

Міністерство освіти і науки України
Сумський державний університет



4844 Методичні вказівки
до курсової роботи
з теми **«Система передачі та відображення
двійково-десяткових чисел»**
із дисциплін **«Цифрова схемотехніка»,**
«Пристрої цифрової електроніки», «Схемотехніка»,
«Схемотехніка телекомунікаційних систем»
для студентів спеціальностей 171 **«Електроніка»,**
172 **«Телекомунікації та радіотехніка»**
і 153 **«Мікро- та наноелектроніка»**
всіх форм навчання

Суми
Сумський державний університет
2020

Методичні вказівки до курсової роботи «Система передачі та відображення двійково-десяткових чисел» із дисциплін «Цифрова схемотехніка», «Пристрої цифрової електроніки», «Схемотехніка» та «Схемотехніка телекомунікаційних систем» / укладачі: О. А. Борисенко, О. В. Бережна. – Суми : Сумський державний університет, 2020. – 41 с.

Кафедра електроніки і комп'ютерної техніки

ЗМІСТ

	С.
1 Вступ і постановка завдання	4
2 Структурна схема системи передачі та відображення двійково-десятькових чисел	5
3 Технічне завдання	8
4 Варіанти завдань	17
5 Виконання завдання	18
Список літератури	21
Додаток А. Зразок оформлення титульного аркуша курсової роботи	22
Додаток Б. Зразок оформлення рецензії на курсову роботу	23
Додаток В. Зразок змісту курсової роботи	24
Додаток Д. Зразок логічного синтезу перетворювача кодів за допомогою логічних функцій	25
Додаток Е. Зразок побудови перетворювача кодів за схемою «дешифратор-шифратор»	36
Додаток Ж. Зразок синтезу пристрою виявлення помилки	40

1 ВСТУП І ПОСТАНОВКА ЗАВДАННЯ

Тема курсової роботи «Система передачі та відображення двійково-десяткових чисел» спрямована на виконання важливого практичного завдання побудування на основі цифрових пристроїв систем передачі та відображення інформації. Фактично немає виробничого процесу, у якому не застосовувалися б такі системи. За їхньою допомогою будують, наприклад, системи збору даних від датчиків тепла, електроенергії, води.

Передача та відображення двійково-десяткових чисел відбувається також під час застосування цифрових пристроїв, які, наприклад, вимірюють частоту, час, тривалість одиночних імпульсів, їхніх фронтів і зрізів або зрушень між ними. Точність і швидкість таких вимірювань значно вища, ніж у разі використання осцилографів із каліброваними розгорненнями, розтяжками, мітками. Водночас досить часто є актуальним завдання не лише передачі та відображення інформації, а й підвищення її завадостійкості та секретності.

Розроблені різні способи розв'язання цього завдання, один із яких запропонований у цій курсовій роботі для системи передачі та відображення цифрових даних. Особливістю такої системи є те, що вона використовує 10 цифр від 0 до 9, заданих у двійково-десятковій формі та відображуваних на семи сегментних індикаторах. Водночас вимагається, щоб система передачі даних ще й була захищеною від завад і несанкціонованого доступу.

У курсовій роботі студентам потрібно розробити систему передачі та відображення однієї двійково-десяткової цифри. Передача та відображення інших двійково-десяткових цифр числа, що створюють додаткові розряди, для кожної з них може відбуватися паралельно за схемою, розробленою для однієї цифри. Відповідно можна організувати паралельні передачу та відображення будь-якої кількості двійково-десяткових цифр з одночасним їхнім висвітленням на індикаторах. Також можна передавати

двійково-десяткові цифри, що належать багаторозрядному числу, послідовно одна за іншою з їхнім висвітленням на одному індикаторі. Такий варіант системи більш економний, але потребує додаткового запам'ятовування відображуваних цифр.

Підвищення завадостійкості цифр, що передаються, відбувається завдяки використанню 6 надлишкових станів у двійково-десяткових числах. Захищеності інформації досягають за допомогою шифрувальних таблиць переходу від двійково-десяткових цифр до двійково-шістнадцяткових. У цих таблицях переходи подані у випадковому порядку, тому інформація в них набуває секретного вигляду. Всі ці питання необхідно комплексно розглянути в курсовій роботі.

2 СТРУКТУРНА СХЕМА СИСТЕМИ ПЕРЕДАЧІ ТА ВІДОБРАЖЕННЯ ДВІЙКОВО-ДЕСЯТКОВИХ ЧИСЕЛ

Структурна схема системи передачі та відображення однієї двійково-десяткової цифри наведена на рисунку 1.1.

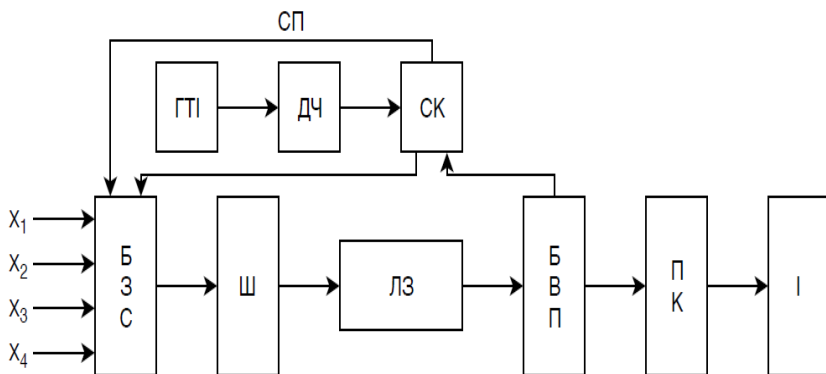


Рисунок 1.1 – Структурна схема системи передачі та відображення двійково-десяткових чисел

Її блоки поділяють на блоки сторони, що передає дані, й блоки сторони, що сприймає ці дані. Відповідно до наведених позначень зазначена система містить на стороні, що передає дані, генератор тактових імпульсів (ГТІ), дільник частоти (ДЧ), систему керування (СК), буферний запам'ятовувальний пристрій (БЗП), шифратор (Ш), лінію зв'язку (ЛЗ). На приймальній стороні містяться блок виправлення помилок (БВП), перетворювач кодів (ПК) чотирирозрядних кодових комбінацій в семирозрядні кодові комбінації, індикатор (І).

Система функціонує так. На вхід буферної запам'ятовувальної схеми (БЗС) надходить і запам'ятовується двійково-десятькова цифра X_1, X_2, X_3, X_4 із 4 розрядів. Вона відповідає десятковій цифрі, що повинна бути переданою ЛЗ і відображеною на індикаторі. Після подання на БЗС зі СК стробувального імпульсу двійково-десятькова цифра з БЗС передається на шифратор, у якому вона перетворюється на вибрану відповідно до шифрувальної таблиці двійково-десятькову цифру (0, 1, ..., 9). Ці цифри в шифрувальних таблицях вибрані випадково, способом довільного переставлення 10 вихідних цифр із 16. З них кожному студенту запропонована лише одна таблиця переходів, що є шифром та одночасно його ключем.

Щоб розшифрувати цей шифр без знання таблиці переходів, потрібно, якщо прагнути максимуму, перевірити всі можливі таблиці переходів. Для комп'ютера це нескладне завдання, але в разі конфіденційної інформації цей шифр може захищати її досить надійно, особливо якщо вона швидко старіє. Цьому допомагає відсутність чіткого тесту для виявлення під час перебирання можливих таблиць переходів зашифрованої цифру, тому що цифрові дані зазвичай не мають очевидного смислового навантаження.

Таким способом чотирирозрядне кодове слово з БЗС у незмінному вигляді подається на шифратор, що перетворює його на чотирирозрядне двійково-десятькове слово, яке не обов'язково відповідатиме вхідному. Після цього одержане

на виході шифратора слово подається на лінію зв'язку, у якій воно може спотворюватися внаслідок дії завад, перетворюючись на інше чотирирозрядне слово.

Якщо це спотворене чотирирозрядне слово за своїм числовим значенням не належатиме до слів, зазначених у таблиці шифру, воно буде визначене як заборонене блоком виправлення помилок. Відповідно цей блок подає сигнал у схему керування про те, що виникла помилка. Зі схеми керування надходить сигнал на БЗС і відбувається повторне надсилання вхідної кодової комбінації. Якщо знову виникає помилка, то знову відбувається надсилання вхідної кодової комбінації. Це може відбуватися до трьох повторів передавання кодової комбінації ЛЗ. Якщо після трьох повторів знову з'являється помилка, то СК виробляє сигнал «Аварія», й система зупиняється.

Після того як БВП ідентифікував сигнал як правильний, він за сигналом із СК надсилає його на перетворювач кодів (ПК), що замість чотирьох двійкових розрядів виробляє сім, які потім надходять на індикатор, щоб висвітити відповідні сегменти.

Подання на БЗС символів для індикації відбувається із заданою частотою, тобто вони періодично (через секунду) змінюються. Проте сам генератор тактових імпульсів генерує імпульси з частотою, заданою студенту у відповідному варіанті, наприклад 100 Гц. Тому потрібний дільник частоти, що змінює частоту сигналу, поданого з ГПІ, – до 1 Гц.

3 ТЕХНІЧНЕ ЗАВДАННЯ

У цій роботі потрібно:

1) перетворити відповідно до заданого варіанта двійково-десяткові цифри на їхні зображення на індикаторі. На приклад, відображувані десяткові цифри можуть мати форму, зазначену на рисунку 3.1.



Рисунок 3.1 – Приклад форм відображуваних десяткових цифр

Проте студент може запропонувати своє зображення десяткових цифр. Заданий варіант наведений у таблиці 3.1;

2) провести логічний синтез шифратора з подальшою його мінімізацією;

3) розробити схему захисту від помилок завдяки надлишковій інформації, що містить у собі двійково-десяткові цифри;

4) спроектувати дільник частоти із заданими в таблиці 3.2 тригерами, що ділить вихідну частоту ГТІ до 1 Гц;

5) провести логічний синтез пристрою, що виявляє помилки у двійково-десяткових цифрах, переданих ЛЗ із подальшою його мінімізацією;

6) синтезувати за схемою «дешифратор – шифратор» комбінаційний перетворювач двійково-десяткових цифр на зображення десяткових цифр для індикації на семисегментному індикаторі.

Таблиця 3.1 – Варіанти шифрувальних таблиць

Варіант 1			Варіант 2			Варіант 3		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0011	0	0000	0000	0	0000	0000
1	0001	0101	1	0001	0001	1	0001	0001
2	0010	0000	2	0010	0010	2	0010	0010
3	0011	1000	3	0011	0011	3	0011	0011
4	0100	0110	4	0100	0100	4	0100	0111
5	0101	0010	5	0101	1001	5	0101	1000
6	0110	0010	6	0110	1000	6	0110	1001
7	0111	0001	7	0111	0101	7	0111	0100
8	1000	1001	8	1000	0110	8	1000	0110
9	1001	0111	9	1001	0111	9	1001	0101
Варіант 4			Варіант 5			Варіант 6		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0011	0	0000	0000	0	0000	0000
1	0001	0100	1	0001	0001	1	0001	0001
2	0010	0101	2	0010	0111	2	0010	0110
3	0011	0110	3	0011	0010	3	0011	0111
4	0100	0111	4	0100	0101	4	0100	0010
5	0101	1000	5	0101	1000	5	0101	0100
6	0110	1001	6	0110	1001	6	0110	0101
7	0111	0010	7	0111	0011	7	0111	1000
8	1000	0001	8	1000	0100	8	1000	1001
9	1001	0000	9	1001	0110	9	1001	0011

Продовження таблиці 3.1

Варіант 7			Варіант 8			Варіант 9		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0000	0	0000	0001	0	0000	0000
1	0001	0001	1	0001	0000	1	0001	0001
2	0010	0010	2	0010	0011	2	0010	0100
3	0011	0011	3	0011	0010	3	0011	0101
4	0100	0100	4	0100	1000	4	0100	0111
5	0101	1000	5	0101	0101	5	0101	1000
6	0110	1001	6	0110	0110	6	0110	1001
7	0111	0110	7	0111	0111	7	0111	0010
8	1000	0111	8	1000	1001	8	1000	0110
9	1001	0101	9	1001	0100	9	1001	0011
Варіант 10			Варіант 11			Варіант 12		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0110	0	0000	0101	0	0000	0011
1	0001	0111	1	0001	0000	1	0001	0101
2	0010	1000	2	0010	0110	2	0010	1000
3	0011	1001	3	0011	0001	3	0011	1100
4	0100	1010	4	0100	0111	4	0100	1111
5	0101	1011	5	0101	0010	5	0101	0010
6	0110	1100	6	0110	1000	6	0110	0100
7	0111	1101	7	0111	1001	7	0111	0110
8	1000	1110	8	1000	1101	8	1000	1110
9	1001	1111	9	1001	1111	9	1001	0000

Продовження таблиці 3.1

Варіант 13			Варіант 14			Варіант 15		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0110	0	0000	1111	0	0000	0101
1	0001	0011	1	0001	0000	1	0001	0011
2	0010	1001	2	0010	1000	2	0010	0000
3	0011	1000	3	0011	1001	3	0011	0001
4	0100	1010	4	0100	1100	4	0100	0100
5	0101	1011	5	0101	0001	5	0101	1001
6	0110	1100	6	0110	0101	6	0110	1101
7	0111	1101	7	0111	0110	7	0111	1110
8	1000	1111	8	1000	1010	8	1000	1111
9	1001	1110	9	1001	0011	9	1001	0110
Варіант 16			Варіант 17			Варіант 18		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0100	0	0000	1011	0	0000	0110
1	0001	0011	1	0001	0100	1	0001	1001
2	0010	0101	2	0010	1000	2	0010	0100
3	0011	0111	3	0011	1010	3	0011	0001
4	0100	1100	4	0100	1001	4	0100	1010
5	0101	1101	5	0101	0001	5	0101	1101
6	0110	1010	6	0110	0010	6	0110	1011
7	0111	1000	7	0111	0011	7	0111	0111
8	1000	0000	8	1000	0110	8	1000	0101
9	1001	0001	9	1001	0111	9	1001	0000

Продовження таблиці 3.1

Варіант 19			Варіант 20			Варіант 21		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	1111	0	0000	0101	0	0000	0011
1	0001	0111	1	0001	1111	1	0001	0111
2	0010	1000	2	0010	0110	2	0010	1001
3	0011	1001	3	0011	1000	3	0011	1100
4	0100	1010	4	0100	0111	4	0100	1111
5	0101	1011	5	0101	0010	5	0101	0011
6	0110	1101	6	0110	0001	6	0110	0101
7	0111	1100	7	0111	1001	7	0111	0110
8	1000	1110	8	1000	1101	8	1000	1110
9	1001	0000	9	1001	0000	9	1001	0000
Варіант 22			Варіант 23			Варіант 24		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0111	0	0000	1101	0	0000	1111
1	0001	0110	1	0001	0000	1	0001	0101
2	0010	1001	2	0010	1110	2	0010	1000
3	0011	1000	3	0011	0001	3	0011	1100
4	0100	1010	4	0100	0111	4	0100	0011
5	0101	0001	5	0101	0010	5	0101	0010
6	0110	1100	6	0110	1000	6	0110	0111
7	0111	1101	7	0111	1001	7	0111	0110
8	1000	1110	8	1000	1101	8	1000	1110
9	1001	1111	9	1001	1111	9	1001	0000

Продовження таблиці 3.1

Варіант 25			Варіант 26			Варіант 27		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	1101	0	0000	0111	0	0000	1000
1	0001	0111	1	0001	0000	1	0001	0101
2	0010	1011	2	0010	0110	2	0010	0011
3	0011	1001	3	0011	0001	3	0011	1100
4	0100	1010	4	0100	0101	4	0100	0000
5	0101	1000	5	0101	1000	5	0101	0010
6	0110	1100	6	0110	0010	6	0110	0100
7	0111	0110	7	0111	1001	7	0111	0110
8	1000	1111	8	1000	1101	8	1000	1110
9	1001	1110	9	1001	1111	9	1001	1111
Варіант 28			Варіант 29			Варіант 30		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0000	0	0000	1000	0	0000	0000
1	0001	0001	1	0001	0001	1	0001	0001
2	0010	0010	2	0010	0110	2	0010	0010
3	0011	0011	3	0011	0000	3	0011	0011
4	0100	1000	4	0100	0111	4	0100	0100
5	0101	1001	5	0101	1111	5	0101	1000
6	0110	1010	6	0110	0101	6	0110	1001
7	0111	1011	7	0111	1001	7	0111	1010
8	1000	1110	8	1000	1101	8	1000	1011
9	1001	1111	9	1001	0010	9	1001	1100

Продовження таблиці 3.1

Варіант 31			Варіант 32			Варіант 33		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	1001	0	0000	0000	0	0000	1111
1	0001	1111	1	0001	0001	1	0001	0000
2	0010	1000	2	0010	0010	2	0010	1000
3	0011	0110	3	0011	0011	3	0011	1001
4	0100	1010	4	0100	1010	4	0100	1100
5	0101	1011	5	0101	1000	5	0101	0001
6	0110	1100	6	0110	1001	6	0110	1001
7	0111	1101	7	0111	1101	7	0111	1011
8	1000	1110	8	1000	1111	8	1000	1101
9	1001	0111	9	1001	0110	9	1001	1110
Варіант 34			Варіант 35			Варіант 36		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0000	0	0000	0001	0	0000	0000
1	0001	0001	1	0001	0011	1	0001	0001
2	0010	0010	2	0010	0100	2	0010	0010
3	0011	0011	3	0011	0101	3	0011	0011
4	0100	0010	4	0100	0110	4	0100	0100
5	0101	1111	5	0101	0111	5	0101	0101
6	0110	0100	6	0110	1000	6	0110	0110
7	0111	1000	7	0111	1001	7	0111	1000
8	1000	1001	8	1000	1010	8	1000	1001
9	1001	1100	9	1001	1011	9	1001	1010

Продовження таблиці 3.1

Варіант 37			Варіант 38			Варіант 39		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0010	0	0000	0000	0	0000	0000
1	0001	0011	1	0001	0001	1	0001	0001
2	0010	0100	2	0010	0100	2	0010	0010
3	0011	0111	3	0011	0101	3	0011	0011
4	0100	1000	4	0100	0111	4	0100	0100
5	0101	1001	5	0101	1001	5	0101	0101
6	0110	1010	6	0110	1010	6	0110	1000
7	0111	1011	7	0111	1011	7	0111	1001
8	1000	1110	8	1000	1110	8	1000	1110
9	1001	1111	9	1001	1111	9	1001	1111
Варіант 40			Варіант 41			Варіант 42		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0000	0	0000	0011	0	0000	0000
1	0001	0001	1	0001	1101	1	0001	0001
2	0010	0010	2	0010	0000	2	0010	0010
3	0011	0011	3	0011	1000	3	0011	0011
4	0100	0100	4	0100	0110	4	0100	0100
5	0101	1000	5	0101	0010	5	0101	1011
6	0110	1001	6	0110	1010	6	0110	1100
7	0111	1010	7	0111	0001	7	0111	1101
8	1000	1011	8	1000	1001	8	1000	1110
9	1001	1110	9	1001	1100	9	1001	1111

Продовження таблиці 3.1

Варіант 43			Варіант 44			Варіант 45		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0000	0	0000	0011	0	0000	0000
1	0001	0001	1	0001	1101	1	0001	0001
2	0010	0010	2	0010	0000	2	0010	0010
3	0011	0011	3	0011	1000	3	0011	0011
4	0100	0100	4	0100	0110	4	0100	0111
5	0101	1011	5	0101	0010	5	0101	1000
6	0110	1100	6	0110	1010	6	0110	1001
7	0111	1101	7	0111	0001	7	0111	1010
8	1000	1110	8	1000	1001	8	1000	1011
9	1001	0101	9	1001	0101	9	1001	0101
Варіант 46			Варіант 47			Варіант 48		
№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄	№ пор.	x ₁ x ₂ x ₃ x ₄	f ₁ f ₂ f ₃ f ₄
0	0000	0000	0	0000	0000	0	0000	1000
1	0001	0001	1	0001	0001	1	0001	0000
2	0010	0110	2	0010	0010	2	0010	1111
3	0011	0111	3	0011	0011	3	0011	1001
4	0100	1010	4	0100	0100	4	0100	1100
5	0101	0100	5	0101	1000	5	0101	0101
6	0110	0101	6	0110	1001	6	0110	0101
7	0111	1000	7	0111	1010	7	0111	0110
8	1000	1001	8	1000	1011	8	1000	1110
9	1001	1110	9	1001	1100	9	1001	0011

Таблиця 3.2 – Варіанти тригерів лічильника

Варіант №	Тип тригера	Варіант №	Тип тригера	Варіант №	Тип тригера
1	T	21	T	41	T
2	D	22	D	42	D
3	DV	23	DV	43	DV
4	JK	24	JK	44	JK
5	T	25	T	45	T
6	D	26	D	46	D
7	DV	27	DV	47	DV
8	JK	28	JK	48	JK
9	T	29	T		
10	D	30	D		
11	DV	31	DV		
12	JK	32	JK		
13	T	33	T		
14	D	34	D		
15	DV	35	DV		
16	JK	36	JK		
17	T	37	T		
18	D	38	D		
19	DV	39	DV		
20	JK	40	JK		

4 ВАРІАНТИ ЗАВДАНЬ

Для кожного студента передбачений свій варіант курсової роботи, якого потрібно дотримуватися. Варіант завдання вибирають за номером прізвища студента в обліковому складі групи. Наприклад, якщо ваше прізвище за обліковим списком під номером 3, ваш варіант третій.

Варіанти роботи мають такі особливості:

1) генератор тактової частоти для кожного варіанта генерує частоту, що дорівнює номеру варіанта, помноженому на частоту 10 Гц. Отже, якщо, наприклад,

номер варіанта дорівнює 10, то тактова частота генератора тактових імпульсів – 100 Гц;

2) для кожного варіанта потрібно використати свою шифрувальну таблицю, а для побудови ДЧ відповідно до таблиці 3.2 – свій тип тригерів;

3) необхідно вибрати семисегментний індикатор.

5 ВИКОНАННЯ ЗАВДАННЯ

У таблиці 3.1 із лівого боку кожного варіанта зазначені десяткові коди символів від 0 до 9 для передавання й подальшого відображення на індикаторі у формі зображень кожного символа. Ці зображення студенти повинні самостійно накреслити так, як вважають за потрібне, але водночас вони повинні чітко відрізнитися одне від одного й бути добре розпізнаваними. Якщо в студента виникають труднощі із зображеннями, йому варто звернутися до викладача за допомогою. Крім того, викладач повинен затвердити накреслені студентом символи.

Із правого боку таблиці 3.1 наведені двійково-десяткові кодові комбінації, що будуть одержані після шифрування десяткових цифр.

Так, наприклад, якщо у варіанті 3 зазначено, що число 4 перетворюється на кодову комбінацію 0111, це означає, що двійково-десятькове число 0100 перетворюється за комбінаційною схемою на число 0111. Унаслідок цього відбувається шифрування цифри 0100 за допомогою її замінювання на цифру 0111. У такій формі вона повинна бути переданою каналом зв'язку на схему відображення інформації.

Отже, студент повинен синтезувати за допомогою логічних функцій перетворювач вхідних двійково-десяткових чисел на відповідні їм двійково-десяткові числа, взяті з шифрувальної таблиці, мінімізувати в ньому кількість необхідних апаратурних витрат за допомогою діаграми Вейча або методу Квайна і підрахувати кількість входів у логічних схемах. Також потрібно оцінити затримку сигналів

у комбінаційній схемі перетворювача та відповідно до цього визначити швидкодію схеми.

У проєктованій системі, крім вищезазначеного шифрувального перетворювача, буде стояти ще один перетворювач кодів, що функціонує за схемою «дешифратор – шифратор», для формування семирозрядних кодових комбінацій. Він спочатку перетворює двійково-десятькову цифру, подану на його вхід, на сигнал на одному з виходів дешифратора, а далі цей сигнал перетворюється за допомогою шифратора на кодову комбінацію із 7 розрядів, що відповідають сегментам підсвічуваного індикатора.

Наприклад, дешифратор спочатку перетворює цифру 0111 на сигнал на одному зі своїх виходів, що потім передається на шифратор, у якому перетворюється на 7 двійкових розрядів кодового слова. У кожному його розряді стоять значення 1 або 0, що в сукупності відповідають формі попередньозображеної студентом десяткової цифри. У реальному пристрої відповідно до значень 1 будуть підсвічуватися сегменти індикатора.

Лічильники для ДЧ будують на тригерах, заданих для кожного варіанта в таблиці 3.2.

Так, варіант 4 ґрунтується на JK-тригерах. Структура лічильника повинна бути з наскрізним переносом. До неї потрібно навести часову діаграму функціонування лічильника, а також опис роботи використовуваних тригерів із таблицею переходів і часовою діаграмою. Необхідно обчислити час перехідних процесів у них та оцінити кількість витрат апаратури за методикою Квайна – підрахувати входи логічних схем тригерів.

Після того як будуть синтезовані й за необхідності мінімізовані схеми окремих блоків розроблюваної системи, потрібно накреслити функціональні схеми цих блоків відповідно до стандарту.

Генератор тактових імпульсів варто взяти у формі стандартної схеми з довідника й розрахувати його елементи так, щоб одержати на виході частоту свого варіанта, наприклад 100 Гц.

Потім функціонування кожного накресленого блоку необхідно докладно описати словами, як і часові діаграми лічильників. Водночас потрібно зазначити, куди та як ідуть сигнали, де вони затримуються і на який час. Також варто навести основні характеристики індикатора: напругу живлення, споживану потужність, розмір і величину яскравості сегментів.

На початку роботи пишуть вступ, у якому досліджують джерела літератури, у яких описані системи, подібні до розроблюваної, зазначаючи сфери їхнього застосування. Наводять посилання на літературу, проводять загальне класифікування. Потім ставлять завдання курсової роботи. Як вступ і завдання можна використати матеріал методичних вказівок або запропонувати своє бачення його.

У кінці роботи потрібно написати висновок, у якому зазначити основні параметри розробленої системи, її переваги, порівнюючи із схожими системами, рекомендовані сфери її застосування.

Зразки синтезу перетворювача кодів за допомогою логічних функцій і за схемою «дешифратор-шифратор» і пристрою виявлення помилок наведені в додатках Д, Е та Ж.

СПИСОК ЛІТЕРАТУРИ

Основна

1. Борисенко О. А. Цифрова схемотехніка : підручник / О. А. Борисенко. – Суми : Сумський державний університет, 2016. – 200 с.
2. Борисенко О. А. Дискретна математика : підручник / О. А. Борисенко. – Суми : Університетська книга, 2019. – 255 с.
3. Рябенький В. М. Цифрова схемотехніка : навч. посіб. / В. М. Рябенький, В. Я. Жуйков, В. Д. Гулий. – Львів : Новий Світ, 2019. – 736 с.

Додаткова

1. Точи Рональд Дж. Цифровые системы. Теория и практика / Дж. Рональд Точи, Нил С. Уидмер ; пер. с англ. – 8-е изд. – Москва : Вильямс, 2004. – 1024 с.
2. Угрюмов Е. П. Цифровая схемотехника : учеб. пособие / Е. П. Угрюмов. – 3-е изд. – Санкт-Петербург : БХВ – Петербург, 2010. – 816 с.
3. Хаханов В. И. Проектирование и верификация цифровых схем на кристаллах. Verilog & System Verilog / В. И. Хаханов, И. В. Хаханова, Е. И. Литвинова, О. А. Гузь. – Харьков : ХНУРЭ, 2010. – 528 с.

ДОДАТОК А
(обов'язковий)

Зразок оформлення титульного аркуша курсової роботи

Міністерство освіти і науки України
Сумський державний університет
Кафедра електроніки і комп'ютерної техніки

Курсова робота
із дисципліни «Пристрої цифрової електроніки»

на тему **«СИСТЕМА ПЕРЕДАЧІ ТА ВІДОБРАЖЕННЯ
ДВІЙКОВО-ДЕСЯТКОВИХ ЧИСЕЛ»**

Студента гр. _____

П. І. П/б

Перевірів викладач, вчене звання

П. І. П/б

Суми
2020

ДОДАТОК Б
(обов'язковий)

Зразок оформлення рецензії на курсову роботу

ВИМОГА ДО РОБОТИ	ВІДПОВІДНІСТЬ ВИМОГАМ		
	Достат.	Частк.	Незад.
1. Відповідність роботи варіанта завдання	ТАК		
2. Повнота виконання завдання		ТАК	
Якість оформлення			
1. Наявність змісту, вступу, основної частини, висновків, списку літератури		ТАК	
2. Правильність оформлення			ТАК
3. Наявність у тексті посилань на літературні джерела		ТАК	
4. Правильність оформлення програми (якщо є)		ТАК	
5. Достатність обсягу основної частини (сторінок)	ТАК		
Якість розрахунків			
1. Правильність розрахунків			ТАК
2. Наявність пояснень до використаних формул		ТАК	
Якість аналізу			
1. Наявність коментарів до розрахунків			ТАК
2. Наявність висновків і рекомендацій	ТАК		
ЗАГАЛЬНІ ВИСНОВКИ ЩОДО РОБОТИ			
1. Робота допущена до захисту			
2. Робота допущена до захисту з доопрацюванням зауважень у рецензії			ТАК

Рецензент _____
дата

_____ підпис

ДОДАТОК В
(обов'язковий)

Зразок змісту курсової роботи

ЗМІСТ

ВСТУП.....	1
1. НАЗВА ПЕРШОГО РОЗДІЛУ.....	4
1.1. Назва підрозділу	4
1.2.	8
2. НАЗВА ДРУГОГО РОЗДІЛУ	12
2.1. Назва підрозділу	12
2.2.	16
3. НАЗВА ТРЕТЬОГО РОЗДІЛУ	20
3.1. Назва підрозділу	20
3.2.	24
ВИСНОВКИ	25
СПИСОК ЛІТЕРАТУРИ.....	26
ДОДАТКИ (якщо вони є).....	27

ДОДАТОК Д
(обов'язковий)

**Зразок логічного синтезу перетворювача кодів
за допомогою логічних функцій**

1 Шифрувальна таблиця

Для побудови перетворювача кодів використовується таблиця Д.1. Вона складається в першій колонці в разі рахунку зліва направо із номерів 10 цифр, пронумерованих від 0 до 9, яким у другій колонці відповідно надані чотирирозрядні двійково-десяткові набори значень змінних $X_1X_2X_3X_4$. У третій колонці цим наборам відповідають чотирирозрядні двійково-шістнадцятирічні набори значень 4 логічних функцій $f_1f_2f_3f_4$. Ці набори є двійково-шістнадцятирічними, тому що кодують 4 двійковими розрядами цифри шістнадцятирічної системи числення. Вони також пронумеровані десятковими номерами в четвертій колонці справа таблиці Д.1.

Таблиця Д.1 – Шифрувальна таблиця

№ 10	$X_1X_2X_3X_4$	$f_1 f_2 f_3 f_4$	№ 16
0	0000	1111	15
1	0001	0000	0
2	0010	1000	8
3	0011	1001	9
4	0100	1100	10
5	0101	0001	1
6	0110	0101	5
7	0111	0110	6
8	1000	1010	10
9	1001	0011	3

Значення логічних функцій $f_1 f_2 f_3 f_4$ отримані випадково, тому в таблиці Д.1 вони шифрують вихідну інформацію, захищаючи її від несанкціонованого доступу. Таблиця Д.1 відповідно буде називатися в подальшому

шифрувальною, а 4-розрядні двійково-шістнадцятирічні комбінації в 3 колонці в разі рахунку зліва шифрами 4-розрядних двійково-десяткових наборів.

2 Синтез логічних функцій

Відповідно до вибраної шифрувальної таблиці Д.1 отримаємо спочатку досконалу диз'юнктивну нормальну форму (ДДНФ) для кожної з 4 логічних функцій $f_1 f_2 f_3 f_4$. Кожна з них реалізується так. Випикується для кожної 1 вибраної з таблиці Д.1 функції $f_1 f_2 f_3 f_4$ кон'юнкції для 4 змінних $X_1 X_2 X_3 X_4$. Таких кон'юнкцій для функції f_1 буде 5, для f_2 – 4, для f_3 – 4, для f_4 – 5. Там де в наборах стоять 0, над відповідними змінними кон'юнкцій ставляться інверсії. Отримані таким способом кон'юнкції з інверсіями об'єднуються знаками диз'юнкції. Відповідно до наведеного алгоритму в таблиці Д.1 подані в ДДНФ логічні функції мають такий вигляд:

$$f_1 = \bar{X}_1 \bar{X}_2 \bar{X}_3 \bar{X}_4 + \bar{X}_1 \bar{X}_2 X_3 \bar{X}_4 + \bar{X}_1 \bar{X}_2 X_3 X_4 + \bar{X}_1 X_2 \bar{X}_3 \bar{X}_4 + X_1 \bar{X}_2 \bar{X}_3 \bar{X}_4;$$

$$f_2 = \bar{X}_1 \bar{X}_2 \bar{X}_3 \bar{X}_4 + \bar{X}_1 X_2 \bar{X}_3 \bar{X}_4 + \bar{X}_1 X_2 X_3 \bar{X}_4 + \bar{X}_1 X_2 X_3 X_4;$$

$$f_3 = \bar{X}_1 \bar{X}_2 \bar{X}_3 \bar{X}_4 + \bar{X}_1 X_2 X_3 X_4 + X_1 \bar{X}_2 \bar{X}_3 \bar{X}_4 + X_1 \bar{X}_2 \bar{X}_3 X_4;$$

$$f_4 = \bar{X}_1 \bar{X}_2 \bar{X}_3 \bar{X}_4 + \bar{X}_1 \bar{X}_2 X_3 X_4 + \bar{X}_1 X_2 \bar{X}_3 X_4 + \bar{X}_1 X_2 X_3 \bar{X}_4 + X_1 \bar{X}_2 \bar{X}_3 X_4.$$

3 Схемна реалізація логічних функцій у ДДНФ

Реалізація логічних функцій у ДДНФ відбувається на базі логічних схем *I* та *АБО*. Схеми *I* дорівнює 1 тоді й тільки тоді, коли на всіх її входах будуть подані 1, а схеми *АБО*, коли на хоча б один її вхід буде подана 1. Тому, щоб отримати на виході схеми *I* 1 потрібно, щоб на її входах там де змінна дорівнює 0, стояв інвертор, який перетворе 0 на 1. Унаслідок цього, якщо, наприклад, перетворюється комбінація 0000, то на всіх входах відповідної кон'юнкції повинні стояти інверсії, які позначаються на її входах кружечками.

Виходи усіх схем I кожної функції об'єднуються схемами АБО. На їхніх виходах будуть отримані двійково-шістнадцятирічні значення, які відповідають значенням двійково-десяткових цифр. Тобто ці логічні схеми в сукупності реалізують перетворювач кодів із вихідних двійково-десяткових цифр у двійково-шістнадцятирічні (рис. Д.1).

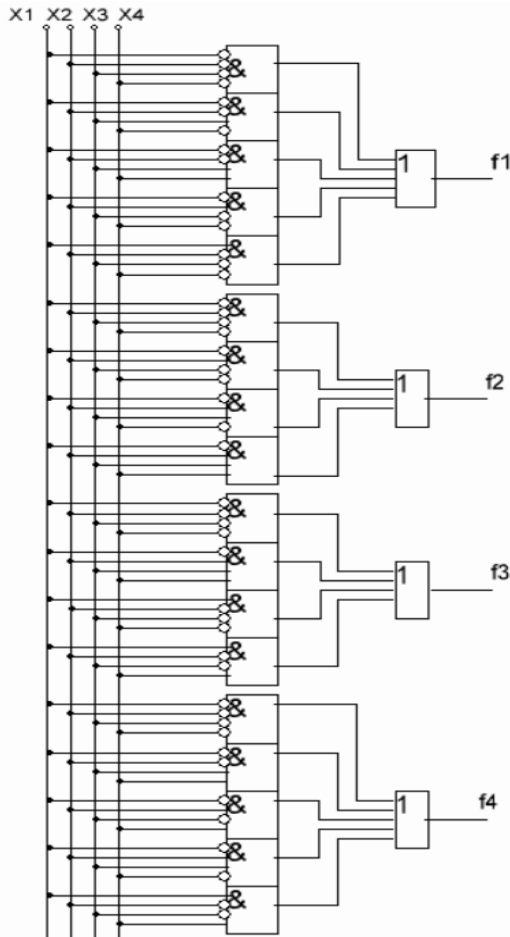


Рисунок Д.1 – Функціональна схема перетворювача кодів

Апаратурні витрати на реалізацію цієї схеми підраховуються за допомогою підсумовування кількості входів, які необхідно витратити для реалізації інверторів, схем I та $АБО$. Для цього в логічних функціях спочатку підраховується кількість схем I та інверторів. Їхня кількість буде дорівнювати відповідно 18 та 47. Кожна схема I має 4 входи, а інвертор – 1.

Відповідно кількість входів для реалізації схем I та інверторів буде дорівнювати $18 \cdot 4 + 47 \cdot 1 = 72 + 47 = 119$. До них треба додати 18 входів, які йдуть на реалізацію схем $АБО$. Тому загальна кількість входів, а отже, й кількість апаратурних витрат у перетворювача кодів, що розглядається, $119 + 18 = 137$.

4 Реалізація перетворювача кодів за допомогою програмно-логічної матриці

Отримані вище схеми логічних функцій f_1, f_2, f_3, f_4 можна реалізувати у вигляді поданої на рисунку Д.2 програмної логічної матриці (ПЛМ). У цій матриці 4 горизонтальним лініям зверху відповідають змінні X_1, X_2, X_3, X_4 , а лініям, що йдуть під кожною з них, їхні інверсії. Цих ліній, очевидно, буде 4. Кружки, на верхній лінії до яких приєднується нижня лінія, означають, що вона реалізує інверсії змінних, пов'язані з верхньою лінією. Вертикальні лінії, що перетинають горизонтальні лінії, кількість яких у випадку, що розглядається, дорівнює 8, зображують кон'юнкції $\&1$ $\&2$, ..., $\&18$. Щоб їх реалізувати, на місті перетину горизонтальної лінії з вертикальною за наявності відповідної змінної з інверсією чи без у відповідній кон'юнкції ставиться точка. Якщо вона ставиться на верхній лінії, то контактує змінна без інверсії. В іншому разі – змінна з інверсією. Цих точок буде стільки, скільки є змінних у кон'юнкції, тобто 4. Унаслідок цього на кожній вертикальній лінії реалізується одна із схем I , яка входить у логічну функцію. Їх буде стільки, скільки кон'юнкцій входить у ту чи іншу логічну функцію, тобто або 5, або 4.

Нижче в ПЛМ під всіма 8 горизонтальними лініями, які реалізують схеми *I*, розташовані 4 горизонтальні лінії, які реалізують схеми *АБО*.

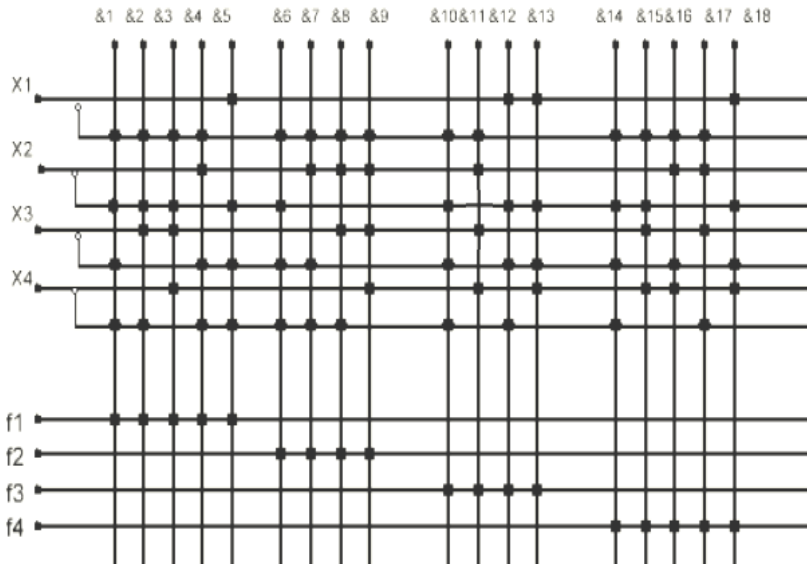


Рисунок Д.2 – ПЛМ логічних функцій

Їх перетинають ті ж самі вертикальні лінії, які перетинали горизонтальні лінії всіх 4 змінних і їхніх інверсій, однак на них тепер подані контакти виходів схем *I* зі входами схем *АБО* для кожної логічної функції у вигляді точок. Це означає, що кожна горизонтальна нижня лінія зображає собою схему *АБО*.

Тепер залишилося підрахувати кількість входів, які йдуть на реалізацію ПЛМ загалом. Для цього спочатку підраховується кількість точок на верхніх горизонтальних лініях без інверсії, а потім на нижніх інверсних лініях під ними. Остання кількість точок помножується на 2, тому що, крім входів схем *I*, ще треба враховувати один вхід інвертора перед кожним із них. Усього точок на верхніх горизонтальних лініях дорівнює $25 + 47 \cdot 2 = 25 + 94 = 119$. Додавши точок на нижніх горизонтальних лініях $f_1 f_2 f_3 f_4$,

кількість яких дорівнює 18, отримаємо загальну кількість точок, тобто $119 + 18 = 137$. Їхня кількість збігається з раніше підрахованою кількістю входів у схемі Д.1 перетворювача кодів на базі логічних схем *I* та *АБО*, а саме – 137.

5 Мінімізація логічних функцій

Отримані в ДДНФ логічні функції потрібно мінімізувати. Для цього будуються діаграми Вейча, кожна з яких має 16 клітинок, так, як це подано на рисунках Д.3 – Д.6. Спочатку в клітинки, які містяться на перехрещенні 4 змінних з інверсіями чи без, що відповідають тій чи іншій кон'юнкції, ставляться 1. Потім шукають клітинки, у яких стоять поряд 8 одиниць, 4 одиниці, 2 одиниці та об'єднують їх. Причому одна й та ж сама клітинка може розміщуватися в різних об'єднаннях. У першому випадку, коли об'єднуються 8 одиниць, шукають 1 змінну, яка їх покриває, коли 4 одиниці – 2 змінні, коли 2 одиниці – 3 змінні. Якщо клітинка з 1 не може об'єднатися з іншими клітинками з одиницями, то вона покривається 4 змінними, тобто мінімізація в цьому разі для відповідної кон'юнкції відсутня.

Так, для функції f_1 діаграма Вейча буде такою.

	X_2	\bar{X}_2	
\bar{X}_1		1	1
		1	1
X_1		1	
	X_4	\bar{X}_4	X_4

Рисунок Д.3 – Таблиця Вейча

Унаслідок мінімізації за допомогою діаграми Вейча логічна функція f_1 матиме більш стислий запис, ніж раніше

$$f_1 = \bar{X}_1 \bar{X}_3 \bar{X}_4 + \bar{X}_1 \bar{X}_2 X_3 + \bar{X}_2 \bar{X}_3 \bar{X}_4.$$

Для функції f_2 діаграма Вейча буде такою.

	X_2	\bar{X}_2		
\bar{X}_1	1	1		X_3
		1	1	\bar{X}_3
X_1				
				X_3
	X_4	\bar{X}_4		X_4

Рисунок Д.4 – Таблиця Вейча

Унаслідок мінімізації запис логічної функції f_2 також скорочується

$$f_2 = \bar{X}_1 X_2 X_3 + \bar{X}_1 \bar{X}_3 \bar{X}_4.$$

Для функції f_3 діаграма Вейча має такий вигляд

	X_2	\bar{X}_2		
\bar{X}_1	1			X_3
		1		\bar{X}_3
X_1		1	1	
				X_3
	X_4	\bar{X}_4		X_4

Рисунок Д.5 – Таблиця Вейча

Після мінімізації за допомогою діаграми Вейча логічна функція f_3 буде записана в такий спосіб:

$$f_3 = \bar{X}_1 X_2 X_3 X_4 + \bar{X}_2 \bar{X}_3 \bar{X}_4 + X_1 \bar{X}_2 \bar{X}_3 .$$

Діаграму Вейча для функції f_4 сформуємо так:

	X_2		\bar{X}_2		
\bar{X}_1		1		1	X_3
	1		1		\bar{X}_3
X_1				1	
					X_3
	X_4	\bar{X}_4		X_4	

Рисунок Д.6 – Таблиця Вейча

Оскільки немає можливості об'єднати клітинки, у яких є поряд хоча б 2 одиниці, запис логічної функції f_4 не скорочується і не змінюється

$$f_4 = \bar{X}_1 \bar{X}_2 \bar{X}_3 \bar{X}_4 + \bar{X}_1 \bar{X}_2 X_3 X_4 + \bar{X}_1 X_2 \bar{X}_3 X_4 + \bar{X}_1 X_2 X_3 \bar{X}_4 + X_1 \bar{X}_2 \bar{X}_3 X_4 .$$

6 Схемна реалізація мінімізованих логічних функцій

Мінімізовані логічні функції $f_1 f_2 f_3 f_4$ реалізуються схемно аналогічно з реалізацією немінімізованих функцій, яка розглянута вище. Схемна реалізація логічних функцій у мінімальних ДНФ наведена на рисунку Д.7.

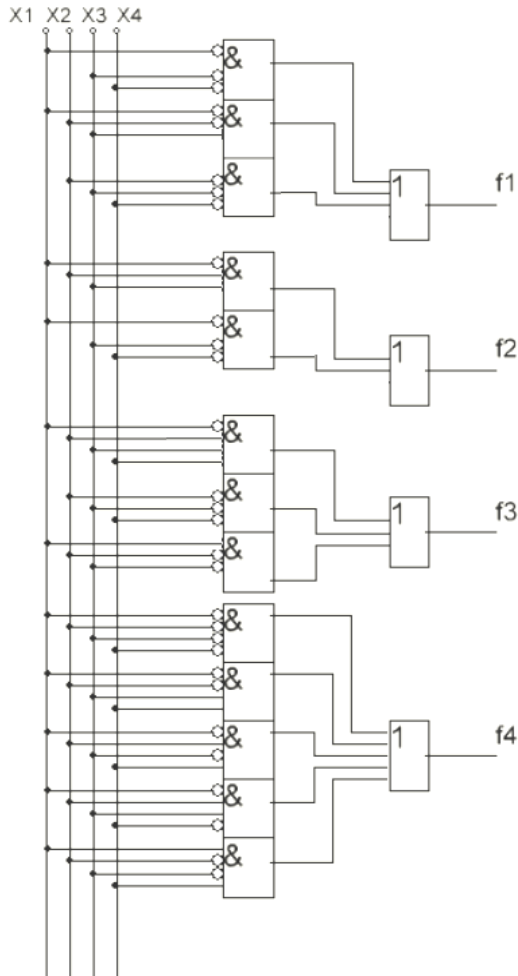


Рисунок Д.7 – Функціональна схема перетворювача кодів із мінімальними ДНФ

Розрахуємо апаратні витрати на реалізацію цієї схеми. Схема перетворювача кодів містить у собі 30 інверторів з 1 входом у кожному та 13 схем I , 7 з яких мають по 3 входи, а 6 – по 4 входи.

Кількість входів для реалізації схем I та інверторів буде дорівнювати $7 \cdot 3 + 6 \cdot 4 + 30 \cdot 1 = 21 + 24 + 30 = 75$.

Кількість входів для реалізації схем АБО дорівнює 13. Тому після проведеної мінімізації загальна кількість входів дорівнює $75 + 13 = 88$. Отже, проведена мінімізація дозволила зменшити апаратні витрати на реалізацію перетворювача кодів з 137 до 88, тобто в 1,56 рази.

7 Програмно-логічна матриця мінімальних ДНФ

Сформовані вище схеми мінімізованих логічних функцій $f_1 f_2 f_3 f_4$ можна реалізувати у вигляді ПЛМ. Вони формуються за аналогією з ПЛМ для ДДНФ на основі отриманих вище мінімальних логічних функцій.

Програмно-логічна матриця для мінімальних ДНФ наведена на рисунку Д.8.

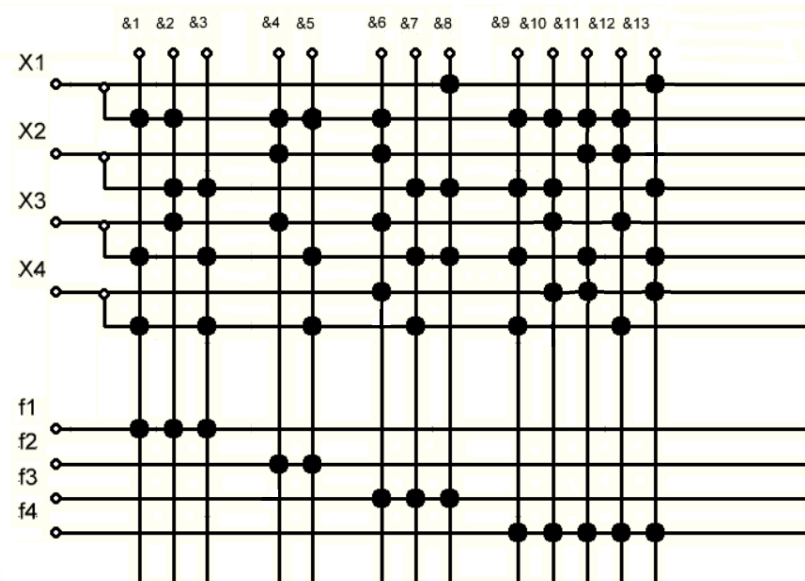


Рисунок Д.8 – ПЛМ для мінімальних ДНФ

Підрахуємо кількість входів у цій матриці відповідно до розглянутого вище правила, коли кількість точок у нижній горизонтальній лінії помножується на 2, а у відповідній верхній – на 1, та додається кількість точок на нижніх горизонтальних лініях для функцій $f_1 f_2 f_3 f_4$.

Відповідно маємо такий результат: $15 + 30 \cdot 2 + 13 = 88$, який збігається з раніше розрахованою кількістю входів у схемі перетворювача кодів із мінімальними ДНФ на базі логічних схем *I* та *АБО*, а саме – дорівнює 88.

Звернемо також увагу на те, що кожна логічна функція $f_1 f_2 f_3 f_4$ на рисунках Д.1 – Д.8 створює неповний лінійний дешифратор, а кожна схема *АБО*, яка об'єднує виходи цього дешифратора, – неповний шифратор з одним виходом. Сумісно виходи цих шифраторів створюють виходи перетворювача кодів, який синтезується. У ньому кожній двійково-десятковій комбінації на вході ставиться у відповідність двійково-шістнадцятирічна комбінація на виході. Аналогічно можна побудувати перетворювач кодів на будь-яке число різних входів і виходів.

ДОДАТОК Е (обов'язковий)

Зразок побудови перетворювача кодів за схемою «дешифратор-шифратор»

Блок перетворювача кодів за схемою «дешифратор-шифратор» повинен перетворити двійково-десяткові номери цифр у номери сегментів десятикових цифр, поданих на рисунку Е.1.



Рисунок Е.1 – Зображення десятикових цифр

Це перетворення відбувається відповідно до таблиці Е.1, яка складається із стовпчика десятикових цифр, стовпчика їхніх двійково-десятикових номерів, стовпчиків відповідних їм номерів дешифратора і шифратора та номерів сегментів.

Таблиця Е.1 – Відображення сегментів на індикаторі

Цифра	Двійково-десятковий номер	Номер виходу дешифратора	Номер входу шифратора	Стан сегмента 1234567
0	0000	15	126	1111110
1	0001	0	24	0011000
2	0010	8	55	0110111
3	0011	9	61	0111101
4	0100	12	89	1011001
5	0101	1	109	1101101
6	0110	5	111	1101111
7	0111	6	56	0111000
8	1000	10	127	1111111
9	1001	3	125	1111101

Блок-схема перетворювача кодів складається з неповного дешифратора, який має 4 входи і 10 виходів, та зв'язаного з ним неповного шифратора з 10 входами та 7 виходами. Їхні виходи та входи відповідними зв'язками з'єднуються між собою (рис. Е.2).

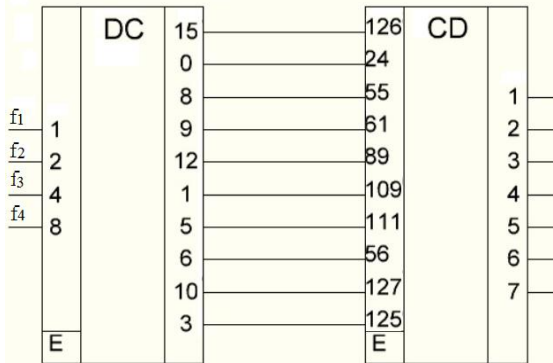


Рисунок Е.2 – Блок-схема перетворювача кодів

На входи дешифратора надходить із шифрувальної таблиці 4-розрядний шифр двійково-десятькової комбінації. Він шифрує одну з 10 цифр від 0 до 9, яку треба відобразити згідно із зображеннями, наведеними на рисунку Е.1. На відповідному цій цифрі виході дешифратора в цей час з'являється 1, а на всіх інших його виходах – 0.

Так, двійково-десятькова комбінація 0000 цифри «0» відповідно до таблиці шифрування варіанта, що розглядається, має шифр 1111, десятковим записом якої є 15, що й є номером виходу дешифратора. Аналогічно пронумеровані й усі інші 9 десяткових цифр. Вони подані у вигляді вихідних номерів дешифратора на рисунку Е.2.

Для відображення цифри 0 на 7-сегментному індикаторі шифратор повинен видати двійкову послідовність 1111110, яка в десятковому вигляді буде дорівнювати 126. Тому позначимо відповідний вхід шифратора як 126 та зв'яжемо його з виходом дешифратора. Аналогічно отримаємо й усі інші дев'ять номерів на виході дешифратора і входи шифратора та зв'яжемо їх між собою.

Номери входів шифратора в десятковому вигляді та 7-розрядні двійкові комбінації, які описують стан кожного сегмента 7-сегментного індикатора, наведені в таблиці Е.1 відповідно у 3 та 4 стовпчиках таблиці в разі рахунку зліва направо. Номери виходів дешифратора в десятковій формі надані у другому стовпчику.

Отримана 1 на виході 15 дешифратора передається лінією зв'язку на вхід 126 шифратора та перетворюється ним у 7-розрядну двійкову комбінацію, одиниці якої вказують на сегменти індикатора, які треба підсвітити. Унаслідок цього буде відображена десяткова цифра 0 і, відповідно, завдання відображення двійково-десяткової цифри, яка прийшла на вхід шифратора, буде розв'язана.

На наступному кроці розроблення перетворювача кодів потрібно відповідно до блок-схеми розробити його функціональну схему.

Ця схема (рис. Е.3) реалізує дешифратор і шифратор відповідно за допомогою схем *I* та схем *АБО*. Дешифратор реалізується на 10 схемах *I*, а шифратор на 7 схемах *АБО*.

Кожний вихід дешифратора *ДС*, номер якого наведений біля відповідного елемента *I*, з'єднується лінією з відповідним входом шифратора *СД*, номер якого наведений на його вході. У разі появи 1 на виході 15, що можливо за одночасної появи чотирьох логічних одиниць на першому елементі *I*, спрацює вхід 126 шифратора і 1 з нього йде на усі схеми *АБО* за винятком нижньої схеми *АБО*. Це означає, що на всіх виходах шифратора, за винятком нижнього, з'являться одиниці, які підсвітять відповідні сегменти. Унаслідок цього відобразиться цифра 0. Звернемо увагу, що кількість входів для кожної схеми *АБО* не може бути більше ніж 10, але їх може не бути зовсім.

Схема перетворювача кодів містить у собі 13 схем *I* з 4 входами в кожній, 22 інвертори з 1 входом у кожному та 7 схем *АБО* з кількістю входів 49. Загальна кількість входів, які характеризують апаратні витрати на реалізацію перетворювача кодів за схемою «дешифратор-шифратор», дорівнює 123.

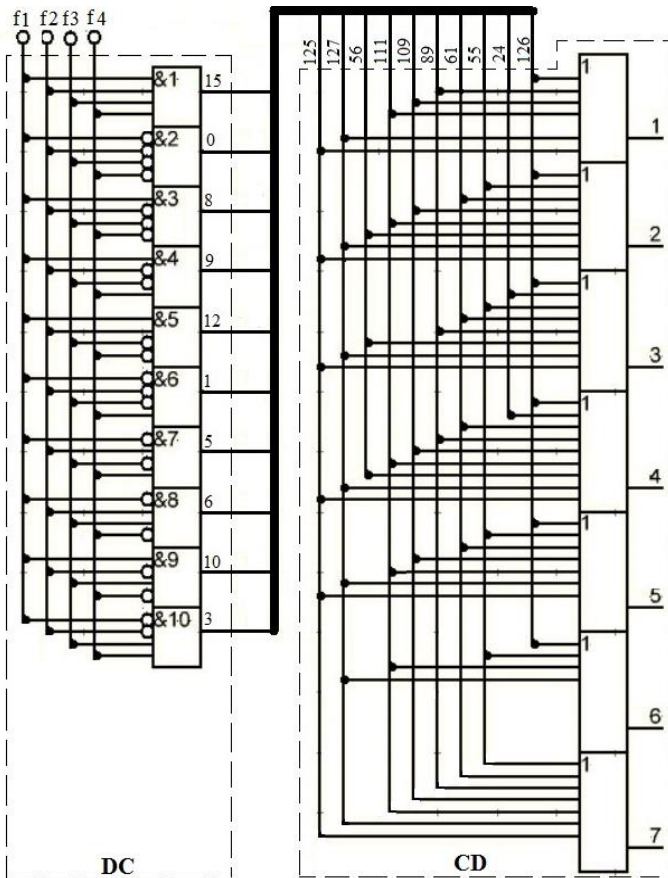


Рисунок Е.3 – Функціональна схема перетворювача кодів

Цю схему можна було б реалізувати через її синтез за допомогою логічних функцій та їхньої оптимізації. Але в цьому разі схема була б неоднорідною та технологічно більш складною для реалізації. Однак все ж її можна зробити логічним синтезом як із мінімізацією, так і без, та порівняти за кількістю апаратних витрат із схемою «дешифратор-шифратор». Рекомендуємо це зробити кожному студенту для свого варіанта синтезу перетворювача кодів, хоча це не обов'язково.

ДОДАТОК Ж (обов'язковий)

Зразок синтезу пристрою виявлення помилок

Блок виявлення помилок складається з неповного дешифратора на 10 заборонених комбінацій 0010 0100 0111 1011 1101 1110. Виходи дешифратора об'єднуються схемою АБО. Унаслідок цього в разі появи забороненої комбінації на виході цієї схеми з'являється сигнал помилки у вигляді логічної одиниці.

Схема блоку виявлення помилок наведена на рисунку Ж.1.

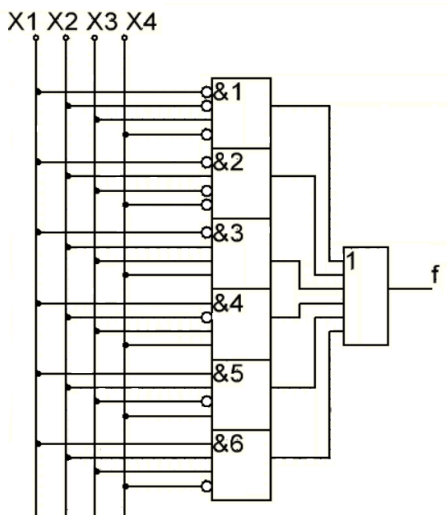


Рисунок Ж.1 – Схема блоку виявлення помилок

Схема складається із 6 схем *I* з 4 входами в кожній, 10 інверторів з 1 входом у кожному та 1 схеми АБО з 6 входами. Відповідно кількість входів для реалізації цієї схеми буде дорівнювати $24 + 10 + 6 = 40$.

Навчальне видання

4844 Методичні вказівки
до курсової роботи
**«Система передачі та відображення
двійково-десяткових чисел»**
із дисциплін **«Цифрова схемотехніка»,**
«Пристрої цифрової електроніки», «Схемотехніка»,
Схемотехніка телекомунікаційних систем
для студентів спеціальностей 171 *«Електроніка»*,
172 *«Телекомунікації та радіотехніка»*
і 153 *«Мікро- та наноелектроніка»*
всіх форм навчання

Відповідальний за випуск А. С. Опанасюк
Редактор І. О. Кругляк
Комп'ютерне верстання О. А. Борисенка

Підписано до друку 15.06.2020, поз.
Формат 60x84/16. Ум. друк. арк. 1,63 Обл.-вид. арк. 1,54. Тираж 5 прим. Зам. N
Собівартість видання грн к.

Видавець і виготовлювач
Сумський державний університет,
вул. Римського-Корсакова, 2, м. Суми, 40007
Свідоцтво суб'єкта видавничої справи ДК № 3062 від 17.12.2007.