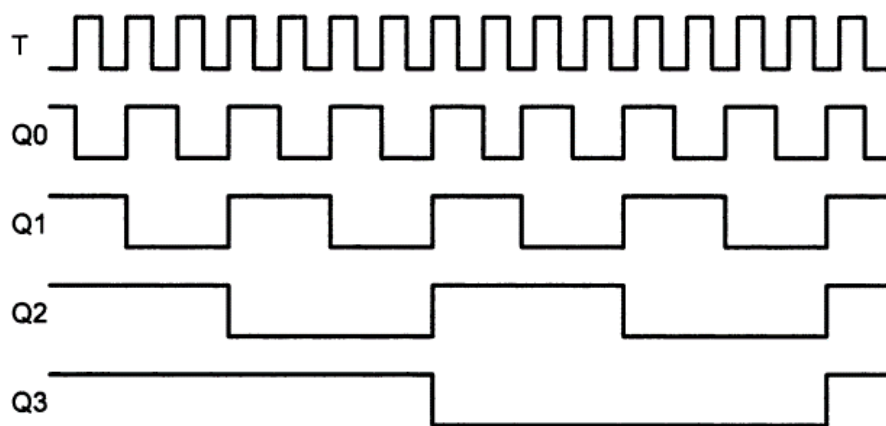


Рис. 5.6. Часові діаграми роботи лічильника на віднімання

Перед початком рахунку виходи всіх тригерів повинні бути встановлені у значення логічної одиниці, для чого на об'єднаний установчий вхід S подається імпульс установки.

На рис. 5.6 зображені часові діаграми роботи лічильника на віднімання, а таблиця станів тригерів асинхронного лічильника, що працює на віднімання, представлена в табл. 5.2.

Табл. 5.2. Таблиця станів тригерів асинхронного лічильника, що працює на віднімання

Число вхідних імпульсів	Стан виходів асинхронного лічильника імпульсів			
	Q3	Q2	Q1	Q0
0	1	1	1	1
1	1	1	1	0
2	1	1	0	1
3	1	1	0	0
4	1	0	1	1
5	1	0	1	0
6	1	0	0	1
7	1	0	0	0
8	0	1	1	1
9	0	1	1	0
10	0	1	0	1
11	0	1	0	0
12	0	0	1	1
13	0	0	1	0
14	0	0	0	1
15	0	0	0	0
16	1	1	1	1

Схема двійкового лічильника імпульсів на D-тригерах, що працює на віднімання, наведена на рис. 5.7. На відміну від раніше розглянутого лічильника на Т-тригерах після нульового стану всіх тригерів, з приходом першого синхроімпульсу, вони встановлюються в стан логічної одиниці.

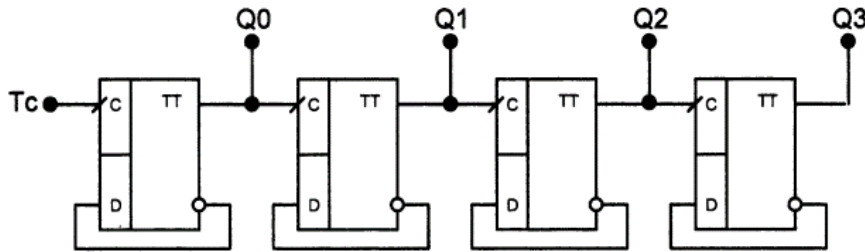


Рис. 5.7. Схема асинхронного лічильника на D-тригерах, що працює на віднімання

Надходження другого синхроімпульсу призводить до зменшення вихідного коду на одиницю. Після надходження восьмого (для трьох розрядних лічильників) або шістнадцятого (для чотирьох розрядних лічильників) імпульсу, всі тригери скидаються, і цикл рахунку повторюється.

Синхронні лічильники з асинхронним перенесенням відрізняються тим, що перемикання розрядів здійснюється одночасно, а сигнал перенесення виконується з деякою затримкою, рис. 5.8. Швидкодія даних лічильників вище, ніж асинхронних, але нижче ніж чисто синхронних.

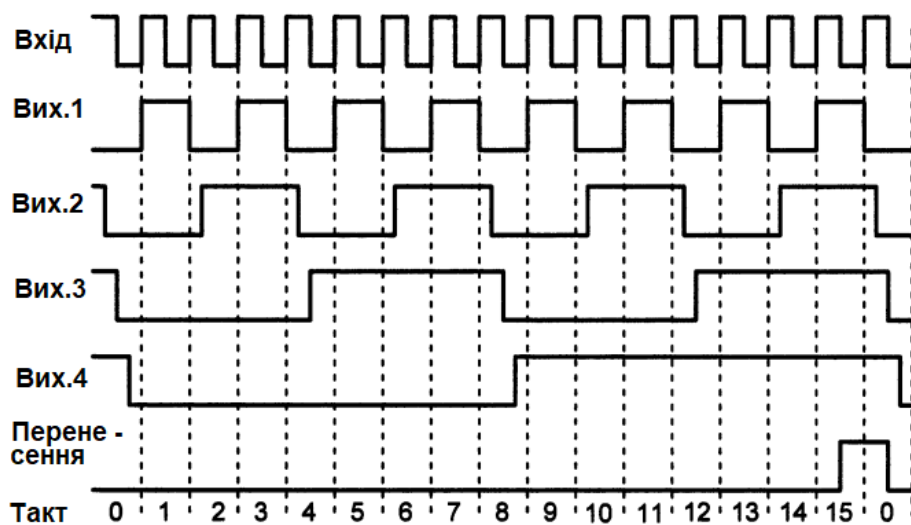


Рис. 5.8. Часова діаграма роботи чотирьох розрядного синхронного лічильника з асинхронним перенесенням з урахуванням затримок

Для об'єднання декількох синхронних лічильників з метою збільшення числа їх розрядів (каскадування) використовується спеціальний вихідний сигнал перенесення. В залежності від принципів формування цього сигналу і від принципів його використання синхронні (паралельні) лічильники діляться на лічильники з асинхронним (послідовним) перенесенням і лічильники з синхронним (паралельним) перенесенням (або повністю синхронні лічильники).

Типовими представниками синхронних лічильників з асинхронним перенесенням є мікросхеми реверсивних лічильників К555ІЕ6 (рахунок від 0 до 9) та К555ІЕ7 (рахунок від 0 до 15), рис. 5.9.

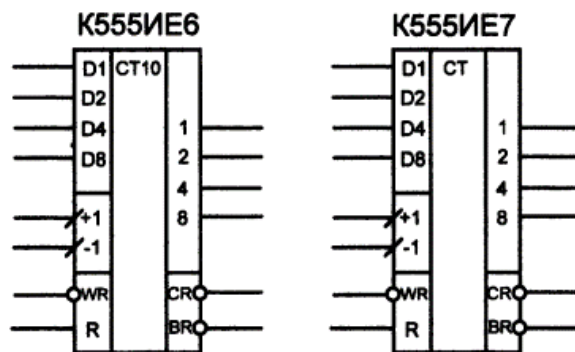


Рис. 5.9. Синхронні лічильники з асинхронним перенесенням

Синхронні лічильники з асинхронним перенесенням знайшли більш широке застосування, ніж асинхронні лічильники. Це подільники частоти, лічильники числа імпульсів, вимірювачі інтервалів часу, формувачі послідовності імпульсів.

Синхронні лічильники (паралельні лічильники) відрізняються від асинхронних тим, що перемикання розрядів в них йде паралельно, без затримки.

Це надшвидкодіючі, але і найскладніші в плані управління лічильники. У таких лічильниках сигнал перенесення виробляється тоді, коли всі виходи лічильника встановлюються в одиницю (при прямому рахунку) або в нуль (при зворотному).

Також при включенні декількох лічильників для збільшення розрядності тактові входи С об'єднуються, а сигнал перенесення подається на вхід дозволу

рахунку кожного наступного лічильника, рис. 5.10.

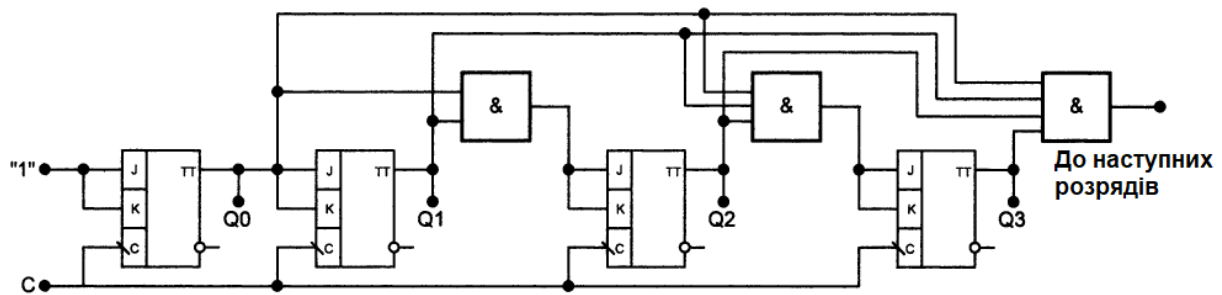


Рис. 5.10. Паралельний підсумовуючий синхронний лічильник на JK-тригерах

Оскільки лічильник має загальну лінію синхронізації, стан тригерів змінюється синхронно, тому тригери, які за синхроімпульсом повинні змінити свій стан, роблять це одночасно, що істотно підвищує швидкодію синхронних лічильників. У серії промислових мікросхем, рис. 5.11, входять кілька типів синхронних лічильників, які розрізняються способом рахунку (двійкові або двійковий-десяткові, реверсивні або нереверсивні) та керуючими сигналами (відсутність або наявність сигналу скидання). Всі лічильники даного типу мають входи перенесення і каскадування.

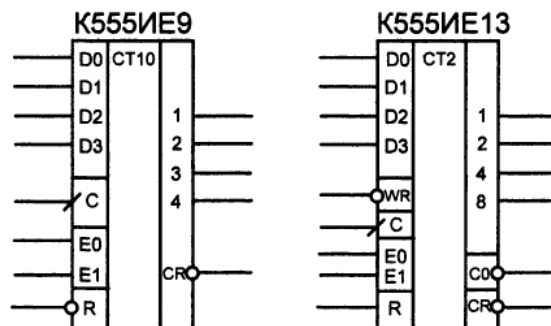


Рис. 5.11. Синхронні лічильники серії ТТЛ

Реверсивний лічильник працює як в прямому, так і в зворотному напрямку лічби імпульсів в залежності від рівня сигналу керування.

На рис. 5.12 показаний приклад схеми асинхронного двійкового реверсивного лічильника. Напрямок лічби здійснюється сигналом на вході V: при $V = 0$ лічильник працює в режимі підсумовування; при $V = 1$ – в режимі віднімання.

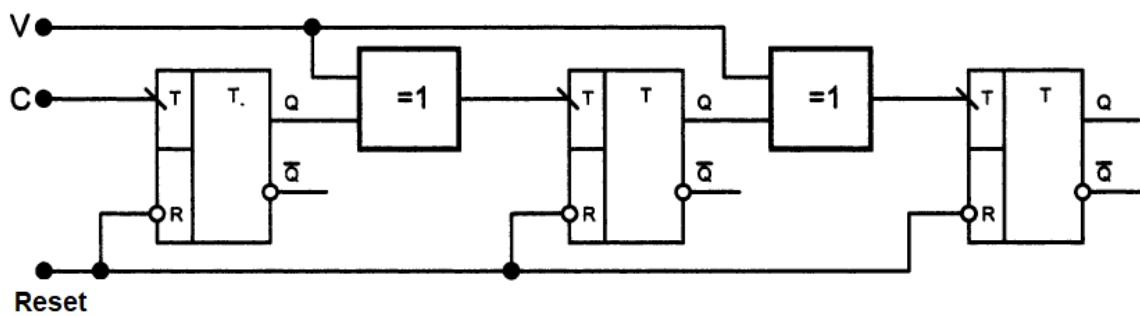


Рис. 5.12. Асинхронний двійковий реверсивний лічильник

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

1. Включити комп'ютер.
2. Запустити програму Proteus 7 Professional.
3. На робочому полі програми зібрати схему для дослідження двійкового підсумовуючого лічильника додавання на JK-тригерах (рис. 5.13).

Скласти таблицю станів роботи двійкового підсумовуючого лічильника (табл. 5.3).

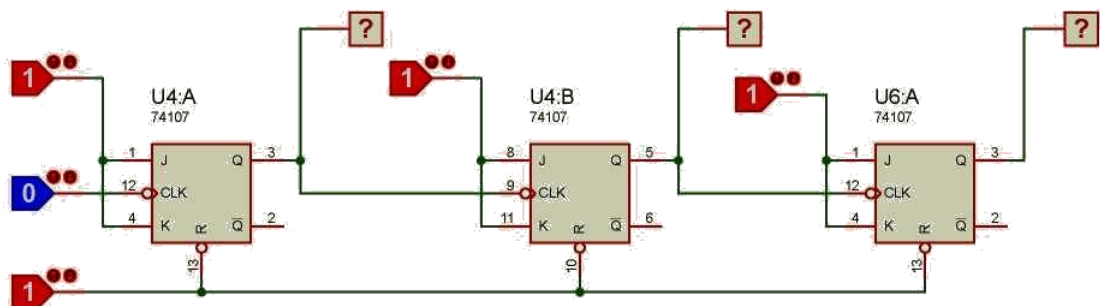


Рис. 5.13. Схема двійкового підсумовуючого лічильника на JK-тригерах

Табл. 5.3. Таблиця станів тригерів двійкового підсумовуючого лічильника

Десятковий еквівалент	Q2	Q1	Q0

4. На робочому полі програми зібрати схему для дослідження двійкового віднімаючого лічильника на JK-тригерах (рис. 5.14). Скласти таблицю станів роботи двійкового віднімаючого лічильника (табл. 5.4).

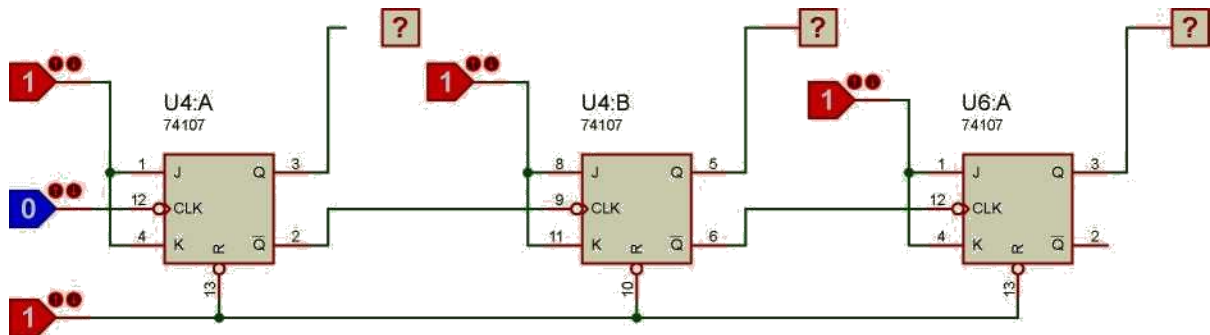


Рис. 5.14. Схема двійкового віднімаючого лічильника на JK-тригерах

Табл. 5.4. Таблиця станів тригерів двійкового віднімаючого лічильника

Десятковий еквівалент	Q2	Q1	Q0

5. На робочому полі програми зібрати схему для дослідження реверсивного чотирьох розрядного лічильника/дільника з можливістю передустановки (рис. 5.15).

6. Написати призначення входів і виходів лічильника:

D0, D1, D2, D3_____

UP_____

DN_____

PL_____

MR_____

Q0, Q1, Q2, Q3_____

TCU_____

TCD_____

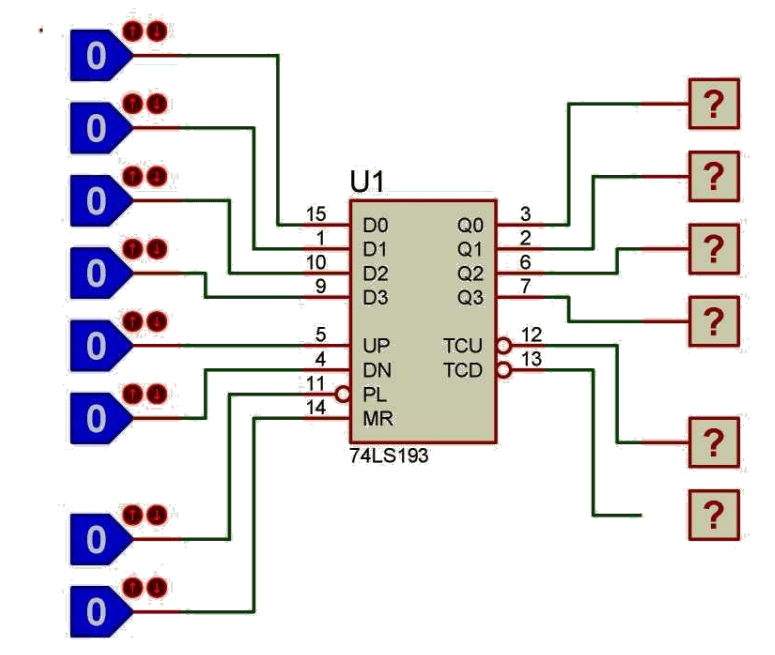


Рис. 5.15. Схема реверсивного чотирьох розрядного лічильника/ділника з можливістю передустановки

7. Провести дослідження всіх режимів роботи мікросхеми SN74LS193 з детальним описом:

8. Зробити висновки по роботі.

Висновки: _____

9. Надати письмові відповіді на контрольні запитання.

КОНТРОЛЬНІ ПИТАННЯ

1. Поясніть, що таке лічильник імпульсів.
2. Наведіть схему і поясніть принцип роботи асинхронного підсумовуючого

лічильника на JK-тригерах.

3. Наведіть схему і поясніть принцип роботи асинхронного віднімаючого лічильника на JK-тригерах.
4. Наведіть схему і поясніть принцип роботи 4-каскадного подільника частоти на T-тригерах.
5. Наведіть схему і поясніть принцип роботи дільника частоти на 8, 10, 12.

Практична робота № 6

Тема: Дослідження основних комбінаційних пристроїв.

Мета роботи: Вивчити основні робочі режими та функції основних комбінаційних пристроїв: дешифраторів, шифраторів, мультиплексорів і демультиплексорів.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Комбінаційні мікросхеми є комбінацією з простих логічних мікросхем. Але, на відміну від послідовних, вони не володіють пам'яттю, а їх вихідні сигнали ніяк не залежать від комбінації попередніх вхідних сигналів.

Дешифратор (декодер, Decoder) – цифровий пристрій, що має n входів і m виходів і перетворює вхідний код в сигнал на одній окремій вихідній лінії. Дешифратор називають повним, якщо він має кількість виходів m , пов'язаних з кількістю розрядів n вхідного двійкового числа співвідношенням $m = 2^n$.

На кожному з його виходів з'являється логічна одиниця тільки в тому випадку, якщо на вхід приходить відповідна комбінація вхідних сигналів.

Функція мікросхем дешифраторів полягає в тому, щоб перетворювати вхідний двійковий код в номер вихідного сигналу, кількість яких відповідає кількості станів двійкового коду, тобто 2^n , де n - кількість розрядів двійкового коду (кількість інформаційних входів дешифратора).

Найпростіший двохранрядний дешифратор, виконаний зі стандартних логічних елементів І і НЕ, показаний на рис. 6.1. Таблиця істинності двохранрядного дешифратора представлена нижче (табл. 6.1).

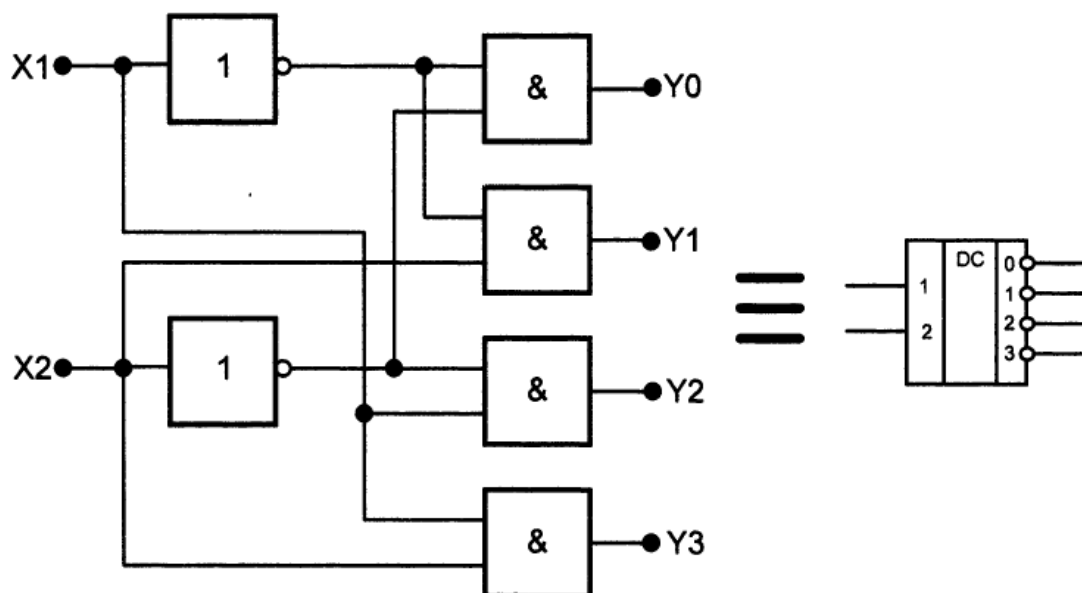


Рис. 6.1. Демонстраційна схема двохранрядного дешифратора та його умовне графічне позначення

Табл. 6.1. Таблиця станів двохранрядного дешифратора

Номер вхідного сигналу	Вхід		Вихід			
	X1	X0	Y3	Y2	Y1	Y0
0	0	0	0	0	0	1
1	0	1	0	0	1	0
2	1	0	0	1	0	0
3	1	1	1	0	0	0

У стандартних серіях мікросхем існують дешифратори на 4, 8 або 16 виходів, відповідно вони мають 2, 3 або 4 входи. Ще відмінності між мікросхемами включають в себе входи управління і типи виходів мікросхем (звичайний вихід або вихід із загальним колектором ЗК).

Входи дешифраторів позначають цифрами, яким відповідає вага розряду двійкового числа (1, 2, 4 або 8), а виходи також позначають цифрами (1, 2, 3...m). Мікросхеми стандартних дешифраторів, наприклад, К555ИД14, К555ИД7, К555ИД3, рис. 6.2, мають інформаційні входи 1, 2, 4, 8, входи

дозволу C1, C2, C3, об'єднані по функції I, а також виходи від 0 до 15. Різниця між цими мікросхемами складається в кількості входів і виходів.

Мікросхеми дешифратори використовують для: дешифрування входних кодів; селекція кодів; мультиплексування ліній.

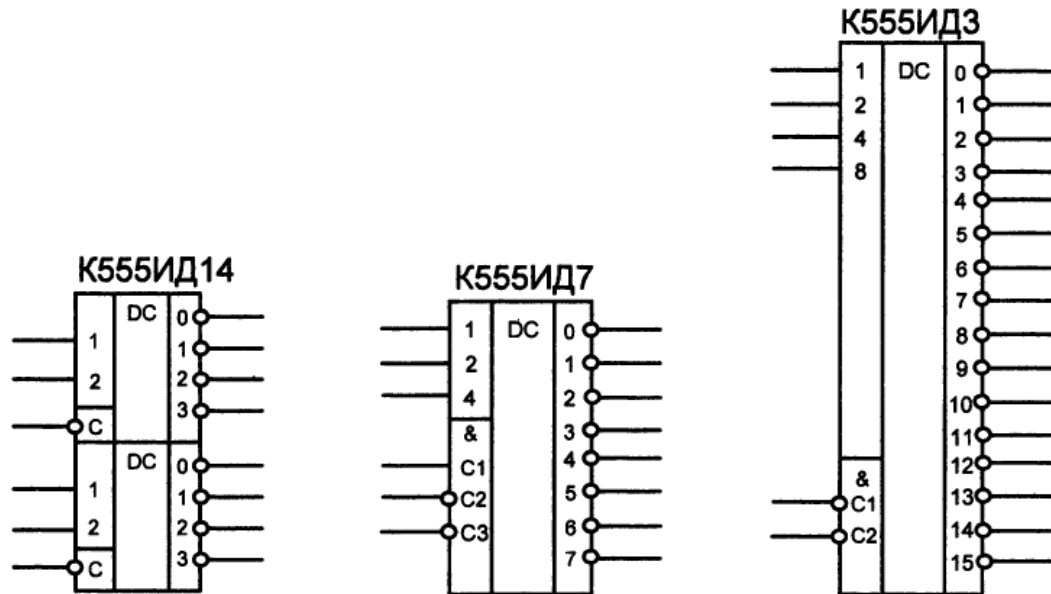


Рис. 6.2. Мікросхеми ТТЛ-дешифраторів

Шифратор – (кодер, Encoder, Coder) цифровий пристрій, що має n входів і m виходів, перетворює позиційний n -розрядний код в m -розрядний код. Шифратор називають повним, якщо виконується умова $n = 2^m$.

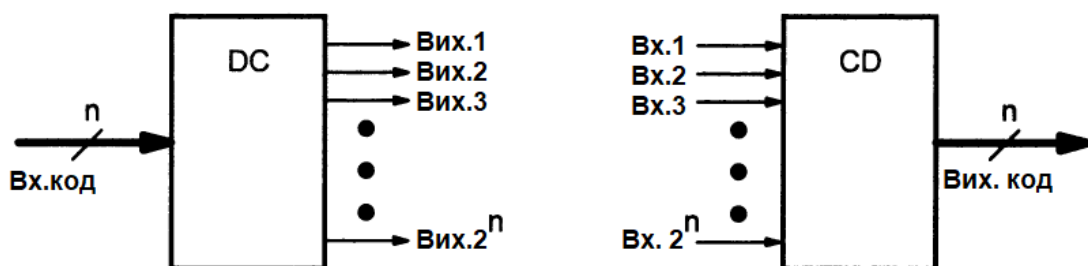


Рис. 6.3. Порівняння функцій дешифратора (зліва) і шифратора (праворуч)

Мікросхеми шифраторів застосовуються значно рідше, ніж дешифратори. Шифратори виконують функцію, зворотну мікросхемам дешифраторів і мають від 4 до 16 (найчастіше 8) інформаційних входів, від 2 до 4 виходів (найчастіше 3), а також кілька входів і виходів дозволу роботи, рис. 6.3.

Приклад побудови шифратора для перетворення десятирозрядного одиничного коду (десяткових чисел від 0 до 9) в двійковий код наведено на рис. 6.4. Передбачається, що сигнал, відповідний логічній одиниці, в кожен момент часу подається тільки на один вхід. Умовне графічне позначення такого шифратора і таблиця відповідності коду наведені на рис. 6.5.

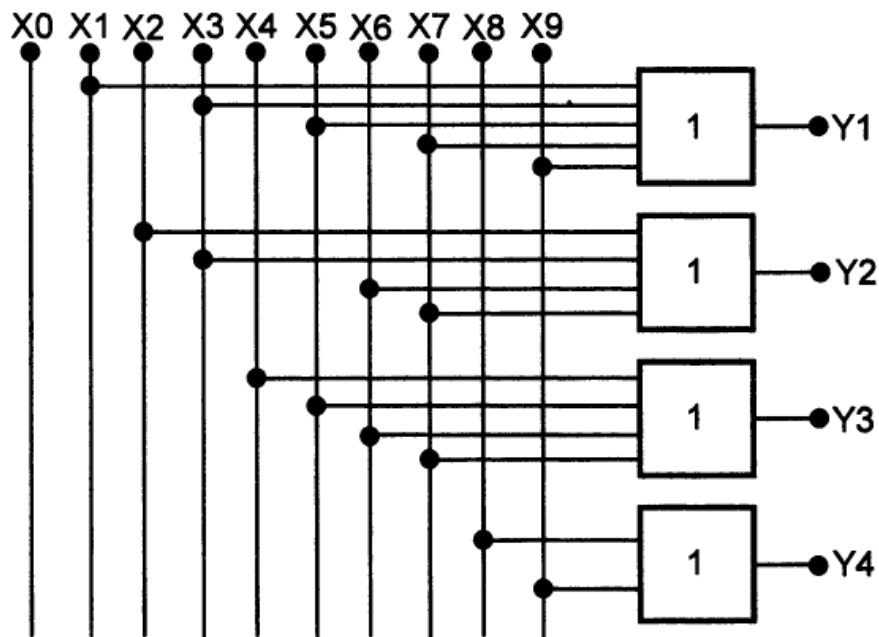


Рис. 6.4. Еквівалентна схема шифратора на основі елементів АБО

Табл. 6.2. Таблиця станів шифратора

Десяткове число	Двійковий код 8 — 4 — 2 — 1			
	Y1	Y2	Y3	Y4
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

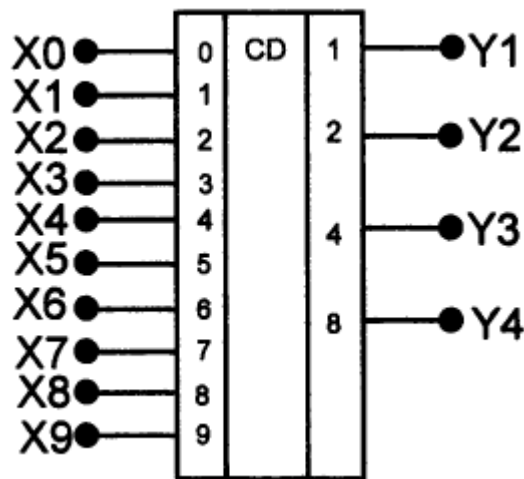


Рис. 6.5. Умовне графічне позначення шифратора

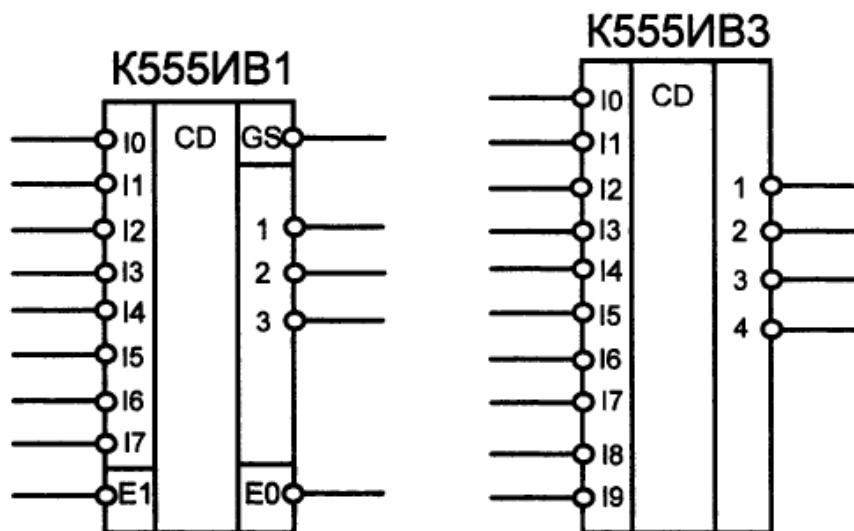


Рис. 6.6. Мікросхеми ТТЛ-шифраторів

Типовими представниками ТТЛ-мікросхем шифраторів є K555IB1 і K555IB3, рис. 6.6.

Мікросхема шифратора K555IB1, рис. 6.6, містить 8 інформаційних входів і три виходи, а також вхід дозволу EI, вихід ознаки приходу будь-якого вхідного сигналу GS і вихід перенесення E0 для об'єднання декількох шифраторів для збільшення їх розрядності. Робота шифратора дозволяється тільки при низькому логічному рівні на вході EI; при високому рівні на ньому на всіх виходах встановлюється рівень логічної одиниці. При відсутності сигналів на входах на виході GS формується логічна одиниця, а на виході E0

- логічний нуль. Таблиця станів шифратора К555ІВ1 представлена нижче (табл. 6.3).

Табл. 6.3. Таблиця станів шифратора К555ІВ1

Входи									Виходи				
ЕІ	0	1	2	3	4	5	6	7	GS	4	2	1	ЕО
1	X	X	X	X	X	X	X	X	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	X	X	X	X	X	X	X	0	0	0	0	0	1
0	X	X	X	X	X	X	0	1	0	0	0	1	1
0	X	X	X	X	X	0	1	1	0	0	1	0	1
0	X	X	X	X	0	1	1	1	0	0	1	1	1
0	X	X	X	0	1	1	1	1	0	1	0	0	1
0	X	X	0	1	1	1	1	1	0	1	0	1	1
0	X	0	1	1	1	1	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	1	0	1	1	1	1

Шифратори застосовують для скорочення кількості сигнальних ліній при передачі сигналів на великі відстані.

Мультиплексор (селектор, Multiplexer) - цифровий пристрій, призначений для почергового передачі на один вихід сигналу з декількох входів.

Мультиплексор має декілька входів – каналів мультиплексора та кілька виходів, які називаються розрядами мультиплексора. Механічний аналог мультиплексора показаний на рис. 6.7, а на рис. 6.8 - електронний варіант його виконання на елементах І та АБО.

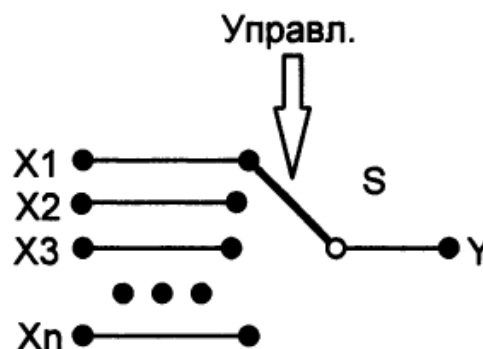


Рис. 6.7. Механічний аналог мультиплексора

Двійковий код, що впливає на адресні входи, відкриває одну з схем І, яка з'єднає з виходом відповідної вхідної лінії. Інформація на виході визначається станом обраного вхідного каналу і не залежить від стану інших каналів.

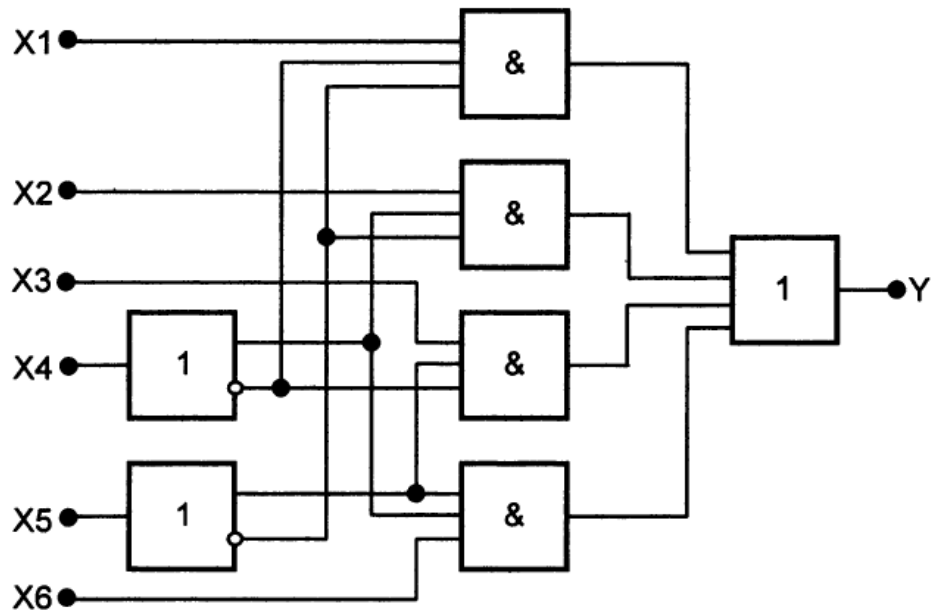


Рис. 6.8. Еквівалентна схему мультіплексора на основі елементів І та АБО

Кількість каналів в мікросхемах мультіплексорах може бути від 2 до 16, а кількість розрядів - від 1 до 4. Управління роботою мультіплексора здійснюється за допомогою адресних входів, на які в двійковому коді надходить номер каналу, з якого в даний момент необхідно передати інформацію на вихід, рис. 6.9.

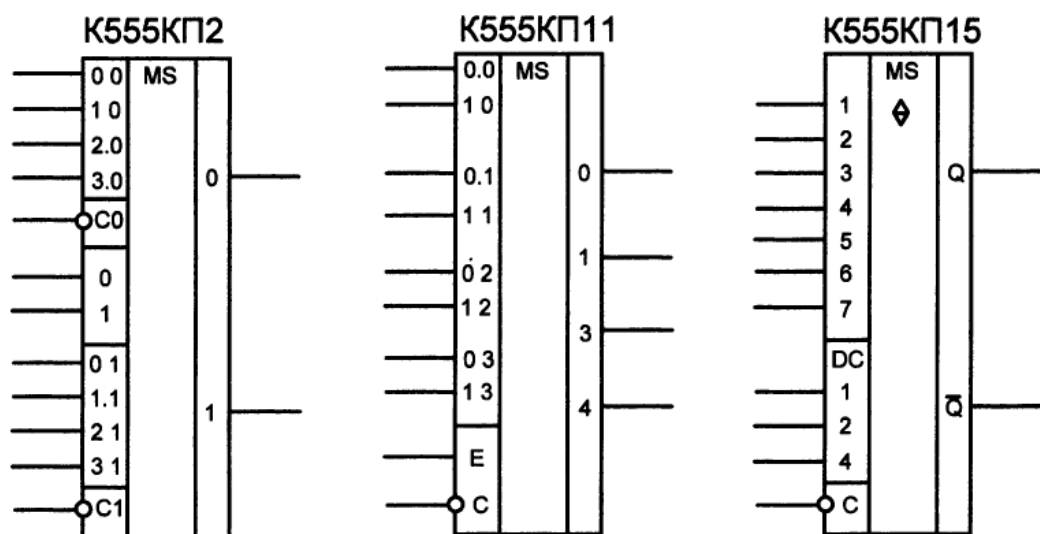


Рис. 6.9. Мікросхеми мультіплексорів

Мікросхеми мультиплексори мають звичайні виходи або 3К, а також входи дозволу С або ЕZ. Часто в одній мікросхемі з'єднані два мультиплексора, наприклад, К555КП2. Роботу мультиплексора розглянемо на прикладі мікросхеми К555КП15.

Мікросхема К555КП15 – восьмиканальний однорозрядних мультиплексор, що дозволяє реалізувати восьмипозиційний перемикач сигналів на один напрямок. Таблиця станів 8-канального мультиплексора К555КП15 представлена у (табл. 6.4).

Табл. 6.4. Таблиця станів 8-канального мультиплексора К555КП15

Входи				Виходи	
4	2	1	С	Q	-Q
X	X	X	1	Z	Z
0	0	0	0	D0	-D0
0	0	1	0	D1	-D1
0	1	0	0	D2	-D2
0	1	1	0	D3	-D3
1	0	0	0	D4	-D4
1	0	1	0	D5	-D5
1	1	0	0	D6	-D6
1	1	1	0	D7	-D7

Сигнали на входах 0-7 позначені D0-D7; Q – прямий вихід, -Q – інверсний; Z – третій стан.

Коли на вході дозволу С високий логічний рівень, то на виходах – прямому та інверсному високоімпендансний стан, тобто сигнал відсутній. Активним є низький логічний рівень на вході С. У цьому випадку сигнал на виходах відповідає сигналу на тому вході, номер якого в двійковому коді задається на входах 1, 2, 4.

Демультимплексор (антімультиплексор, Demultiplexer) – цифровий пристрій, призначений для перемикання сигналу з одного інформаційного входу на один з m інформаційних виходів. Демультимплексор виконує функцію, зворотну функції мультиплексора.

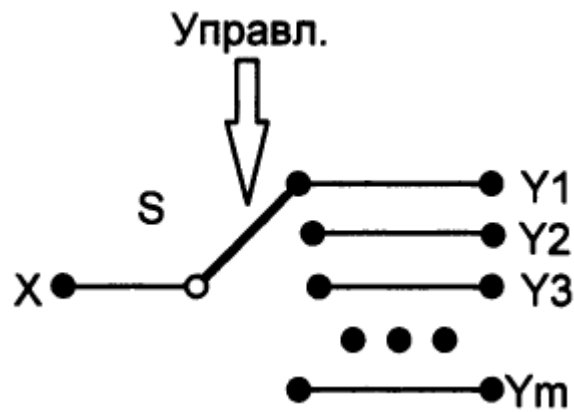


Рис. 6.10. Механічний аналог демультиплексор

Номер виходу, на який в кожен такт машинного часу передається значення вхідного сигналу, визначається адресним кодом. Адресні входи m і інформаційні входи n пов'язані співвідношенням $n = 2^m$. На рис. 6.10 показаний механічний аналог демультиплексор. Демультиплексори, рис. 6.11, застосовують для: комутації окремих ліній і багаторозрядних шин; перетворення послідовного коду в паралельний; реалізації логічних функцій.

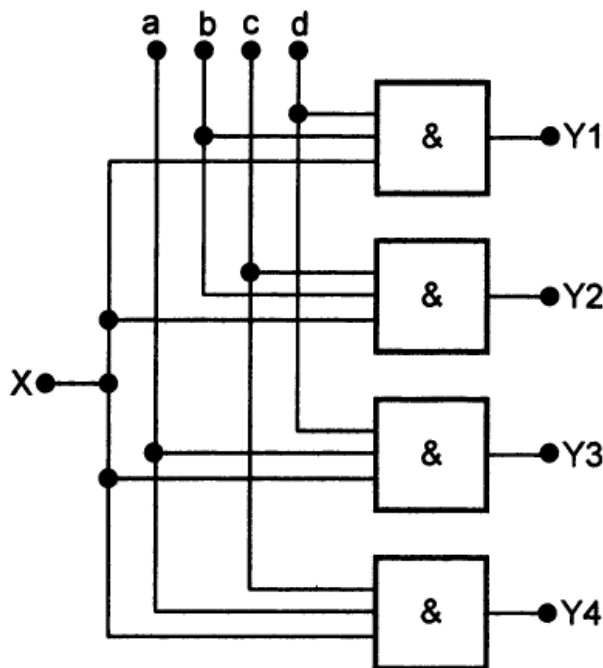


Рис. 6.11. Еквівалентна схема демультиплексора

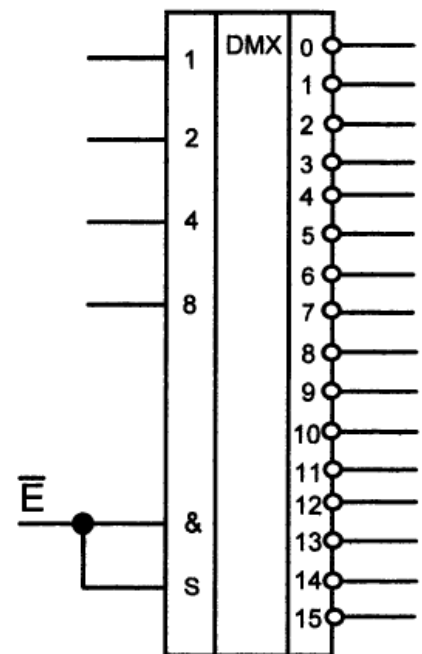


Рис. 6.12. Умовне зображення дешифратора 155ИД3, що працює в режимі демультиплексора

Для збільшення кількості каналів демультіплексори можна об'єднувати. Як самостійні вироби на інтегральних мікросхемах демультіплексори не випускають. Їх функцію реалізують на дешифратор, що мають входи стробування.

Щоб показати, що дешифратор працює в режимі демультіплексора, в середньому полі умовного графічного позначення ставляться літери DMX або DMS (Demultiplexer), рис. 6.12.

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

1. Включити комп'ютер.
2. Запустити програму Proteus 7 Professional.
3. На робочому полі програми зібрати схему для дослідження лінійного дешифратора на два входи $X1$ і $X2$ та чотири прямих виходи (f_0, f_1, f_2, f_3) (рис. 6.13).

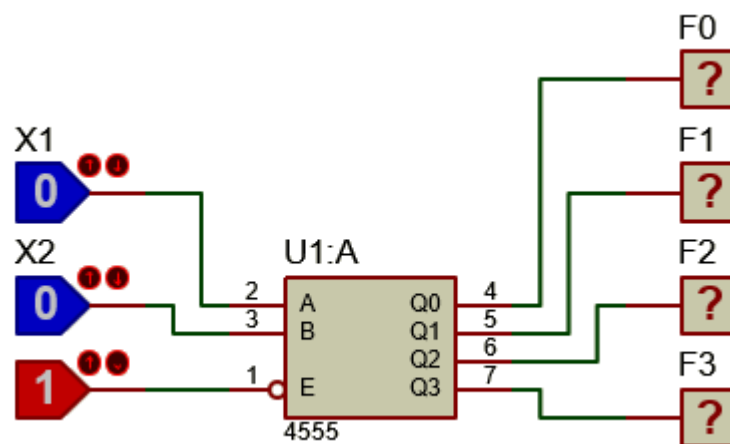


Рис. 6.13. Схема лінійного дешифратора на два входи $X1$ і $X2$ та чотири прямих виходи ($F0, F1, F2, F3$)

4. Експериментальним шляхом зняти таблицю станів роботи дешифратора (табл. 6.5).

Табл. 6.5. Таблиця станів лінійного дешифратора на два входи X_1 і X_2 та чотири прямих виходи (F_0, F_1, F_2, F_3)

X_2	X_1	F_0	F_1	F_2	F_3
0	0				
0	1				
1	0				
1	1				

5. На робочому полі програми зібрати схему для дослідження лінійного дешифратора на два входи X_1 і X_2 та чотири інверсних виходи (Q_0, Q_1, Q_2, Q_3) (рис. 6.14) та експериментальним шляхом зняти таблицю станів роботи дешифратора (табл. 6.6).

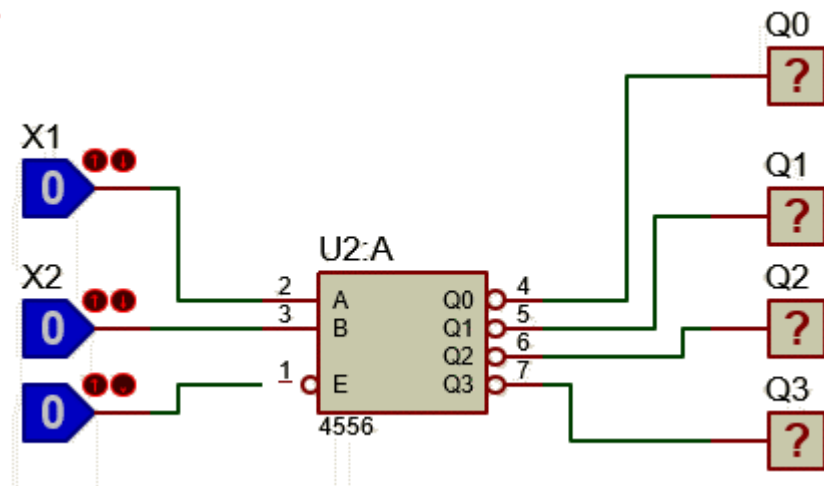


Рис. 6.14. Схема лінійного дешифратора на два входи X_1 і X_2 та чотири інверсних виходи (Q_0, Q_1, Q_2, Q_3)

Табл. 6.6. Таблиця станів лінійного дешифратора на два входи X_1 і X_2 та чотири прямих виходи (Q_0, Q_1, Q_2, Q_3)

X_2	X_1	Q_0	Q_1	Q_2	Q_3
0	0				
0	1				
1	0				
1	1				

6. На робочому полі програми зібрати схему для дослідження двійкового шифратора на 8 входів та 3 виходи (рис. 6.15) та експериментальним шляхом зняти таблицю станів роботи шифратора (табл. 6.7).

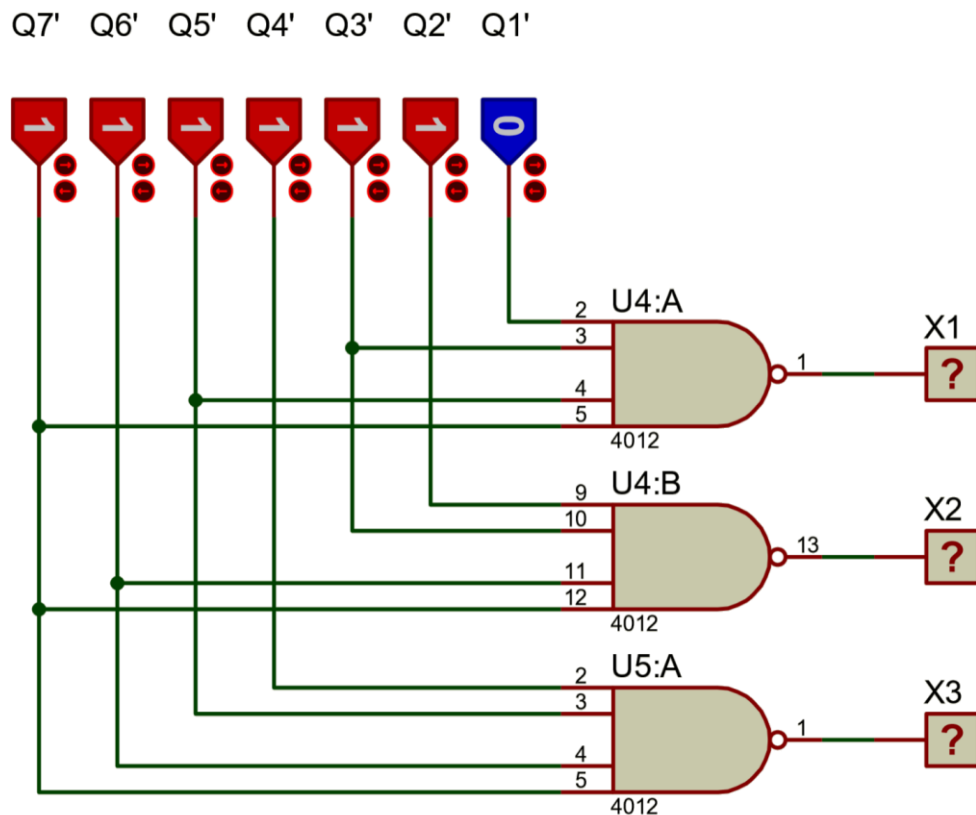


Рис. 6.15. Схема двійкового шифратора на 8 входів та 3 виходи

Таблиця 6.7 - Таблиця станів двійкового шифратора на 8 входів та 3 виходи

Активний вхід	Виходи		
	X_3	X_2	X_1
Q1			
Q2			
Q3			
Q4			
Q5			
Q6			
Q7			

7. На робочому полі програми зібрати схему для дослідження мультиплексора (рис. 6.16).

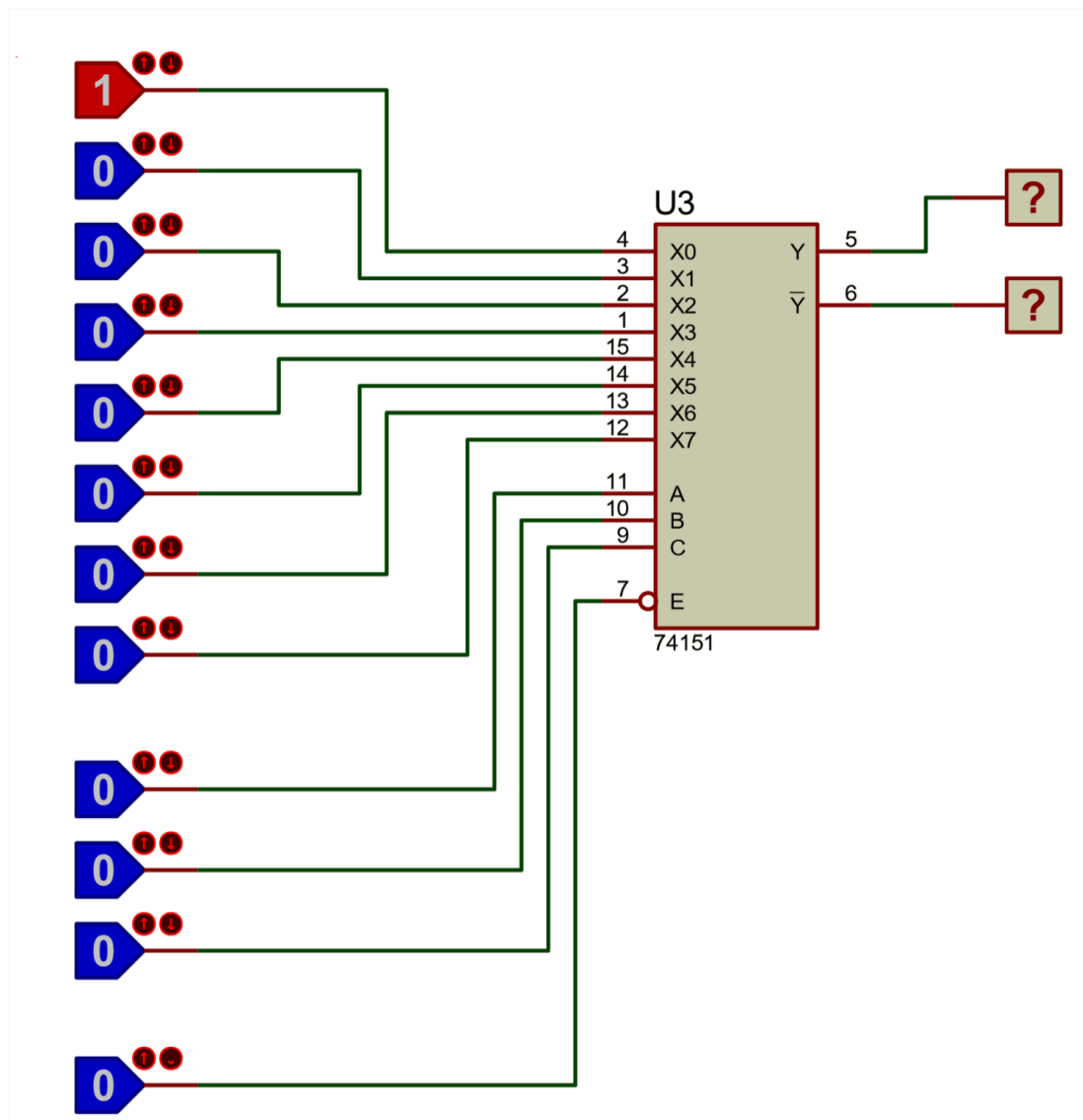


Рис. 6.16. Схема для дослідження мультиплексора

Провести тестування роботи мультиплексора і описати принцип його роботи:

8. На робочому полі програми зібрати схему для дослідження демультиплексора з внутрішнім дешифратором (рис. 6.17).

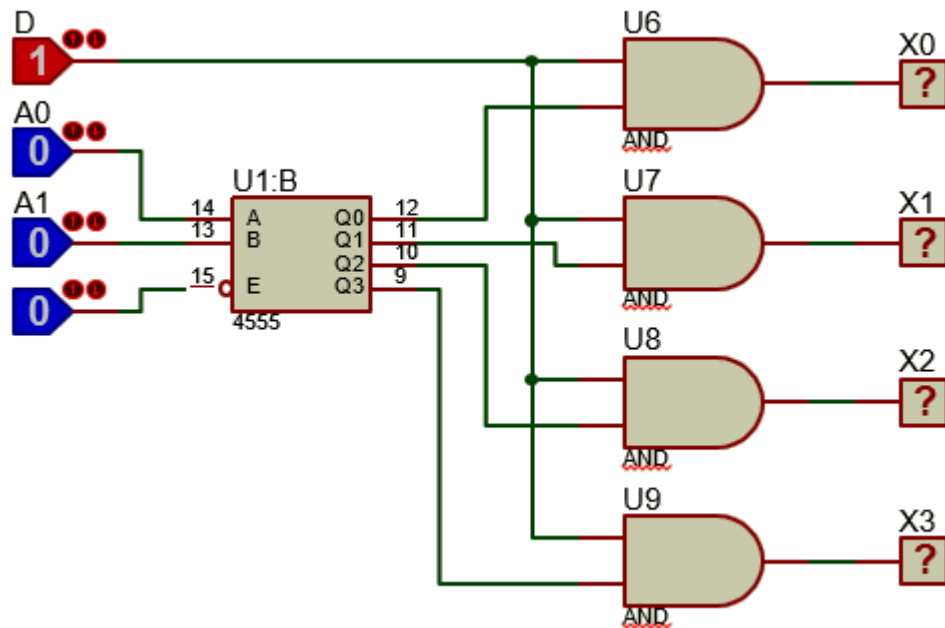


Рис. 6.17. Схема для дослідження демультиплексора з внутрішнім дешифратором

Провести тестування роботи демультиплексора і описати принцип його роботи:

9. Зробити висновки по роботі.

Висновки: _____

9. Надати письмові відповіді на контрольні запитання.

КОНТРОЛЬНІ ПИТАННЯ

1. Поясніть, що таке дешифратор.
2. Наведіть умовне графічне позначення дешифратора на електричних схемах
3. Поясніть принцип роботи лінійного дешифратора 2х4 (наведіть таблицю істинності).
4. Для виконання яких операцій в комп'ютерах використовують дешифратори?
5. Поясніть, що таке шифратор.
6. Наведіть умовне графічне позначення шифратора на електричних схемах.
7. Поясніть принцип роботи двійкового шифратора на 8 входів та 3 виходи (наведіть таблицю істинності).
8. Для виконання яких функцій в комп'ютерах використовують шифратори?
9. Поясніть, що таке мультиплексор.
10. Наведіть умовне графічне позначення мультиплексора на електричних схемах
11. Поясніть принцип роботи мультиплексора:
12. Для виконання яких операцій в комп'ютерах використовують мультиплексори?
13. Поясніть, що таке демультиплексор.
14. Наведіть умовне графічне позначення демультиплексора на електричних схемах
15. Поясніть принцип роботи демультиплексора.
16. Для виконання яких функцій використовують демультиплексори?

Практична робота № 7

Тема: Вивчення архітектури персонального комп'ютера за допомогою програми AIDA64.

Мета роботи: За допомогою програмної утиліти AIDA64 (Everest) вивчити архітектуру персонального комп'ютера (ПК), ознайомитися з основними пристроями ПК, ознайомитися з основними характеристиками пристроїв ПК.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Основні терміни та визначення

ACPI (Advanced Configuration and Power Interface) – вдосконалений інтерфейс конфігурації та керування живленням) – відкритий промисловий стандарт, вперше випущений в грудні 1996 року і розроблений спільно компаніями HP, Intel, Microsoft, Phoenix і Toshiba, який визначає загальний інтерфейс для виявлення апаратного забезпечення, керування живленням і конфігурації материнської плати і пристроїв. Завдання ACPI – забезпечити взаємодію між операційною системою, апаратним забезпеченням і BIOS материнської плати.

DMI (Direct Media Interface) – послідовна шина розроблена Intel для під'єднання південного мосту материнської плати (ICH) до північного мосту. Магістраль DMI у деяких сучасних процесорах використовується для під'єднання чіпсета до процесора (для Core i3, Core i5 і деяких серій Core i7).

COM-port – асинхронні послідовні порти (що позначаються COM1 - COM3). Через них раніше приєднувалися миша, модем тощо.

DMI (Desktop Management Interface) – інтерфейс програмування додатків (Application Programming Interface – API), що дозволяє програмному забезпеченню збирати дані про характеристики комп'ютера. Специфікація DMI розроблена консорціумом Distributed Management Task Force (DMTF),

очолюваному фірмою Intel. Даний інтерфейс дозволяє користувачеві отримати інформацію про апаратне забезпечення ПК.

FSB (Front Side Bus) – системна шина (магістраль) в архітектурі корпорації Intel, що зв'язує процесор з чіпсетом. Часто використовується як загальна назва для магістралі, що сполучає процесор і чіпсет.

Game-port – порт для ігрових пристроїв (для підключення джойстика).

ICH – контролер–концентратор введення–виведення – південний міст (southbridge) – забезпечує взаємодію між ЦП і жорстким диском, картами PCI, інтерфейсами IDE, SATA, USB і пр. Також іноді до чіпсетів відносять мікросхему Super I/O, яка підключається до південного мосту і відповідає за низькошвидкісні порти RS232, LPT, PS/2.

IEEE – 1394 (FireWire) – інтерфейс для передачі великих обсягів відео інформації в реальному часі (для підключення цифрових відеокамер, зовнішніх жорстких дисків, сканерів та іншого високошвидкісного обладнання). Інтерфейсом FireWire оснащені всі відеокамери, що працюють в цифровому форматі. Може використовуватися і для створення локальних мереж.

LPT – паралельні порти, до них звичайно підключаються принтери.

MCH (Memory Controller Hub) – контролер –концентратор пам'яті – північний міст (northbridge) – забезпечує взаємодію центрального процесора (ЦП) з пам'яттю і відеоадаптером (PCI Express). У нових чіпсетах часто є інтегрована відеопідсистема. Контролер пам'яті може бути інтегрований в процесор (Athlon64, Athlon II, Phenom II, Core i7, Core i5, Core i3).

PCI (Peripheral Component Interconnect) – паралельна шина для підключення різних периферійних пристроїв. В даний час інтенсивно витісняється з комп'ютерної техніки шиною PCI –Express.

PCI –Express – це послідовний інтерфейс, розроблений організацією PCI– SIG на чолі Intel і призначений для використання як локальної шини замість PCI.

PS/2 – асинхронні послідовні порти для підключення клавіатури і

маніпулятора миша.

QPB (Quad – Pumped Bus) – 64-бітна процесорна шина забезпечує зв'язок процесорів Intel з північним мостом чіпсета. Характерною її особливістю є передача чотирьох блоків даних (і двох адрес) за такт. Таким чином, для частоти FSB, рівної 200 МГц, ефективна частота передачі даних буде еквівалентна 800 МГц (4 x 200 МГц).

QPI (Quick Path Interconnect) – послідовна шина типу точка–точка для з'єднання процесорів між собою і з чіпсетом, розроблена фірмою Intel. QPI створювався у відповідь на розроблену раніше фірмою AMD шину HyperTransport. Використовується в топових процесорах Intel Core i7 і деяких Core i5.

SPD (Serial Presence Detect) – специфікація, що описує технологію запису, зберігання й зчитування інформації про характеристики 168-контактних модулів DIMM.

USB (Universal Serial Bus) – універсальний інтерфейс для підключення 127 пристроїв (цей інтерфейс може розташовуватися на передній або бічній стінці корпусу).

VGA (Video Graphics Array) – вихід контролера графічного адаптера (відеокарти) для підключення монітора.

iRDA – інфрачервоні порти призначені для бездротового підключення кишенькових або блокнотних ПК або стільникового телефону до настільного комп'ютера. Зв'язок забезпечується за умови прямої видимості, дальність передачі даних не більше 1 м. Якщо в ПК немає вбудованого iRDA адаптера, то він може бути виконаний у вигляді додаткового зовнішнього пристрою (USB iRDA адаптера), що підключається через USB –порт.

HT (HyperTransport) – послідовна двунаправленна магістраль служить для зв'язку процесорів AMD сімейства починаючи з K8 один з одним, а також з чіпсетом. У багатьох чіпсетах використовують HT для зв'язку між мостами.

Таймінги оперативної пам'яті. Схема таймінгів включає в себе затримки **CL– RCD–RP–RAS** відповідно. Для роботи з пам'яттю необхідно

для початку вибрати чіп, з яким ми будемо працювати. Робиться це командою CS (Chip Select). Потім вибирається банк і рядок. Перед початком роботи з будь-яким рядком необхідно її активувати. Робиться це командою вибору рядка RAS (Row Address Strobe), при виборі рядка вона активується. Потім потрібно вибрати стовпець командою CAS (Column Address Strobe) – ця ж команда ініціює читання.

- CL (Cas Latency) – мінімальний час між подачею команди на читання (CAS) і початком передачі даних (затримка читання).

- RCD (RAS to CAS delay) – час, необхідний для активізації рядка банку, або мінімальний час між подачею сигналу на вибір рядка (RAS) і сигналу на вибір стовпця (CAS).

- RP (Row Precharge) – час, необхідний для попереднього заряду банку (precharge). Іншими словами, мінімальний час закриття рядка, після чого можна активувати новий рядок банку.

- RAS (Active to Precharge) – мінімальний час активності рядка, тобто мінімальний час між активацією рядка (її відкриттям) і подачею команди на предзаряд (початок закриття рядка). Рядок не може бути закритий раніше цього часу.

- CR (Command Rate) – час, необхідний для декодування контролером команд і адрес. Інакше кажучи, мінімальний час між подачею двох команд. При значенні 1T команда розпізнається 1 такт, при 2T – 2 такти, 3T – 3 такти.

Це всі основні таймінги. Решта таймінгів мають менший вплив на продуктивність.

Чіпсет (chip set) – набір мікросхем, спроектованих для спільної роботи з метою виконання набору певних функцій. Так, в комп'ютерах чіпсет виконує роль сполучного компонента, що забезпечує спільне функціонування підсистем пам'яті, ЦП, введення–виведення та інших. Чіпсети зустрічаються і в інших ристроях, наприклад, в радіоблоках стільникових телефонів. Чіпсет складається з двох основних мікросхем (іноді вони об'єднуються в один чіп) «північний міст» і «південний міст».

Утиліта AIDA64

AIDA64 (Everest) – програма для перегляду інформації про апаратну і програмну конфігурації комп'ютера (). Програма аналізує конфігурацію комп'ютера і видає детальну інформацію про встановлені в системі пристрої – процесора, системні плати, відеокарти, аудіокарти, модулі пам'яті і так далі, а також інформацію про їхні характеристики, набори команд та їх режими роботи, їх виробників, встановлене програмне забезпечення, конфігурації операційної системи і встановлених драйверів.

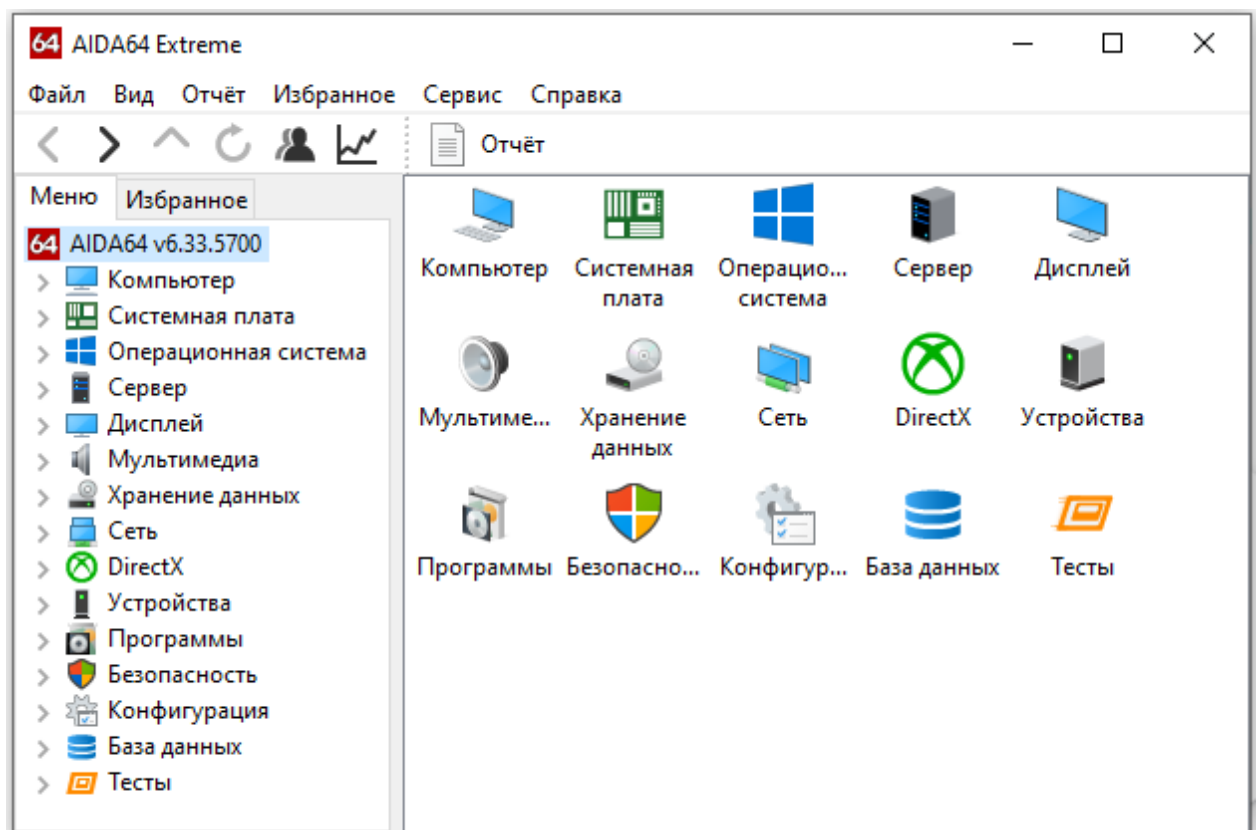


Рис. 7.1. Графічний інтерфейс утиліти AIDA64

У програмі є такий набір тестів:

- читання з пам'яті;
- запис в пам'ять;
- копіювання в пам'яті;
- затримка пам'яті;

- CPU Queen – тестує продуктивність процесора в цілочисельних операціях при вирішенні класичної «задачі з ферзями»;
- CPU PhotoWorxx – тестує продуктивність блоків цілочисельних арифметичних операцій, множення, а також підсистеми пам'яті при виконанні ряду стандартних операцій з RGB–зображеннями;
- CPU ZLib – тестує продуктивність процесора і підсистеми пам'яті при створенні архівів формату ZIP за допомогою популярної відкритої бібліотеки ZLib. Використовує цілочисельні операції;
- CPU AES – тестує швидкість процесора при виконанні шифрування за криптоалгоритмом AES. Здатний використовувати низько рівневі команди шифрування процесорів VIA C3 і C7, що дозволяє останньому бути одним з лідерів тесту, перевершуючи по продуктивності ряд багатоядерних процесорів Intel і AMD;
- FPU Julia – тестує продуктивність блоків процесора, що виконують операції з плаваючою комою, в обчисленнях з 32–розрядною точністю. Моделює кілька фрагментів фрактала Жюліа. При можливості використовує інструкції MMX, SSE і 3DNow!;
- FPU Mandel – тестує продуктивність блоків процесора, що виконують операції з плаваючою комою, в обчисленнях з 64–розрядною точністю шляхом моделювання декількох фрагментів фрактала Мандельброта. Здатний використовувати інструкції SSE2.
- FPU SinJulia – ускладнений варіант тесту FPU Julia. Тестує продуктивність блоків процесора, що виконують операції з плаваючою комою, в обчисленнях з 80–розрядною точністю. Використовує інструкції x87, призначені для обчислення тригонометричних і показових функцій.

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Для виконання роботи на досліджуваному комп'ютері повинна бути встановлена програма AIDA64 або Everest актуальної версії.

1. Ознайомитися з сумарною інформацією про комп'ютер. Для цього в лівому меню у списку «Комп'ютер» слід вибрати пункт «Сумарна інформація», після чого в правому вікні з'явиться список основних параметрів досліджуваного комп'ютера. Виписати:

- тип комп'ютера;
- тип операційної системи;
- ім'я комп'ютера;
- ім'я користувача;
- тип центрального процесора (ЦП);
- тип системної плати;
- тип чіпсета системної плати;
- кількість і тип оперативної (системної) пам'яті;
- тип відеоадаптера;
- тип монітора;
- тип і обсяг дискового накопичувача (жорсткого диска – ЖД);
- перерахувати інші пристрої введення–виведення.

2. Ознайомитися з центральним процесором комп'ютера. Для цього в лівому меню у списку «Системна плата» вибрати пункт «ЦП», після чого в правому вікні з'явиться список основних параметрів ЦП досліджуваного комп'ютера. Виписати основні властивості ЦП:

- тип ЦП;
- назва процесора (псевдонім) ЦП;
- кількість ядер;
- степпінг ЦП;
- набори інструкцій;
- вихідна частота;
- розмір і характеристики кеш–пам'яті ЦП;
- фізичні параметри ЦП.

Отримати відомості про реальну частоту процесора, для цього в списку

«Комп'ютер» вибрати пункт «Розгін». У даному пункті в реальному масштабі часу відображається поточна частота процесора. Виписати поточну частоту процесора. Порівняти вихідну частоту процесора з поточною частотою.

3. Ознайомитися з материнською платою. Для цього в лівому меню у списку «Системна плата» вибрати пункт «Системна плата», після чого в правому вікні з'явиться список основних параметрів материнської плати досліджуваного комп'ютера. Виписати:

- назву материнської плати і фірму виробника;
- властивості системної шини (FSB, HT, QPB);
- властивості шини пам'яті;
- назва чіпсета;
- фізичну інформацію про системну плату.

4. Ознайомитися з властивостями модулів оперативного запам'ятовуючого пристрою (ОЗП). Для цього в лівому меню у списку «Системна плата» вибрати пункт «SPD». Виписати властивості модулів ОЗП і основні таймінги пам'яті, для різних частот. Якщо встановлені різні модулі пам'яті, виписати параметри для кожного з них.

5. Ознайомитися з чіпсетом материнської плати. Для цього в лівому меню у списку «Системна плата» вибрати пункт «чіпсет».

5.1 Ознайомитися з властивостями «північного моста» чіпсета. Для цього у верхньому вікні пункт «Північний міст». Перерахувати контролери, вбудовані в «північний міст». Виписати:

- назву «північного моста»;
- підтримувані швидкості системної шини (FSB, HT, QPB);
- підтримувані типи оперативної пам'яті;
- тип контролера пам'яті;
- максимальний обсяг оперативної пам'яті;
- основні таймінги пам'яті (CR, tRAS, tRP, tRCD, CL, tREF).

Порівняти характеристики ОЗП з отриманими в попередньому пункті.

5.2 Ознайомитися з властивостями «південного моста» чіпсета. Для цього у верхньому вікні пункт «Південний міст». Перерахувати пристрої, що містяться в «південному мосту». Зробити висновки.

6. Ознайомитися з системою зберігання даних – постійними запам'ятовуваними пристроями (ПЗП). Для цього в лівому меню у списку «Збереження даних» вибрати пункт «Збереження даних Windows», після чого в правому верхньому вікні з'явиться список всіх можливих ПЗП досліджуваного комп'ютера. У роботі слід розглянути параметри жорсткого диска і оптичного DVD накопичувача. Виписати їх основні характеристики, такі як:

- назва ЖД;
- виробник;
- ємність;
- швидкодію;
- інтерфейс підключення;
- фізичні параметри:
 - форм–фактор (розмір в дюймах);
 - кількість пластин (дисків);
 - вага;
 - швидкість обертання.

7. Ознайомитися з наявними на платі портами введення–виведення. Для цього в розділі «Комп'ютер» вибрати пункт «DMI». У даному пункті з розділу «Системні роз'єми» виписати наявні на материнській платі роз'єми. З розділу «Роз'єми портів» виписати роз'єми для підключення зовнішніх пристроїв введення–виведення, для кожного вказати тип порту.

8. Провести тестування швидкодії ОЗП. Для цього перейти в розділ «Тест» і вибрати відповідні пункти. Для початку тестування слід натиснути кнопку «Оновити», або клавішу «F5» на клавіатурі. Провести наступні тести ОЗП:

- читання з пам'яті – тестує швидкість пересилання даних з ОЗП до

- процесора;
- запис в пам'ять;
- копіювання в пам'яті – тестує швидкість пересилання даних з одних комірок пам'яті в інші через кеш процесора;
- затримка пам'яті – тестує середній час зчитування процесором даних з ОЗП.

Записати результати тестування. Порівняти продуктивність досліджуваної системи з продуктивністю еталонних систем. Виписати найбільш близькі по продуктивності системи. Зробити висновки.

9. За результатами попередніх пунктів побудувати структурну схему ПК. У схемі повинні бути відображені всі пристрої, що входять в ПК, з їх назвами і основними параметрами.

КОНТРОЛЬНІ ПИТАННЯ

1. Основні принципи побудови ЕОМ, структура Дж. фон Неймана.
2. Намалюйте структурну схему ПК, пояснити призначення всіх компонентів.
3. Центральний процесор, основні характеристики.
4. Система пам'яті. Склад, призначення.
5. Системна магістраль. Визначення, призначення, параметри.
6. Основні внутрішні шини ПК.
7. «Північний міст». Склад, призначення.
8. «Південний міст». Склад, призначення.
9. Пристрої введення–виведення ПК.
10. Чинники, що впливають на продуктивність ПК.
11. Які пристрої до яких портів можуть підключатися ?

Практична робота № 8

Тема: Дослідження роботи комп'ютера. Емулятор EMU8086.

Мета роботи: Ознайомитись із програмними способами емуляції роботи персонального комп'ютерів в Emu8086.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Процесор містить 14 швидкодіючих 16-розрядних чарунок пам'яті, які називаються регістрами (див. таблицю 10.1).

Регістри поділяють на регістри загального призначення, сегментні, індексні, вказівники та регістр стану процесора і регістр вказівник команд. До регістрів загального призначення (AX, BX, CX, DX) можна звертатися як безпосередньо, так і побайтово (до молодшого та старшого розряду окремо), наприклад, регістр AX складається з молодшого розряду AL та старшого — AH. Індексні регістри (SI, DI), призначені для зберігання індексів при роботі з рядковими даними. Регістри-вказівники (BP, SP) — використовуються для роботи зі стеком.

Сегментні регістри використовуються для зберігання адрес відповідних сегментів пам'яті. Сегментом називається область пам'яті, яка починається з адреси кратної 16 та містить дані однієї структури. Розрізняють три головні сегменти програми: сегмент коду (CS) — містить інструкції програми; сегмент даних (DS) — містить дані, задані програмістом; сегмент стеку (SS) — область пам'яті, доступ до якої організований за принципом LIFO (last in, first out — останній зайшов, перший вийшов). Також існує розширений сегмент (ES), який використовується, наприклад, при роботі з рядковими даними.

Регістр стану (регістр прапорів) визначає стан процесору після виконання кожної команди. Для мікропроцесорів Intel 8086 активними є 9 біт з 16. Ці біти називають прапорами стану. Регістр-вказівник команд (IP) містить адресу команди (в сегменті коду). Під час виконання програми значення вказівника збільшується на розмір виконаної команди. Існують

команди, які змінюють значення ІР для реалізації переходів всередині програми.

Табл. 8.1. Регістри процесору Intel 8086

Ім'я	English	Призначення
Регістри загального призначення		
AX	Accumulator	Акумулятор (AH, AL), надшвидкий регістр, зв'язаний безпосередньо з АЛП (арифметико-логічним пристроєм) мікропроцесора.
BX	Base	Базовий регістр (BH, BL), використовується при розширеній адресації.
CX	Counter	Регістр-лічильник (CH, CL), керує числом повторень у циклах.
DX	Data	Регістр даних (DH, DL), використовується для введення/виведення даних та для обчислень з подвійною точністю, або великими числами при недостатній розрядності акумулятора.
Індексні регістри		
SI	Source Index	Індексування джерела (SI)/приймача(DI), використовуються при обробці рядкових даних та індексування масивів даних.
DI	Destination Index	
Регістри – вказівники		
BP	Base Pointer	Вказівник бази
SP	Stack Pointer	Вказівник стека
Сегментні регістри		
CS	Code Segment	Сегмент коду, не можна змінити напряму.
SS	Stack Segment	Сегмент стеку.
DS	Data Segment	Сегмент даних.
ES	Extension Segment	Розширений сегмент даних.
Вказівник команди		
IP	Instruction Pointer	Зміщення команд, програмно не доступний.
Регістр прапорів		
FLAGS		Інформація про поточний стан процесора.

Програма на мові Асемблера є послідовністю операторів, що описують виконувані дії. Оператором (рядком) початкової програми може бути або команда, або псевдооператор (директива) Асемблера.

Команди представляють коротку нотацію (запис) системи команд. На відміну від команд псевдооператори повідомляють (транслятору), що йому робити з командами і даними, які вводяться в програму.

Команда може включати до 4-х полів наступного виду:

[мітка:] мнемокод [операнд] [; коментар]

Приклад команди з усіма полями:

exit: MOV CX, DX ; помістити вміст регістра DX в регістр CX

Мітка, команда та операнд відокремлюється один від одного символом пробілу або табуляції, коментар починається від символу «;» і до кінця рядка. Коментар не відноситься до машинного коду і ігнорується транслятором. Не можна використати як мітку імена регістрів і мнемокоди, крім того мітка повинна розпочинатися з букви, але може містити цифри і спеціальні символи : ?, @, /, _, \$ і точку, проте точка може бути тільки першим символом мітки. Мітки в основному використовують для команд передачі керування і вони не є обов'язковими.

Мова Асемблера може містити змінні, які визначаються за допомогою директив: DB – визначає байт, DW – слово, DD – подвійне слово. Синтаксис:

ім'я DB значення

Вимоги до імені змінної такі ж, як і для мітки. Значення може бути: числовим (14, 4Bh, 1000112b), декілька чисел, рядокта масивом, елементи якого розділяються комою, оператор «?» (який задає невизначену змінну) та DUP (дублювання даних вказану кількість раз). Змінна являє собою область пам'яті, яка помічена певним ім'ям. Зазвичай всі змінні розміщують в сегменті даних.

```
text_string  db  'Dobrogo dnya!'      ;рядјr символів
number       dw  7                    ;number=7 та розміром 2 байти
tab          db  1,2,3,4,5,6,7        ;масив даних
null         db  ?                    ;невизначена змінна
table_512    dw  512 dup(0)           ;масив з 512 слів заповнених нулями.
mas          db  8 dup ('a', 'b')     ;масив: 8 разів повторюється 'ab'
```

Як операнд в команді може застосовуватись константа, яка може бути:

- двійковою, як послідовність цифр 0 і 1, що закінчуються буквою B, наприклад, 10111010B;
- десятковою, в звичній десятковій системі числення з необов'язковою буквою D на кінці, наприклад, 129d або просто 129;
- шістнадцятковою, як послідовність цифр від 0 до 9 і букв від A до F, що закінчується буквою H. Якщо шістнадцяткова константа розпочинається з букви, то така константа доповнюється першим символом - цифрою від 0 до 9, наприклад, 0E23h (в даному випадку перша цифра інформує Асемблер про те, що E23 число, а не ідентифікатор або змінна);
- літералом, у вигляді рядка букв, цифр і інших символів у лапках а.

Мнемокоди можуть мати від 2 до 6 букв, при трансляції мнемокод перетворюється в числове значення усередині транслятора.

Важливою особливістю машинних команд є те, що вони не можуть маніпулювати одночасно 2-ма операндами, що знаходяться в оперативній пам'яті (ОЗП). Це означає, що в команді тільки 1 операнд може вказувати на осередок ОЗП, інший операнд має бути або регістром, або безпосереднім значенням. З цієї причини можливі наступні поєднання операндів в команді:

- регістр – регістр;
- регістр – пам'ять;
- пам'ять – регістр;
- регістр – безпосередній операнд;
- пам'ять – безпосередній операнд;
- сегментний регістр – регістр.

Для команд характерно, що за наявності двох операндів перший з них є приймачем, а другий – джерелом. Результат операції зберігається за першою адресою, ось чому перший операнд ніколи не може бути безпосереднім операндом або константою.

Команда MOV – команда пересилки даних. Формат команди:

MOV приймач, джерело

Дія: копіює вміст джерела в приймач, джерело не змінюється.

Приймач може бути регістром, пам'яттю. Джерело може бути регістром, пам'яттю, безпосереднім значенням.

MOV AX, 7 ; до регістру AX буде записано число 7.

MOV AX, ABC ; до регістр AX буде записано значення за адресою ABC.

MOV ABC, 82 ; за адресою ABC буде занесене число 82.

MOV DS, BX ; регістр DS матиме значення регістра BX

Стек – це структура пам'яті, яка використовується для тимчасового зберігання інформації. Програма може помістити дані в стек (PUSH) або забрати їх звідти (POP). Наприклад:

PUSH AX ; у стек розмістити вміст регістру AX

PUSH abc ; у стек розмістити значення змінної abc

PUSH 1234h ; у стек розмістити число 1234h

POP BX ; значення зі стеку розмістити у BX

POP [1234h]; значення зі стеку розмістити за адресою 1234h

POP qwerty ; значення зі стеку розмістити у змінну qwerty

Для створення програми на мові Асемблера необхідне певне програмне забезпечення: компілятор, компоновщик (редактор зв'язків) та налагоджувач (дебагер). Існує велика кількість пакетів подібного програмного забезпечення.

Програма пишеться у будь-якому текстовому редакторі та зберігається з довільним ім'ям та розширенням ASM. Такий файл називається вихідним модулем. За допомогою компілятора вихідний модуль перетворюється в об'єктний (файл з тим же ім'ям та розширенням OBJ), паралельно може створюватися і файл лістингу, який має розширення LST. Об'єктний модуль за допомогою редактора зв'язків перетворюється на виконуєму програму з розширенням COM або EXE.

Emu8086 поєднує у собі потужний редактор вихідного коду, асемблер, дізасемблер, програмний емулятор (віртуальний ПК) з відладчиком.

Візуальний інтерфейс дуже простий в роботі. Можна переглядати зміст регістрів, ознак, пам'яті під час виконання програми.

Арифметико-логічний пристрій (АЛП) показує внутрішню роботу центрального процесора (CPU).

Емулятор виконує програми на віртуальному ПК, який повністю виключає можливість доступу з вашої програми до реальних апаратних засобів, таким як жорсткі диски та пам'ять.

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Розробити програму на Асемблері для виконання таких завдань:

1. За адресою рівному дню і місяцю Вашого народження (наприклад, 23 травня – 2305) занести рік Вашого народження, представивши його як шістнадцятиричне число.

2. Визначити змінну `const`, присвоївши їй день і місяць Вашого народження у форматі описаному вище. Відправити це значення до стека. Розмістити це значення зі стека у регістр `CX`.

3. Призначити змінній `fiо` Ваше прізвище, ім'я, по батькові. У пам'яті відразу після `fiо` розмістити символ, код якого в ASCII-кодах визначити, як Ваш варіант плюс 14. А потім через пропуск дату Вашого народження у форматі `ddmmuuuu` (наприклад, 23051987).

4. Визначити адресу `fiо`. Визначити адресу сегмента, зміщення та фізичну адресу кожного слова `fiо`. Розмістити в регістр `AL` середню букву ПІБ (значення округлити до цілого). Заповнити таблицю 8.2 даними.

Табл.8.2. Адреса сегмента, зміщення та фізичної адреса

Слово в рядку	Адреса сегмента	Зміщення	Фізична адреса

У звіті до кожного завдання повинен бути screenshot середовища `Emu8086`, в якому повинні відображатись текст програми, регістри, стек та данні з оперативної пам'яті.

Приклад виконання завдання

```
; Task1
; дата народження - 23/05/1987
org 100h
mov [2305h], 1987h
```

```

; Task2
push const
pop cx

; Task3. Вapиaнт - 30
lea bx, fio
mov [bx+22], 44; y fio 22 cИMBOЛИ
mov [bx+23], ' '
mov [bx+24], 2305h
mov [bx+26], 1987h

; Task4. Середня буква ПІВ має номер 11 і є «a»
mov al, [bx+11]
ret

Const dw 2305d
fio db "Sidorov Ivan Petrovich"

```

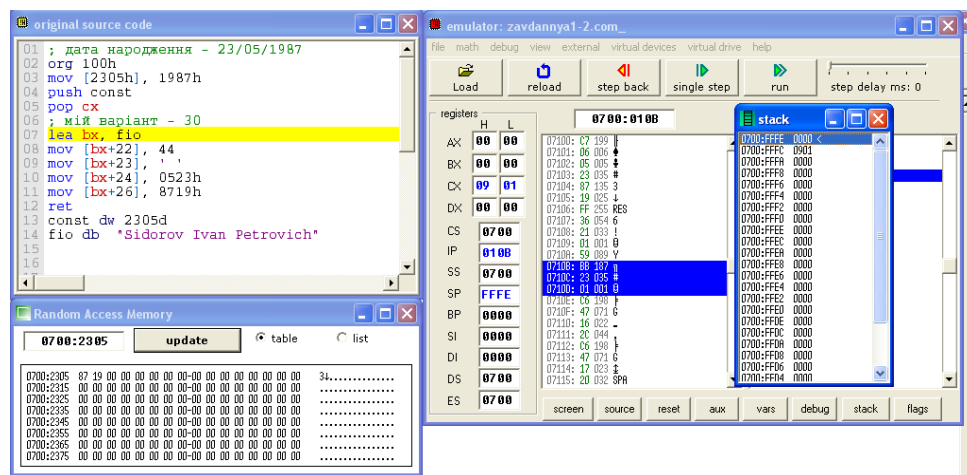


Рис.8.1. Копія екрану Еми8086 виконання програми (завдання 1 – 3)

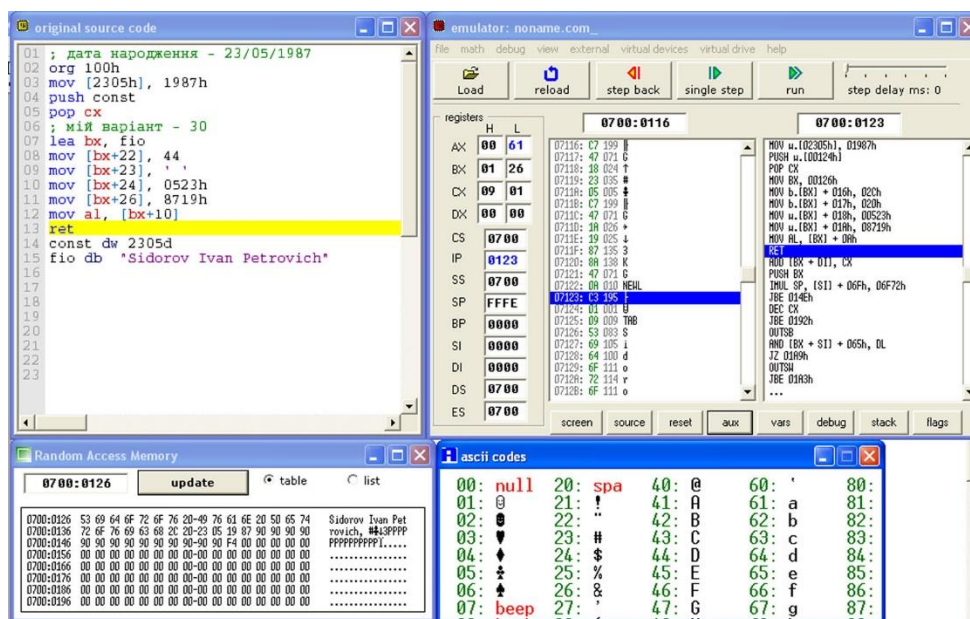


Рис. 8.2. Копія екрану Еми8086 виконання програми (завдання 4)

Табл. 8.3. Виконання завдання 4

Слово в рядку	Адреса сегмента	Зміщення	Фізична адреса
Sidorov	0700	0126	07126
Ivan	0700	012E	0712E
Petrovich	0700	0133	07133

КОНТРОЛЬНІ ПИТАННЯ

1. Для чого використовують програмний емулятор Emu8086?
2. Роботу якого пристрою можна спостерігати при роботі в Emu8086?
3. Перелічить основні види регістрів Intel8086.
4. Перелічить регістри загального призначення Intel8086.
5. Перелічить індексні регістри Intel8086.
6. Перелічить регістри–вказівники Intel8086.
7. Перелічить сегментні регістри Intel8086.
8. Для чого потрібен регістр стану (регістр прапорів)?
9. Вкажіть структуру команди Асемблера.
10. Які бувають директиви визначення змінних, масивів тощо?
11. Які можуть бути числові дані в Асемблері?
12. Яким чином визначають рядок?
13. Чим можуть буди операнди в команді? Перерахуйте можливі їх поєднання в команді.
14. Опишіть команду MOV.
15. Опишіть команди стека.

Практична робота № 9

Тема: Складання комп'ютера.

Мета роботи: Ознайомитись із особливостями складання і модернізації персональних комп'ютерів.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

Складання та розбирання комп'ютерів виконується у такій послідовності:

1. Захист від електростатичного розряду

1.1. Працюючи з відкритим корпусом комп'ютера, ви повинні вжити заходи, що виключають можливість електростатичного розряду через сигнальні ланцюги. Для цього торкніться ділянки його шасі, наприклад кришки блоку живлення. При цьому потенціали тіла і загального дроту комп'ютера зрівнюються.

1.2. Складніший спосіб рівномірного розподілу потенціалів між вами і компонентами комп'ютера – це застосування захисного електростатичного комплекту. У комплект входить браслет та килимок, забезпечений дротами для підключення до шасі. При роботі з комп'ютером підкладіть килимок під системний блок. Після цього з'єднаєте його дротом з шасі й вдягніть антистатичний браслет.

1.3. Покладіть на антистатичний килимок вийняті з комп'ютера елементи: накопичувачі на жорстких, плати адаптерів, системну плату, модулі пам'яті і процесор.

1.4. Якщо у вас немає килимка, розміщуйте вийняті схеми та пристрої прямо на столі. Плати адаптерів завжди тримайте за металевий кронштейн, яким вони кріпляться до корпусу. Кронштейн сполучений із загальним дротом плати, і можливий електростатичний розряд не приведе до ушкодження компонентів адаптера. Якщо у плати немає металевого кронштейна (як, наприклад, у системної плати), акуратно тримайте її за

краї й не торкайтеся встановлених на ній компонентів.

2. Установка системної плати

2.1. Перед установкою в комп'ютер нової системної плати необхідно змонтувати на ній процесор та модулі пам'яті.

2.2. Щоб встановити на системну плату процесор і радіатор, виконаєте ряд дій.

- Витягніть нову плату з антистатичного пакету, в який вона упакована, і положіть її згори на пакет або на антистатичний килимок, якщо він у вас є.
- Встановіть процесор. Послідовність дій при виконанні цієї процедури залежить від типу роз'єму процесора - socket або slot.
- Для роз'ємів типу *SOCKET* знайдіть на процесорі контакт 1: зазвичай один з кутів мікросхеми злегка скошений або помічений точкою, біля нього і знаходиться цей контакт. Потім знайдіть контакт 1 в ZIF-гнезді для процесора, що знаходиться на системній платі. Тепер треба підняти важіль і помістити мікросхему в роз'єм, поєднавши контактні виводи з відповідними отворами. Якщо процесор в роз'єм не входить, перевірте, чи правильно він орієнтований і чи співпадають контакти. Коли процесор увійде як слід, опустіть затискаючий важіль, щоб зафіксувати мікросхему в гнізді.
- Більшість систем тепловідведення закріплюються або на самому процесорі, або кріпляться до гнізда одним або декількома затискачами. Перед установкою тепловідводу слід змастити процесор теплопровідною пастою. Тоді між ним та тепловідводом не буде повітря, яке ускладнює відведення тепла від процесора.

2.3. Прочитайте в документації виробника плати, як правильно встановити на платі перемички для роботи з конкретним процесором. У документації має бути схема, що показує розташування перемичок, й таблиця з варіантами їх установки для різних типів процесорів.

3. Установка модулів пам'яті

3.1. У сучасних платах використовується модуль пам'яті – DIMM. Ці модулі встановлюються в роз'єми з найменшими номерами.

3.2. Часто модулі встановлюються парами, а іноді навіть по чотири. Тому перед установкою рекомендую ще раз заглянути в документацію до плати; там повинно бути сказано, які роз'єми і в якому порядку заповнювати першими і як встановити модулі того типу, який використовує ваша плата.

4. Закріплення системної плати в корпусі

4.1. Огляньте призначені для стійок отвори в платі. Якщо навкруги є металевий кант, отвір призначений для металевої стійки, а якщо канта немає – для пластикової. Тепер металеві стійки треба угвинтити в отвори в шасі корпусу так, щоб вони розташовувалися навпроти отворів, на платі.

4.2. Пластикові стійки вставляються знизу в саму плату. Натисніть – і вони з клацанням стануть на місце.

4.3. У системних платах ATX не використовуються пластикові стійки. Плата кріпиться за допомогою семи гвинтів (рис. 9.1).

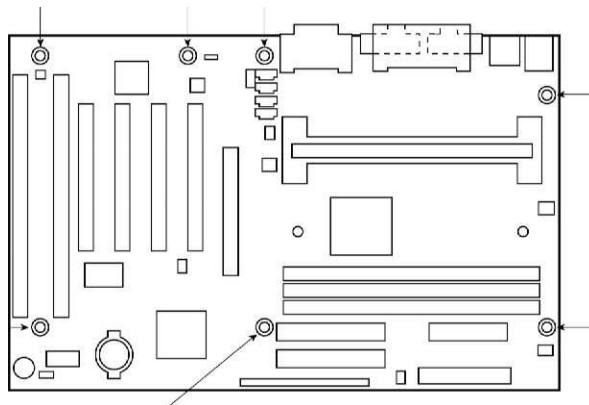


Рис. 9.1 Місця кріплення системної плати ATX

4.4. Візьміть гвинти і пластикові шайби і пригвинтіть плату до шасі.

4.5. Встановіть задню панель роз'ємів введення-виведення.

4.6. Встановіть шасі з системною платою в направляючі корпусу.

Простежте за тим, щоб роз'єми введення-виведення системної плати

співпали з відповідними отворами задньої панелі. Системна плата повинна без особливих зусиль встати на призначене їй місце.

4.7. У платах з пластмасовими стійками перевірте, щоб усі стійки потрапили у відповідні прорізи. Якщо необхідно, трохи посувайте плату із сторони в сторону. При правильній установці плати усі отвори для гвинтів в платі і шасі корпусу співпадають.

4.8. Тепер пригвинтіть шасі з системою платою до корпусу комп'ютера.

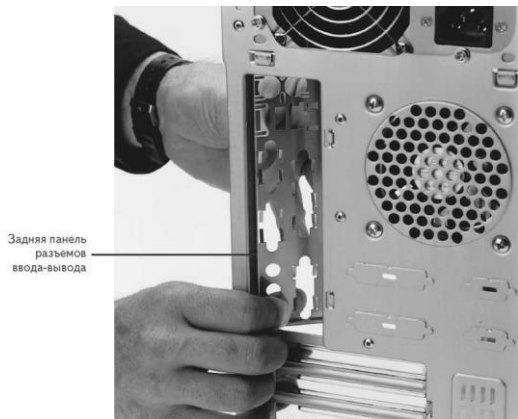


Рис. 9.2. Установка задньої панелі



5. Підключення живлення

5.1. Встановити блок живлення досить просто: треба лише помістити його у відповідний відсік корпусу і пригвинтити декількома гвинтами.

5.2. У ATX-платах для підключення живлення використовується тільки один роз'єм, який підключається одним-єдиним способом.

5.3. У багатьох системах для охолодження процесора використовується вентилятор, його теж слід підключити. Для подачі живлення до вентилятора існує спеціальний роз'єм – прямо на системній платі.

6. Підключення до системної плати кабелів від пристроїв введення-виведення та інших з'єднувачів

6.1. Підключіть пристрої з інтерфейсом IDE : накопичувачі на жорсткому

диску, DVD-ROM. Вони підключаються плоским IDE-кабелем до розташованих на платі 40-контактних роз'ємів головного і підпорядкованого IDE-контролерів. Зазвичай жорсткий диск підключається до головного контролера, а DVD-ROM – до підлеглого.

6.2. Підключіть пристрої з інтерфейсом Serial ATA: накопичувачі на жорсткому диску, DVD-ROM. Вони підключаються плоским ATA кабелем до розташованих на платі 7-контактних роз'ємів. Зазвичай жорсткий диск підключається до SATA0, а DVD-ROM – до SATA3.

6.3. Підключіть до плати кнопки та індикатори передньої панелі комп'ютера, а також внутрішній гучномовець. Якщо на платі не позначені місця підключення відповідних дротів, скористайтеся схемою, приведеною в документації до плати.

7. Установка накопичувачів

7.1. Зніміть направляючі з накопичувача (якщо вони встановлені). Помістіть накопичувач у відповідний відсік корпусу. Перед цим не забудьте встановити в потрібне положення усі перемички і перемикачі на накопичувачі. Прикрутіть гвинтами (частіше всього чотирма) накопичувач до корпусу.

7.2. Підключіть інтерфейсний кабель та кабель живлення (як до накопичувача, так і до системної плати).

8. Установка плат розширення

8.1. Акуратно візьміть плату за краї, не торкаючись мікросхем та електричних з'єднань. Опустіть її нижній край з нанесеними на нього металевими контактами у відповідний роз'єм. З силою натисніть на верхній край плати, щоб вона стала на місце. Гвинтом прикрутите плату до корпусу комп'ютера.

8.2. Тепер підключіть до вставленої плати усі необхідні кабелі.

9. Закриваємо корпус і підключаємо зовнішні кабелі

10. Підключіть комп'ютер до електричної мережі. Запустіть систему.

Складання та розбирання комп'ютерів у середовищі «IT Essentials Virtual Desktop»

Програма має 3 режими роботи:

- Learn Mode (режим навчання)
- Test Mode (тестовий режим)
- Explore Mode (огляд режиму) - допомагає візуально моделювати всі деталі ПК.

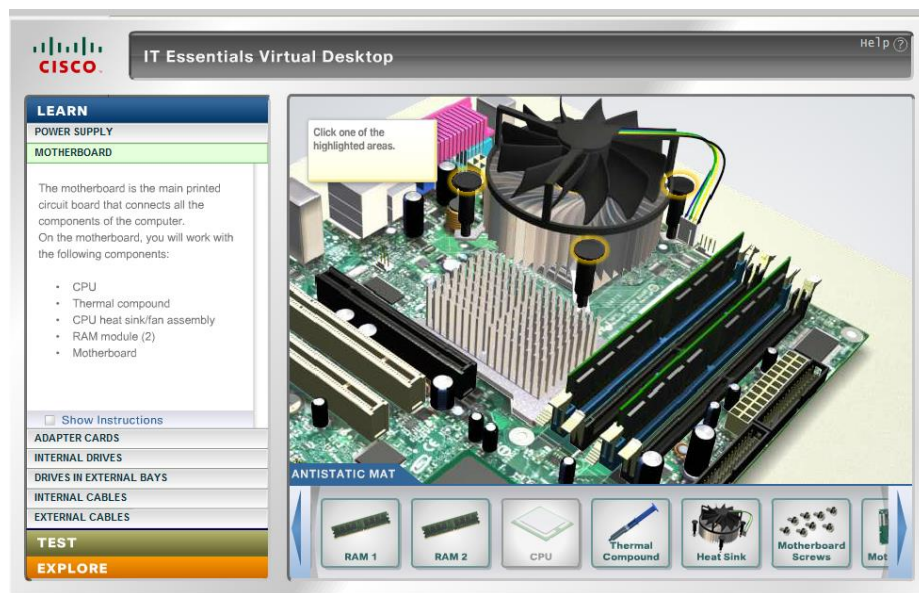


Рис. 9.4. Режим LEARN. Етап установки вентилятора охолодження процесора

Learn Mode складається з 7 вкладок: напруга живлення, материнська плата, адаптер карти, внутрішні локальні диски, відсіки для зовнішніх дисків, внутрішніх кабелів та зовнішніх кабелів. Learn Mode дає чотири внутрішніх види на робочому столі. Тестовий режим дає всі компоненти для установки, але вони не розділені на шари.

У навчальному режимі, представлена інформація про те, які компоненти складають кожен шар. Натиснувши на кнопку «Показати Інструкції», ви побачите крок за кроком список інструкцій, які допоможуть вам встановити кожен компонент. Після установки кожного компонента в шарі, ви будете

повідомлені, що установка цього шару буде завершена. Потім ви можете вибрати наступний шар в головному меню навігації.

Антистатичний килимок (панель інструментів) розташований в нижній частині головного вікна. Він містить всі необхідні деталі будуть збирати на віртуальному робочому столі. Використовуйте ліву та праву стрілки навігації, щоб перейти компонентів на антистатичний килимок.



Рис. 9.5. Режим LEARN. Етап установки материнської плати до системного блоку



Рис. 9.6. Режим LEARN. Пункт роботи з External Cables

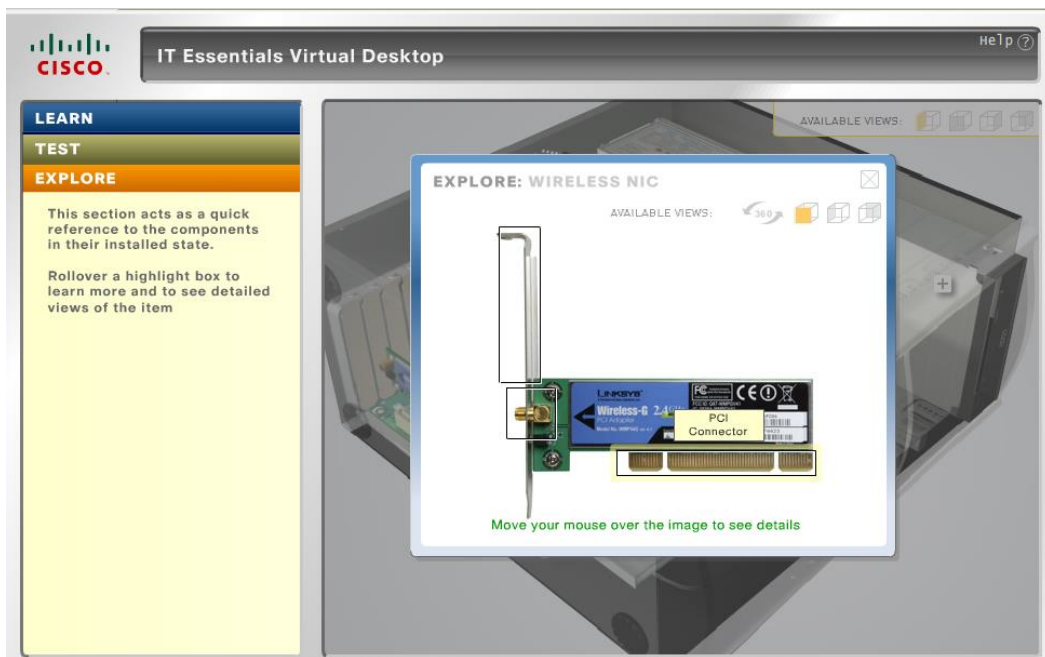


Рис. 9.7. Режим Explore. Зовнішній вигляд Wireless NIC

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

1. Встановити програму «IT Essentials Virtual Desktop» (<https://goo.gl/QTMyeC>) та запустити її у Internet Explorer. Виконати повне складання віртуального комп'ютера.
2. Під керівництвом викладача виконати розбирання комп'ютера. Вжити заходи щодо захисту від електростатичного розряду. Записати конфігурацію комп'ютера, включаючи апаратні положення перемичок та перемикачів, схеми кабельних з'єднань.
3. Ознайомитися з елементами та модулями, з яких складається системний блок. Занести у звіт перелік встановлених елементів, зазначивши точні моделі елементів та їх серійні номери (табл. 9.1).
4. Здійснити складання системного блоку.

Табл. 9.1. Зразок оформлення конфігурації комп'ютера

№ пп	Найменування елементу	Точне найменування	Серійний номер
1	Процесор	Intel Pentium DC (Model G860 LGA1155) 3.0GHz	MC230034A4633
2	Материнська плата	ASUS P8H61-M LX3 R2.0/C/S	CAM0CS054278
3	Оперативна пам'ять		
4	Жорсткий диск		
5	Відеоадаптер		
6	Оптичний дисковод		
9			
10			
11	Корпус	noName	—
12	Блок живлення	EuPower 380W	CT223549U12

КОНТРОЛЬНІ ПИТАННЯ

1. Які правила техніки безпеки мають бути дотриманні при збиранні (модернізації) комп'ютера?
2. Де вказується модель елемента (модуля) системного блоку та серійний номер? Наведіть приклад позначення.
3. Які інформаційні написи зустрічаються на елементах та модулях системного блоку?
4. Які існують види материнських плат і у чому їх відмінність?
5. На що потрібно звернути увагу під час вибору корпусу?
6. Як здійснюється охолодження центрального процесора?
7. Для чого потрібна термопаста?
8. На що потрібно звернути увагу при виборі відеокарти?
9. Які типи оперативної пам'яті існують і чи можуть вони працювати разом у одній системі?
10. Які бувають жорсткі диски та у чому їхня принципова різниця?
11. Чому так важливо щільно закріпити усі кабелі у корпусі?

Практична робота № 10

Тема: Дослідження порядку запуску комп'ютера.

Мета роботи: Ознайомитись із етапами початкового завантаження комп'ютера, знати можливі несправності та методи їх діагностики.

ОСНОВНІ ТЕОРЕТИЧНІ ВІДОМОСТІ

BIOS (англомовний акронім – англ. Basic Input/Output System – базова система введення/виведення) – є набором спеціальних команд-інструкцій, підпрограм обчислювального пристрою для ініціалізації компонентів його персональної платформи, необхідних для первинного завантаження та подальшої роботи. Компонентами, які здатні працювати незалежно від операційної системи, як реагувати і що робити за певних умов є процесор, системна логіка (чипсет), оперативна пам'ять, пристрої введення-виведення (відеокарта, клавіатура тощо) та інші.

Фактично, BIOS – перше програмне забезпечення, що виконується процесором. Оскільки на початковому етапі завантаження комп'ютера зовнішні пристрої недоступні, BIOS, в загальному випадку, зберігається незалежним від живлення персональної платформи чином – в NVRAM-пам'яті (від англ. Non Volatile, – не тимчасова). Для цього, як правило, використовується одна або декілька мікросхем пам'яті – пристроїв постійного зберігання даних, які розташовані на системній платі.

Також термін BIOS використовується стосовно:

- мікросхем пам'яті або будь-яких носіїв інформації, що містять програму BIOS
- панельок на системних платах, призначених для мікросхем пам'яті з програмою BIOS

На застарілих платформах BIOS зберігався рівними частинами на двох мікросхемах: на одній – дані парних адрес (Even BIOS), на іншій – з непарних

адрес (Odd BIOS). Це було пов'язано із різною шириною адресних шин процесора та мікросхем пам'яті.

Для того, щоб зберігати BIOS або його частину незалежним від живлення чином на різних етапах еволюції персонального комп'ютера використовувались всі можливі засоби NVRAM: магнітні носії різних гатунків[джерело?] та мікросхеми пам'яті постійного зберігання – ROM (PPROM, OTPROM). На сучасних платформах зазвичай використовуються тільки мікросхеми, дані в яких зберігаються і після вимкнення живлення. З їх числа:

- EPROM (від англ. Erasable Programmable Read Only Memory) – мікросхема постійної пам'яті, вміст якої стирається за допомогою ультрафіолетового опромінення.
- EEPROM (від англ. Electrically Erasable Programmable Read Only Memory) – мікросхема постійної пам'яті, що стирається за допомогою спеціального електричного сигналу.
- Flash ROM – спеціальний різновид EEPROM, який може бути перезаписаний просто в комп'ютері без додаткових пристроїв на зразок програматора.

Від 2006 року на персональних платформах почав використовуватись SPI-інтерфейс. SPI-мікросхеми мають всього вісім контактів, з яких лише чотири задіяні для передачі даних.

Виробники BIOS. Найбільшого поширення набули програмні продукти компанії Phoenix Technologies Ltd. Компанія продає виробникам платформ BIOS двох торгових марок: PhoenixBIOS та AwardBIOS.

Другий по використанню – AMIBIOS, розроблений компанією American Megatrends Inc.

Також існує і безліч інших BIOS (наприклад, AST, BIOSSTAR, COMPAQ, C&T, DTK, Philips, Quadtel та інші), які в основному набули поширення наприкінці 80-х, на початку 90-х років минулого століття.

1.1 Принцип роботи BIOS

Старт BIOS. Одразу після подачі живлення системний контролер починає генерувати тактові імпульси і подає сигнал RESET (скидання) на усі компоненти системи. Центральний процесор комп'ютера починає виконувати програму BIOS, яка міститься у відповідній мікросхемі. Власне, ця стадія стосується лише частини BIOS, що відповідає за первинне завантаження і називається Boot-блок. Програма завантажувального блоку обчислює контрольні суми усього вмісту BIOS і у випадку їх невідповідності (зазвичай при пошкодженні програмного коду BIOS) припиняє виконання штатного коду і видає сигнал помилки (звуковий сигнал).

Відновлення BIOS. При негараздах BIOS, що можуть виникнути в результаті яких-небудь дій користувача або збою апаратури, управління передається спеціальній процедурі, на яку покладено функцію відновлення – Crisis Recovery. Ця процедура покликана в аварійному порядку прочитати з дискети, інколи навіть з жорсткого диска, файл BIOS, а потім записати його в мікросхему замість пошкодженого коду, тим самим відновивши стан персональної платформи до нормального.

Початкова стадія. На цьому етапі виконується початкове тестування всіх вузлів та компонентів комп'ютера, яке називається POST (англ. Power-On Self Test – самотестування після подачі живлення). Окрім цього, метою процедури POST є робота з програмними ресурсами персональної платформи: обчислення обсягу оперативної пам'яті, пошук та ініціалізація відео системи, послідовних та паралельних портів, накопичувачів на гнучких та жорстких дисках, додаткових пристроїв, що підключені до PCI та USB шин абощо.

Етапи ініціалізації та перевірки працездатності відстежуються засобами діагностики BIOS. Для цього процедури POST при переході від одного до іншого пристрою щоразу посилають у діагностичний порт (Manufacturing Test Port) спеціальні сигнали, що називаються POST-кодами. Деякі з них дублюються відповідними звуковими сигналами. В разі, коли виникають

помилки, завантаження комп'ютера припиняється до усунення несправності. Про характер несправності можна зробити висновки, судячи з останнього POST-коду або звукового сигналу.

У своїй роботі процедури POST керуються налаштуванням BIOS, читаючи їх із CMOS-пам'яті – особливого різновиду пам'яті, призначеного для зберігання апаратної конфігурації комп'ютера. Крім того, тут також знаходяться всі налаштування BIOS, які може змінювати користувач – характеристики оперативної пам'яті (таймінги), частота роботи процесора, параметри жорсткого диска і ін.

Фінальна стадія. На фінальній стадії виконується те, задля чого власне й розроблявся BIOS. У наперед заданий (один і той же для всіх персональних платформ) програмний сегмент записуються процедури обробки операцій введення та виведення даних. Це дозволяє операційній системі, коли вона перейме управління від BIOS, послуговуватись бібліотеками програм в оперативній пам'яті, що вже заздалегідь розміщені там.

Якби всі програми самостійно намагалися опікуватися периферійними пристроями та містили б в собі подібні інструкції, то вони працювали вельми не ефективно та займали б забагато місця. Окрім того, кожен новий пристрій потребував би повної модифікації існуючих програм. Щоб уникнути подібних проблем, велику частину роботи по обробці даних переклали на BIOS. Це, напевно, не вирішило всіх проблем, але щонайменше значно спростило їх вирішення.

Хоча сучасні операційні системи практично не використовують або взагалі не використовують можливості BIOS по обробці операцій введення-виведення, з розвитком технічного прогресу роль BIOS зовсім не зменшується. З введенням у дію стандарту ACPI одна із першочергових задач BIOS – підготовка та передача операційній системі методів керування ресурсами персональної платформи.

Фінальна стадія завершується завантаженням операційної системи. Управління передається програмі, що знаходиться в Boot-секторі

(завантажувальному секторі) дискети, жорсткого диска, компакт-диска) або віддаленого носія, вказаного по мережі. Далі управління беруть на себе вбудовані механізми операційної системи.

1.2 POST

POST (англ. Power-On self-test) – програма самотестування комп'ютера, яку виконує центральний процесор після подачі живлення або отримання команди RESET. На сучасних персональних платформах окрім діагностичних задач на POST покладається також налаштування апаратних ресурсів комп'ютера та взаємодія з користувачем, що може вказати деякі параметри та обрати ті чи інші можливості для подальшої роботи.

POST складається з послідовних кроків, спрямованих на перевірку всіх вузлів та компонентів комп'ютера, кожний з яких відмічається контрольними точками або POST-кодами. Для візуалізації POST-кодів використовується один або декілька послідовних чи паралельних портів вводу-виводу, або ж Manufacturing Test Port – діагностичний порт 80h. В цьому випадку для відображення контрольних точок потрібен додатковий пристрій: POST-контролер, або як його ще називають діагностична POST-карта, за допомогою якої здійснюється моніторинг всіх етапів виконання POST. Це стає можливим за рахунок того, що POST-контролер перехоплює діагностичні повідомлення (коди) та наочно відображає їх цифровому індикаторі.

Крім візуальних засобів, процедури POST використовують також звукові повідомлення про ситуації, що склалися, та негаразди, які зафіксовані в процесі виконання. Деякі з них носять інформаційний характер і повідомляють про перебіг ініціалізації регістрів, контролерів чи окремих пристроїв; інші – слугують сигналом аварійного завершення POST внаслідок помилок або збоїв систем та вузлів комп'ютера.



Рис. 10.1. POST-карта, дозволяє діагностувати неполадки на стадії запуску комп'ютера

1.3 Установки конфігурації системи

При кожному запуску або скидання системи програма BIOS перевіряє установки конфігурації системи, щоб визначити, які типи додаткових апаратних пристроїв є в системі. ПК оснащений спеціальною пам'яттю RAM з додатковим живленням від батарейки CR2032. Ця пам'ять використовується для зберігання інформації про розширеної конфігурації системи та називається CMOS RAM.

Акумулятор для живлення CMOS RAM виглядає як невелика монетка, що вставляється в спеціальне гніздо на материнській платі поруч з чіпом ROM BIOS, яку легко впізнати. Але деякі системні плати не мають замінних акумуляторів для живлення CMOS RAM. У таких системних платах живлення CMOS RAM та годинника реального часу (RTC, real-time clock) забезпечується незамінюваною батарейкою, розрахованої на термін служби 10 років та вбудованої в чіп годинника реального часу. Функція годинника реального часу надає для системи інформацію про поточний час та дату.

Так як установки конфігурації системи є для системи основним джерелом інформації про встановлене додаткове устаткування, то вони повинні бути необхідним чином налаштовані, щоб правильно відображати обладнання, що

використовується в системі. Сучасні ПК можуть автоматично переконфігурувати самі себе після установки нового обладнання. Ця можливість називається Plug and Play. Завдяки PnP користувачеві не потрібно виконувати конфігурацію системи, переставляючи перемички на материнській платі або редагувати вміст CMOS RAM, як це робилося в попередніх системах. Це можливою завдяки тому, що системна BIOS і роз'єми розширення і плати розширення розроблені таким чином, що операційна система може конфігурувати їх автоматично.

У процесі завантаження операційної системи BIOS PnP виконує перевірку, які пристрої встановлені в комп'ютері. Пристрої PnP можуть надавати BIOS інформацію про свої характеристики і про те яким чином взаємодіяти з ними. Ця інформація зберігається в CMOS RAM, щоб система могла працювати з пристроєм.

1.4 Утиліта CMOS Setup

Під час процесу завантаження системи BIOS виводить на екран повідомлення. Щоб користувач мав можливість отримати доступ до утиліти для налаштування CMOS, яка називається CMOS Setup, необхідно натиснути певну клавішу або комбінацію клавіш. Ці клавіші різні для комп'ютерів різних виробників, але зазвичай використовуються клавіші <Delete>, <F2> та <Esc>. Якщо клавіша для входу в утиліту CMOS Setup не було натиснуто протягом певного часу, то BIOS продовжує процес завантаження комп'ютера.

Якщо ж клавіша натиснута в межах встановленого періоду, то завантаження системи зупиняється й відкривається екран утиліти CMOS Setup, де користувач може змінити установки конфігурації системи. Значення, що вводяться в утиліті CMOS Setup, зберігаються в конфігураційних регістрах CMOS RAM, які перевіряються при кожному завантаженні системи. Кожна модель системної плати має специфічну BIOS, розроблену на основі застосовуваного в ній чіпсета. Тому кожна утиліта CMOS Setup повинна працювати з функціями, специфічними для чіпсета даної розробки системної

На рис. 10.2 показаний приклад головного вікна утиліти налаштування CMOS.



Наступними за частотою налаштування є опції **Power Management Setup** (Налаштування управління енергоспоживанням), **PnP / PCI Configuration** (Конфігурація PnP / PCI), **Integrated Peripherals** (Вбудовані периферійні пристрої) та **Password Maintenance Services** (Служба управління паролями). Утиліта налаштування CMOS для BIOS конкретної системної плати може містити такі ж опції, як показано на рис. 10.2, опції, які виконують такі ж

функції, але під іншими назвами, додаткові опції, а деякі з показаних опцій можуть бути відсутні.

Утиліти налаштування CMOS для BIOS нових системних плат містять дві опції: **Auto Configuration** (Автоматична конфігурація) та **Default Settings** (Налаштування за замовчуванням), призначені допомогти користувачеві з налаштуванням параметрів CMOS. Вибравши опцію автоматичної конфігурації, користувач передає завдання конфігурації параметрів системи утиліті налаштування CMOS. У більшості випадків автоматичне конфігурування налаштовує стандартні пристрої на ефективну роботу на базовому рівні, але при цьому продуктивність системи не оптимізована. Для оптимізації продуктивності системи необхідно відключити опцію автоконфігурування і вказати необхідні значення параметрів вручну. Функція автоконфігурування має дві опції: **Auto Configure with Power-On Defaults** (Встановити налаштування за замовчуванням при включенні системи) та **Auto Configure with BIOS Defaults** (Встановити параметри BIOS за замовчуванням).

Екрани меню налаштування CMOS різних виробників BIOS в значній мірі стандартизовані і дозволяють конфігурувати однакові основні параметри. Зокрема, через них можна встановити системний годинник та календар, параметри жорстких дисків, а також вказати типи помилок, при виникненні яких під час процедури POST процес завантаження системи буде зупинений.

На комп'ютерах, випущених кілька 10 років тому, встановлювалася BIOS фірми Award версії 4.51PG. Сучасні комп'ютери використовують BIOS фірми Award версій 6.0 або 6.0PG. AMI пропонує BIOS версії 1.24. Остання поширена версія - 1.45 - практично повністю копіює попередню з дещо зміненим інтерфейсом.

1.5 Графічний інтерфейс UEFI-BIOS

У далекому минулому, коли програмісти розробляли BIOS, ніхто з них не замислювався над тим, скільки ще їй доведеться працювати. Але час минав, техніка змінювалася, і багато новинок доводилося вписувати в стару систему

але почали з'являтися проблеми. Одна з них полягає ось у чому, сучасні жорсткі диски вже досягли розмірів 3Тб, а стара BIOS бачить тільки 2Тб. Виходить, що комп'ютери, оснащені BIOS, обмежені в обсягах пам'яті, крім цього кожен виробник материнських плат робив свої інтерфейси, що плунав користувачів.



Рис. 10.3. Вікно головного меню графічного інтерфейсу UEFI-BIOS

Все зовсім інакше виглядає із застосуванням **UEFI**: обсяг жорсткого диска необмежений, єдиний для всіх інтерфейс. В UEFI є безліч нових додаткових функцій, які були недоступні в старих версіях BIOS, це наприклад резервне копіювання даних та прибрані деякі зайві функції, що вже не використовуються.

Перша кнопка **GreenPower**, вона зеленого кольору, в центрі неї зображена лампа. Вона призначена для настройки енергозбереження процесора. Друга кнопка «**Службова програма**», після натискання на неї відкриється додаткове меню з допоміжними програмами **UEFI Bios** (рис. 4): «Стан системи» (показує яке обладнання встановлено в ПК); «Спеціальні настройки для професіоналів»; «Можливість установки нової версії UEFI»; «Установка пароля доступу до налаштувань та до комп'ютера»;

«Завантаження» (визначається черговість завантажувальних пристроїв);
«Збереження всіх змін».



Рис. 10.4. Меню Utilities - Службові програми

Третя кнопка – меню налаштувань (рис.10.3). Вона призначена для розгону процесора. Четверта кнопка «Налаштування» (рис. 10.3), містить підменю з кнопками додаткових налаштувань (рис. 10.5).



Рис. 10.5. Меню Settings – Настройки

Меню Settings (рис. 10.5) містить такі кнопки: «Тестування оперативної пам'яті»; «Live Update», дає можливість оновити UEFI, через Інтернет; «HDD Backup» – резервне копіювання даних жорсткого диска; «Зміна інтерфейсу UEFI».

Переваги UEFI:

1. Підтримує жорсткі диски величезного обсягу. BIOS для управління жорстким диском, використовував програму Master Boot Record, вона містила в собі, всю інформацію про розділи диска. Але, у неї був один великий недолік, розміри записів в ній були всього по 32 біта, виходить, що контролювати BIOS міг тільки 4 мільярди секторів, що в загальній сумі складає 2Тб. UEFI працює зі стандартом GPT, що дає можливість підтримки жорстких дисків об'ємом до 8 млрд Тб.

2. Вбудована BIOS. Материнським платам з UEFI не потрібна BIOS, тому що в ній є своя вбудована BIOS, яка називається – модуль підтримки сумісності. Тому ті програми, яким для роботи потрібен був BIOS, спокійно можуть працювати і на комп'ютерах з UEFI.

3. Просте управління. В меню налаштувань все можна робити за допомогою мишки, раніше BIOS управління було можливо тільки з клавіатури.

4. Швидкість завантаження ОС. Приблизно 30-60 секунд йде на вашому комп'ютері з моменту включення в мережу і до повного завантаження Windows. З UEFI завантаження відбувається набагато швидше, зараз вже досягнутий рекорд - 2 секунди.

5. Вбудована система. UEFI сама по собі є операційною системою, вона чимось схожа на DOS, тому що виконує текстові команди. Вона може допомогти розібратися в причинах відмови завантаження основної «Операційної Системи», якщо таке сталося, але працювати в ній можуть тільки досвідчені користувачі.

6. Додаткові програми. В UEFI за бажанням можна додавати програми. Дається можливість установки додатків (утиліти та прості ігри).

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

1. Увімкніть монітор та комп'ютерну систему.

2. Для спостереження повідомлень, що надходять від комп'ютера в процесі запуску, використовуйте клавішу Pause / Break. Комп'ютер припиняє завантаження і дає можливість уважно прочитати повідомлення. Для продовження запуску використовуйте клавішу ENTER.

3. Визначте версію BIOS.

4. Вкажіть протестований обсяг пам'яті.

5. Дані, що визначають склад комп'ютерної системи та її налаштування, на екрані монітора відображаються в таблиці System Configuration. Призупинивши запуск за допомогою клавіші PAUSE / BREAK, вивчіть таблицю та встановіть:

- скільки жорстких дисків має комп'ютерна система і який їхній обсяг;
- скільки послідовних і паралельних портів є в наявності;
- до якого типу відносяться мікросхеми, розміщені в банках пам'яті;

Продовжить запуск клавішею ENTER.

6. Визначте тип установленної операційної системи.

7. Дочекавшись закінчення запуску операційної системи. Перезавантажте комп'ютер.

8. Запишіть порядок початкового завантаження комп'ютера, відзначте, що є кінцевим пунктом кожного етапу.

9. Заповніть табл. 10.1.

Табл 10.1. Зразок оформлення конфігурації комп'ютера

Елемент конфігурації	Маркування, тип	Додаткові характеристики	Значення
BIOS			
Процесор		Тип, тактова частот	
Оперативна пам'ять		Тип, обсяг	
Жорсткий диск		Кількість обсяг	
Порти вводу / виводу		Кількість: послідовні, паралельні	
Тип операційної системи			

КОНТРОЛЬНІ ПИТАННЯ

1. Що таке BIOS?
2. Як відбувається самотестування?
3. Які ознаки справного жорсткого диску, які звуки він створює при правильній роботі?
4. Як скинути налаштування BIOS, якщо комп'ютер перестав завантажуватись?
5. Що таке UEFI і у чому відмінності від звичайного BIOS?
6. Яким чином можна збільшити частоту роботи процесора?
7. Який баланс потрібно зберігати при розгоні системи і чому?
8. Яким чином відбувається збільшення частоти відеокарти?
9. Можливо розігнати будь яку відеокарту чи лише спеціальні версії?
10. Звукові повідомлення BIOS, які вони бувають і для чого використовуються. Наведіть приклади для різних версій BIOS.
11. Як можна досягти кращого охолодження комп'ютерної системи?

ПЕРЕЛІК ПИТАНЬ ДЛЯ САМОСТІЙНОЇ РОБОТИ СТУДЕНТА

1. Загальні відомості про комп'ютерні сигнали.
2. Параметри реальних імпульсних сигналів.
3. Графічний спосіб завдання логічних функцій.
4. Мінімізації логічної функції за допомогою діаграми Вейча.
5. Цифрові компаратори двійкових кодів. Мажоритарні пристрої.
6. Пристрої контролю на парність та на непарність.
7. Тригери в інтегральному виконанні
8. Синтез тригерів із заданим законом функціонування та оцінка його характеристик
9. Регістри в інтегральному виконанні.
10. Лічильники в інтегральному виконанні.
11. Мікросхеми з відкритим колектором.
12. Логічні елементи транзисторно-транзисторної логіки з трьома вихідними станами
13. Принцип побудови фільтра сигналів за допомогою операційного підсилювача.
14. Області застосування аналого-цифрових та цифро-аналогових перетворювачів.
15. Області застосування генераторів напруги, що лінійно змінюються.
16. Таймери в інтегральному виконанні.
17. Прямий доступ до пам'яті. Призначення і функції чипсету в мікропроцесорній системі.
18. Принципи побудови схемного та мікропрограмного засобів управління.
19. Побудова запам'ятовувальних пристроїв заданої організації на мікросхемах пам'яті різного типу.
20. Розподіл та організація віртуальної пам'яті на основі сторінкового розподілу.
21. Архітектура 32-розрядного мікропроцесора.
22. Машинне подання команд різних форматів.

23. Паралельні структури обчислювальних систем.
24. Трансп'ютери технології й способи міжпроцесорного обміну даними.
25. Мультиядерні процесори *ARM*-архітектури.
26. Мультиядерні процесори на базі *MIPS*-сумісних ядер.
27. Основні концепції технології віртуалізації (*POWER*)
28. Компоненти і функції *POWER Hypervisor*.
29. Компоненти і функції архітектури *HACMP*.
30. Компоненти топології кластера.
31. Багатопроцесорні та багатомашинні обчислювальні системи.

ТЕСТИ ДЛЯ САМОКОНТРОЛЮ

1. Яка кількість основних інформаційних шин входить в системну магістраль мікропроцесорної системи?

- 1) три шини
- 2) дві шини
- 3) чотири шини
- 4) вісім шин

2. Для того, щоб інформація зберігалася довгий час її, потрібно записати:

- 1) на жорсткий диск
- 2) у реєстри процесора
- 3) у оперативну пам'ять
- 4) до кеш пам'яті.

3. Для чого служить реєстр ознак?

- 1) для зберігання прапорців результатів виконаних операцій
- 2) для зберігання кодів спеціальних команд
- 3) для зберігання коду адреси
- 4) для визначення режиму роботи мікропроцесорної системи.

4. Яка основна перевага сегментації пам'яті?

- 1) сегментація спрощує завдання адреси операнда
- 2) сегментація спрощує структуру процесора
- 3) сегментація спрощує перемикання між сегментами даних і між сегментами програм
- 4) сегментація дозволяє збільшити об'єм пам'яті системи

5. На пам'яті якого типу організовано системне ОЗП?

- 1) FLASH
- 2) SRAM
- 3) DDRAM
- 4) PROM

6. Режим переривання використовують коли:

- 1) з'являється необхідність негайної передачі від пристрою введення/виведення до МП
- 2) необхідно здійснити обмін інформацією без участі МП

- 3) потрібно відключити МП від системних шин на час передачі даних
- 4) для перевірки МП пристрою введення/виведення на готовність приймати/передавати інформацію.

7. Стек:

- 1) запам'ятовує копії інформації між процесором та основною пам'яттю
- 2) зберігає черги даних з порядком вибірки слів таким же, як й порядок їх надходження
- 3) зберігає дані, які не дуже часто змінюються
- 4) використовується в якості зовнішньої пам'яті.

8. Який компонент системи забезпечує початковий старт комп'ютера?

- 1) BIOS
- 2) процесор
- 3) оперативна пам'ять
- 4) чипсет.

9. Для чого застосовується мультиплексування шин?

- 1) Для зменшення кількості шин
- 2) Для збільшення пропускної спроможності
- 3) Для збільшення кількості шин
- 4) Для зменшення пропускної спроможності

10. Зовнішня пам'ять - це:

- 1) Пам'ять, що призначена для тривалого зберігання програм і даних
- 2) Накопичувач на гнучких магнітних дисках
- 3) Пам'ять, що призначена для тимчасового зберігання програм і даних
- 4) Накопичувач на оптичних магнітних дисках.

11. Який принцип роботи стекової пам'яті?

- 1) перший записаний код читається першим
- 2) перший записаний код читається останнім
- 3) вміст стекової пам'яті не міняється за час роботи системи
- 4) стекова пам'ять прискорює роботу пам'яті векторів переривань.

12. Який режим обміну забезпечує найбільшу швидкість передачі інформації?

- 1) обмін з прямим доступом до пам'яті

- 2) програмний обмін
- 3) обмін за перериванням
- 4) усі режими однакові за швидкодією

13. Пріоритетне переривання полягає в тому, що:

- 1) у векторі вказується конкретна адреса пристрою
- 2) визначається пристрій, якій має пріоритет в обслуговуванні
- 3) єдиний пристрій введення/виведення потребує обслуговування МП
- 4) швидкодіючий пристрій обмінюється з МП по одному слову в порядку пріоритету

14. Яка шина використовується тільки для відеоадаптера?

- 1) ISA
- 2) VESA
- 3) PCI
- 4) AGP.

15. Як називається пристрій, що відповідає за виконання арифметичних, логічних і операцій управління, записаних в машинному коді?

- 1) мікропроцесор
- 2) оперативна пам'ять
- 3) система введення/виведення
- 4) пристрій керування.

16. Назвіть правильні характеристики зовнішньої пам'яті:

- 1) енергонезалежна, повільна, може зберігати великий об'єм інформації
- 2) енергозалежна, швидка, невелика за об'ємом
- 3) повільна, енергозалежна
- 4) енергонезалежна, повільна, невелика за об'ємом.

17. Яка функція конвеєра?

- 1) прискорити виконання логічних операцій
- 2) розпаралелювання виконання арифметичних операцій
- 3) зменшити кількість команд процесора
- 4) прискорення вибірки команд.

18. Яка архітектура забезпечує більш високу швидкодію?

- 1) Фон-нейманівська
- 2) Гарвардська

- 3) швидкодія не залежить від архітектури
- 4) Пристонська.

19. Який тип обміну даними найбільш небажаний для роботи швидких пристроїв введення/виведення?

- 1) програмне введення/виведення
- 2) векторне переривання
- 3) ПДП
- 4) радіальне переривання.

20. Векторне переривання полягає в тому, що:

- 1) у векторі вказується конкретна адреса пристрою
- 2) визначається пристрій, якій має пріоритет в обслуговуванні
- 3) МП визначає готовий периферійний пристрій до виконання операцій введення/виведення до початку програмної передачі даних
- 4) швидкодіючий пристрій обмінюється з МП по одному слову в порядку пріоритету

21. Сигнал READY:

- 1) показує, що пам'ять або зовнішній пристрій готові до обміну з мікропроцесором
- 2) використовується, як вихід підтвердження векторного переривання
- 3) формується зовнішнім пристроєм для запиту захоплення шин МП
- 4) підтверджує режим захоплення шин МП.

22. Яка з шин представлена на материнській платі не більше ніж одним роз'ємом?

- 1) ISA
- 2) AGP
- 3) PCI
- 4) EISA

23. ADDR bus розшифровується як:

- 1) шина адреси
- 2) шина даних
- 3) шина управління
- 4) системна магістраль.

24. Яка з приведених операцій не вимагає проведення циклу обміну інформацією?

- 1) читання даних з пам'яті
- 2) усі операції вимагають проведення циклу обміну
- 3) запис даних в пам'ять
- 4) читання команди з пам'яті.

25. У якій пам'яті зберігається вміст регістра ознак при перериванні?

- 1) у стековій пам'яті
- 2) у пам'яті векторів переривань
- 3) у пам'яті програм початкового запуску
- 4) у пам'яті пристроїв, підключених до магістралі.

26. Який режим обміну використовується найчастіше?

- 1) обмін за перериваннями
- 2) усі режими використовуються однаково часто
- 3) обмін з прямим доступом до пам'яті
- 4) програмний обмін.

27. Для чого використовується вектор переривання?

- 1) для скидання програми
- 2) для переривання центрального процесора
- 3) для виклику підпрограми обробки
- 4) для повернення у програмно-керований режим роботи.

28. Для зменшення втрат часу при обміні масивами даних доцільно застосувати:

- 1) векторне переривання
- 2) пріоритетне переривання
- 3) режим прямого доступу до даних
- 4) програмно-керований обмін.

29. Сигнал INTR:

- 1) підтверджує режим захоплення шин МП
- 2) використовується, як вхід запиту векторного переривання
- 3) використовується, як вихід підтвердження векторного переривання
- 4) формується зовнішнім пристроєм для запиту захоплення шин МП.

30. Контролер жорсткого диска називають:

- 1) IDE
- 2) FDC
- 3) LPT
- 4) COM.

31. При шинній структурі зв'язку сигнали між пристроями передаються:

- 1) по одних і тих же лініях зв'язку, але в різний час
- 2) по різних лініях зв'язку, але в один і той же час
- 3) по одних і тих же лініях зв'язку і в один і той же час
- 4) по різних лініях зв'язку.

32. Який тип обміну забезпечує більш високу швидкість передачі інформації?

- 1) синхронний
- 2) асинхронний
- 3) не можна сказати однозначно
- 4) синхронний обмін з можливістю асинхронного обміну.

33. Виберіть вірне твердження:

- 1) пристрій введення-виведення завжди має безліч адрес на магістралі
- 2) пристрій введення-виведення може мати одну адресу на магістралі
- 3) пристрій введення-виведення призначений виключно для двонаправленого обміну із зовнішніми пристроями
- 4) пристрій введення-виведення нічим не відрізняється від модуля пам'яті.

34. Який сенс вкладається в термін «архітектура» ПК?

- 1) визначення фізичної системи ПК
- 2) розгляд апаратних засобів ПК
- 3) розгляд програмного забезпечення
- 4) розгляд програмної моделі ПК.

35. Який з режимів обміну дозволяє виконувати операції в обхід процесора?

- 1) програмне введення/виведення
- 2) ПДП
- 3) процедура переривання
- 4) poling.

36. Сигнал HOLD:

- 1) підтверджує режим захоплення шин МП
- 2) використовується, як вхід запиту векторного переривання
- 3) використовується, як вихід підтвердження векторного переривання
- 4) формується зовнішнім пристроєм для запиту захоплення шин МП.

37. Мікропроцесорний пристрій - це:

- 1) функціонально і конструктивно закінчений виріб, що складається з декількох мікросхем, до складу яких входить мікропроцесор, призначений для виконання певного набору функцій : отримання, обробка, передача, перетворення інформації і управління
- 2) одиниця виміру (квант) тривалості виконання команди
- 3) пристрій, по якому передається адреса елементу пам'яті або блоку введення-виводу
- 4) електронна схема, що виконує функції цілого пристрою та розташовується на одній інтегральній схемі.

38. Яка структура шин адреси і даних забезпечує більшу швидкодію?

- 1) мультиплексуєма
- 2) немультимплексуєма
- 3) двонаправлена
- 4) швидкодія від типу структури не залежить.

39. Яка команда використовується для повернення з програмного переривання?

- 1) команда умовного переходу
- 2) команда безумовного переходу
- 3) команда переходу з поверненням
- 4) спеціальна команда повернення з переривання.

40. По якій з системних шин передаються коди команд?

- 1) по інформаційній шині
- 2) по шині даних
- 3) по шині адреси
- 4) по шині керування.

41. Поясніть значення аббревіатури IRQ 1:

- 1) переривання номер 1

- 2) лінія переривання номер 1 на системній платні
- 3) вектор переривань з адресою 0001H
- 4) мітка початку підпрограми переривань.

42. Сигнал HLDA:

- 1) використовується, як вхід запиту векторного переривання
- 2) використовується, як вихід підтвердження векторного переривання
- 3) формується зовнішнім пристроєм для запиту захоплення шин МП
- 4) підтверджує режим захоплення шин МП.

43. Скільки мінімум біт необхідно для розміщення числа 32000?

- 1) 32
- 2) 16
- 3) 14
- 4) 15.

44. Архітектура ЕОМ - це:

- 1) опис структури і функцій ЕОМ на рівні, достатньому для розуміння принципів роботи і системи команд ЕОМ
- 2) Опис деталей технічної і фізичної будови комп'ютера
- 3) Опис будови пристроїв введення-виведення
- 4) Опис програмного забезпечення необхідного для роботи ЕОМ.

45. Який регістр визначає адресу поточної виконуваної команди?

- 1) спеціалізований регістр
- 2) будь-який з адресних регістрів
- 3) регістр-акумулятор
- 4) регістр показчик стека.

46. Який принцип лежить в основі динамічної пам'яті?

- 1) заряд/розряд паразитної ємності
- 2) включення/виключення тригерів
- 3) перетворення енергії
- 4) інжекція електронів у затвор польового транзистора за рахунок створення напруженості електричного поля у тонкому діелектрику.

47. Що означає поняття «32-розрядний процесор»?

- 1) 32-разрядна шина даних

- 2) 32-розрядні регістри загального призначення
- 3) 32-розрядна шина адреси
- 4) 32 розрядна системна магістраль.

48. У чому основне призначення режиму Protected Mode?

- 1) у використанні механізму захисту програм і даних
- 2) у мультипрограмних можливостях
- 3) у створенні віртуального простору
- 4) перехід до енергозберігаючого режиму.

49. Послідовний порт інакше називають ... порт:

- 1) IDE
- 2) PS/2
- 3) LPT
- 4) COM

50. Яка пам'ять втрачає дані при відключенні живлення?

- 1) EEPROM
- 2) FLASH
- 3) SRAM
- 4) PROM

51. Як називається шина, в якій передача даних може виконуватися в обох напрямках?

- 1) двонаправлена
- 2) однонаправлена
- 3) шина з вихідним сигналом
- 4) мультиплексуєма.

52. Для чого призначені регістри процесора?

- 1) для виконання арифметичних операцій
- 2) для тимчасового зберігання інформації
- 3) для прискорення вибірки команд з пам'яті
- 4) для управління перериваннями.

53. Яке розділення функцій між внутрішніми регістрами процесора?

- 1) призначення регістрів залежить від типу процесора
- 2) усі регістри виконують одні і ті ж функції

- 3) половина регістрів використовується для даних, половина - для адресації
- 4) кожен регістр виконує свою індивідуальну функцію.

54. На якому принципі заснована робота статичної пам'яті?

- 1) на роботі тригерів
- 2) на заряді паразитної ємності
- 3) на роботі логіки
- 4) інжекція електронів у затвор польового транзистора за рахунок створення напруженості електричного поля у тонкому діелектрику.

55. Завдяки якій процедурі в осередках DRAM підтримується постійний рівень заряду?

- 1) переривання
- 2) ПДП
- 3) регенерації
- 4) інжекції заряду.

56. Суть «Гарвардської архітектури» полягає в тому що:

- 1) пам'ять програм відокремлена від пам'яті даних, для того щоб одночасно виконувати вибірку команд та даних
- 2) пам'ять програм поєднана з пам'яттю даних, для того щоб одночасно виконувати вибірку команд та даних
- 3) обробка виконується декількома процесорами з спільною пам'яттю
- 4) застосовується матриця однакових процесорних елементів зі спільною системою керування.

57. Як організована багатозадачність?

- 1) за допомогою декількох каналів
- 2) виділенням для кожної програми кванта часу
- 3) передачею управління
- 4) перехід до виконання наступної задачі після завершення попередньої.

58. Що таке порт?

- 1) пристрій введення-виведення
- 2) пристрій зв'язку магістралі з системною пам'яттю
- 3) буфер магістралі усередині процесора
- 4) зовнішній пристрій, з яким здійснюється сполучення.

59. Що таке виконавча адреса?

- 1) адреса початку сегменту
- 2) адреса поточної виконуваної команди
- 3) розмір сегменту
- 4) зміщення відносно початку сегменту.

60. На пам'яті якого типу організована кеш-пам'ять?

- 1) FLASH
- 2) SRAM
- 3) DRAM
- 4) FRAM.

61. Кеш-пам'ять:

- 1) запам'ятовує копії інформації між процесором та основною пам'яттю
- 2) зберігає черги даних з порядком вибірки слів таким же, як й порядок їх надходження
- 3) зберігає дані, які не дуже часто змінюються
- 4) представляє пам'ять з послідовним доступом.

62. Для чого служить регістр ознак?

- 1) для зберігання прапорців результатів виконаних операцій
- 2) для зберігання кодів спеціальних команд
- 3) для зберігання коду адреси
- 4) для визначення режиму роботи мікропроцесорної системи.

63. При якому типі переривань їх загальна кількість може бути більшою?

- 1) при векторних перериваннях
- 2) при радіальних перериваннях
- 3) максимальне число переривань постійне для будь-якого типу
- 4) максимальне число переривань не обмежено.

64. Що відрізняє процесори з RISC-архітектурою від процесорів з CISC - архітектурою?

- 1) тактова частота
- 2) можливість паралельного виконання декількох команд
- 3) системою команд
- 4) спосіб звернення до пам'яті команд.

65. Яка пам'ять не змінює свого змісту в ході виконання програми?

- 1) пам'ять даних
- 2) регістри МК
- 3) енергонезалежна пам'ять даних
- 4) пам'ять програм.

66. Час и дату системи можна встановити за допомогою

- 1) кеш-пам'яті
- 2) південного мосту
- 3) утиліти настройки CMOS
- 4) північного мосту.

67. На якому етапі завантаження операційної системи BIOS взаємодіє з пристроями PnP системи?

- 1) під час процедури POST
- 2) на етапі ініціалізації
- 3) на початковому етапі завантаження операційної системи
- 4) під час налаштування CMOS/

68. Який контакт мікросхеми застосовується для правильної орієнтації мікросхеми процесора при установці в сокет?

- 1) контакт 0
- 2) контакт 1
- 3) контакт 10
- 4) контакт 8

69. Для завантаження операційної системи комп'ютера перший набір інструкцій зберігається в

- 1) CMOS
- 2) ROM BIOS
- 3) CPU
- 4) RAM

70. На початкових стадіях процесу завантаження операційної системи перевірка комп'ютера виконується

- 1) центральним процесором
- 2) утилітою CMOS Setup
- 3) процедурою POST
- 4) контролером переривань

71. Яка з перерахованих фраз правильно відображає послідовність завантаження операційної системи?

- 1) виконання POST, ініціалізація, самозавантаження
- 2) ініціалізація, самозавантаження, виконання POST
- 3) самозавантаження, виконання POST, ініціалізація

4) ініціалізація, виконання POST, самозавантаження.

72. При старті комп'ютера на моніторі виводиться повідомлення Non-system disk or disk error...Про що воно повідомляє?

- 1) не виявлений системний диск і операційна система не може завантажитись
- 2) системний диск пошкоджений і операційна система буде завантажена з іншого диску
- 3) пошкоджена операційна система
- 4) не вірно настроєний CMOS SETUP

73. Для запуску системної плати джампер Clear CMOS повинний знаходитись у положенні

- 1) Normal
- 2) Clear
- 3) в жодному з положень
- 4) On

74. Про яку несправність повідомляє 1 довгий 1 короткий сигнал під час процедури POST

- 1) несправність оперативної пам'яті
- 2) несправність системного таймера
- 3) помилка перевірки контрольної суми мікросхеми BIOS
- 4) несправність відеоадаптера.

75. Про яку несправність повідомляє 1 довгий 1 короткий сигнал, що повторюється, під час процедури POST

- 1) помилка ініціалізації оперативної пам'яті
- 2) помилка ініціалізації центрального процесора
- 3) помилка читання даних з мікросхеми BIOS
- 4) несправність відеоадаптера.

76. Про що повідомляє 1 короткий сигнал під час процедури POST

- 1) проблеми з блоком клавіатури на системній платі
- 2) помилка ініціалізації центрального процесора
- 3) помилка читання даних з мікросхеми BIOS
- 4) помилок не виявлено.

77. Для чого потрібна термопаста

- 1) для виключення повітряного простору, що утворюється між процесором та радіатором
- 2) для передавання тепла від процесора до радіатора
- 3) для захисту від перегрівання процесора
- 4) для щільного та м'якого прилягання радіатора до поверхні.

78. При старті комп'ютера на моніторі виводиться повідомлення "BIOS checksum error". Про що воно повідомляє?

- 1) несправність оперативної пам'яті
- 2) несправність системного таймера
- 3) помилка перевірки контрольної суми мікросхеми BIOS
- 4) несправність відеоадаптера.

79. Яка правильна послідовність дій після процедури POST?

- 1) виявлення та ініціалізація відеоадаптера, визначення типу процесора, визначення типу та обсягу установленої в системі оперативної пам'яті
- 2) визначення типу процесора, виявлення та ініціалізація відеоадаптера, визначення типу та обсягу установленої в системі оперативної пам'яті
- 3) визначення типу процесора, визначення типу та обсягу установленої в системі оперативної пам'яті, виявлення та ініціалізація відеоадаптера
- 4) виявлення та ініціалізація відеоадаптера, визначення типу та обсягу установленої в системі оперативної пам'яті, визначення типу процесора.

80. Технологія самодіагностики S.M.A.R.T. стосується

- 1) самодіагностики HDD
- 2) самодіагностики DRAM
- 3) самодіагностики CPU
- 4) самодіагностики CMOS-пам'яті.

81. S.M.A.R.T. - це мікропрограма, що перебуває в

- 1) контролері диска
- 2) BIOS
- 3) операційній системі
- 4) процедурі POST.

82. Процедура Power-On Self Test перевіряє справність

- 1) контролера HDD
- 2) операційної системи
- 3) мікросхеми CMOS
- 4) слоту PCІe.

83. На екран монітора виводиться текстове повідомлення «Memory Configuration error». Про що повідомляє BIOS?

- 1) помилка контролю парності оперативної пам'яті
- 2) не працює або збоїть контролер пам'яті
- 3) помилка оперативної пам'яті
- 4) помилка в конфігурації пам'яті.

84. На екран монітора виводиться текстове повідомлення «Memory test fail». Про що повідомляє BIOS?

- 1) помилка контролю парності оперативної пам'яті
- 2) не працює або збоїть контролер пам'яті

- 3) помилка оперативної пам'яті
- 4) помилка в конфігурації пам'яті.

85. На екран монітора виводиться текстове повідомлення «OFFENDING SEGMENT». Про що повідомляє BIOS?

- 1) помилка контролю парності оперативної пам'яті
- 2) не працює або збоїв контролер пам'яті
- 3) помилка оперативної пам'яті
- 4) помилка в конфігурації пам'яті.

86. На екран монітора виводиться текстове повідомлення «RAM PARITY ERROR - CHECKING FOR SEGMENT..». Про що повідомляє BIOS?

- 1) помилка контролю парності оперативної пам'яті
- 2) не працює або збоїв контролер пам'яті
- 3) помилка оперативної пам'яті
- 4) помилка в конфігурації пам'яті.

87. Який інтерфейсів дозволяє підключати різні зовнішні пристрої до переносних і настільних комп'ютерів без відключення живлення комп'ютера?

- 1) FireBall
- 2) SATA
- 3) USB
- 4) PCIe.

88. Які дві функції виконуються частиною North Bridge набору чіпсету материнської плати?

- 1) обмін даними між процесором і звуковою картою
- 2) забезпечення роботи процесора, пам'яті та відеоадаптера
- 3) обмін даними між процесором і портами введення/виведення
- 4) доступ до плати відеоадаптера.

89. Північний міст призначений для

- 1) забезпечення взаємодії на материнській платі між чіпсетом материнської плати та її компонентів
- 2) обміну даними між процесором і портами введення/виведення
- 3) забезпечення роботи процесора, пам'яті та відеоадаптера, а також для визначення частоти системної плати
- 4) доступу до ОЗП

90. Технологія PnP дозволяє ...

- 1) синхронізувати роботу комп'ютера та зовнішніх пристроїв
- 2) відновити BIOS у випадку збоїв
- 3) новим пристроям автоматично налаштовуватись під конфігурацію

даного комп'ютера

- 4) завантажувати комп'ютер у випадку не критичних помилок POST.

91. Для чого призначений роз'єм DVI?

- 1) інтерфейс, який використовується для передачі відеосигналу в цифровому вигляді
- 2) інтерфейс, що застосовується для передачі цифрового відео та аудіо сигналу
- 3) роз'єм паралельного інтерфейсу, за допомогою якого здійснюється підключення принтера до персонального комп'ютеру
- 4) інтерфейс, який використовується для захоплення відеосигналу та введення його в комп'ютер.

92. Що це означає повідомлення на екрані монітора «Signal out of range»?

- 1) проблема виникла через погане підключення VGA кабелю
- 2) проблема виникла з причини переходу монітора в режим очікування
- 3) проблема виникла через неправильні налаштування роздільної здатності
- 4) проблема виникла через вихід із ладу інвертора LCD монітора.

93. На якому виході дешифратора з'явиться сигнал, якщо на входи подати код 0110?

- 1) Y6
- 2) Y5
- 3) Y8
- 4) Y0.

94. Який код треба подати на адресні входи дешифратора, що має 24 виходи, щоб зробити активним вихід номер 13?

- 1) 01101
- 2) 10010
- 3) 10110
- 4) 01111.

95. Який код треба подати на адресні входи дешифратора, що має 24 виходи, щоб зробити активним вихід номер 15?

- 1) 01101
- 2) 10010
- 3) 10110
- 4) 01111.

96. Мультиплексор підключає _____ на вихід

- 1) кожен парний вхід
- 2) один з входів, у залежності від адреси
- 3) кожен непарний вхід

4) кожен з входів послідовно.

97. Скільки адресних входів повинен мати мультиплексор-селектор 12 до 1?

- 1) 6
- 2) 3
- 3) 8
- 4) 4

98. З якого інформаційного входу мультиплексора пройде сигнал, якщо на адресні входи подати код 101?

- 1) D0
- 2) D0, D1
- 3) D5
- 4) D4, D5.

99. Двійковий суматор склав два числа. Перше число було 01101100. Сума склала 10001110. Яким було друге число?

- 1) 00100010
- 2) 01011000
- 3) 01011100
- 4) 01010011

100. Частота проходження імпульсів на виході трьох розрядного підсумовуючого лічильника послідовного рахунку

- 1) Зменшується у 8 разів
- 2) Зменшується у 6 разів
- 3) Збільшується у 30 разів
- 4) збільшується у 6 разів.

101. Шести розрядний лічильник знаходиться в режимі віднімання. Початковий стан 010110. В який стан перейде лічильник після надходження дев'яти лічильних імпульсів?

- 1) 001101
- 2) 101100
- 3) 011010
- 4) 111000.

102. Чому дорівнює ємність лічби п'ятирозрядного підсумовуючого двійково-десятькового лічильника?

- 1) 99999
- 2) 11111
- 3) 10000
- 4) 5000.

103. Визначити кількість тригерів, необхідну для побудови лічильника імпульсів з числом станів $K = 26$

- 1) 3
- 2) 5
- 3) 7
- 4) 26.

104. Для чого у двійково-десяткових лічильниках застосовується зворотний зв'язок?

- 1) для підвищення точності підрахунку імпульсів
- 2) для перетворення числового імпульсу у двійково-десятковий
- 3) для вилучення надлишкових станів
- 4) для перетворення десятикового коду у двійково-десятковий.

105. Двійковий лічильник використаний як подільник частоти. Вкажіть, на яку величину поділить лічильник вхідний сигнал частотою "f" якщо вихідний сигнал буде зніматися з виходу 23?

- 1) $f/2$
- 2) $f/4$
- 3) $f/8$
- 4) $f/16$.

106. Як називають лічильник, який може працювати як за схемою додавання, так і за схемою віднімання?

- 1) реверсивним
- 2) універсальним
- 3) паралельно-послідовним
- 4) зсуву.

107. Що таке реверсивний лічильник?

- 1) його код зростає при надходженні тактових імпульсів
- 2) він може рахувати на додавання або на віднімання
- 3) лічильник з високою швидкістю
- 4) лічильник з паралельним записом вхідної послідовності
- 5) він завжди рахує тільки на зменшення.

108. Як відрізняються один від одного сусідні вихідні сигнали лічильника?

- 1) вони мають частоти, що відрізняються у 2 рази
- 2) вони зсунуті один від одного на один такт
- 3) вони мають протилежну полярність
- 4) вони повністю однакові.

109. Який принцип покладений у режим роботи дільника частоти на задане число?

- 1) Лічильник починає рахувати в зворотню сторону при досягненні заданого коду
- 2) Лічильник починає рахувати у два рази швидше при досягненні заданого коду
- 3) Лічильник перестає рахувати при досягненні заданого коду
- 4) Лічильник скидається в нуль по досягненні заданого коду.

110. Десяткове число 132 була переведено в шістнадцятковий код. Як буде виглядати цей код?

- 1) B6
- 2) 7A
- 3) 84
- 4) D2.

111. Шістнадцятковий код числа B7 був переведений у двійкову форму, як виглядатиме результат?

- 1) 11001001
- 2) 10111110
- 3) 10110111
- 4) 11000111.

112. Десяткове число 132 було переведено в шістнадцятковий код. Як буде виглядати цей код?

- 1) B6
- 2) 7A
- 3) 84
- 4) D2.

113. Шістнадцятковий код числа B7 був переведений у двійкову форму, як виглядатиме результат?

- 1) 11001001
- 2) 10111110
- 3) 10110111
- 4) 11000111.

114. Число 123 з десятикової системи числення переведено у двійкову. Якому значенню у двійковій системі числення воно відповідає?

- 1) 1111011
- 2) 1101011
- 3) 1011011
- 4) 1110011.

115. У синхронних тригерах допускається змінювати інформаційні сигнали

- 1) коли надходять тактові імпульси

- 2) коли тактові імпульси блокують вхідні кола і запобігають зміні стану тригера
- 3) коли не надходять тактові імпульси
- 4) при наявності на інверсному виході тригера високого рівня.

116. Для побудови Т-тригера використовують

- 1) синхронний RS-тригер з зворотним зв'язком
- 2) D-тригер з зворотним зв'язком з прямого виходу
- 3) RS-тригер з зворотним зв'язком
- 4) JK-тригер, що з'єднаний з D-тригером.

117. Для того, щоб із JK-тригера отримати Т-тригер, необхідно:

- 1) $J=S$, $K=R$, $C=1$
- 2) $J=S$, $K=R$, C - імпульс лічби
- 3) $J=R$, $K=S$, $C=0$
- 4) $J=K=1$, на вхід C - імпульс лічби.

118. Що треба подати на входи RS тригера, щоб встановити тригер в 0?

- 1) $S=0$ $R=1$
- 2) $S=1$ $R=1$
- 3) $S=0$ $R=0$
- 4) $S=1$ $R=0$.

119. Що треба подати на входи тригера JK, щоб встановити тригер в 1?

- 1) $J=0$ $K=1$
- 2) $J=1$ $K=1$
- 3) $J=0$ $K=0$
- 4) $J=1$ $K=0$

КРИТЕРІЇ ОЦІНКИ ЗНАНЬ СТУДЕНТІВ

Шкала оцінки знань студента

Оцінка за національною 4-бальною шкалою	Рейтинг студента, бали	Оцінка за шкалою ECTS	Визначення оцінки ECST
Відмінно	90-100	A	ВІДМІННО – відмінне виконання лише з незначною кількістю помилок
Добре	82-89	B	ДУЖЕ ДОБРЕ – вище середнього рівня з кількома помилками
	75-81	C	ДОБРЕ – в загальному правильна робота з певною кількістю грубих помилок
Задовільно	66-74	D	ЗАДОВІЛЬНО – непогано, але зі значною кількістю недоліків
	60-65	E	ДОСТАТНЬО – виконання задовольняє мінімальні критерії
Незадовільно	35-59	FX	НЕЗАДОВІЛЬНО з можливістю повторного складання іспиту
	01-34	F	НЕЗАДОВІЛЬНО – необхідна серйозна подальша робота для повторного складання іспиту

Розподіл балів, які отримують студенти

Атестація 1

Поточне тестування, практичні роботи та самостійна робота								
T1	T2	T3	T4	T5	T6	T7	T8	T9
-	5	5	5	5	5	5	5	-

Атестація 2

Поточне тестування практичні роботи та самостійна робота								Підсумковий тест (залік)	Сума
T10	T11	T12	T13	T14	T15	T16	T17		
5	5	5	5	5	5	5	-	30	100

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

Основні

1. Бабич Н.П. Компьютерная схемотехника / Н.П.Бабич, И.А. Жуков. – К.: МК-Пресс, 2004. – 276 с.
2. Дэвид М., Хэррис С., Хэррис Л. Цифровая схемотехника и архитектура компьютера. Morgan Kaufman, 2017. 772 с.
3. Злобін Г. Г., Рикалюк Р.Є. Архітектура та апаратне забезпечення комп'ютерів. Навч. посібн. К.: Каравела. 2016. 224с.
4. Матвієнко, М.П. Архітектура комп'ютерів [Текст]: навч. посібник / М.П, Матвієнко, В.П, Розен, О.М. Закладний. – К.: Видавництво Ліра-К, 2013. – 264 с.
5. Матвієнко, М.П. Комп'ютерна логіка [Текст]: учебное пособие / М.П, Матвієнко. – К.: Видавництво Ліра-К, 2012. – 288 с.
6. Сенько В. І. Панасенко М. В., Сенько Є. В. Електроніка і мікропроцесорна техніка. К. Каравела. 2015. 676 с.
7. Сенько В. І., Панасенко М. В., Сенько Є. В. Електроніка і мікросхемотехніка. Том 3. Цифрові пристрої. Підручник. К.: Каравела. 2017. 400с.
8. Цирульник С. М. Азаров О. Д., Крупельницький Л. В., Трояновська Т. І. Мікропроцесорна техніка. Навчальний посібник. Вінниця: ВНТУ, 2017. 123с.
9. Якименко Ю. І. Терещенко Т. О. Мікропроцесорна техніка: Підручник. К.: «Політехніка», «Кондор», 2017. 440с.

Допоміжні

10. Бойко, В.И. Схемотехніка електронних систем. Мікропроцесори та мікроконтролери. [Текст]: Підручник / В.И. Бойко, А.М. Гуржій, В.Я Жуйков. – К.: Вища школа, 2004. Основи схемотехніки електронних схем / Бойко В.І., Жуйко В.Я. та інші. Підручник. – К.: Вища школа, 2004. – 526 с.
11. Брукс Ч. Дж. CompTIA A+. Устройство, настройка, обслуживание и ремонт ПК. СПб.: БХВ-Петербург, 2010. 1232 с.
12. Кори Сандлер. Ремонт персонального компьютера, 7-е издание. : Пер. с англ./ Кори Сандлер. – М.: Издательский дом «Вильямс», 2004. – 656 с.
13. Чураков, А.Я. Архітектура ЕОМ [Текст]: посібник/ А.Я. Чураков, С.В. Шаров, О.В. Строкань. – Мелітополь: РВЦ МДПУ, 2012. – 195 с.

14. Угрюмов, Е.П. Цифровая схемотехника. [Текст]: учебное пособие – СПб.: БХВ Петербург, 2001. – 180 с.

Інформаційні ресурси

1. Тестові завдання (внутрішній сайт ВНАУ – Електронна система Сократ)
2. Методичні розробки (внутрішній сайт ВНАУ – Електронна система Сократ).
3. TOP500 Becomes a Petaflop Club for Supercomputers [Електронний ресурс]. URL: <https://www.top500.org>
4. SIV - System Information Viewer [Електронний ресурс]. URL: <http://rh-software.com>
5. PC Building Simulator. [Електронний ресурс]. URL: https://store.steampowered.com/app/621060/PC_Building_Simulator

ПРИКЛАД ОФОРМЛЕННЯ ТИТУЛЬНОЇ СТОРІНКИ ЗВІТУ

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕКОНОМІКИ ТА ПІДПРИЄМНИЦТВА
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ЕКОНОМІЧНОЇ КІБЕРНЕТИКИ**

**КОМП'ЮТЕРНА СХЕМОТЕХНІКА ТА АРХІТЕКТУРА
КОМП'ЮТЕРІВ**

**ЗВІТ
з практичної роботи № 10
на тему**

ДОСЛІДЖЕННЯ ПОРЯДКУ ЗАПУСКУ КОМП'ЮТЕРА

**здобувача вищої освіти
першого (бакалаврського) рівня
галузі знань 12 «Інформаційні технології»
спеціальності 122 «Комп'ютерні науки»**

**Виконав: студент групи 21-КН
Петренко П. П.**

**Перевірів: к.т.н., доцент
Цирульник С. М.**

Вінниця – 2021

ДЛЯ НОТАТОК

ДЛЯ НОТАТОК

НАУКОВО-МЕТОДИЧНЕ ВИДАННЯ

Цирульник Сергій Михайлович

КОМП'ЮТЕРНА СХЕМОТЕХНІКА ТА АРХІТЕКТУРА КОМП'ЮТЕРІВ

Методичні вказівки
для проведення практичних занять
та самостійної роботи
для здобувачів вищої освіти
першого (бакалаврського) рівня
галузі знань 12 «Інформаційні технології»
спеціальності 122 «Комп'ютерні науки»
денної та заочної форми навчання

Набір і редагування авторські

Технічний редактор: *Цирульник Сергій Михайлович*
Верстка

Підписано до друку Формат 60x84/16.
Папір офсетний. Друк різнографічний.
Тираж 50 прим.

Віддруковано у редакційно-видавничому відділі
Вінницького національного аграрного університету
21008, м. Вінниця, вул. Сонячна, 3