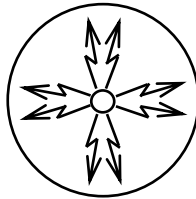


О. А. Борисенко

ЦИФРОВА СХЕМОТЕХНІКА

Підручник



Міністерство освіти і науки України
Сумський державний університет

О. А. Борисенко

ЦИФРОВА СХЕМОТЕХНІКА

Підручник

Рекомендовано вченою радою Сумського державного університету



Суми
Сумський державний університет
2016

УДК 681.3
ББК 32.973
Б82

Рецензенти:

А. С. Опанасюк – доктор фізико-математичних наук, професор, завідувач кафедри електроніки і комп'ютерної техніки Сумського державного університету;

О. Г. Пономарьов – доктор фізико-математичних наук, професор, завідувач відділу фізики пучків заряджених частинок Інституту прикладної фізики НАНУ (м. Суми)

*Рекомендовано до видання
вченою радою Сумського державного університету
як підручник
(протокол № 4 від 13 жовтня 2016 року)*

Борисенко О. А.

Б82 Цифрова схемотехніка : підручник / О. А. Борисенко.
– Суми : Сумський державний університет, 2016. –
200 с.

ISBN 978-966-657-642-5

Підручник «Цифрова схемотехніка» рекомендований для ВНЗ III–IV рівнів акредитації напрямів підготовки 6.172 «Телекомунікації та радіотехніка» і 6.171 «Електроніка» денної та заочної форм навчання з дисциплін «Цифрова схемотехніка» і «Пристрої цифрової електроніки».

До підручника входять як теоретичні питання, так і великий обсяг ілюстративного матеріалу у вигляді структурних та функціональних схем цифрових пристроїв. У кінці лекцій і в додатках наведені запитання і задачі, що можуть використовуватися для практичних і самостійних занять, а також модульного та сесійного контролю студентів.

УДК 681.3
ББК 32.973

ISBN 978-966-657-642-5

© Борисенко О. А., 2016
© Сумський державний університет, 2016

Зміст

С.

Вступ	6
ТЕМА 1. Загальні положення	7
1.1. Цифрова схемотехніка	7
1.2. Аналогові та цифрові пристрої	9
1.3. Класи цифрових пристроїв	10
ТЕМА 2. Базові схеми	14
2.1. Елементарні логічні схеми	14
2.2. Побудова схем за правилом де Моргана	22
2.3. Логічні базиси	25
2.4. Спеціальні логічні елементи	31
2.5. Логічні схеми в булевому базисі	41
ТЕМА 3 Дешифратори	45
3.1. Загальні поняття	45
3.2. Лінійні дешифратори	49
3.3. Каскадні дешифратори	52
3.4. Пірамідальні дешифратори	56
3.5. Неповні дешифратори	59
3.6. Матричні дешифратори	63
3.7. Двоступінчасті дешифратори	66
3.8. Підсумовувальні дешифратори	68
ТЕМА 4. Шифратори	79
4.1. Повні шифратори	79
4.2. Неповні шифратори	81
4.3. Пріоритетні шифратори	85
ТЕМА 5. Перетворювачі кодів	88
5.1. Загальні поняття	88
5.2. Перетворювач за логічними функціями	89
5.3. Перетворювач дешифратор – шифратор	91
ТЕМА 6. Цифрові комутатори	94
6.1. Мультиплексори	94
6.2. Логічні схеми на мультиплексорах	98

6.3. Каскадні мультиплексори	101
6.4. Демультимплексори	103
ТЕМА 7. Комбінаційні суматори	106
7.1. Суматори чисел	106
7.2. Півсуматори	109
7.3. Однорозрядні суматори	113
7.4. Суматори одиниць	117
ТЕМА 8. Схеми порівняння	121
8.1. Схеми порівняння цифр на рівність	121
8.2. Схеми порівняння чисел на рівність	122
8.3. Схеми порівняння цифр на нерівність	125
8.4. Схеми порівняння чисел на нерівність	129
ТЕМА 9. Тригери	131
9.1. Загальні поняття про тригери	131
9.2. Типи тригерів	133
9.3. Асинхронні <i>RS</i> -тригери	135
9.4. Синхронні <i>RS</i> -тригери	142
ТЕМА 10. Одноступінчасті тригери	146
10.1 Одноступінчасті <i>D</i> -тригери	146
10.2. Одноступінчасті <i>DV</i> -тригери	148
10.3. Одноступінчасті <i>T</i> -тригери	149
ТЕМА 11. Двоступінчасті тригери	153
11.1. Узагальнена схема	153
11.2. Двоступінчасті <i>RS</i> -тригери	155
11.3. Двоступінчасті <i>JK</i> -тригери	158
11.4. Двоступінчасті <i>T</i> - і <i>D</i> -тригери	161
ТЕМА 12. Регістри	164
12.1. Нагромаджувальні регістри	164
12.2. Регістри зсуву	168
12.3. Розподільники імпульсів	169
ТЕМА 13. Лічильники імпульсів	172
13.1. Загальні поняття	172
13.2. Підсумовувальні лічильники	174

13.3. Віднімальні лічильники.....	176
13.4. Реверсивні лічильники.....	178
13.5. Лічильники з довільним перерахунком	179
Післямова.....	181
Додаток А	182
Додаток Б.....	189
Додаток В_	194
Список літератури	196

Вступ

Сьогодні цифрові схеми за своїм значенням займають особливе місце. Вони використовуються у складі як побутових, так і промислових пристроїв, наприклад у пристроях керування, телезв'язку, комп'ютерах. Важко знайти сьогодні сферу людського життя, де б не використовувалися цифрові схеми. Тому фахівці з їх розроблення і експлуатації необхідні сьогодні й будуть потрібні у подальшому.

Цей підручник якраз і націлений на те, щоб дати базові знання у сфері схемотехніки майбутнім фахівцям, які спеціалізуються у галузі розроблення та виробництва цифрових схем. Цьому допомагає те, що матеріал у підручнику викладений стисло, але в той самий час зберігає глибину матеріалу, що викладається.

Підручник розрахований для викладання курсів «Цифрова схемотехніка» та «Пристрої цифрової електроніки» напрямів підготовки 6.172 «Телекомунікації та радіотехніка» і 6.171 «Електроніка» денної та заочної форм навчання, а також для самостійної роботи над курсом. Хоча насамперед він буде корисним викладачам відповідних дисциплін, тому що в ньому поряд із лекційним матеріалом зосереджена велика кількість питань і задач для підсумкового модульного й сесійного контролю.

ТЕМА 1. Загальні положення

1.1. Цифрова схемотехніка

Цифровую схему, пристроєм чи автоматом називають будь-яку схему, побудовану на елементах, що працюють за релейною логікою – двійковою чи багатозначною.

Наука, яка ґрунтується на синтезі цифрових схем, називається цифровою схемотехнікою.

Цифрові пристрої можуть бути за своєю природою різними – електронними, механічними, гідравлічними, пневматичними, але, як це впливає з їх попереднього визначення, основним для них є релейна логіка їх роботи, тобто дискретна зміна станів їх елементів.

Однак на сьогодні найбільшого поширення набули поки що електронні цифрові пристрої, оскільки для них характерна велика швидкодія, вони досить надійні, дешеві, мають малі габарити. З ними стикаємося усюди. Це комп'ютери, мобільні телефони, цифрове телебачення, схеми керування побутовою технікою тощо.

Але це не означає, що інші типи цифрових пристроїв не потрібні в житті. Там, де велика радіоактивність, не обійтися без релейних схем та відповідної автоматики, якщо потрібна пожежна безпека, наприклад на шахтах чи в газосховищах, то необхідно впроваджувати пневматичні цифрові автомати, а там, де проводяться

підводні роботи, необхідно використовувати цифрову гідроавтоматику.

Поява електронних цифрових пристроїв у користуванні людини викликала необхідність розроблення їх схем, що привело до виникнення нової професії схемотехніка в галузі цифрової електроніки та відповідної дисципліни «Цифрова схемотехніка», яка вивчається сьогодні у вищих навчальних закладах.

Сьогодні, як і на самому початку розвитку цифрової техніки, знання роботи і структури цифрових схем мають першорядне значення, хоча аспекти їх практичного застосування на цей час дещо змінилися. Треба вміти не тільки виготовляти і налагоджувати цифрові схеми, а також вміти їх ще й проектувати, зокрема за допомогою комп'ютерних програм.

Контрольні запитання і завдання

1. Яку схему називають цифровим пристроєм?
2. Чому на практиці сьогодні використовують здебільшого цифрові схеми?
3. Які можуть використовуватися ще цифрові пристрої, крім електронних?
4. У чому виявляється перевага електронних цифрових пристроїв над іншими їх типами, а в чому недолік?

1.2. Аналогові та цифрові пристрої

Електронні пристрої поділяють на дві великі групи – аналогові та цифрові. Аналогові пристрої під час роботи використовують в основному неперервні сигнали, а цифрові пристрої — сигнали, що мають перервну структуру, тобто мають імпульсну форму. Вони або є, або їх немає. На початку розвитку електроніки переважали аналогові пристрої, тому що на той час вони були більш швидкодіючими і мали менші габарити. Технологія їх виготовлення була добре відпрацьована.

Поряд цим розроблялися цифрові схеми на лампах або реле, де використовувалися цифрові сигнали, як правило, у вигляді появи або відсутності напруги або струму на їх елементах. Із часом цифрові пристрої стали все більше застосовуватись, а їх елементна база все більш покращувалася. Спочатку використовувалися для побудови цифрових пристроїв навісні транзисторні схеми, потім уся цифрова схема виготовлялася на одному кристалі, а у подальшому були розроблені мікропроцесори, що дозволило у кінцевому підсумку сконструювати сучасні комп'ютери.

Основна перевага цифрових схем – це їх висока точність роботи, але вони довго поступалися за швидкодією аналоговим пристроям, які теж весь час поліпшувалися. Сьогодні цифрові пристрої витісняють аналогові пристрої. Так, наприклад, хоча аналогове телебачення і зв'язок взагалі ще застосовується на практиці, але вже сьогодні їм на зміну приходять

цифрове телебачення і цифровий зв'язок. Але це не означає, що зникне аналогова техніка. Цього не станеться хоча б тому, що основою цифрових схем є аналогові елементи. Крім того, різні підсилювачі, особливо потужні, потребують для своєї реалізації аналогових пристроїв. Також є такі застосування електронних пристроїв, де аналогові схеми є більш ефективними, ніж цифрові. Наприклад, там, де необхідна дуже велика швидкодія, а точність не має особливого значення.

Контрольні запитання і завдання

1. Чим аналогові схеми відрізняються від цифрових схем?
2. У чому виявляється перевага аналогових схем, а в чому цифрових? Проаналізуйте.
3. Чому на практиці використовують сьогодні в основному цифрові схеми?

1.3. Класи цифрових пристроїв

Усі цифрові схеми можна поділити на елементарні схеми і побудовані на них більш складні схеми – пристрої, прилади чи автомати. Також схеми конструктивно поділяють на комбінаційні схеми і схеми з пам'яттю, а за логікою своєї роботи — на двійкові та багатозначні.

Комбінаційні схеми і схеми з пам'яттю

Цифрові схеми (прилади, пристрої, автомати) поділяють на 2 великих класи – схеми без пам'яті, або *комбінаційні* схеми, і схеми з *пам'яттю*. У комбінаційних схемах вихідні значення сигналів визначаються тільки поточними сигналами на вході. Функціонування такої схеми не залежить від її внутрішнього стану, який у неї є лише один. Приклади комбінаційних автоматів, з якими ми ознайомимося далі, – це дешифратори, шифратори, мультиплексори.

Схеми з пам'яттю мають більше ніж один внутрішній стан, в одному з яких схема може знаходитися як завгодно довго, поки не зміняться вхідні сигнали. Вихідні сигнали схем при цьому залежать від їх вхідних сигналів і станів, що були в попередні моменти часу. Зміна станів схем із пам'яттю зазвичай відбувається через певні наперед задані проміжки (інтервали) часу, які називаються тактами. Прикладами схем із пам'яттю можуть бути тригери, лічильники, регістри.

Двійкові та багатозначні цифрові схеми

На сьогодні більшість цифрових пристроїв користуються двома видами сигналів – або 1, або 0, тому що їх технічно значно легше реалізувати ніж у разі, коли кількість сигналів буде більшою двох. Відповідно і пристрої з двома значеннями сигналів будуть простішими і дешевшими пристроїв, що користуються кількістю сигналів більше двох, наприклад, трьома. Такі

цифрові пристрої називають багатозначними пристроями чи схемами. Крім того, двійкові пристрої будуть через простоту своєї реалізації ще і значно надійнішими. Тому на цей час цифрова техніка в основному базується на двозначній логіці. При цьому фізична природа цих сигналів може бути різною. Це і світлові хвилі, і тиск води або повітря, і електричні сигнали у вигляді рівнів напруги електричного струму. Останні сигнали дістали на сьогодні найбільшого поширення. Вони широко використовуються для кодування 0 і 1 під час схемної реалізації логічних функцій. Це пов'язано з тим, що електричні сигнали легко передавати і зберігати, а схеми, які їх використовують, мають велику швидкодію.

Схеми з позитивною і негативною логікою

Зазвичай 1 в цифрових пристроях кодується на практиці високим рівнем фізичного сигналу, наприклад, напруги, а 0 – низьким. Таке кодування називають *позитивною* логікою, а відповідні схеми - пристроями з позитивною логікою. Якщо кодування 1 і 0 проводиться у зворотному порядку, то його називають *негативною* логікою, а пристрої, що використовують цю логіку, – пристроями з негативною логікою. Позитивна логіка поширена більше, ніж негативна, тому що краще сприймається розробниками і користувачами цифрової апаратури. Але і негативна логіка трапляється у практиці проектування цифрових пристроїв. Тому використовують як пристрої з позитивною, так і негативною логікою.

Контрольні запитання і завдання

1. На які основні класи поділяють електронні схеми?
2. Чим відрізняються комбінаційні цифрові схеми від схем із пам'яттю?
3. Яка різниця між двійковими і багатозначними цифровими схемами?
4. У чому різниця між позитивною і негативною логікою цифрових схем?

ТЕМА 2. Базові схеми

2.1. Елементарні логічні схеми

Елементарними логічними схемами, або логічними елементами, називають найпростіші схеми, з яких складають будь-які цифрові схеми, але самі вони при цьому не можуть бути одержані за допомогою інших логічних схем.

Елементарними логічними елементами будуть елементи, що реалізують функції *НІ*, *І*, *АБО*, а також символи 0 і 1. Елементи *НІ* ще називають інверторами, елементи *І* – кон'юнкторами, елементи *АБО* – диз'юнкторами, елементи 0 і 1 – константами 0 і 1.

Константи

Константами називають такі логічні елементи, що не мають входів і не змінюють свого значення.

Здебільшого константа 1 при позитивному кодуванні реалізується за допомогою високого рівня фізичного параметра, наприклад напруги, а константа 0 – відсутністю цього значення або його низьким рівнем, хоча можливе й зворотне кодування, що називається негативним. При позитивному кодуванні зазвичай константа 0 моделює землю, а константа 1 – напругу джерела живлення *E*.

Інвертор

Інвертором називається логічний елемент, що має один вхід і один вихід, на якому виникає 0 тільки тоді, коли на вході інвертора буде 1, а 1, коли на його вході буде 0.

Інвертор реалізує логічну функцію *НІ* за допомогою одиничних або нульових значень напруги, струму чи інших фізичних параметрів. Логіка його роботи зображена в табл. 2.1.

Таблиця 2.1 – Логіка роботи інвертора

x	$f = \bar{x}$
0	1
1	0

На функціональних схемах інвертор зображується прямокутником, у якого вхід зліва, вихід справа (рис. 2.1 а, б). На вихідній або вхідній лінії в місці її з'єднання з прямокутником зображується кружок – символ інверсії. Стрілку на вхідних і вихідних лініях ставити заборонено.

Зображення інвертора може бути повернуто на 90° так, що вхід буде зверху, а вихід знизу (рис. 2.1 в, г). Інші повороти заборонені.

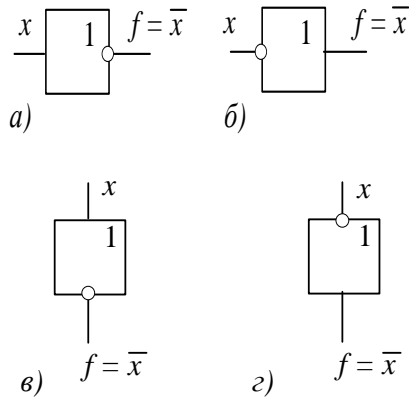


Рисунок 2.1 а, б, в, г – Зображення інвертора

У релейно-контактній логіці функцію *НИ* реалізує контакт x , який або замкнений, що відповідає його перебуванню в стані 1, або розімкнений, коли контакт знаходиться в стані 0 (рис. 2.2). Джерело напруги E , в цей час відображає високий його рівень.

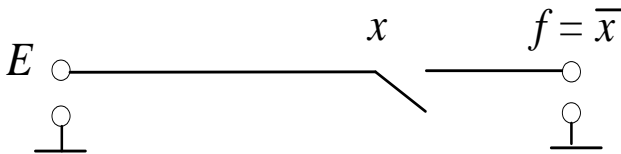


Рисунок 2.2 – Інвертор у релейному виконанні

Часова діаграма роботи схеми *НИ* зображена на рис. 2.3.

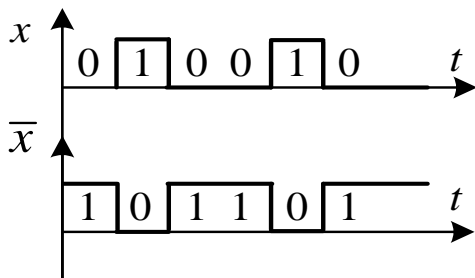


Рисунок 2.3 – Часова діаграма роботи інвертора

Кон'юнктор

Кон'юнктором (схемою *I*, схемою кон'юнкції, клапаном) називається логічний елемент із декількома входами та одним виходом, на якому з'являється 1 лише тоді, коли є сигнали 1 на всіх його входах.

Функціонує кон'юнктор відповідно до табл. 2.2.

Таблиця 2.2 – Логіка роботи кон'юнктора

x_1	x_2	$f = x_1 \cdot x_2$
0	0	0
0	1	0
1	0	0
1	1	1

Кон'юнктор реалізує операцію логічного множення:

$$f = x_1 \cdot x_2 = x_1 \wedge x_2 = x_1 \& x_2 = x_1 x_2.$$

Зображується він так, як це показано на рис. 2.4.

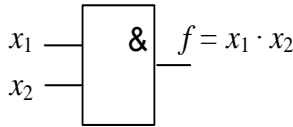


Рисунок 2.4 – Зображення кон'юнктора

У релейному вигляді кон'юнктор зображений на рисунку 2.5, а часова діаграма його роботи наведена на рисунку 2.6. Функцію його змінних x_1 і x_2 виконують контакти, які можуть бути або в положенні замкнено, тобто 1, або розімкнено – 0.

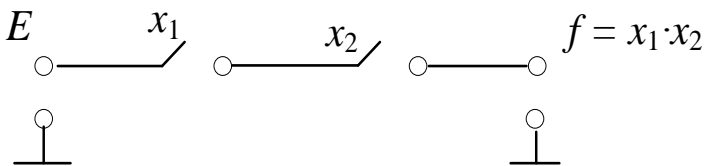


Рисунок 2.5 – Кон'юнктор у релейному виконанні

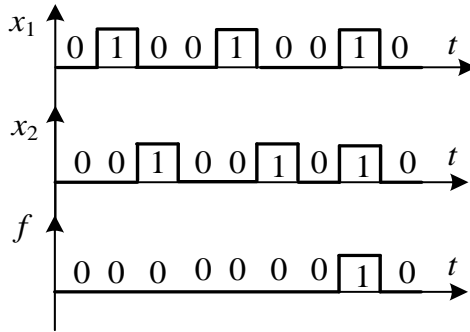


Рисунок 2.6 – Часова діаграма роботи кон'юнктора

Диз'юнктор

Диз'юнктором (схемою диз'юнкції, схемою АБО) називається такий логічний елемент, у якого з'являється на виході 0 лише тоді, коли на всіх його входах будуть нулі.

Функціонує диз'юнктор відповідно до табл. 2.3.

Таблиця 2.3 – Логіка роботи диз'юнктора

x_1	x_2	$f = x_1 \vee x_2$
0	0	0
0	1	1
1	0	1
1	1	1

Диз'юнктор реалізує логічну операцію *АБО* (логічне додавання):

$$f = x_1 + x_2 = x_1 \vee x_2 .$$

Функціональна схема диз'юнктора подана на рис. 2.7, релейний варіант – на рис. 2.8, часова діаграма роботи – на рис. 2.9.

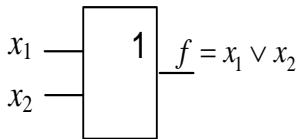


Рисунок 2.7 – Зображення диз'юнктора

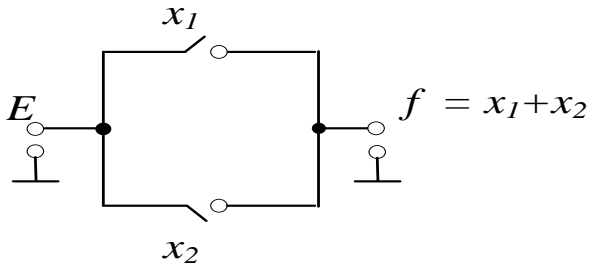


Рисунок 2.8 – Диз'юнктор у релейному виконанні

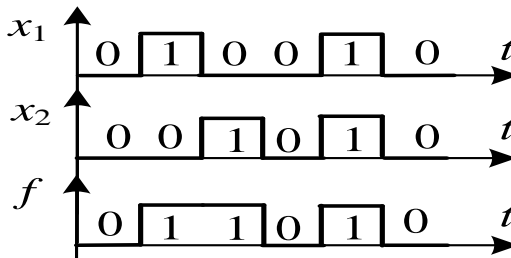


Рисунок 2.9 – Часова діаграма роботи диз'юнктора

Контрольні запитання і завдання

1. Які ви знаєте базові логічні елементи цифрових схем?
2. Які логічні елементи називають константами?
3. Наведіть релейну схему інвертора, його часову діаграму та опишіть її роботу.
4. Наведіть релейну схему кон'юнктора, його часову діаграму та опишіть її роботу.
5. Наведіть релейну схему диз'юнктора, його часову діаграму та опишіть її роботу.

2.2. Побудова схем за правилом де Моргана

Правило де Моргана

Важливе значення в цифровій схемотехніці має правило де Моргана. З одного боку, воно зводиться до того, що логічна сума двох змінних з інверсіями дорівнює інверсії логічного добутку цих змінних, а з іншого – логічний добуток двох змінних з інверсіями дорівнює інверсії логічної суми цих змінних. Тобто

$$\bar{x} \cdot \bar{y} = \overline{x + y} \quad ; \quad \bar{x} + \bar{y} = \overline{x \cdot y} .$$

Реалізація схеми *I* на основі правила де Моргана

Для реалізації схеми *I* досить часто використовується модифікована формула де Моргана

$$x \cdot y = \overline{\bar{x} + \bar{y}} .$$

Вона дозволяє функціонально замінити операцію *I* операцією *АБО* та інверсіями над змінними, що може бути у ряді випадків технічно легше реалізувати на практиці. Функціонування такої схеми наведено в табл. 2.4, а її зображення – на рис. 2.10.

Таблиця 2.4 – Логіка роботи елемента І за правилом де Моргана

x	y	\bar{x}	\bar{y}	$\bar{x} + \bar{y}$	$\overline{\bar{x} + \bar{y}} = x \cdot y$
0	0	1	1	1	0
0	1	1	0	1	0
1	0	0	1	1	0
1	1	0	0	0	1

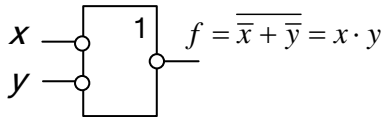


Рисунок 2.10 – Реалізація елемента І за правилом де Моргана

Реалізація схеми АБО на основі правила де Моргана

У цьому випадку використовується правило де Моргана у вигляді формули

$$x + y = \overline{\bar{x} \cdot \bar{y}}.$$

Функціонування схеми проходить відповідно до табл. 2.5.

Таблиця 2.5 – Логіка роботи елемента АБО за правилом де Моргана

x	y	\bar{x}	\bar{y}	$\bar{x} \cdot \bar{y}$	$\overline{\bar{x} \cdot \bar{y}} = x + y$
0	0	1	1	1	0
0	1	1	0	0	1
1	0	0	1	0	1
1	1	0	0	0	1

Реалізується схема АБО за правилом де Моргана у вигляді схеми, поданій на рис. 2.11.

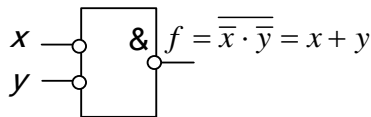


Рисунок 2.11 – Реалізація елемента АБО за правилом де Моргана

Контрольні запитання і завдання

1. Сформулюйте правило де Моргана у загальному вигляді.
2. Сформулюйте правило де Моргана для функції I .
3. Як реалізувати схему I за правилом де Моргана?
4. Сформулюйте правило де Моргана для функції АБО.

2.3. Логічні базиси

Функціонально повні логічні базиси

Набір логічних елементів, за допомогою якого можна реалізувати будь-яку логічну функцію, називають функціонально повним логічним, або універсальним базисом.

Найбільш відомим із цих базисів буде набір, що складається з логічних елементів *НИ, І, АБО*, а також констант 0 і 1. Нелегко знайти цифрову схему, де б не використовувався цей базис. Він називається *булевим*, тому що його вперше дослідив англійський вчений Буль.

Крім булевого базису, на практиці використовують ще два універсальні базиси, один з яких складається з елементів *І, НИ*, а інший – *АБО, НИ*. Ці базиси були названі відповідно до прізвищ вчених, які їх запропонували. Перший із них називається базисом «штрих Шеффера», а другий – базисом «стрілка Пірса». Вони відповідно позначаються як $A|B = \overline{A \cdot B}$ і $A \downarrow B = \overline{A + B}$. Як бачимо, ці базиси мають у своєму складі два логічних елементи, тобто на один елемент менше, ніж має булевий базис. Це означає, що булевий базис є надмірним, у той самий час як інші базиси є мінімальними. При цьому вони залишаються універсальними. Якщо видалити із цих базисів хоча б один логічний елемент, то він перетвориться в неповний базис, за допомогою якого вже не можна буде

реалізувати будь-яку логічну функцію, а відповідно і схему цифрового пристрою.

Нижче, у табл. 2.6, для порівняння між собою наведена логіка функціонування елементів I , $A \vee B$, $A|B$, $A \downarrow B$ для трьох змінних. Аналогічно будуються таблиці істинності для більшої кількості елементів.

Таблиця 2.6 – Логіка роботи базових елементів цифрових схем

Вхід			АБО	I	A B	A↓B
x ₁	x ₂	x ₃				
0	0	0	0	0	1	1
0	0	1	1	0	1	0
0	1	0	1	0	1	0
0	1	1	1	0	1	0
1	0	0	1	0	1	0
1	0	1	1	0	1	0
1	1	0	1	0	1	0
1	1	1	1	1	0	0

Логічні схеми на основі елемента «штрих Шеффера»

Як було показано вище, елемент «штрих Шеффера» має вигляд $A|B = \overline{A \cdot B}$, тобто він реалізує операцію логічного множення з інверсією (*I-НІ*). Тому його функціональна схема буде мати вигляд, наведений на рис. 2. 12:

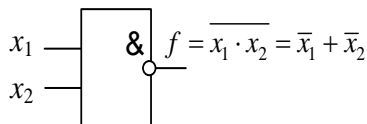


Рисунок 2.12 – Схема елемента «штрих Шеффера»

Елемент «штрих Шеффера» створює функціонально повний логічний базис, тому що з його допомогою можна виконати логічні операції інверсії, додавання і множення.

Для реалізації інвертора на елементі «штрих Шеффера» треба лише перемкнути його входи між собою, і тоді, при подаванні на вхід елемента сигналу 1, на виході буде сигнал 0 і зворотно, при подачі на вхід 0, на виході буде 1 (див. рис. 2.13). У такому разі буде побудований елемент *НІ*. Для одержання констант 0 і 1 потрібно лише постійно подавати на входи побудованого на елементі «штрих Шеффера» інвертора 1 або 0. Тоді на виході відповідно буде постійно знаходитися 0 або 1.

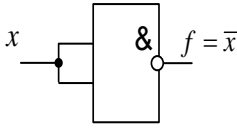


Рисунок 2.13 – Інвертор на елементі «штрих Шеффера»

Реалізація схеми *I* за допомогою елемента «штрих Шеффера» показана на рис. 2.14, а схеми АБО на рис. 2.15.

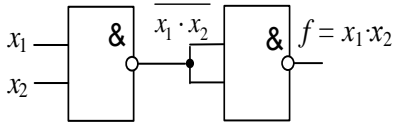


Рисунок 2.14 – Схема *I* на елементах «штрих Шеффера»

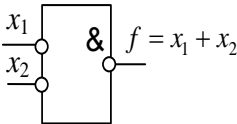
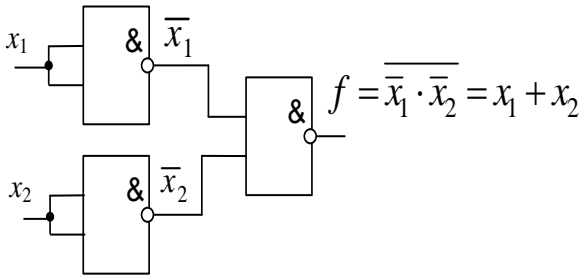


Рисунок 2.15 – Схеми АБО на елементах «штрих Шеффера»

Схеми на основі елемента «стрілка Пірса»

Елемент «Стрілка Пірса» $A \downarrow B = \overline{A + B}$ реалізує операцію множення з інверсією (АБО – НІ) і також як і елемент «штрих Шеффера» створює функціонально повний (універсальний) мінімальний базис. Схеми цього елемента показана на рис. 2.16.

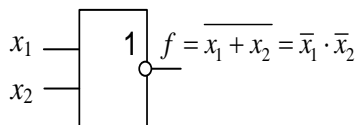


Рисунок 2.16 – Елемент «стрілка Пірса»

Виконання інверсії за допомогою цього елемента наведено на рис. 2.17. Як бачимо, для цього потрібно лише перемкнути входи цього елемента між собою.

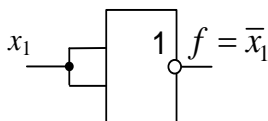


Рисунок 2.17 – Інвертор на елементі «стрілка Пірса»

Операція логічного множення I подана на рис. 2.18. Для її виконання потрібно три елементи «стрілка Пірса», два з яких виконують функції схеми НІ.

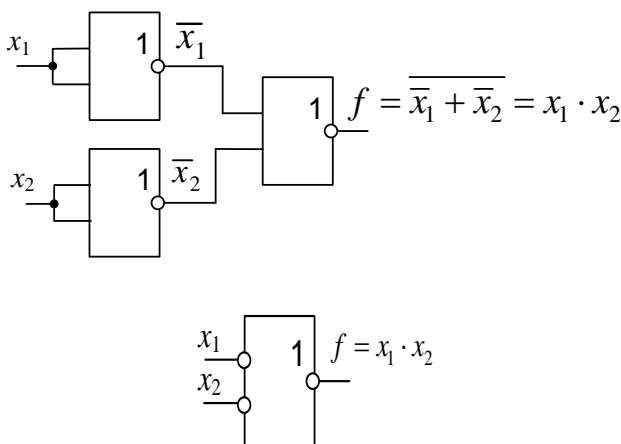


Рисунок 2.18 – Схема І на елементах «стрілка Пірса»

Простіше реалізується на елементах «стрілка Пірса» схема логічного додавання АБО, тому що потребує для своєї реалізації всього два елементи «стрілка Пірса» (див. рис. 2.19).

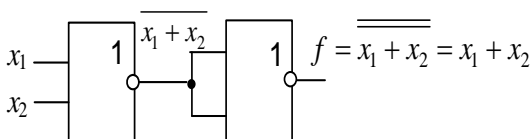


Рисунок 2.19 – Схема АБО на елементах «стрілка Пірса»

Розглянуті вище логічні схеми мали два входи, хоча таких входів на практиці, звичайно, трапляється більше. Але для кінцевого результату проектування це не має особливого значення, тому що на практиці діє простий принцип суперпозиції, відповідно до якого кожна змінна, яка надходить на вхід логічної схеми вважається

вихідною змінною, тобто функцією від двох або більше інших змінних. Тому всі операції, які відбуваються в цифровій схемі, можна зобразити як такі, що виконуються лише над двома змінними. Тобто кінцевий результат операцій над багатьма змінними можна зобразити як операції над двома змінними.

Контрольні запитання і завдання

1. Який набір елементів буде функціонально повним?
2. Що таке булевий базис?
3. Розкажіть, як працює елемент «штрих Шеффера». Наведіть його типове зображення і часову діаграму роботи.
4. Опишіть роботу елемента «стрілка Пірса» та наведіть його часову діаграму і графічну схему.
5. Реалізуйте елемент «стрілка Пірса» на елементах «штрих Шеффера».
6. Реалізуйте елемент «штрих Шеффера» на елементах «стрілка Пірса».

2.4. Спеціальні логічні елементи

На практиці, крім розглянутих вище базових логічних елементів, також широко використовують спеціальні логічні елементи. Серед них найбільш уживані «сума за модулем 2», «мажоритарний елемент», «виключне АБО». Вони реалізуються на базових елементах, розглянутих у попередній лекції. У них сигнал на виході дорівнює 1 тоді, коли на вході схеми «виключне АБО» тільки одна із всіх вхідних змінних дорівнює 1, коли у мажоритарному

елементі кількість вхідних змінних, що утримують 1, буде більше кількості змінних, які утримують 0, а також коли в елементі «сума за модулем 2» кількість одиниць в змінних буде непарною. Відповідне функціонування цих логічних елементів показано в табл. 2.7.

Таблиця 2.7 – Логіка роботи спеціальних елементів

x_1 x_2 x_3	$M2$	<i>Мажоритарний елемент</i>	<i>Виключне АБО</i>
0 0 0	0	0	0
0 0 1	1	0	1
0 1 0	1	0	1
0 1 1	0	1	0
1 0 0	1	0	1
1 0 1	0	1	0
1 1 0	0	1	0
1 1 1	1	1	0

Крім цих основних спеціальних логічних елементів, на практиці, хоча і меншою мірою, використовуються також логічні елементи «рівнозначність» та «імплікація», логіка роботи яких буде наведена нижче.

Синтез спеціальних елементів у булевому базисі відбувається за загальними правилами синтезу логічних схем у ньому: знаходиться ДДНФ або ДКНФ відповідних логічних функцій, проходить їх мінімізація, якщо це можливо, і потім на основі синтезованої логічної функції будується логічна схема спеціального елемента.

Проведемо відповідний синтез цих елементів і почнемо з елемента «сума за модулем 2». Він ще називається елементом «нерівнозначності» і на схемах позначається як «M2», або як «сума за mod 2». Цей елемент досить поширений у цифровій схемотехніці, особливо широко він використовується в схемах самоконтролю цифрових пристроїв. Логіка його роботи для двох змінних подана в табл. 2.8. Відповідна до цієї таблиці логічна функція має вигляд:

$$f = x_1 \oplus x_2 = \bar{x}_1 x_2 + x_1 \bar{x}_2 .$$

Таблиця 2.8 – Логіка роботи елемента «M2»

x_1	x_2	$f = x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0

На рис. 2.20 наведено стандартне позначення цього логічного елемента, а функціональні схеми його реалізації показані на рис. 2.21 і 2.22.

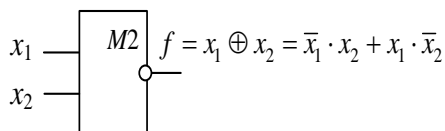


Рисунок 2.20 – Елемент «сума за модулем 2»

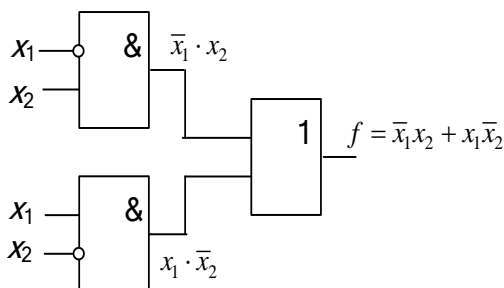


Рисунок 2.21 – Реалізація елемента «сума за модулем 2» у булевому базисі

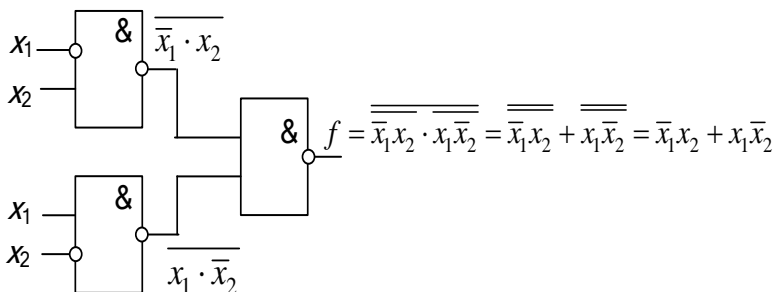


Рисунок 2.22 – Реалізація елемента «сума за модулем 2» на елементах «штрих Шеффера»

«Мажоритарний елемент» базується на функції трьох змінних, що формує вихідний сигнал за принципом голосування «два з трьох». Відповідні значення мажоритарної функції y для трьох змінних a b c подані у табл. 2.9.

Таблиця 2.9 – Мажоритарна функція

a	b	c	y
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

Користуючись таблицею 2.9, запишемо логічне рівняння для вихідний функції y вигляді ДДНФ:

$$y = \bar{a}bc \vee a\bar{b}c \vee ab\bar{c} \vee abc.$$

Після мінімізації

$$y = ab + bc + ac,$$

а після використання правила де Моргана

$$y = \overline{\overline{ab} \cdot \overline{bc} \cdot \overline{ac}}.$$

Використовуючи останнє рівняння, будемо схему мажоритарного елемента (рис. 2. 23).

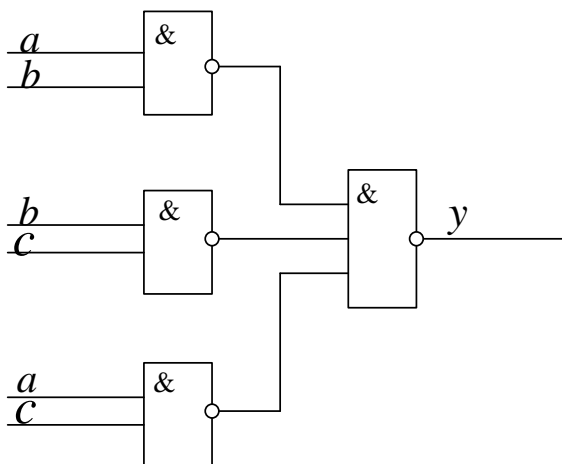


Рисунок 2.23 – Схема «мажоритарний елемент»

Елемент «виключне АБО» формує вихідний сигнал, що дорівнює 1, тоді, коли лише один з аргументів дорівнює 1. Коли 2 чи більше аргументів дорівнюють 1, тоді на його виході з'являється 0. Ця логіка відображена у табл. 2.10.

Таблиця 2.10 – Логіка роботи елемента
«виключне АБО»

<i>a</i>	<i>b</i>	<i>c</i>	<i>y</i>
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Користуючись табл. 2 10, запишемо логічне рівняння для вихідної змінної у вигляді ДДНФ:

$$y = \overline{a}\overline{b}c \vee \overline{a}b\overline{c} \vee a\overline{b}\overline{c}.$$

Ця ДДНФ одночасно є мінімальною ДНФ, тобто її не можна у подальшому мінімізувати. За допомогою правила де Моргана запишемо попередню логічну функцію без операції диз'юнкції:

$$y = \overline{\overline{abc} \cdot \overline{abc} \cdot abc}.$$

Використовуючи її, реалізуємо функцію y на елементах *I-НІ* («штрих Шеффера») (див. рис. 2.24).

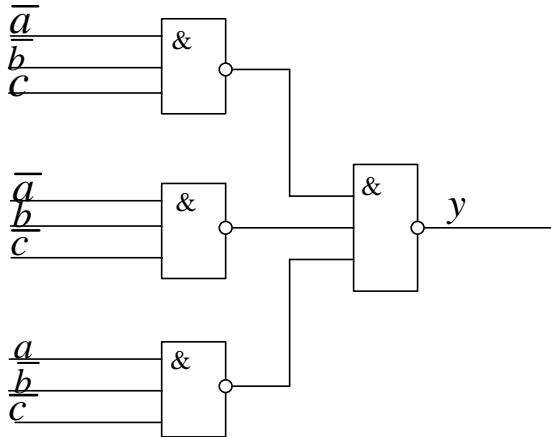


Рисунок 2.24 – Схема елемента «виключне АБО»

Елемент «рівнозначність» реалізує функцію рівнозначності

$$a \sim b = \overline{a} \overline{b} + ab.$$

Вона дорівнює одиниці у разі рівності значень вхідних змінних і дорівнює нулю у разі їх нерівності, інверсна щодо функції «сума за модулем 2». Правило функціонування цього елемента подано в табл. 2.11, а одна із можливих його схем надана на рис. 2.25.

Таблиця 2.11 – Логіка роботи елемента «рівнозначність»

a	b	$a \sim b$
0	0	1
0	1	0
1	0	0
1	1	1

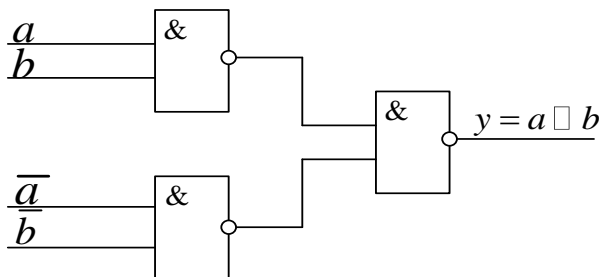


Рисунок 2.25 – Схема елемента «рівнозначність»

Елемент «імплікація» $a \rightarrow b$ – це функція двох змінних, значення істинності якої показано в табл. 2. 12.

Таблиця 2.12 – Логіка роботи елемента «імплікація»

a	b	$a \rightarrow b$
0	0	1
0	1	1
1	0	0
1	1	1

Запишемо логічне рівняння для вихідної змінної y у вигляді ДКНФ:

$$y = \bar{a} + b.$$

Застосувавши до одержаного виразу правило де Моргана, маємо

$$y = \overline{a\bar{b}}.$$

Реалізуємо функцію імплікації на елементі «штрих Шеффера». Його зображення подано на рис.2.26.

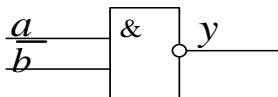


Рисунок 2.26 – Елемент «імплікація»

Контрольні запитання і завдання

1. Реалізуйте елемент «сума за модулем 2» у базисі «стрілка Пірса».
2. Реалізуйте «мажоритарний елемент» у булевому базисі.
3. Реалізуйте схему «виключне АБО» у базисі «штрих Шефера».
4. Побудуйте елемент «рівнозначності» на елементах «імплікація».

2.5. Логічні схеми в булевому базисі

На практиці досить часто складні логічні функції подаються в аналітичній формі за допомогою елементарних логічних функцій, серед яких знаходяться не тільки булеві функції, а і більш складні, які можна, у свою чергу, подати за допомогою булевих функцій. Тоді виникає завдання перетворення складних логічних функцій у булеві функції.

Для того щоб перейти до булевого базису, необхідно подані вихідні логічні функції розкласти на елементарні логічні функції. Порядок виконання операцій перетворення в логічних функціях такий: дужки, кон'юнкція; диз'юнкція; імплікація, рівнозначність, нерівнозначність, сума за модулем 2.

Розглянемо на прикладах реалізацію логічних функцій у різних базисах. Необхідно реалізувати в булевому базисі цифрову логічну схему, що реалізує логічну функцію

$$F = (A + \bar{B}) \rightarrow (C \sim A).$$

Враховуючи, що $C \sim A = AC + \bar{A}\bar{C}$, одержимо

$$F = (A + \bar{B}) \rightarrow (C \sim A) = \overline{A + \bar{B}} + AC + \bar{A}\bar{C} = \bar{A}B + AC + \bar{A}\bar{C}.$$

Цю формулу можна одержати й інакше, використавши табл. 2.13.

Таблиця 2.13 – Функціонування логічної схеми

A	B	C	\bar{B}	$A + \bar{B}$	$C \sim A$	F
0	0	0	1	1	1	1
0	0	1	1	1	0	0
0	1	0	0	0	1	1
0	1	1	0	0	0	1
1	0	0	1	1	0	0
1	0	1	1	1	1	1
1	1	0	0	1	0	0
1	1	1	0	1	1	1

Відповідно до цієї таблиці функція F у булевому базисі після її мінімізації буде мати такий вигляд:

$$F = \bar{A}\bar{C} + \bar{A}B + AC.$$

Контрольні запитання і завдання

1. Який існує порядок виконання логічних операцій під час перетворення логічних функцій?
2. Для чого потрібно перетворювати логічні функції у булеву форму?
3. Яка послідовність операцій перетворення логічних функцій у булеві функції?
4. Як перевірити правильність перетворення логічних функцій у булеві за допомогою таблиць істинності?

ТЕМА 3 Дешифратори

3.1. Загальні поняття

Дешифратором називається цифровий пристрій, в якому в робочому стані активний сигнал виникає на одному і тільки одному виході.

Дешифратори дискретних сигналів відносять до одних з основних вузлів систем автоматичного керування, телефонних і телеграфних систем комунікації, ЕОМ, систем зв'язку і телемеханіки. На практиці широко застосовують декілька структур дешифраторів, серед яких найбільш прості лінійні. Ці структури відрізняються одна від одної кількістю каскадів та елементів, швидкістю, характером навантаження на джерела входних сигналів. Також вони різняться кількістю логічних елементів, часом затримки сигналів, коефіцієнтами розгалуження логічних елементів, завадостійкістю, надійністю.

Дешифратори бувають також *повні* і *неповні*. У повному дешифраторі є n входів і 2^n виходів. У неповному кількість виходів буде меншою 2^n . Отже, якщо повний дешифратор має 5 входів, то кількість виходів у ньому буде 32, а якщо 10, то – 1024. Як бачимо, кількість виходів у повному дешифраторі зростає за експонентою відносно кількості входів.

Умовне позначення дешифратора – DC (decoder). Наприклад, якщо в повному дешифраторі є 3 входи, то кількість його виходів дорівнює 8 (рис. 3.1).

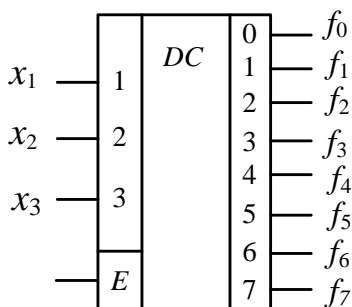


Рисунок 3.1 – Стандартне позначення дешифратора

Входи дешифратора називають ще адресними входами, тому що вони задають адрес виходу, що активізується.

Крім входів і виходів, дешифратор звичайно має вхід E , що вмикає дешифратор. При $E = 1$ дешифратор перебуває в робочому стані. При $E = 0$ на всіх виходах встановлюються неактивні сигнали, що дорівнюють нулю, незалежно від того, є чи немає на його входах сигналів.

Будь-який дешифратор функціонує відповідно до табл. 3.1, в якій для 3 змінних x_1, x_2, x_3 наведено 8 можливих функцій – f_0, f_1, \dots, f_7 . Вони відповідають конститuentам 1.

Таблиця 3.1 – Лінійний дешифратор із $n = 3$ входами

№	x_1	x_2	x_3	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	0	0	0	1	0	0	0	0	0	0	0
2	0	0	1	0	1	0	0	0	0	0	0
3	0	1	0	0	0	1	0	0	0	0	0
4	0	1	1	0	0	0	1	0	0	0	0
5	1	0	0	0	0	0	0	1	0	0	0
6	1	0	1	0	0	0	0	0	1	0	0
7	1	1	0	0	0	0	0	0	0	1	0
8	1	1	1	0	0	0	0	0	0	0	1

Реалізація логічних функцій

Оскільки на кожному виході дешифратора з будь-якої структурою реалізується конституента одиниці (див. табл. 3.1), то очевидно, що за його допомогою може бути реалізована будь-яка логічна функція f з кількістю змінних, що дорівнює кількості входів дешифратора. Для цього виходи дешифратора, які відповідають конституентам 1, необхідно з'єднати за допомогою схеми АБО.

Приклад. Необхідно реалізувати функцію, задану в диз'юнктивній нормальній формі, за допомогою дешифратора:

$$f = abc\bar{d} + \bar{a}b\bar{c}d.$$

Для цього перетворимо її в ДДНФ і подамо у вигляді сум конститuent 1:

$$f = abc\bar{d} + ab\bar{c}\bar{d} + \bar{a}b\bar{c}d = f_{13} + f_{12} + f_5.$$

Відповідні одержаним конститuentам 1 виходи дешифратора об'єднаємо схемою АБО. У результаті на її виході буде одержана функція f (див. рис.3.2).

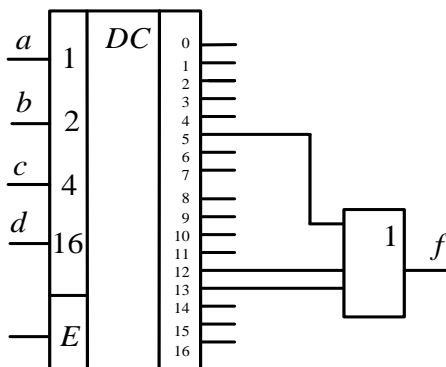


Рисунок 3.2 – Реалізація логічної функції

Звернемо увагу на те, що у своєму складі будь-яка логічна функція має дешифратор – повний чи неповний. Тому підняття швидкодій, надійності й завадостійкості дешифраторів є одночасно підвищенням цих показників і для логічних функцій.

Контрольні запитання і завдання

1. Який пристрій називається дешифратором?
2. Де на практиці використовують дешифратори?
3. Нарисуйте стандартне позначення дешифратора, що використовується на функціональних схемах.
4. Які сигнали подаються на адресні входи дешифратора і чому їх називають адресними?
5. Як можна реалізувати логічну функцію на дешифраторі?

3.2. Лінійні дешифратори

Дешифратори, в яких логічні схеми І чи АБО розміщені у вигляді одного каскаду, називаються лінійними.

Функціонування лінійного дешифратора описується системою n логічних функцій f_i , $i = 0, 1, \dots, 2^n - 1$, як наприклад, таких, що впливають з табл. 3.1.

З цієї таблиці одержимо такі логічні рівняння для кожного з виходів дешифратора:

$$f_0 = \bar{x}_1\bar{x}_2\bar{x}_3, \quad f_1 = \bar{x}_1\bar{x}_2x_3, \quad f_2 = \bar{x}_1x_2\bar{x}_3, \quad f_3 = \bar{x}_1x_2x_3, \\ f_4 = x_1\bar{x}_2\bar{x}_3, \quad f_5 = x_1\bar{x}_2x_3; \quad f_6 = x_1x_2\bar{x}_3, \quad f_7 = x_1x_2x_3.$$

За цими рівняннями на рис. 3.3 для трьох вхідних змінних x_1, x_2, x_3 будується комбінаційна схема лінійного дешифратора, що містить відповідно

$2^n = 2^3 = 8$ виходів. Якщо на входи цього дешифратора при $E = 1$ подати набір змінних 000, то на входах відповідного елемента I з'являться 3 одиниці 1, отже, на його виході буде 1, а на кожному з виходів інших елементів – 0. Функція f_0 відповідно буде дорівнювати 1. Аналогічно до цього подача на вхід дешифратора будь-якого іншого набору вхідних змінних приведе до появи на виході відповідного йому елемента I одиничного сигналу, а на всіх інших виходах елементів дешифратора з'являться нулі.

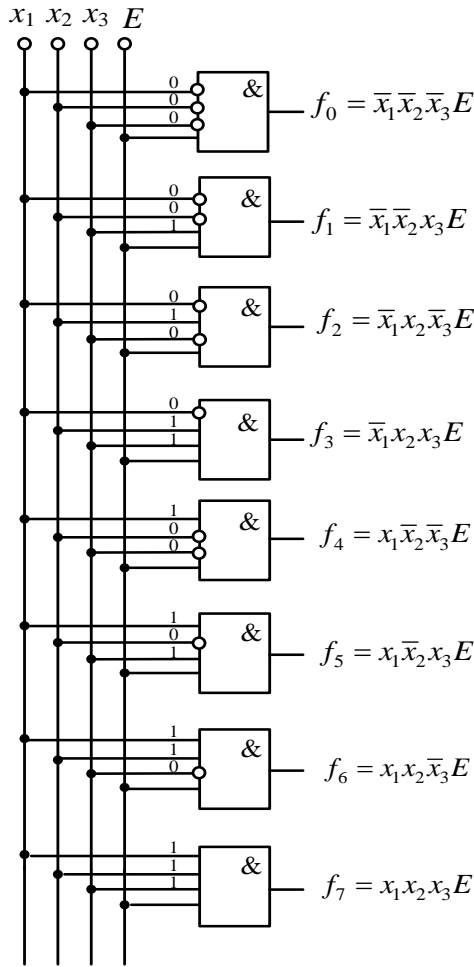


Рисунок 3.3 – Схема повного лінійного дешифратора

Кількість входів у схемах I , потрібних для реалізації лінійного дешифратора,

$$N = (n + 1) \cdot 2^n .$$

Для $n = 3$; число $N = 4 \cdot 2^3 = 32$.

Лінійний дешифратор – це найбільш швидкодіюча схема дешифратора, тому що сигнал в ній затримується тільки на одній схемі I . Через це він широко застосовуваний на практиці.

Контрольні запитання і завдання

1. Який дешифратор називається лінійним?
2. Якими логічними функціями описується робота лінійного дешифратора?
3. Нарисувати функціональну схему лінійного дешифратора.
4. Як оцінити кількість входів, що має лінійний дешифратор?

3.3. Каскадні дешифратори

Каскадним дешифратором називається дешифратор, що має у своїй структурі більше одного каскаду логічних схем I .

Хоча лінійні дешифратори є найбільш швидкодіючими, однак їх реалізація у вигляді мікросхем із великою розрядністю входних слів важка, оскільки потрібно виготовити для однієї мікросхеми з n входами, що виконує функції дешифратора, 2^n контактних площинок для реалізації його виходів. А ще потрібні контактні площинки для входів, блока живлення,

сигналів керування мікросхемою в цілому. Тому часто ставиться завдання виготовлення дешифратора з великою кількістю входів і виходів на основі мікросхем з обмеженою кількістю контактів. Для цього схема дешифратора складається з декількох каскадів лінійних дешифраторів. Найчастіше ця кількість дорівнює двом. При цьому перший каскад керує мікросхемами – дешифраторами другого каскаду, другий – мікросхемами дешифраторами третього каскаду і т.д.

Для прикладу на рис. 3.4 поданий каскадний дешифратор, який реалізується двома каскадами за допомогою дешифраторів, що мають 2 входи і 4 виходи. Дешифратор першого каскаду, який є один, дешифрує 2 вхідних розряди. Відповідно один із його 4 виходів дає дозвіл на роботу одного з 4 дешифраторів другого каскаду. В результаті на одному з виходів лише цього дешифратора з'явиться сигнал. На виходах інших трьох дешифраторів у цей час поява сигналів буде заборонена. Оскільки дешифраторів у другому каскаді 4, то сумарно дешифратор має 16 виходів. При цьому тільки на одному з цих 16 виходів буде знаходитися одиничний сигнал.

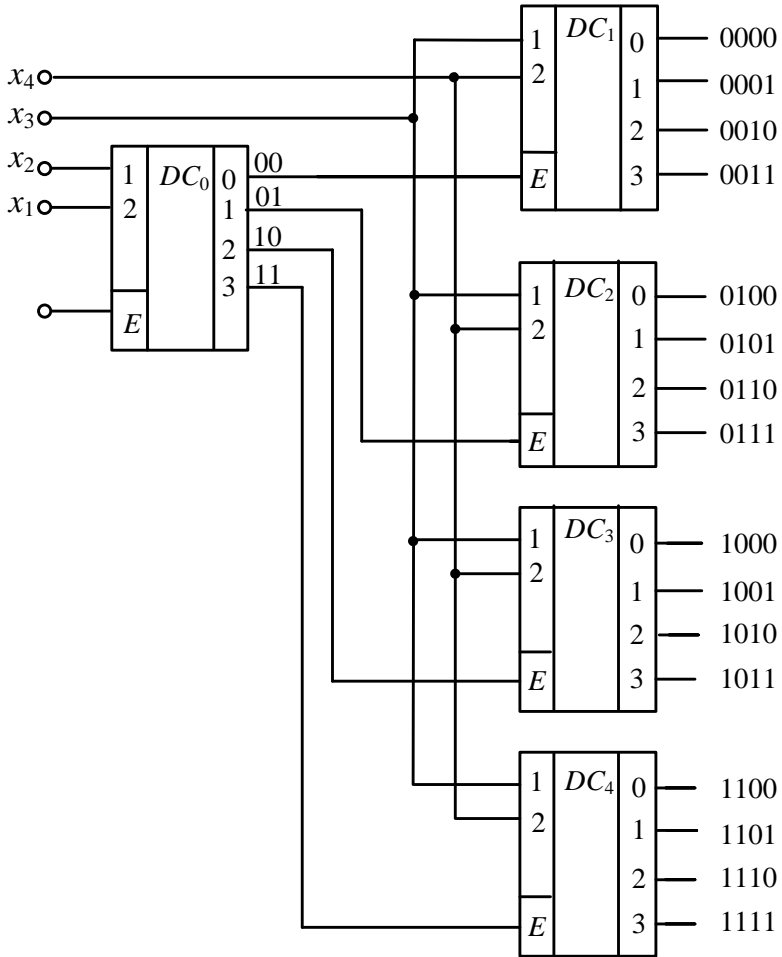


Рисунок 3.4 – Схема двокаскадного дешифратора

Розглянемо більш детально роботу наведеної схеми. На дешифратор DC_0 першого каскаду заведені розряди x_1, x_2 комбінації двійкового коду, яка дешифрується.

Кожен із входів E дешифраторів $DC_1 - DC_4$ зв'язаний з одним із виходів дешифратора DC_0 . Тому в кожний момент часу тільки один дешифратор, на який подається сигнал дозволу з дешифратора DC_0 , дешифрує розряди x_3, x_4 . Тобто він дешифрує два старших розряди x_3, x_4 вхідної кодової комбінації, що подаються на його входи.

Кожен дешифратор другого каскаду можна ввімкнути аналогічно з другим і т. д., і таким чином зібрати схему, яка дешифрує двійкові комбінації з чотирма розрядами. Недоліком каскадного з'єднання дешифраторів є збільшення часу проходження сигналів дешифратора від входу до виходу, тому що на кожному каскаді дешифратора відбувається певна затримка вхідного сигналу.

Контрольні запитання і завдання

1. Який дешифратор називається каскадним?
2. Чому виникла потреба в каскадних дешифраторах?
3. Поясніть роботу двокаскадного дешифратора за його функціональною схемою.
4. Які переваги має каскадний дешифратор порівняно з лінійним?

3.4. Пірамідальні дешифратори

Каскадний дешифратор, одержаний за допомогою каскадного з'єднання двох або більше лінійних дешифраторів, реалізований на логічних схемах I з двома виходами і одним входом, називається пірамідальним дешифратором.

На рисунку 3.5 наведена функціональна схема пірамідального дешифратора з трьома каскадами, трьома входами і вісьма виходами, а на рис. 3.6 – функціональна.

Його перевагою є те, що він використовує для своєї реалізації схеми I з двома входами, а недоліком – додаткова затримка сигналу на другому і третьому каскадах.

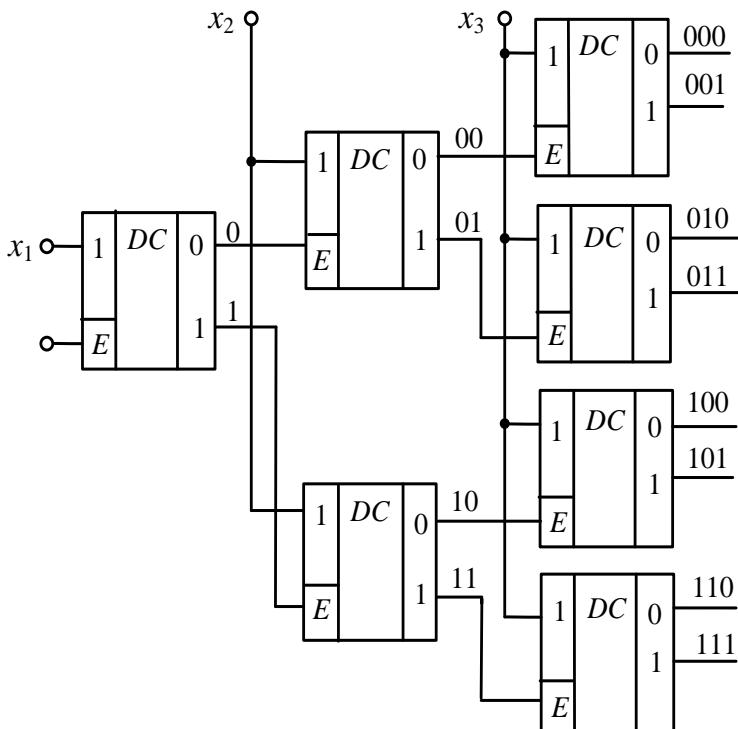


Рисунок 3.5 – Схема пірамідального дешифратора з трьома каскадами

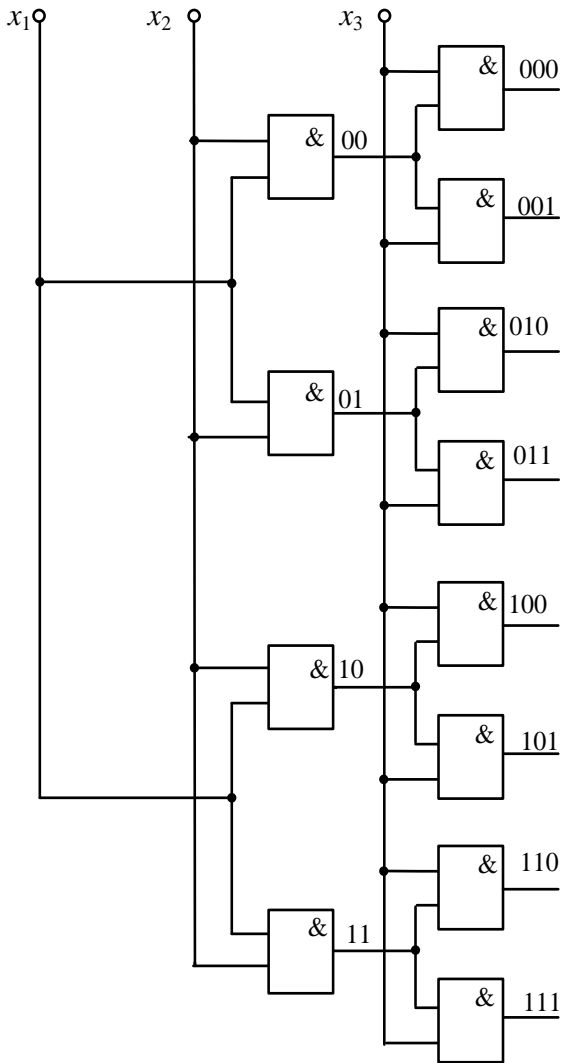


Рисунок 3.6 – Схема пірамідального дешифратора з двома каскадами

Контрольні запитання і завдання

1. Який дешифратор називається пірамідальним?
2. Які переваги має пірамідальний дешифратор?
3. Опишіть роботу пірамідального дешифратора за його функціональною схемою.
4. Яку кількість входів має пірамідальний дешифратор залежно від його розрядності? Порахуйте на конкретному прикладі.

3.5. Неповні дешифратори

Дешифратори, які розпізнають менше 2^n кодових комбінацій, називаються неповними.

Це означає, що дешифратори можуть бути повними, коли використовують всі можливі набори змінних, і неповними, якщо певні вхідні набори змінних не використовують. Ця обставина дозволяє за необхідності спростити дешифратор.

У цьому разі неробочим наборам ставиться у відповідність будь-яке з двох можливих значень виходів повного дешифратора – 0 або 1, тому що цього виходу в неповному дешифраторі в реальності не існує. Такі дешифратори використовують, наприклад, для перетворення двійково-десяткових цифр до відповідних їм сигналів. Приклад однієї з таблиць функціонування такого дешифратора наведений нижче у табл. 3.2.

Таблиця. 3.2 – Логіка роботи неповного дешифратора

x_3	x_2	x_1	x_0	F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9
0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	0	0
0	0	1	1	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1	0	0	0	0	0
0	1	0	1	0	0	0	0	0	1	0	0	0	0
0	1	1	0	0	0	0	0	0	0	1	0	0	0
0	1	1	1	0	0	0	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	0	0	0	0	1
1	0	1	0	*	*	*	*	*	*	*	*	*	*
1	0	1	1	*	*	*	*	*	*	*	*	*	*
1	1	0	0	*	*	*	*	*	*	*	*	*	*
1	1	0	1	*	*	*	*	*	*	*	*	*	*
1	1	1	0	*	*	*	*	*	*	*	*	*	*
1	1	1	1	*	*	*	*	*	*	*	*	*	*

Відповідно до таблиці істинності одержимо за допомогою діаграми Вейча або методу Квайна для кожної функції неповного дешифратора логічне рівняння з урахуванням неповністю визначених наборів, що дозволяє провести мінімізацію дешифратора за апаратурними витратами. Це для цього прикладу будуть такі рівняння:

$$F_0 = \bar{x}_3\bar{x}_2\bar{x}_1\bar{x}_0, \quad F_5 = x_2\bar{x}_1x_0,$$

$$F_1 = \bar{x}_3\bar{x}_2\bar{x}_1x_0, \quad F_6 = x_2x_1\bar{x}_0,$$

$$F_2 = \bar{x}_2x_1\bar{x}_0, \quad F_7 = x_2x_1x_0,$$

$$F_3 = \bar{x}_2x_1x_0, \quad F_8 = x_3\bar{x}_0,$$

$$F_4 = x_2\bar{x}_1\bar{x}_0, \quad F_9 = x_3x_0.$$

Одержані рівняння використаємо для побудови схеми дешифратора, наведеної на рис. 3.7. Вона складена з логічних елементів *I* з чотирма, трьома і двома входами, тобто у цьому разі використовують елементи *I* з меншою ніж 4 кількістю входів, що приводить до економії апаратурних витрат, необхідних для реалізації неповного дешифратора.

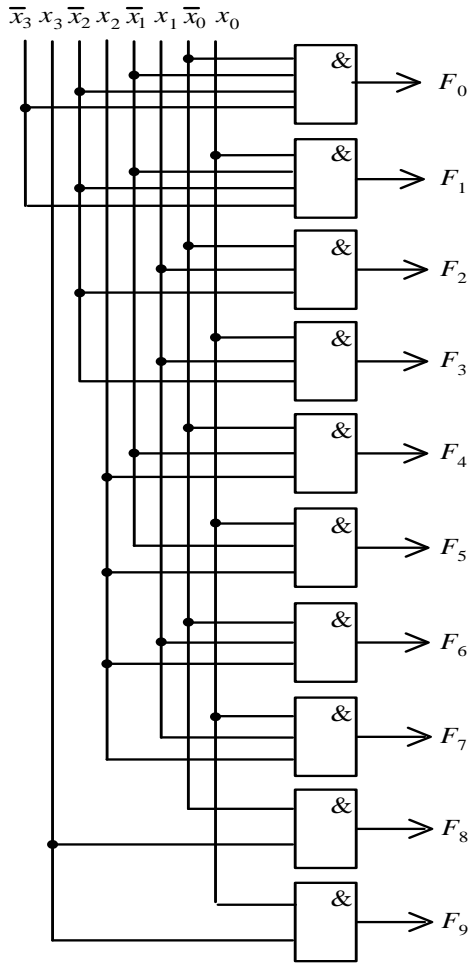


Рисунок 3.7 – Схема неповного дешифратора

Контрольні запитання і завдання

1. Який дешифратор називається неповним і чому?
2. Чому неповний дешифратор можна мінімізувати за кількістю апаратурних витрат?
3. Як мінімізуються апаратурні витрати в неповному дешифраторі?
4. Опишіть роботу неповного дешифратора за його схемою.

3.6. Матричні дешифратори

Матричним дешифратором називається неповний дешифратор, який схемно реалізує збіг елементів прямокутної матриці з будь-якою позиційною значністю.

Використовується в багатьох електронних пристроях, наприклад у засобах відображення інформації. Вирішує питання дешифрування багатозначних позиційних кодів, поданих у вигляді матриць.

Математична модель реалізується за допомогою логічних двійкових функцій. Кожному виходу матричного дешифратора у загальному вигляді відповідає одна з N двійкових функцій від двох змінних:

$$f_{\gamma} = x_i y_j, \quad i = 1, 2, \dots, k, \quad j = 1, 2, \dots, n, \quad \gamma = 1, 2, \dots, N.$$

Для матричного дешифратора, поданого як приклад на рис. 3.8, це будуть такі функції:

$$f_1 = x_1 y_1, f_2 = x_2 y_1, f_3 = x_3 y_1,$$

$$f_4 = x_1 y_2, f_5 = x_2 y_2, f_6 = x_3 y_2,$$

$$f_7 = x_1 y_3, f_8 = x_2 y_3, f_9 = x_3 y_3.$$

Реалізується дешифратор на схемах I з двома входами, на один з яких подається відповідний сигнал з одного боку матриці, а на другий — з іншого. У результаті використовують в їх схемах тільки двовходові схеми I , що робить їх технологічними і надійними.

На рисунку 3.8 подана функціональна схема дешифратора на 6 входів.

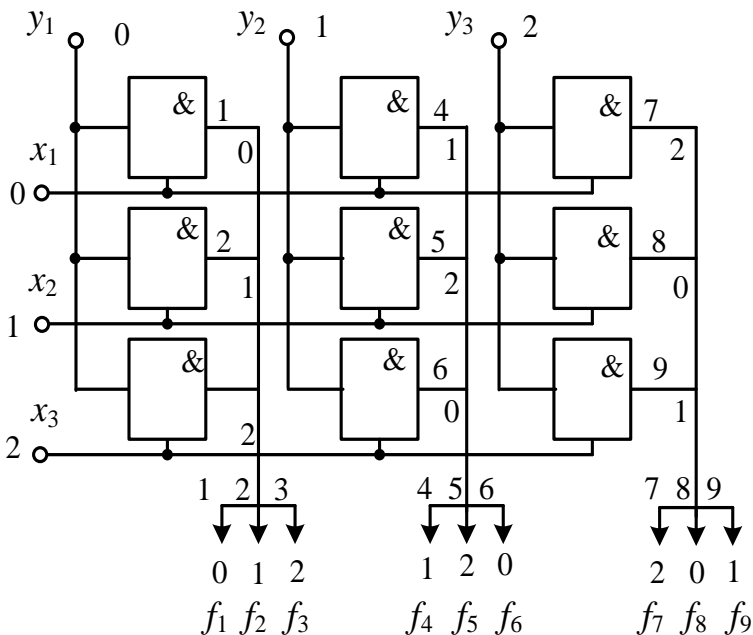


Рисунок 3.8 – Матричний дешифратор

Вхідні сигнали у матричному дешифраторі з’являються одночасно на одному із входів зверху і одному із входів зліва. Відповідно активізується одна із 9 схем I матриці та на її виході з’явиться 1.

Кількість схем I в дешифраторі $N = n \cdot k$, де n – кількість його входів зліва, а k – зверху. У прикладі, що розглядається, $n = 3$, $k = 3$. Тому кількість схем I буде дорівнювати $N = n \cdot k = 3 \cdot 3 = 9$. Відповідно кількість входів матричного дешифратора у прикладі дорівнює 18, а кількість виходів – 9.

Контрольні запитання і завдання

1. Який пристрій називається матричним дешифратором?
2. У яких випадках він застосовується?
3. Чому під час реалізації матричного дешифратора використовують двовходові схеми I ?
4. Як обчислити кількість схем I в матричному дешифраторі?

3.7. Двоступінчасті дешифратори

Двоступінчастими дешифраторами називають дешифратори, в яких перший ступінь має в своєму складі звичайні лінійні дешифратори DC_1 і DC_2 , а другий містить матричний дешифратор.

Старші й молодші розряди вхідної кодової комбінації подаються в однаковій або близькій до неї кількості на дешифратори DC_1 , DC_2 і дешифруються кожним із них окремо, як це зображено, наприклад, на рис. 3.9. Сигнали з виходів лінійних дешифраторів DC_1 і DC_2 в ньому подаються на входи матричного дешифратора і активізують один із його виходів, що і є ознакою дешифрування вхідної кодової комбінації. У цьому разі як лінійні дешифратори, так і матричні дешифратори використовують для побудови схеми I з двома входами, що приводить до підвищення рівня технологічності цифрової схеми під час реалізації дешифратора.

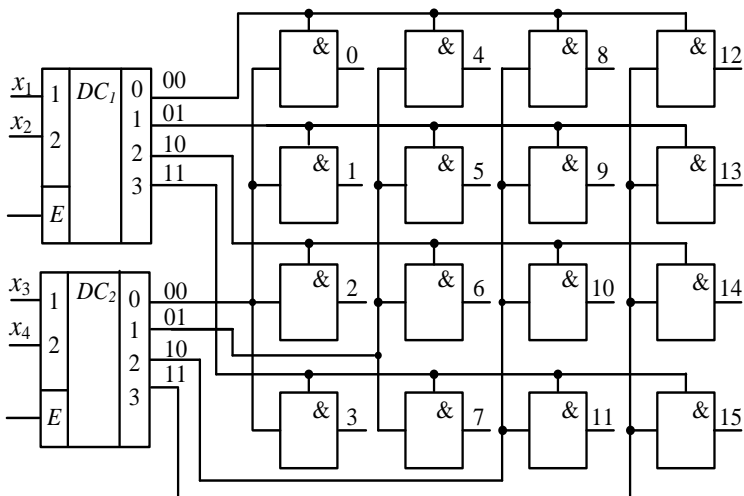


Рисунок 3.9 – Двоступінчастий дешифратор

Основними перевагами двоступінчастих дешифраторів, як це бачимо з наведеної вище схеми, є простота, однорідність і надійність структури.

Контрольні запитання і завдання

1. Який пристрій називається двоступінчастим дешифратором?
2. У яких випадках він застосовується?
3. Чому під час реалізації другий ступінь використовує двовходові схеми I ? Як обчислити кількість схем I у двоступінчастому дешифраторі?

3.8. Підсумовувальні дешифратори

Пристрій, в якому для дешифрування використовується інформація про кількість одиниць у комбінації, що дешифрується, називається підсумовувальним дешифратором, або дешифратором з підсумовуванням одиниць.

Інколи шляхом одержання додаткової інформації про кодові комбінації, які дешифруються, можна зменшити апаратурні витрати на побудову дешифраторів. Досить зручною операцією для цієї мети є операція підрахунку кількості одиниць у вхідній кодовій комбінації, що дешифрується. Далі кількість одиниць використовується для економії апаратурних затрат, яких потребує дешифратор.

Підсумовувальні дешифратори значно зменшують кількість апаратурних витрат порівняно зі звичайними дешифраторами і тому, якщо не враховувати апаратуру на реалізацію суматорів, є більш економними. Отже, вони можуть застосовуватися під час конструюванні дешевої і надійної апаратури зі зниженим енергоспоживанням.

Є багато випадків на практиці, коли кількість одиниць підрахована вже до дешифрування кодової комбінації, наприклад при передаванні інформації за допомогою коду Бергмана. Але навіть якщо є потреба в суматорі одиниць, то його можна одержати комбінаційним шляхом, наприклад за допомогою матричних комбінаційних дешифраторів. Правда, загальна кількість апаратури у такому випадку значно зросте. Однак економія апаратури

при цьому все рівно відбудеться. Крім того, це дозволяє будувати завадостійкі дешифратори.

Розглянемо спосіб побудови таких економних дешифраторів на основі підсумовування одиниць на прикладі дешифратора, що дешифрує 16 комбінацій, наведених у табл. 3.3. У ній для кожної комбінації знайдена кількість одиниць, які вона має. Це дає можливість розбити всі 16 двійкових комбінацій на 5 груп, відповідно до кількості утримуваних у них одиниць – нуля, однієї, двох, трьох, чотирьох.

Таблиця 3.3 – Логіка роботи підсумовувального дешифратора

№ пор.	Кодова комб. $x_1x_2x_3x_4$	Кільк. одиниць	№ пор.	Кодова комб. $x_1x_2x_3x_4$	Кільк. одиниць
0	0000	0	8	1000	1
1	0001	1	9	1001	2
2	0010	1	10	1010	2
3	0011	2	11	1011	3
4	0100	1	12	1100	2
5	0101	2	13	1101	3
6	0110	2	14	1110	3
7	0111	3	15	1111	4

У таблиці 3.4 показані ці групи. Вони мають комбінації з однаковою кількістю одиниць і нулів, тому створюють *рівноважні* коди, або коди з *постійною* вагою. Символи S_0, S_1, S_2, S_3, S_4 у табл. 3.4 при цьому надають відповідних ознак групам комбінацій, що мають 0, 1, 2, 3, 4 одиниці.

Таблиця 3.4 – Групування кодових комбінацій

S_0	S_1	S_2	S_3	S_4
$x_1x_2x_3x_4$	$x_1x_2x_3x_4$	$x_1x_2x_3x_4$	$x_1x_2x_3x_4$	$x_1x_2x_3x_4$
0000	0001	0011	0111	1111
	0010	0101	1011	
	0100	1001	1101	
	1000	0110	1110	
		1010		
		1100		

Припустимо тепер, що для будь-якої кодової комбінації, що дешифрується, відомі числа 0, 1, 2, 3, 4, утримуваних нею одиниць. Це дозволяє дешифрувати кодову комбінацію з тієї чи іншої групи кодових комбінацій за допомогою відповідного дешифратора рівноважних кодів, який можна побудувати на основі звичайного неповного лінійного дешифратора, де відкидаються елементи, налаштовані на дешифрування комбінацій з більшою або меншою кількістю одиниць, ніж задано в групі (див. рис. 3. 10).

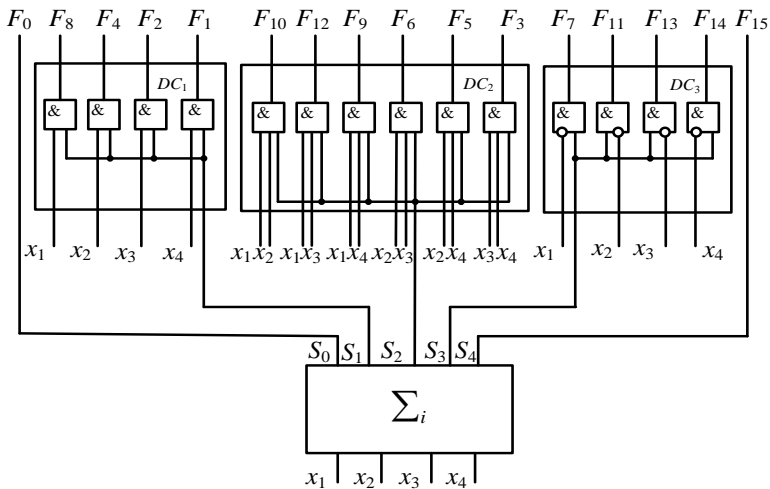


Рисунок 3.10 – Підсумовувальний дешифратор для чотирьох змінних із суматором одиниць

Групи S_0 і S_4 при цьому утримують по одній комбінації і тому для їх дешифрування не має потреби мати певні додаткові елементи, крім відповідних виходів суматора, які можна вважати виродженими дешифраторами на один вхід.

Вихідні функції всіх п'яти рівноважних дешифраторів від чотирьох змінних $x_1x_2x_3x_4$ будуть мати такий вигляд:

$$F_0 = S_0,$$

$$\begin{aligned} F_1 &= x_4 S_1, F_2 = x_3 S_1, F_4 = x_2 S_1, F_8 = x_1 S_1, \\ F_3 &= x_3 x_4 S_2, F_5 = x_2 x_4 S_2, F_9 = x_1 x_4 S_2, F_6 = x_2 x_3 S_2, \\ F_{10} &= x_1 x_3 S_2, F_{12} = x_1 x_2 S_2, \\ F_7 &= \bar{x}_1 S_3, F_{11} = \bar{x}_2 S_3, F_{13} = \bar{x}_3 S_3, F_{14} = \bar{x}_4 S_3, \end{aligned}$$

$$F_{15} = S_4.$$

Їх кількість, очевидно, дорівнює 16.

Для схемної реалізації цих функцій підсумовувальний дешифратор повинен мати суматор одиниць із 5 виходами або просто 5 додаткових входів до 4 змінних, що дешифруються, на одному з яких виникає сигнал, який відповідає тій чи іншій групі S_0, S_1, S_2, S_3, S_4 . Відповідно до цього сигналу дешифрується вхідна комбінація, що надійшла на входи підсумовувального дешифратора. Цю роботу виконує той чи інший із 5 рівноважних дешифраторів.

У загальному випадку вихідні функції, реалізовані підсумовувальним дешифратором, будуть мати вигляд $F_0, F_1, F_2, \dots, F_{2^n-1}$. Загальна ж кількість груп $S_i, i = 0, 1, \dots, n$, буде, очевидно, дорівнювати $n+1$.

У розглянутому прикладі для чотирьох змінних x_1, x_2, x_3, x_4 дешифратор має три лінійних дешифратори рівноважних кодів DC_1, DC_2, DC_3 для груп S_1, S_2, S_3 і два вироджених дешифратори S_0 і S_1 , кожний для одної кодової комбінації у вигляді виходів із суматора. Вихідні функції $F_0 - F_{15}$ підсумовувального дешифратора розміщуються у порядку, показаному на рис. 3.10: $F_0; F_8 F_4 F_2 F_1; F_{10} F_{12} F_6 F_5 F_3 F_9; F_7 F_{11} F_{13} F_{14} F_{15}$.

Кількість апаратурних витрат у цифрових схемах зазвичай визначається за величиною кількості входів у їх схемах. Такий підрахунок називається підрахунком апаратурних витрат за Квайном. У цьому прикладі для реалізації всіх рівноважних дешифраторів необхідно мати $1 + 2 \times 4 + 3 \times 6 + 2 \times 4 + 1 = 36$ входів, тоді ж як для звичайного лінійного дешифратора необхідно мати $16 \times 4 = 64$ входи.

Кількість схем I рівноважного дешифратора в загальному випадку визначається кількістю рівноважних комбінацій – сполучень у групі:

$$C_n^i = \frac{n!}{i!(n-i)!}, \quad i = 0, 1, \dots, n,$$

де n – кількість розрядів у рівноважних комбінаціях групи; i – кількість одиниць у них. У таблиці 3.5 наведені значення C_n^i для $n = 2 \div 8$.

Таблиця 3.5 – Кількість сполучень

n	i									
	0	1	2	3	4	5	6	7	8	
2	1	2	1							
3	1	3	3	1						
4	1	4	6	4	1					
5	1	5	10	10	5	1				
6	1	6	15	20	15	6	1			
7	1	7	21	35	35	21	7	1		
8	1	8	28	56	70	56	28	8	1	

Якщо тепер обчислити добуток

$$Q = (i+1)C_n^i,$$

то він визначить кількість входів схем I у відповідному рівноважному дешифраторі.

У таблиці 3.6 наведена кількість входів у дешифраторах рівноважних комбінацій і у відповідних їм лінійних дешифраторах. Кількість входів у схемах I рівноважних дешифраторів визначається кількістю одиниць у комбінаціях, що дешифруються. Добуток кількості схем I на кількість входів у них задає кількість апаратних витрат на рівноважний дешифратор за Квайном.

Таблиця 3.6 – Кількість входів

n	Лінійний дешифратор	Дешифратор сполучень
2	8	4
3	24	12
4	64	34
5	160	80
6	384	194
7	896	434
8	2048	998

Відповідно вираз

$$S = \sum_{i=1}^{n-1} (i+1)C_n^i$$

визначить кількість входів усіх схем I , що входять взагалі до підсумовувального дешифратора.

Додаткова мінімізація апаратурних витрат дешифраторів сполучень можлива за рахунок використання інверсних значень вхідних змінних. Якщо в комбінації кількість одиниць менша від кількості нулів, то використовують прямі значення вхідних змінних, якщо ж більша кількість нулів, то – інверсні значення.

Для прикладу, що розглядався вище,

$$S = 2C_4^1 + 3C_4^2 + 4C_4^3 = 8 + 18 + 16 = 42.$$

Цю кількість входів можна зменшити, якщо замість дешифрування за одиницями використовувати дешифрування за нулями, коли їх кількість у комбінації буде меншою за кількість одиниць. Тоді

$$S = 2C_4^1 + 3C_4^2 + 2C_4^3 = 8 + 18 + 8 = 34.$$

Як бачимо, спосіб побудови дешифраторів із суматорами дозволяє у ряді випадків більше ніж удвічі зменшити кількість апаратурних витрат на дешифратори сполучень щодо відповідних їм лінійних дешифраторів. Це робить його досить ефективним у

багатьох застосуваннях, особливо там, де швидкодія дешифраторів не має особливого значення.

Однак загальна кількість апаратурних витрат у підсумовувальному дешифраторі збільшиться, коли в ньому будуть ще враховані апаратурні витрати, яких потребує суматор одиниць. Як один із варіантів його можна реалізувати у вигляді матричного суматора. Зростання апаратури в матричних суматорах проходить не так стрімко, як в дешифраторах сполучень, тому за значної кількості вхідних змінних їх вплив на загальну кількість апаратурних витрат у дешифраторі з суматором буде зменшуватися. Однак швидкодія цих дешифраторів дещо зменшується, що можна віднести до їх недоліків.

Контрольні запитання і завдання

1. Який пристрій називається підсумовувальним дешифратором?
2. У яких випадках рекомендується застосування підсумовувальних дешифраторів?
3. Чому у підсумовувальних дешифраторах використовують рівноважні дешифратори?
4. Як обчислити кількість схем I у рівноважному і підсумовувальному дешифраторах?

ТЕМА 4. Шифратори

4.1. Повні шифратори

Шифратором називається пристрій, що перетворює вхідний сигнал одного із його входів на кодову комбінацію на його виходах.

Функціонує зворотно до функціонування дешифратора, тобто перетворює унітарний код на двійковий. Повний шифратор має 2^n входів і n виходів. Умовне позначення повного шифратора показано на рис. 4.1. Він має 8 входів і 3 виходи.

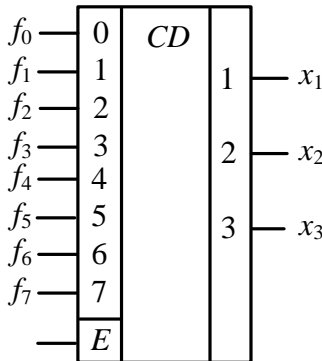


Рисунок 4.1 – Повний шифратор

У таблиці 4.1 показана логіка функціонування цього шифратора, а на рис. 4.2 – функціональна схема, що його реалізує.

Таблиця 4.1 – Функціонування повного шифратора

f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	x_1	x_2	x_3
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

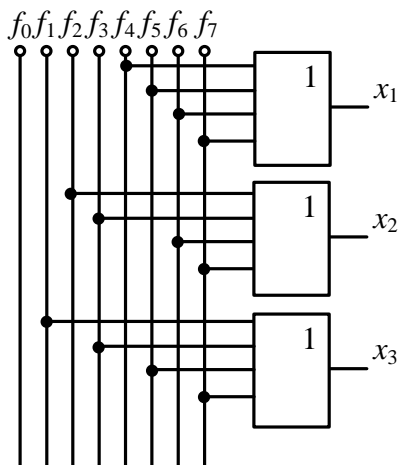


Рисунок 4.2 – Схема повного шифратора

Контрольні запитання і завдання

1. Який пристрій називається шифратором?
2. У яких випадках він застосовується?
3. Опишіть роботу шифратора за функціональною схемою.
4. Яку кількість входів має шифратор залежно від його розрядності? Порахуйте на конкретному прикладі.

4.2. Неповні шифратори

На практиці досить часто використовують *неповні* шифратори. Одне з таких застосувань неповного шифратора є введення за його допомогою даних із клавіатури. Під час натиснення клавіші клавіатури, з'єднаної з шифратором, на його виході з'являється цифра в двійково-десятковому коді, що далі передається

на цифровий пристрій. У цьому разі достатньо використати неповний шифратор 10×4 , що перетворює десяткові цифри $0, 1, \dots, 9$ на чотирирозрядні двійково-десяткові цифри $0000, 0001, \dots, 1001$.

Функціонування такого шифратора буде проходити відповідно до табл. 4.2.

Таблиця 4.2 – Функціонування неповного шифратора

Збуджений вхід	В и х і д			
	x_3	x_2	x_1	x_0
F_0	0	0	0	0
F_1	0	0	0	1
F_2	0	0	1	0
F_3	0	0	1	1
F_4	0	1	0	0
F_5	0	1	0	1
F_6	0	1	1	0
F_7	0	1	1	1
F_8	1	0	0	0
F_9	1	0	0	1

З таблиці 4.2 випливає, що

$$x_0 = F_1 \vee F_3 \vee F_5 \vee F_7 \vee F_9,$$

$$x_1 = F_2 \vee F_3 \vee F_6 \vee F_7,$$

$$x_2 = F_4 \vee F_5 \vee F_6 \vee F_7,$$

$$x_3 = F_8 \vee F_9.$$

Це означає, що реалізація шифратора може проводитися за аналогією до тієї самої схеми, як і на рис. 4.2, тобто вона буде реалізуватися на елементах АБО. Однак шифратор можна реалізувати і на елементах І. Для цього необхідно виразити одержані співвідношення через операцію І-НІ, використовуючи для цього правило де Моргана. Тоді одержані вище рівняння будуть мати такий вигляд:

$$x_0 = \overline{\overline{F_1} \cdot \overline{F_3} \cdot \overline{F_5} \cdot \overline{F_7} \cdot \overline{F_9}} E,$$

$$x_1 = \overline{\overline{F_2} \cdot \overline{F_3} \cdot \overline{F_6} \cdot \overline{F_7}} E,$$

$$x_2 = \overline{\overline{F_4} \cdot \overline{F_5} \cdot \overline{F_6} \cdot \overline{F_7}} E,$$

$$x_3 = \overline{\overline{F_8} \cdot \overline{F_9}} E.$$

На основі цих виразів будується шифратор, показаний на рис. 4.3.

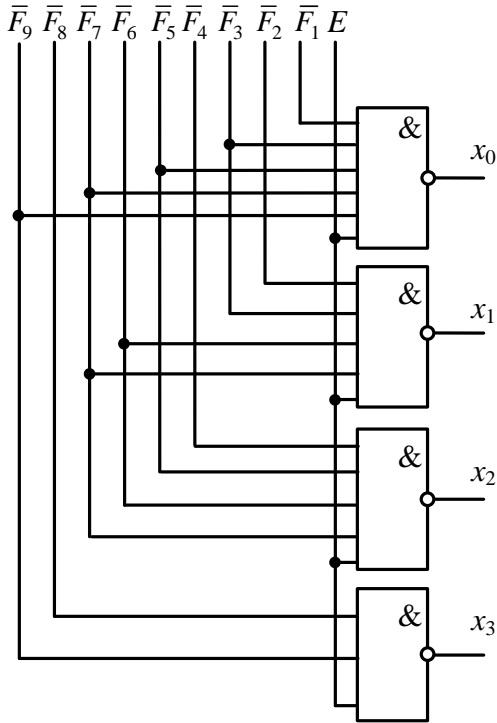


Рисунок 4.3 – Шифратор 10×4

Контрольні запитання і завдання

1. Який пристрій називається неповним шифратором?
2. У яких випадках він застосовується?
3. Опишіть роботу неповного шифратора за функціональною схемою.
4. Яку кількість входів має шифратор залежно від його розрядності? Порахуйте на конкретному прикладі.

4.3. Пріоритетні шифратори

Крім звичайних шифраторів, на практиці використовують схеми, що вирішують більш складні завдання шифрування. До таких шифраторів відносять, наприклад, *пріоритетні* шифратори, які виконують у двійковій комбінації пошук старшого розряду, що утримує, наприклад, 1. Тобто якщо є комбінація 0110, то старшим розрядом, який утримує 1, буде, очевидно, другий розряд.

У своїй структурі такий пріоритетний шифратор містить блок виділення старшої одиниці у вигляді сигналу на одному з його виходів і звичайний повний або неповний шифратор (див. рис. 4.4).

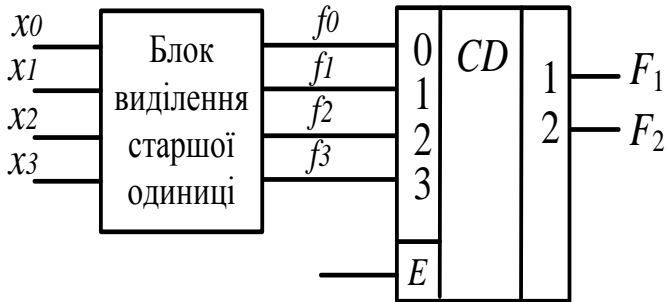


Рисунок 4.4 – Структура пріоритетного шифратора

Схема блока виділення старшої одиниці складається для кожного розряду вхідного слова зі схеми диз'юнкції

та двох інверторів, пов'язаних між собою так, як це показано на рис. 4.5.

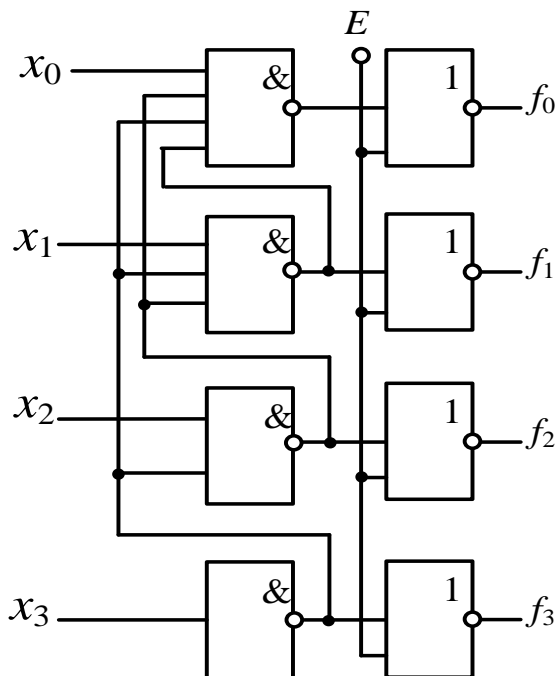


Рисунок 4.5 – Функціональна схема пріоритетного шифратора

Із цієї схеми випливає, що старша одиниця вхідного слова, де б вона не стояла, проходить на свій вихід без змін. Будь-які нулі при цьому породжують на виходах схем *I-HI* відповідних розрядів 1 і тому не впливають на молодші розряди. Потім ці одиниці інвертуються за допомогою відповідних інверторів, і на виходах цих

інверторів з'являються нулі. Старша 1 на схемі *I-III*, на яку вона подається, перетворюється на 0, а потім за допомогою відповідного їй інвертора – на 1. Усі молодші та старші розряди при цьому будуть видавати нулі.

Таким чином, лише на одному з розрядів блока виділення старшої одиниці буде 1, а на інших – будуть розміщені нулі. Ця одиниця відповідно до рис. 2.14 подається на шифратор, за допомогою якого перетворюється на відповідний двійковий адрес розряду, де вона стоїть. Він і є результатом роботи пріоритетного шифратора.

Контрольні запитання і завдання

1. Який пристрій називається пріоритетним шифратором?
2. Коли появляється необхідність у пріоритетному шифраторі?
3. Опишіть роботу пріоритетного шифратора за його структурною схемою.
4. Опишіть роботу пріоритетного шифратора за функціональною схемою.

ТЕМА 5. Перетворювачі кодів

5.1. Загальні поняття

Перетворювачем кодів називається пристрій, що перетворює одну кодову комбінацію на іншу.

Перетворювачі кодів – це найбільш поширені на сьогодні схеми цифрової схемотехніки, до яких відносять шифратори, дешифратори, логічні схеми, постійні запам'ятовувальні пристрої (ПЗП) і навіть комп'ютер, тому що він відіграє роль перетворювача вхідних сигналів, що надходять, наприклад, із клавіатури, на вихідні сигнали на екрані монітора. Стандартне позначення перетворювача кодів наведено на рис. 5 1.

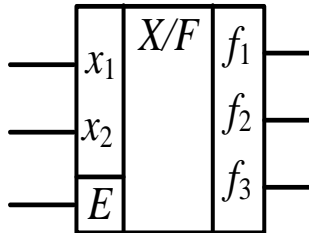


Рисунок 5.1 – Позначення перетворювача кодів

Якщо перетворювач кодів реалізується за допомогою комбінаційних схем (автоматів), то такий перетворювач називається комбінаційним. Його перевагами є простота реалізації і велика швидкодія. Основний його недолік – великі апаратні витрати під час реалізації складних цифрових схем. Тому такі перетворювачі

використовують у разі, коли реалізуються досить прості їх схеми. У більш складних випадках на практиці застосовують перетворювачі кодів із пам'яттю, але і в них широко використовують комбінаційні схеми. Необхідно відзначити, що завжди можна комбінаційний перетворювач кодів реалізувати як автомат із пам'яттю, а автомат із пам'яттю – як комбінаційний автомат.

Є два підходи до синтезу комбінаційних перетворювачів кодів. Перший реалізується у вигляді системи логічних функцій, що одержують із двійкової таблиці істинності, а другий – використовує систему дешифратор – шифратор.

Контрольні запитання і завдання

1. Який пристрій називається перетворювачем кодів?
2. У чому полягає перевага комбінаційних перетворювачів кодів щодо перетворювачів із пам'яттю, а в чому – їх недолік?
3. Де застосовують перетворювачі кодів?
4. Які є підходи до синтезу перетворювачів кодів?

5.2. Перетворювач за логічними функціями

Логічним перетворювачем кодів називається перетворювач, що працює відповідно до таблиці істинності.

У загальному випадку логічний перетворювач кодів задається за допомогою логічних функцій його функціонування, як це показано в табл. 5.1.

Таблиця 5.1 – Логічний перетворювач

x_1	x_2	f_1	f_2	f_3
0	0	0	0	0
0	1	0	0	1
1	0	0	1	1
1	1	1	1	1

Маючи таку таблицю, легко одержати відповідні логічні функції f_1 , f_2 , f_3 , а далі з їх допомогою можна синтезувати відповідну цифрову схему перетворювача. Його перевага – велика швидкодія, одержана за рахунок комбінаційних схем, що реалізують логічні функції.

Контрольні запитання і завдання

1. Який перетворювач кодів називається табличним?
2. У чому полягають перевага і недолік табличного перетворювача кодів?
3. Синтезуйте за наведеною вище таблицею 5.1 відповідний перетворювач кодів.
4. Порахуйте кількість входів на одержаній схемі.

5.3. Перетворювач дешифратор – шифратор

Перетворювач кодів, в основу роботи якого покладено дешифрування і шифрування, називається перетворювачем кодів дешифратор – шифратор.

Головними чинниками схеми, що реалізує перетворення кодових слів, є зв'язки між дешифратором і шифратором. Це показано на прикладі перетворювача кодів на рис. 5.2. У ньому виходи дешифратора зв'язуються з його входами, як це подано у табл. 5.2.

*Таблиця 5.2 – Перетворювач
дешифратор – шифратор*

0	0
1	1
2	3
3	7

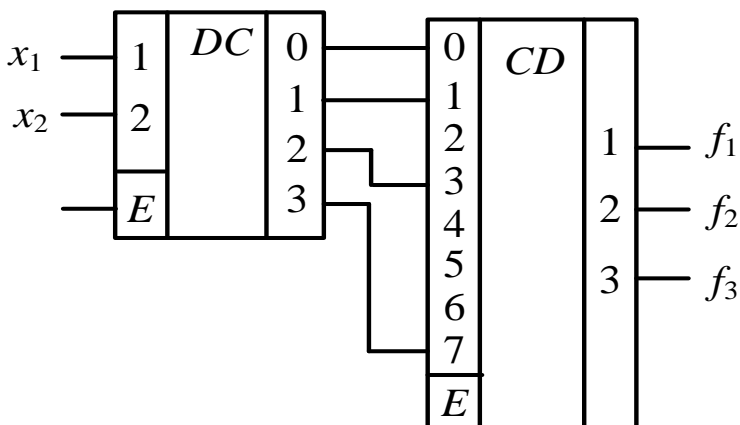


Рисунок 5.2 – Перетворювач коду

Кількість входів у цьому перетворювачі дорівнює кількості входів дешифратора, яких два, а кількість виходів – кількості виходів шифратора, тобто три.

Сам по собі дешифратор це є також перетворювач кодів, тому що перетворює вхідну комбінацію в сигнал на своєму виході, який можна розглядати як відповідну двійкову комбінацію з однією одиницею і рештою нулів. Також і шифратори відносять до перетворювачів кодів тому, що вони перетворюють одиницю на своєму вході на відповідну їй кодову комбінацію на своїх виходах.

Перетворювачі кодів створюють постійні запам'ятовувальні пристрої. У них у шифраторах розміщуються комірки пам'яті, а дешифратори використовують як елементи адресації до цих комірок. Такі пристрої сьогодні широко використовують у цифровій техніці, тому на них є великий попит.

Контрольні запитання і завдання

1. Який пристрій називається перетворювачем кодів дешифратор – шифратор?
2. У чому полягають переваги перетворювача кодів дешифратор – шифратор?
3. Опишіть за функціональною схемою роботу перетворювача кодів дешифратор – шифратор.
4. Яку кількість входів має перетворювач дешифратор – шифратор залежно від його розрядності? Порахуйте на конкретному прикладі.

ТЕМА 6. Цифрові комутатори

6.1. Мультиплексори

Мультиплексор – це пристрій, що здійснює перемикання одного із декількох його входів на вихід.

Завданням мультиплексора є під'єднання одного з його входів до виходу. Його стисла назва – *MS* або *MUX*. Залежно від сфери призначення мультиплексор ще називають комутатором, селектором – *SL*, селектором – мультиплексором.

У мультиплексорі є група інформаційних і група адресних входів, а також вхід, що дозволяє або забороняє роботу мультиплексора. Номер вибраного для під'єднання входу при цьому відповідає кодовій комбінації, яка подається на адресні входи мультиплексора.

Стандартне позначення мультиплексора з чотирма інформаційними D_0 – D_3 і двома A_1 , A_2 адресними входами наведено на рис. 6.1.

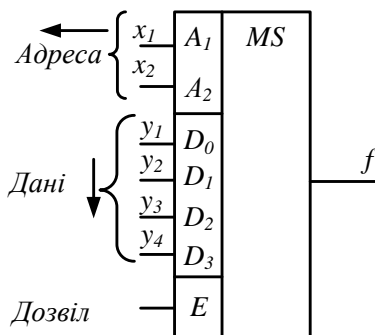


Рисунок 6.1 – Позначення мультиплексора

Структурна і функціональна схеми мультиплексора

Структурна схема мультиплексора подана на рис. 6.2. Вона містить дешифратор DC , що перетворює вхідну адресу на сигнал на одному із своїх 4 виходів. Цей сигнал дає дозвіл на проходження одного з 4 сигналів, які подаються на входи D_0-D_3 мультиплексора через одну з 4 двовходових схем I на схему $АБО$, що об'єднує всі 4 двовходові схеми I . Тому вибраний інформаційний сигнал, після того як він пройде схему I , з'явиться на виході схеми $АБО$ і далі за наявності сигналу дозволу E – на виході мультиплексора у вигляді сигналу f .

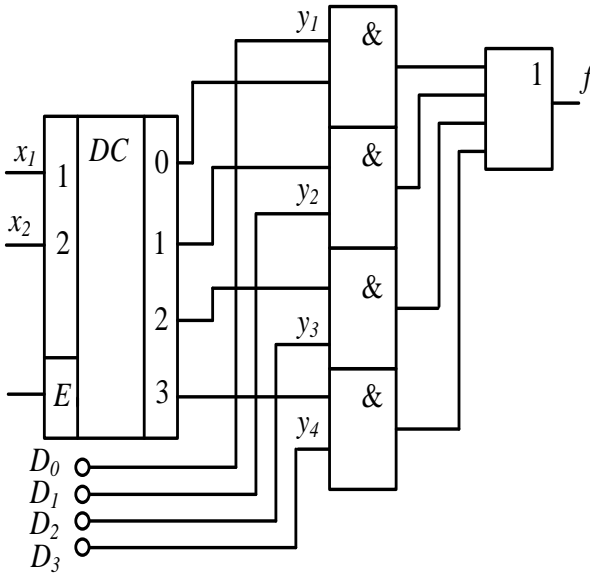


Рисунок 6.2 – Структурна схема мультиплексора

Функціональна схема мультиплексора на рис. 6.3 реалізує структурну схему, зображену на рис. 6.2. Її робота описується за допомогою логічної рівності

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

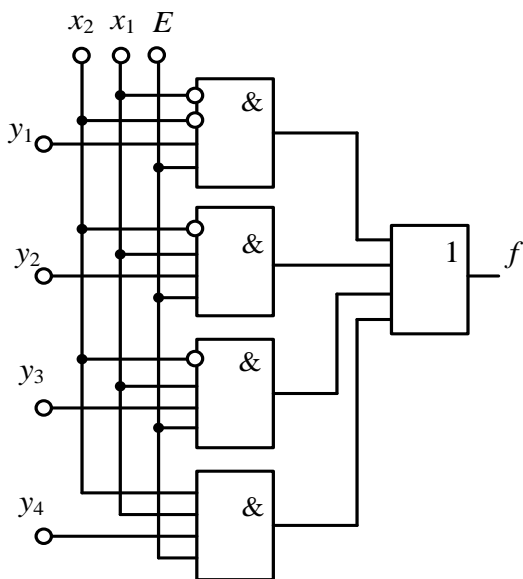


Рисунок 6.3 – Функціональна схема мультиплексора

Великого поширення дістав на практиці мультиплексор 2 в 1 (рис. 6.4). Він передає інформацію з одного із двох входів на один вихід.

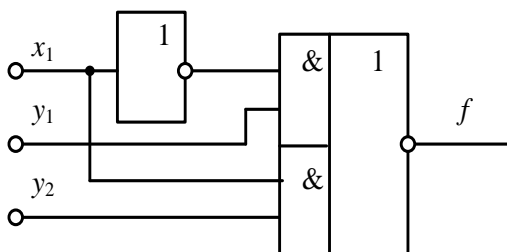


Рисунок 6.4 – Мультиплексор 2 в 1

Контрольні запитання і завдання

1. Який пристрій називається мультиплексором?
2. Яке завдання вирішує мультиплексор?
3. Опишіть роботу мультиплексора за його функціональною схемою.
4. Яку кількість входів має мультиплексор залежно від його розрядності? Порахуйте на конкретному прикладі.

6.2. Логічні схеми на мультиплексорах

Мультиплексор, що функціонально реалізує логічну функцію, називається логічною схемою на мультиплексорі.

Мультиплексор, так само як і дешифратор, може використовуватися як логічний елемент для реалізації будь-якої логічної функції з кількістю аргументів, що дорівнює кількості адресних входів мультиплексора, на яких функція дорівнює одиниці.

Побудуємо для прикладу мультиплексор, що реалізує логічну функцію, задану в табл. 6.1.

Таблиця 6.1 – Логічна функція

№ пор.	x_1	x_2	x_3	f
0	0	0	0	0
1	0	0	1	0
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	1

З цією метою інформаційні входи мультиплексора, номери яких збігаються з номерами наборів у табл. 6.1, де функція дорівнює одиниці, з'єднаємо з високим рівнем напруги, що відображає сигнал 1, а входи, які збігаються з наборами, де функція дорівнює нулю, – із землею, що відображає сигнал 0. Тоді при появі, наприклад адреси 011, спрацює третій вхід

мультиплексора, через який сигнал, що відповідає 1, буде поданий на його вихід.

Реалізація відповідної логічної функції на мультиплексорі наведена на рис. 6.5.

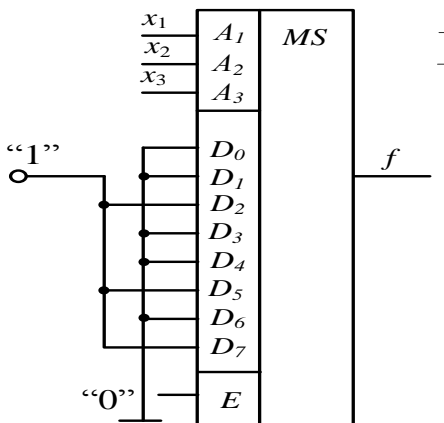


Рисунок 6.5 – Логічна схема на мультиплексорі

Контрольні запитання і завдання

1. Який пристрій називається логічною схемою на мультиплексорі?
2. Яке завдання вирішує логічна схема на мультиплексорі?
3. Опишіть роботу логічної схеми на мультиплексорі за її функціональною схемою.
4. Яку кількість адресних та інформаційних входів потребує мультиплексор під час реалізації на ньому логічної схеми з 4 змінними?

6.3. Каскадні мультиплектори

Мультиплектор, що складається з декількох каскадів, називається каскадним мультиплектором.

В окремих випадках кількість входів, що комутуються в мультиплекторі, перевищує кількість входів одного стандартного мультиплектора. Тоді на допомогу приходить каскадне вмикання декількох стандартних мультиплекторів і створення відповідної схеми їх комутації (див. рис. 6.6). При такому вмиканні інформаційні входи розбиваються на групи, кожна з яких заведена на свій мультиплектор першого каскаду. Отже, для цього випадку побудови мультиплектора половина адресних входів заводиться паралельно на всі мультиплектори першого каскаду, а інша половина – на мультиплектор другого каскаду. Тоді при появі сигналу E на всіх виходах мультиплекторів першого каскаду виникають сигнали, що відповідають інформаційним входам, номери яких задають адреси, що надходять на першу половину адресних входів. Ці сигнали далі проходять на інформаційні входи мультиплектора другого каскаду. Цей мультиплектор за допомогою другої половини адресних сигналів, що подаються на його адресні входи, комутує один з інформаційних входів на свій вихід. У результаті відбувається комутація одного з вхідних сигналів на вихід, що і є завданням мультиплектора.

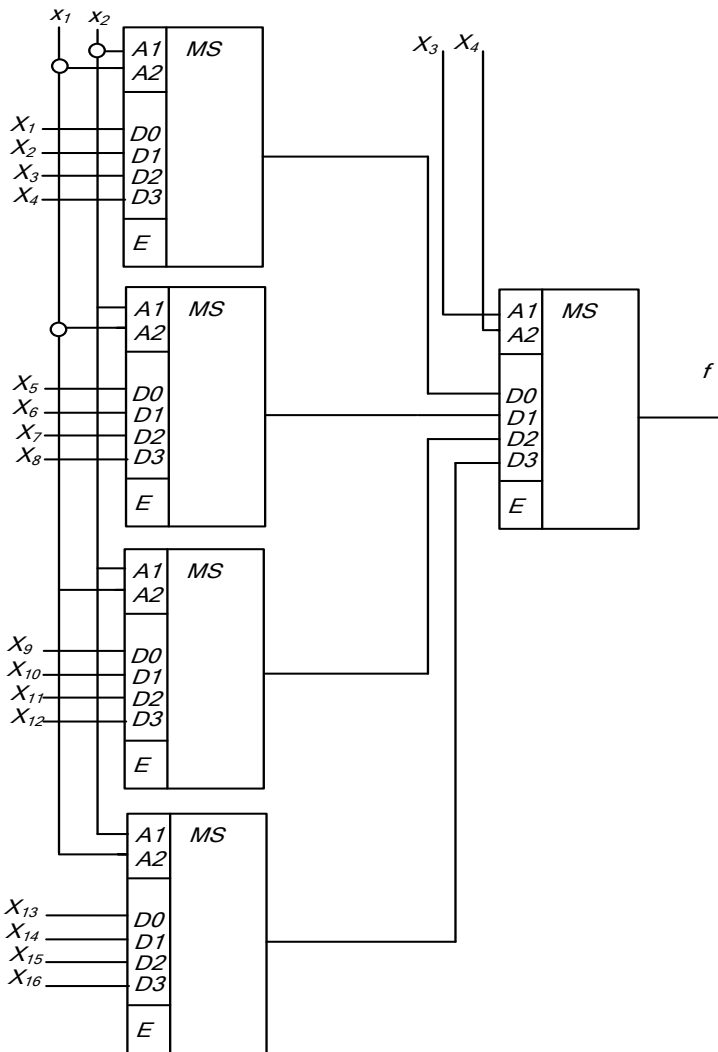


Рисунок 6.6 – Каскадне вмикання мультиплексора

Контрольні запитання і завдання

1. Який пристрій називається каскадним мультиплексором?
2. Яке завдання вирішує каскадний мультиплексор?
3. Опишіть роботу каскадного мультиплексора за функціональною схемою?
4. Яку кількість входів має каскадний мультиплексор залежно від його розрядності? Порахуйте на конкретному прикладі.

6.4. Демультиплексори

Схема, що реалізує завдання, зворотне до завдання мультиплексора, називається демультиплексором.

Позначення демультиплексора з двома адресними входами і одним інформаційним входом подано на рис. 6.7.

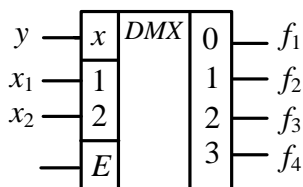


Рисунок 6.7 – Позначення демультиплексора

У цьому випадку один інформаційний вхід під дією двох адресних сигналів повинен перемикатися на один з чотирьох виходів.

Структурна схема демультиплексора

Структурна схема демультиплексора, що відповідає рис. 6.7, наведена на рис. 6.8. До її складу входять дешифратор і схеми I , кількість яких дорівнює кількості виходів дешифратора, тобто 4. У цьому разі 2 адресні сигнали подаються на входи дешифратора. Відповідно до них на одному з виходів дешифратора виникає сигнал, який вмикає ту чи іншу схему I , що дозволяє інформаційному сигналу пройти на відповідний вихід.

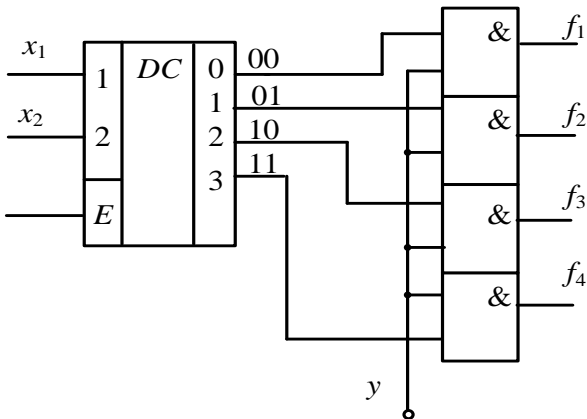


Рисунок 6.8 – Схема демультиплексора

Контрольні запитання і завдання

1. Який пристрій називається демультимплексором?
2. Як позначається демультимплексор?
3. Опишіть роботу демультимплексора за функціональною схемою?
4. Яку кількість входів має демультимплексор залежно від його розрядності? Порахуйте на конкретному прикладі.

ТЕМА 7. Комбінаційні суматори

7.1. Суматори чисел

Основні поняття

Суматором називається цифровий пристрій, який підсумовує два числа, що виникають на його входах.

Зазвичай суматори підсумовують двійкові числа. Їх поділяють на *однорозрядні*, які підсумовують цифри в одному розряді числа, і *багаторозрядні*, що підсумовують одночасно цифри декількох розрядів числа або всіх його розрядів. Останні мають підвищену швидкодію, але й потреби в апаратурі у них підвищені. Завдяки своїй підвищеній швидкодії вони на сьогодні найбільш поширені в цифровій техніці.

Однорозрядні суматори підсумовують тільки цифри одного розряду. Тому вони є складовими елементами багаторозрядних суматорів. Вони можуть бути з трьома входами, на два з яких подаються сигнали у вигляді 1 чи 0 з однойменних розрядів, а на третій – перенесення з попереднього розряду, і з двома входами без входу для перенесення. Останні ще називаються *півсуматорами*. Вони використовуються насамперед для побудови суматора нульового розряду багаторозрядних суматорів.

Суматори послідовної дії

У суматорах послідовної дії двійкові цифри числа подаються послідовно, починаючи з молодшого розряду. Результат на виході також видається в послідовному коді, починаючи з молодшого розряду. При цьому перенесення затримується на час підсумовування і подається на вхід суматора синхронно з парою вхідних сигналів наступного розряду (див. рис. 7.1).

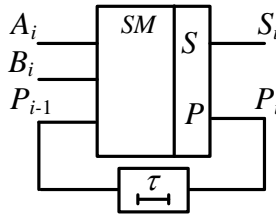


Рисунок 7.1– Схема суматора послідовної дії

Відповідно підсумовування буде проходити послідовно розряд за розрядом, починаючи з молодшого розряду, до того часу, поки не надійде останній старший розряд. На цьому підсумовування закінчується. Час складання багаторозрядних чисел буде не меншим суми затримок часу у всіх розрядах. Перевагою цього типу суматора є невеликі апаратурні витрати, потрібні на його реалізацію, а недоліком – низька швидкодія.

Суматори паралельної дії

Суматор паралельної дії оперує багаторозрядними числами і створюється однорозрядними суматорами, кількість яких відповідає розрядності чисел, що підсумовуються (рис. 7.2). Як однорозрядний суматор молодшого нульового розряду у цьому разі можна взяти півсуматор.

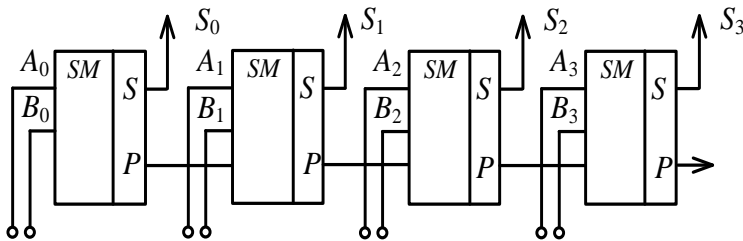


Рисунок 7.2 – Суматор паралельної дії

Суматор працює таким чином. На перші входи однорозрядних суматорів подаються цифри першого числа, що підсумовуються, і одночасно з ним на другі входи – другого. Підсумовування проходить одночасно у всіх однорозрядних суматорах, і тому результат виникає на виходах паралельно із затримкою на час перенесення між розрядами.

Контрольні запитання і завдання

1. Який пристрій називається суматором?
2. У чому виявляються переваги однорозрядного і багаторозрядного суматорів?
3. Опишіть роботу послідовного суматора за функціональною схемою.
4. Опишіть роботу багаторозрядного суматора за функціональною схемою.

7.2. Півсуматори

Півсуматором називається однорозрядний суматор із двома входами.

Він стоїть зазвичай в нульовому розряді багаторозрядних суматорів. Тому на нього не подається сигнал перенесення. Відповідно він менш складний, ніж повноцінний однорозрядний суматор, і, як правило, потребує для своєї реалізації менше апаратних витрат.

Функціональна схема півсуматора

Півсуматор позначається так, як це показано на рис. 7.3.

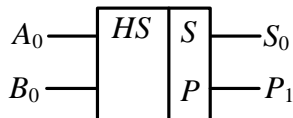


Рисунок 7.3 – Стандартне позначення півсуматора

Функціонування півсуматорів задається за допомогою таблиці істинності (табл. 7.1).

Таблиця 7.1 – Логіка роботи півсуматора

A_0	B_0	S_0	P_1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Їй відповідають дві функції:

$$S_0 = \bar{A}_0 B_0 + A_0 \bar{B}_0 = A_0 \oplus B_0,$$

$$P_0 = A_0 B_0.$$

Ці функції реалізовані на рис. 7.4 у вигляді відповідної схеми.

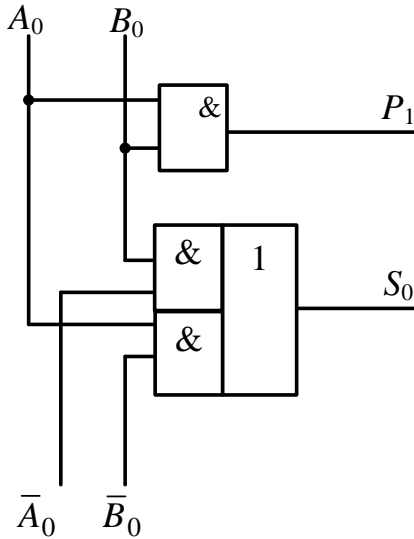


Рисунок 7.4 – Функціональна схема півсуматора

Спрощений півсуматор

Перетворимо вираз для півсуматора таким чином:

$$\begin{aligned}
 S_0 &= \bar{A}_0 B_0 + A_0 \bar{B}_0 = \bar{A}_0 B_0 + A_0 \bar{B}_0 + A_0 \bar{A}_0 + B_0 \bar{B}_0 = \\
 &= \bar{A}_0 (A_0 + B_0) + \bar{B}_0 (A_0 + B_0) = \\
 &= (\bar{A}_0 + \bar{B}_0) (A_0 + B_0) = (A_0 + B_0) \overline{A_0 B_0}, \\
 P_1 &= A_0 + B_0.
 \end{aligned}$$

За допомогою одержаного виразу побудуємо півсуматор, функціональна схема якого подана на рис. 7.5.

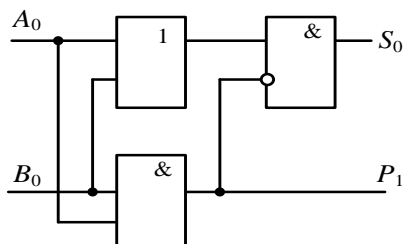


Рисунок 7.5 – Півсуматор після перетворення

Як бачимо, він має в собі менше елементів і входів, ніж суматор на рис. 7.4. Тому його необхідно використовувати тоді, коли потрібні економія апаратурних затрат і більш висока надійність. Швидкодія цих схем однакова. Схема півсуматора на рис. 7.4 використовується тоді, якщо вона будується на стандартних схемах без інверторів.

Контрольні запитання і завдання

1. Який пристрій називається півсуматором?
2. У чому полягають переваги півсуматорів на рис. 7.4 і 7.5?
3. Опишіть за функціональними схемами роботу півсуматорів на рис. 7.4 і 7.5.
4. Яку кількість входів мають півсуматори на рис. 7.4 і 7.5? Порахуйте на конкретному прикладі.

7.3. Однорозрядні суматори

Основні поняття

Суматор одного розряду з трьома входами називається однорозрядним суматором.

Його стандартне позначення подане на рис. 7.6.

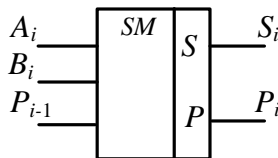


Рисунок 7.6 – Позначення однорозрядного суматора

Відповідно суматор із багатьма розрядами – багаторозрядний суматор – складається з ряду однорозрядних суматорів (див. рис. 7.2).

Функціональна схема

Однорозрядний суматор функціонує відповідно до таблиці 7.2.

Таблиця 7.2 – Робота суматора

A_i	B_i	P_{i-1}	S_i	P_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Згідно з цією таблицею маємо такі логічні функції, що відображають логіку функціонування однорозрядного суматора:

$$S_i = \bar{A}_i \bar{B}_i P_{i-1} + \bar{A}_i B_i \bar{P}_{i-1} + A_i \bar{B}_i \bar{P}_{i-1} + A_i B_i P_{i-1},$$

$$P_i = A_i B_i + A_i P_{i-1} + B_i P_{i-1}.$$

Відповідно до цих функцій суматор має функціональну схему, наведену на рис. 7.7.

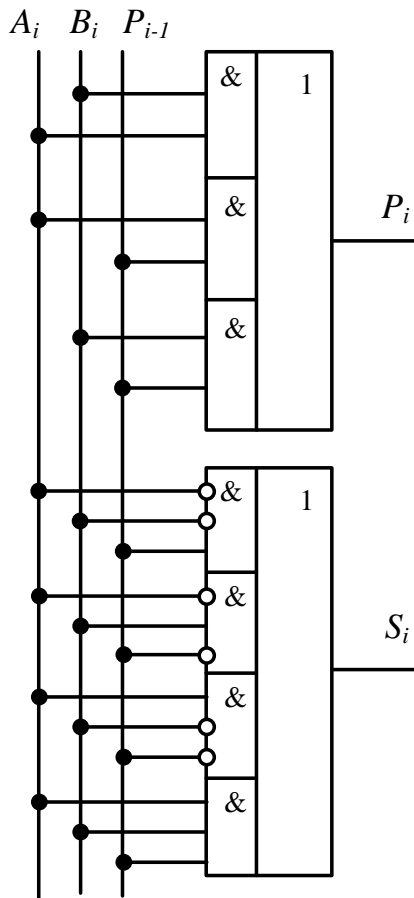


Рисунок 7.7 – Функціональна схема однорозрядного суматора

Однорозрядний суматор на двох півсуматорах

Можна створити також однорозрядний суматор за допомогою двох півсуматорів і однієї схеми АБО. Для цього побудуємо табл. 7.3.

Таблиця 7.3 – Робота суматора на двох півсуматорах

A_i	B_i	S'_i	P'_i	S'_i	P_{i-1}	S_i	P''_i
0	0	0	0	0	0	0	0
0	1	1	0	0	1	1	0
1	0	1	0	1	0	1	0
1	1	0	1	1	1	0	1

Відповідно до цієї таблиці побудуємо структурну схему однорозрядного суматора на двох півсуматорах (див. рис. 7.8).

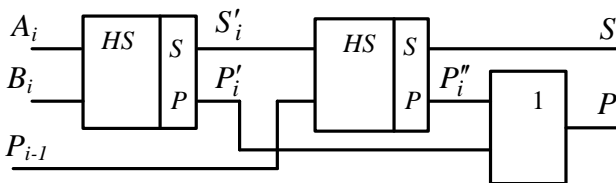


Рисунок 7.8 – Суматор на двох півсуматорах

Контрольні запитання і завдання

1. Який пристрій називається однорозрядним суматором?
2. Яке завдання вирішує однорозрядний суматор?
3. Опишіть роботу однорозрядного суматора за функціональною схемою на рис. 7.7.
4. Опишіть роботу однорозрядного суматора за функціональною схемою на рис. 7.8.

7.4. Суматори одиниць

Суматори, що обчислюють кількість одиниць в двійкових комбінаціях, називаються суматорами одиниць.

Вони розв'язують задачу підрахунку кількості одиниць в двійкових комбінаціях, тобто обчислюють їх вагу. Ця задача розв'язується в спеціальних цифрових пристроях, наприклад, біноміальних, що використовуються для лічби, стиснення і захисту інформації. Також підрахунок кількості одиниць використовують при побудові кодувальних і декодувальних пристроїв під час завадостійкого кодування, наприклад рівноважного.

Вище суматори одиниць використовували в підсумовувальних дешифраторах. Вони можуть бути комбінаційними та з елементами пам'яті. Однак найбільш прості і швидкодіючі суматори одиниць одержують при побудові на матричних суматорах. Їх

особливістю є те, що їх складають з однорозрядних матричних суматорів.

Однорозрядний матричний суматор

Комбінаційні матричні дешифратори можуть бути побудовані на матричних дешифраторах. При цьому використовується матрична арифметика, як це показано у табл. 7.4.

Таблиця 7.4 – Додавання

+	0	1
0	0	1
1	1	2

На рисунку 7.9 показана реалізація матриці додавання двійкових цифр у вигляді цифрової схеми:

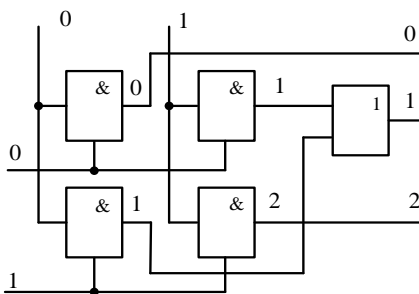


Рисунок 7.9 – Схема реалізації матриці додавання

Схеми *I* дешифратора при цьому виконують роль матричного дешифратора, а схема *АБО* перетворює його на матричний суматор.

Багаторозрядний матричний суматор

На рисунку 7.10 показана схема багаторозрядного суматора одиниць двійкового числа, реалізована на однорозрядних матричних суматорах. У такому суматорі стоять матричні суматори, подібні до показаного на рис. 7.9. Останній підсумовує в старшому розряді дві змінні зі значеннями 0 або 1 і відповідно має три виходи – 0, 1, 2, на одному з яких виникає сигнал. Далі стоять матричні суматори, на один із входів яких подається одна двійкова змінна, що теж набуває значення 0 або 1, а кількість виходів на молодших суматорах поступово збільшується у цьому разі від 3 до 5. У кінці стоїть шифратор, який перетворює сигнал на одному з виходів останнього однорозрядного суматора на відповідну кодову комбінацію.

Змінні, про які йде мова, реалізуються просторово на входах матричного суматора, як це показано на рис. 7.9. Якщо на одному із двох входів є сигнал, то на іншому вході його не буде. Тобто поява на двох входах матричного суматора одночасно двох сигналів 11 заборонена.

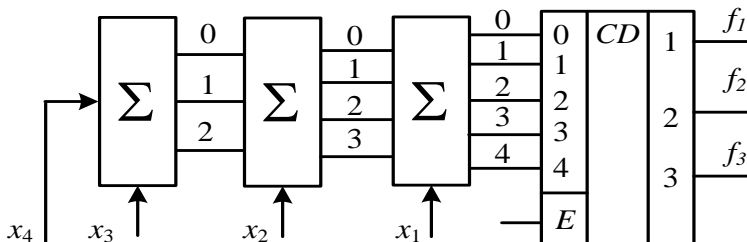


Рисунок 7.10 – Структурна схема суматора одиниць

Контрольні запитання і завдання

1. Який пристрій називається однорозрядним матричним суматором?
2. Яке завдання вирішує однорозрядний матричний суматор?
3. Опишіть роботу однорозрядного матричного суматора за функціональною схемою на рис. 7.9.
4. Опишіть роботу матричного суматора одиниць за структурною схемою на рис. 7.10.

ТЕМА 8. Схеми порівняння

8.1. Схеми порівняння цифр на рівність

Схема, що порівнює цифри на рівність, називається схемою порівняння цифр.

На практиці задача порівняння цифр відповідних розрядів двійкових чисел A і B розв'язується так, як це впливає з табл. 8.1.

Таблиця 8.1 – Рівність цифр

$a_i \ b_i$	\bar{F}_i	F_i
0 0	0	1
0 1	1	0
1 0	1	0
1 1	0	1

Вона показує, що логічна функція під час порівняння значень a_i, b_i одного розряду двох двійкових чисел A і B на рівність, буде мати такий вигляд:

$$F_i = \bar{a}_i \bar{b}_i + a_i b_i,$$

або

$$F_i = \overline{\bar{a}_i b_i + a_i \bar{b}_i}.$$

Тобто порівняння цифр двох чисел на рівність в одному розряді відбувається за допомогою функції рівнозначності, або інверсії відповідної їй функції за модулем 2.

Контрольні запитання і завдання

1. Який пристрій називається схемою порівняння цифр на рівність?
2. Яку задачу розв'язує схема порівняння цифр на рівність?
3. Опишіть логічну функцію схеми порівняння цифр на рівність.
4. Побудуйте за логічною функцією схему порівняння цифр на рівність.

8.2. Схеми порівняння чисел на рівність

Пристрій, що порівнює числа на рівність, називається схемою порівняння чисел на рівність.

Стандартне позначення схеми порівняння на рівність двох чисел A і B із чотирма розрядами наведено на рис. 8.1.

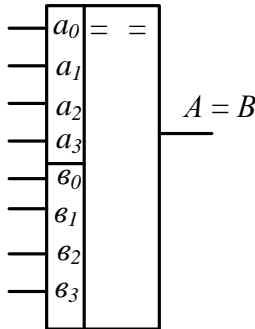


Рисунок 8.1 – Схема порівняння на рівність двох чисел

Алгоритм на рівність двох двійкових чисел A і B складається із операцій порівняння всіх однойменних їх розрядів. Їх рівність буде тоді, коли цифри всіх їх розрядів будуть однакові. У цьому випадку ознакою рівності буде значення одиниці логічної функції

$$R = F_0 F_1, \dots, F_i, \dots, F_{n-1},$$

де $F_0, F_1, \dots, F_i, \dots, F_{n-1}$ – логічні функції порівняння двох цифр в одному розряді числа. Відповідно

$$F_i = \bar{a}_i \bar{b}_i + a_i b_i.$$

Для чисел із двома розрядами

$$R = F_0 F_1.$$

Схема, яка реалізує цю функцію для нульового і першого розрядів, показана на рис. 8.2. На цій схемі поява нуля буде свідчити, що порівнювані числа рівні, а одиниці – нерівні.

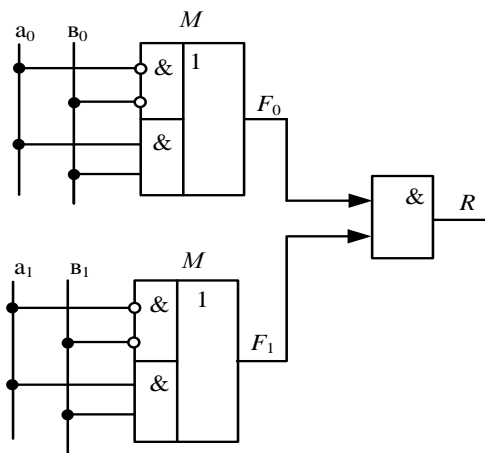


Рисунок 8.2 – Порівняння на рівність
двох двійкових чисел

Контрольні запитання і завдання

1. Який пристрій називається схемою порівняння чисел на рівність?
2. Яку задачу розв'язує схема порівняння чисел на рівність?
3. Опишіть за функціональною схемою на рис. 8.2 роботу схеми порівняння двох двійкових чисел на рівність.
4. Як позначається схема порівняння чисел на рівність?

8.3. Схеми порівняння цифр на нерівність

Пристрій, що порівнює цифри на нерівність, називається схемою порівняння цифр на нерівність.

Порівняння на нерівність в одному розряді двох двійкових цифр a_i , b_i чисел A і B виконується відповідно до табл. 8.2.

Таблиця 8.2 – Порівняння цифр на нерівність

a_i	b_i	$F_{a_i > b_i}$	$F_{a_i < b_i}$
0	0	0	0
0	1	0	1
1	0	1	0
1	1	0	0

У цій таблиці кожна цифра a_i , b_i чисел A і B може в i -му розряді набути одного з двох значень – 1 або 0, а логічні функції $F_{a_i > b_i}$, $F_{a_i < b_i}$ одержують при цьому відповідні результати порівняння цих значень. Ці функції для одного розряду мають таку форму:

$$F_{a_i > b_i} = a_i \bar{b}_i,$$

$$F_{a_i < b_i} = \bar{a}_i b_i.$$

Щоб побудувати схему для реалізації функції нерівності двох багаторозрядних двійкових чисел A і B , спочатку одержимо схему для порівняння цифр на нерівність і відповідно рівність їх однойменних розрядів. Для цього проводимо логічне перетворення двох вищезазначених логічних функцій на нерівність для одного розряду $\bar{a}_i b_i$ таким чином:

$$\bar{a}_i b_i = \bar{a}_i a_i + \bar{a}_i b_i = \bar{a}_i (a_i + b_i) = \overline{a_i + (a_i + b_i)}.$$

Відповідно одержуємо

$$a_i \bar{b}_i = \overline{b_i + (a_i + b_i)}.$$

Аналогічно одержуємо перетворення і для виразу $a_i \bar{b}_i$:

$$a_i \bar{b}_i = b_i \bar{b}_i + a_i \bar{b}_i = \bar{b}_i (a_i + b_i) = \overline{b_i + (a_i + b_i)}.$$

Використаємо ці перетворені функції для побудови схеми та для порівняння цифр a_i , b_i чисел A і B в одному розряді. Відповідна схема буде такою, як це показано на рис. 8.3.

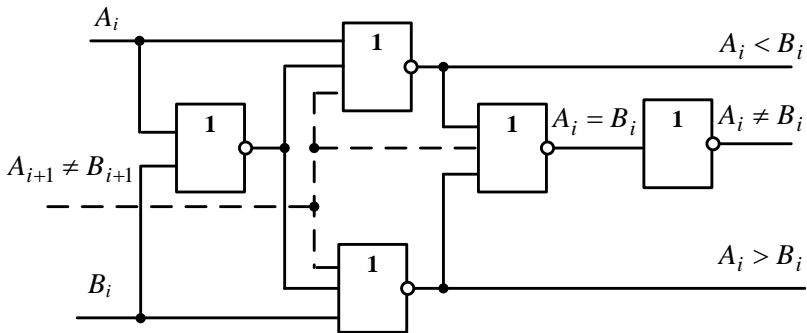


Рисунок 8.3 – Функціональна схема на нерівність в одному розряді

З цієї схеми можна одержати сигнали на нерівність цифр $a_i < b_i$ і $a_i > b_i$. Дійсно, з цього випливає, якщо $a_i = 1$, $a_i > b_i = 0$, і на пунктирний вхід надходить 0, то результуючий сигнал з'явиться на інверсному виході нижньої схеми АБО, тому що на всі три її входи на цей час будуть подаватися нулі. Відповідно на її виході буде надана інформація, що $a_i > b_i$. На інверсному виході верхньої схеми АБО на цей час сигналу не буде, тому що на один із її трьох входів подається сигнал $a_i = 1$. При зворотній подачі сигналів, тобто $a_i = 0$ і $b_i = 1$, сигнал 1 відповідно з'явиться зверху, а знизу буде 0. Тобто надійде інформація, що $a_i < b_i$.

Якщо ж $a_i = b_i = 1$, то тоді з'явиться одиниця на інверсному виході першої середньої схеми АБО. Вона подається через верхню і нижню схеми АБО на

відповідні інвертори, з виходів яких у вигляді нулів потрапляє на другу середню схему АБО і на її інверсному виході знову з'являється в вигляді 1. Це сигналізує, що цифри на вході схеми рівні. Одночасно на виході інвертора, що йде після другої середньої схеми АБО, з'явиться сигнал нерівності 0.

За рівності $a_i = b_i = 0$ на інверсному виході першої середньої схеми АБО з'явиться 1, що приведе до появи нулів на інверсних виходах верхньої та нижньої схем АБО. Ці нулі приводять до появи 1 на інверсному виході другої середньої схеми АБО, символізуючи тим самим рівність порівнюваних цифр, і одночасно ця 1 перетворюється на 0 інвертором, що дає інформацію про рівність цифр молодшому розряду, деблокуючи тим самим його роботу. Ця інформація передається на вхід відповідної схеми порівняння, виділений штриховою лінією.

Контрольні запитання і завдання

1. Який пристрій називається схемою порівняння цифр на нерівність?
2. Яку задачу розв'язує схема порівняння цифр на нерівність?
3. Опишіть роботу схеми порівняння цифр.
4. Напишіть логічні функції, що виконують операції порівняння двох цифр на нерівність.

8.4. Схеми порівняння чисел на нерівність

Пристрій, що порівнює числа на нерівність, називається схемою порівняння чисел на нерівність.

В основу порівняння чисел на нерівність покладені функції їх порівняння. Для двох чисел A і B вони будуть мати такий вигляд:

$$F_{A > B} = a_{n-1} \bar{b}_{n-1} + a_{n-2} \bar{b}_{n-2} r_{n-1} + \dots + a_0 \bar{b}_0 r_1, \dots, r_{n-1},$$

$$F_{A < B} = \bar{a}_{n-1} b_{n-1} + \bar{a}_{n-2} b_{n-2} r_{n-1} + \dots + \bar{a}_0 b_0 r_1, \dots, r_{n-1},$$

де $r_i = a_i b_i + \bar{a}_i \bar{b}_i$ — ознака рівності цифр i -го розряду, $i = 0, 1, 2, \dots, n-1$.

Із цих функцій випливає, що наявність однакових цифр в i -му розряді приведе до рівності $r_i = 1$, а це повинно відповідно до логіки порівняння дозволити далі проводити аналіз цифр молодших розрядів. Однак як тільки цифри певного розряду стають різними, порівняння чисел повинно закінчитися, і тоді одна з наведених вище формул дасть відповідь, яке з порівнюваних чисел — A і B — буде більшим, а яке — меншим.

Числа порівнюються порозрядно, починаючи з старшого розряду. Порівняння відбувається до появи в якому-небудь розряді різних цифр — або 0, або 1. Після цього фіксується відповідна нерівність цифр: або $a_i > b_i$, або $a_i < b_i$, і порівняння чисел на цьому

закінчується. Якщо ж до останнього нульового розряду не було одержано сигналів про нерівність, то це означає, що порівнювані числа однакові.

Реалізація наведених вище формул можлива за допомогою схеми, наведеної на рис. 8.3. Кількість таких схем дорівнює кількості розрядів у числах, що порівнюються. Їх виходи послідовно з'єднуються між собою за входами зі штриховою лінією. Наявність на цих входах одиниць блокує роботу схем порівняння молодших розрядів. Це означає, якщо в одному із старших розрядів порівнюваних чисел виявлена нерівність цифр, то на цьому процедура порівняння чисел повинна закінчитися. Відповідь на запитання, яке з двох чисел більше, а яке менше, дасть схема того розряду порівнюваних чисел, де одержана нерівність.

Контрольні запитання і завдання

1. Який пристрій називається схемою порівняння чисел на нерівність?
2. Яку задачу розв'язує схема порівняння чисел на нерівність?
3. Опишіть роботу схеми порівняння чисел.
4. Напишіть логічні функції, що виконують операції порівняння двох чисел на нерівність.
5. Опишіть алгоритм роботи порівняння чисел на нерівність.

ТЕМА 9. Тригери

9.1. Загальні поняття про тригери

Тригером називається елементарний цифровий автомат з пам'яттю, що має два стійких стани – нульовий і одиничний, в які він може переходити під дією вхідних сигналів.

У загальному випадку входів у тригера може бути один і більше – до п'яти, а виходів всього два – *прямий* та *інверсний*. Прямий вихід тригера будемо ще називати *одиничним*, а інверсний – *нульовим*. Характерною властивістю тригера є те, що після переходу в інший стан він залишається в ньому як завгодно довго і після того, як вхідний сигнал зникне. Цим тригер відрізняється від комбінаційних схем, в яких сигнал на виході тримається до того часу, поки є сигнал на вході.

Тригери – це особливі елементи цифрових схем, що мають фундаментальне значення для побудови всієї цифрової техніки. Без них не може обійтися більш-менш складна цифрова схема, тому що на основі тригерів будуються регістри, лічильники, оперативна пам'ять, суматори і багато інших подібних цифрових пристроїв.

Кожному стану тригера відповідають двійкові сигнали на його прямому та інверсному виходах: одиничному стану відповідають 1 на прямому виході і нуль на інверсному виході, нульовому стану тригера

відповідають зворотні сигнали – 0 на прямому і 1 на інверсному виході. Стани тригера з сигналами на його виходах 00 і 11 заборонені.

Узагальнена схема тригера

На рис. 9.1 наведена *узагальнена* структурна схема тригера, де *ЗЕ* – запам'ятовувальний елемент (елементарний тригер); *Q* – прямий (одичний) і \bar{Q} – інверсний (нульовий) виходи *тригера*; *ПК* – пристрій керування; A_1, \dots, A_n – інформаційні входи тригера; *C* – тактовий (синхронний) вхід. Інформаційні входи тригера відрізняються від синхронного входу тим, що на інформаційні входи сигнали надходять в будь-які моменти часу, а на синхронні – лише у визначені моменти. Якраз у ці моменти і можлива дія інформаційних сигналів. Із виходів тригера на *ПК* заведені ланцюги зворотного зв'язку, за допомогою яких передаються на *ПК* сигнали про стан, в якому знаходиться тригер.

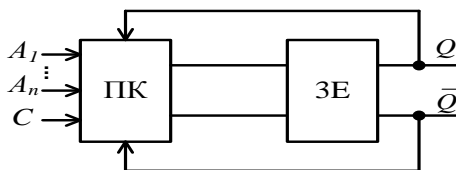


Рисунок 9.1 – Узагальнена структурна схема тригера

Контрольні запитання і завдання

1. Який пристрій називають тригером?
2. Що таке стан тригера? Скільки дозволених і заборонених станів є у тригера? Чим вони відрізняються між собою? Назвіть ці стани.
3. Скільки входів і виходів є у тригера? Чим вони відрізняються між собою і як називаються?
4. Яка особливість роботи тригера відрізняє його від комбінаційних схем?
5. Наведіть приклади застосування тригерів у цифрових пристроях.
6. Опишіть роботу узагальненої структури тригера.
7. Яка функція в узагальненому тригері покладається на *ПК*, а яка – на *ЗЕ*?

9.2. Типи тригерів

Асинхронні та синхронні тригери

Тригери класифікують за способами запису інформації на *асинхронні* та *синхронні*.

Асинхронним тригером називають тригер, в якому запис інформації здійснюється в будь-який момент часу.

Синхронним тригером називають тригер, в якому запис інформації проходить під час дії тактового імпульсу (ТІ).

Тактовий імпульс у синхронному тригері подається на вхід *С*. Тоді *ПК* пропускає сигнали на входи тригера *T* без перешкод. Коли немає синхронізувального сигналу

на вході С, вхідні сигнали не можуть потрапити на входи ЗЕ.

Прозорі та непрозорі тригери

Усі тригери можна поділити на два класи – *прозорі та непрозорі*.

Прозорим тригером називають тригер, в якому перехід до нового стану може проходити через заборонені стани, і відповідно ці стани відображаються на їх виходах.

Цей процес негативно впливає на роботу цифрового пристрою, де використовується такий тригер. Тому для більш стабільної та надійної роботи цифрових пристроїв пропонуються різні схеми тригерів, які б не дозволяли знімати з них інформацію під час зміни їх стану, що ускладнює роботу тригера та його загальну схему.

Тригер, що видає інформацію про свій стан лише в момент його стабільного перебування в одному з дозволених станів, називають непрозорим.

Це на сьогодні найбільш використовувані схеми тригерів, хоча вони і значно складніші за схеми прозорих тригерів.

Контрольні запитання і завдання

1. Наведіть класифікацію тригерів.
2. Зазначте принципову відмінність синхронних тригерів від асинхронних.
3. Які тригери називають прозорими, а які – непрозорими?

4. Чому непрозорі тригери складніші за прозорі тригери?

9.3. Асинхронні RS-тригери

Елементарним тригером (запам'ятовувальним елементом) називається тригер, побудований на основі двох логічних елементів І чи АБО і відповідно має два виходи – прямий (одичинний) Q та інверсний (нульовий) \bar{Q} і два входи: R (RESET) – вхід, за яким тригер устанавлюється в нульовий стан, і S (SET) – вхід устанавлення тригера в одичинний стан. Такий тригер ще називається елементарним RS-тригером.

Такий елементарний RS-тригер називають ще одичинчастим RS-тригером. Його стандартне позначення показано на рис. 9.2.

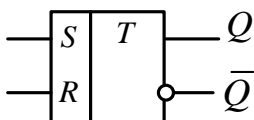


Рисунок 9.2 – Позначення елементарного RS-тригера

У цих тригерах відсутній *ПК* і тому їх характерною ознакою є те, що вони можуть змінювати свій стан в будь-який момент часу. Тобто вони є асинхронними тригерами.

Асинхронний *RS*-тригер на схемах *АБО*

Його функціональна схема наведена на рис. 9.3.

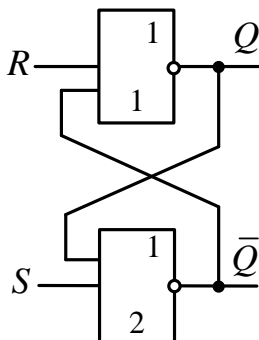


Рисунок 9.3 – Елементарний *RS*-тригер на схемах *АБО*

Ця схема функціонує відповідно до табл. 9.1.

Таблиця 9.1 – Робота RS-тригера
на схемах АБО

S	R	$Q_{(t+1)}$
0	0	Q_t
0	1	0
1	0	1
1	1	-

Працює тригер таким чином. На вхід S -тригера подається одиничний сигнал, який інвертується відповідною другою схемою АБО, тобто її вихідний сигнал набуває значення 0 і в такому вигляді надходить на першу схему АБО, на яку в цей час подається з нульового входу R сигнал, що дорівнює також нулю. Два нульових сигнали на входах цієї схеми АБО створюють на її інверсному виході 1. У результаті на прямому виході тригера буде 1, а на інверсному – 0. Такий стан тригера буде *одиничним* і позначається у відповідному стовпчику табл. 9.1 як 1.

Якщо ж на вхід R подається одиничний сигнал, а на вхід S – нульовий, то одиничний сигнал інвертується на відповідній йому першій схемі АБО і в інвертованому вигляді, тобто у вигляді 0, проходить на перший вхід

другої схеми АБО, на другий вхід якої на цей час за правилами функціонування тригера подається 0. У результаті на обох входах цієї схеми з'являються нулі і відповідно на її виході буде 1. Тобто на прямому виході тригера буде 0, а на інверсному – 1. Цей стан вже буде нульовим і позначається у відповідному стовпчику табл. 9.1 як 0.

Якщо ж сигнали на входи тригера не надходять, тобто на цих входах знаходяться нулі, то тригер буде перебувати в незмінному стані. Подання ж двох одиничних сигналів на входи S і R забороняється.

Відповідно до розглянутої процедури установалення тригерів у стани 1 і 0 показані наочно за допомогою часової діаграми на рис. 9.4.

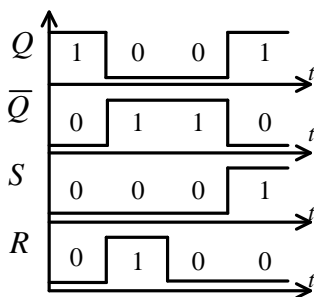


Рисунок 9.4 – Часова діаграма елементарного RS-тригера на схемах АБО

Асинхронний *RS*-тригер на схемах *I*

Функціональна схема тригера на схемах *I* показана на рис. 9.5.

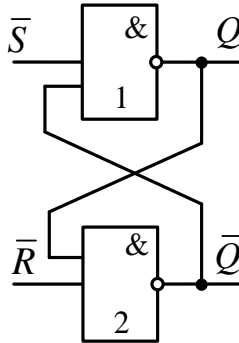


Рисунок 9.5 – Функціональна схема елементарного *RS*-тригера на схемах *I*

Вона керується нулями, що у цьому разі будуть активними сигналами, а одиниці відповідно – пасивними сигналами. Ця особливість інформаційних сигналів для такого випадку позначається на схемі знаками інверсії над символами S і R . Функціонує схема відповідно до табл. 9.2.

Таблиця 9.2 – Робота RS-тригера на схемах I

\bar{S}	\bar{R}	$Q_{(t+1)}$
0	0	–
0	1	1
1	0	0
1	1	Q_t

Із рисунка 9.5 бачимо, якщо на входи тригера надійдуть одиниці, то тригер залишиться у попередньому стані. Якщо ж на входи однієї зі схем I надійде 0, то після інвертування на її виході з'явиться 1. Ця 1 подається на один із входів другої схеми I і сумісно з 1, яка вже є на її іншому вході, встановить її в нуль. Відповідно буде одержаний прямий чи інверсний стан тригера. Часова діаграма роботи цього тригера наведена на рис. 9.6.

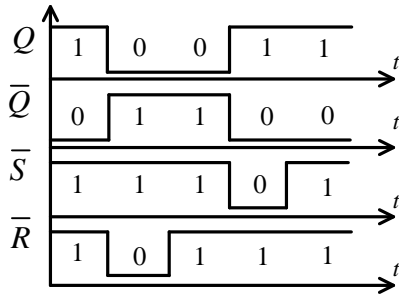


Рисунок 9.6 – Часова діаграма елементарного RS -тригера на схемах I

Контрольні запитання і завдання

1. Наведіть класифікацію елементарних тригерів.
2. Зазначте принципову відмінність елементарного тригера на схемах $АВО$ від тригера на схемах I .
3. Як позначається елементарний тригер?
4. Поясніть роботу елементарного RS -тригера на схемах I за часовою діаграмою.
5. Поясніть роботу елементарного RS -тригера на схемах $АВО$ за часовою діаграмою.
6. Поясніть роботу елементарного RS -тригера за його таблицею функціонування.
7. Скільки входів і виходів має елементарний тригер?
8. Які сигнали в тригерах є пасивними, а які – активними?

9.4. Синхронні *RS*-тригери

Розглянуті вище елементарні тригери відносять до *асинхронних*, тому що вхідні сигнали можуть діяти на них в будь-які моменти часу. Якщо ж потрібно щоб сигнали діяли на ці тригери лише у фіксовані моменти часу, то тоді необхідно вводити додатково до них *ПК*. Цей пристрій сумісно з елементарним тригером створює більш складну схему тригера – *синхронний RS*-тригер.

Особливістю синхронних тригерів є те, що в них на відміну від асинхронних тригерів з'являється синхронізувальний вхід. Наявність синхронізувального входу в синхронних *RS*-тригерах дозволяє відсікати несталі процеси на їх загальних входах від входів елементарних тригерів, що входять до їх складу, оскільки за відсутності синхронізувального сигналу вони не реагують ні на які вхідні сигнали. Це дозволяє тригеру працювати більш надійно. Тому такі тригери частіше використовують на практиці.

Схеми синхронних *RS*-тригерів

Синхронний *RS*-тригер може використовувати елементарний тригер або на схемах *I* (див. рис. 9.7), або на схемах *АБО* (див. рис. 9.8). До цих пристроїв додається на рис. 9.7 пристрій керування на двох схемах *I* з інверсією для елементарного тригера на схемах *I* і на рис. 9.8 – без інверсії для елементарного тригера на схемах *АБО*.

На відміну від асинхронних тригерів із *I-HI* синхронні тригери завжди працюють лише в прямих кодах, тобто під час встановлення тригера в нульовий стан на вхід його установа в одиницю R подається 1, а на вхід установа в нуль S – 0, а при встановленні тригера в одиничний стан вхідні сигнали подаються зворотно – на вхід S подається 1, а на вхід R – 0.

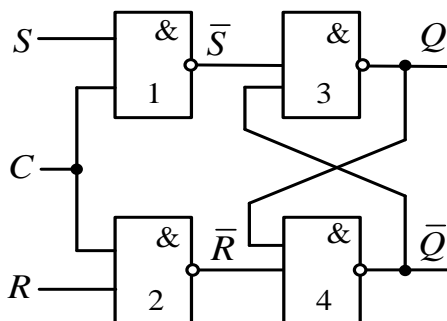


Рисунок 9.7 – Синхронний RS-тригер на схемах *I-HI*

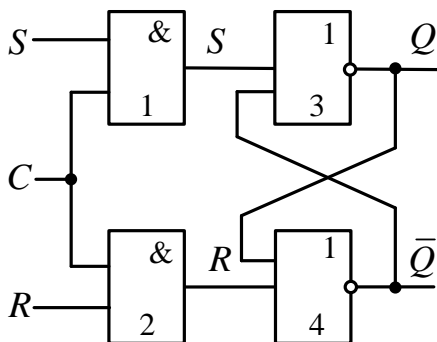


Рисунок 9.8 – Синхронний RS-тригер на схемах *АБО-НІ*

Керуючі сигнали можуть виникнути на входах схем I пристрою керування в будь-який час, але на входах елементарного тригера виникають лише за наявності тактового імпульсу. За його відсутності тригери можуть бути установлені в стан 1 чи 0 тільки за допомогою асинхронних входів Ra і Sa .

Стандартне позначення такого синхронного тригера з асинхронними входами наведено на рис. 9.9.

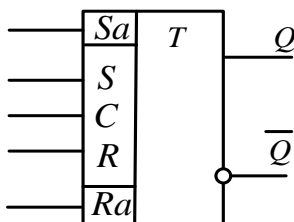


Рисунок 9.9 – Позначення асинхронного RS -тригера

Прозорість схем синхронних RS -тригерів

Характерною особливістю синхронних RS -тригерів є те, що впродовж всього часу, поки є синхронізувальний сигнал, зміни інформаційних сигналів на їх входах можуть подіяти на елементарні тригери таким чином, що на їх виходах можуть виникнути на певний час заборонені стани. Тому така схема буде *прозорою* за S - і R -входами при $C = 1$. Це є основним недоліком розглядуваних RS -тригерів, хоча для їх реалізації потрібні мінімальні апаратні затрати.

Заборонені стани виникають як наслідок того, що час затримки сигналу схемами $АБО$ чи I в елементарних

тригерах може бути різним. Тому якщо на одному виході елементарного тригера виникає сигнал, то на іншому він ще може триматися деякий час. Це призведе до того, що на виходах тригера з'являться або два 0, або дві 1, тобто у тригері у цей час виникнуть заборонені стани і він буде працювати неправильно. Таке явище, коли сигнали на виходах тригерів змінюються з різною швидкістю, називають *гонками*. Ці гонки тригерів дуже небезпечні для правильної роботи цифрових схем, які їх використовують. Тому на практиці завжди ставиться завдання уникнення гонок.

Необхідно додати, що гонки взагалі – це поширене явище, яке відносять не тільки до цифрових схем з пам'яттю.

Комбінаційні схеми, наприклад дешифратори, теж можуть мати гонки, коли певний час на двох їх виходах виникають одиниці або на всіх виходах вони зникають.

Контрольні запитання і завдання

1. Як працює синхронний *RS*-тригер на схемах *АБО* ? Накресліть його часову діаграму і поясніть її.
2. Як працює синхронний *RS*-тригер на схемах *I* ? Зобразіть його часову діаграму і поясніть її.
3. Розкажіть про особливості роботи синхронних тригерів із асинхронними входами. Визначте їх переваги і недоліки.
4. Чому синхронні тригери під час дії синхронізувальних сигналів стають прозорими?

ТЕМА 10. Одноступінчасті тригери

Розглянуті вище елементарні синхронні та асинхронні тригери відносять до одноступінчастих. Крім того, є і більш надійні двоступінчасті тригери, які на сьогодні дістали широкого застосування на практиці. Однак спочатку розглянемо одноступінчасті тригери як більш прості. Це означає, що в кожному з них є всього один елементарний *RS*-тригер та його схема керування.

Крім синхронних *RS*-тригерів, на практиці широко використовують одноступінчасті *D*-, *DV*- і *T*- тригери, які мають інші схеми реалізації, ніж *RS*-тригери, що надає їм нових можливостей. Розглянемо кожен із цих типів тригерів детальніше.

10.1 Одноступінчасті *D*-тригери

Тригер з одним інформаційним і одним синхронізувальним входом називається D-тригером.

Функціональна схема *D*-тригера показана на рис. 10.1, а у табл. 10.1 поданий алгоритм його функціонування.

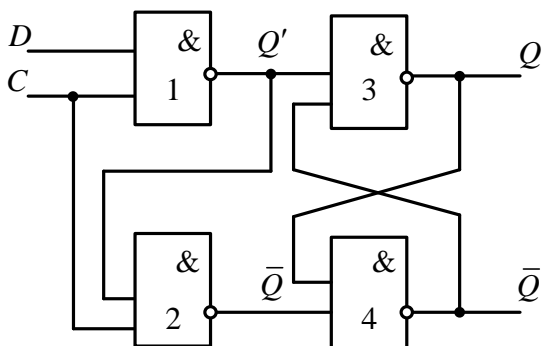


Рисунок 10.1 – Функціональна схема D -тригера

Таблиця 10.1 – Логіка роботи D -тригера

D	$Q_{(t+1)}$
0	0
1	1

Стандартне позначення D -тригера подане на рис. 10.2, а часова діаграма його роботи – на рис. 10.3.

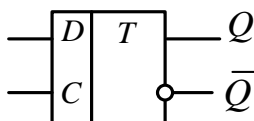


Рисунок 10.2 – Стандартне позначення D -тригера

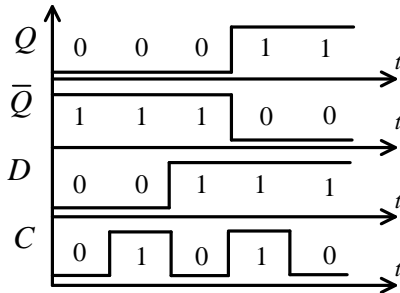


Рисунок 10.3 – Часова діаграма роботи D -тригера

Контрольні запитання і завдання

1. Як працює одноступінчастий D -тригер? Нарисуйте часову діаграму його роботи та поясніть її.

2. Розкажіть про особливості роботи одноступінчастих D -тригерів. Визначте їх переваги і недоліки.

4. Дайте визначення D -тригера.

5. Скільки входів і виходів має D -тригер?

6. Опишіть роботу функціональної схеми D -тригера.

10.2. Одноступінчасті DV -тригери

Тригером DV -типу називають D -тригер із додатковим інформаційним входом V , який при $V = 1$ функціонує аналогічно тригеру D -типу, а при $V = 0$ зберігає початковий стан незалежно від зміни інформації на вході D .

Функціональна схема одноступінчастого DV -тригера наведена на рис. 10.4.

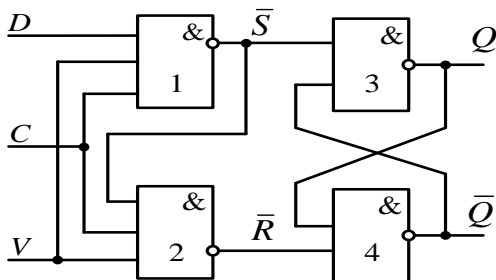


Рисунок 10.4 – Функціональна схема DV -тригера

Тригер DV -типу, так само як і тригер D -типу, широко використовується під час побудови лічильних схем, регістрів зсуву, а також у пристроях керування.

Контрольні запитання і завдання

1. Як працює одноступінчастий DV -тригер? Зобразіть часову діаграму його роботи та поясніть її.
2. Розкажіть про особливості роботи одноступінчастих DV -тригерів. Визначте їх переваги і недоліки.
3. Дайте визначення DV -тригера.
4. Опишіть роботу функціональної схеми DV -тригера.
5. Чим D -тригер відрізняється від DV -тригера?

10.3. Одноступінчасті T -тригери

Якщо виходи RS -тригера з'єднати ланцюгом зворотного зв'язку з лінією затримки, як це показано на рис. 10.4, то одержимо T -тригер, що працює так, як це показано на часовій діаграмі рис. 10.5.

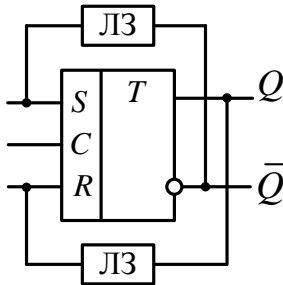


Рисунок 10.4 – T-тригер

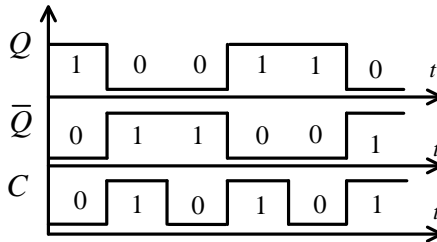


Рисунок 10.5 – Часова діаграма роботи T-тригера

Із діаграми бачимо, що за переднім фронтом кожного тактового імпульсу, який подається на синхронізувальний вхід тригера, проходить зміна його стану на зворотний. При цьому частота зміни станів тригера буде удвічі меншою частоти тактових імпульсів, тобто тактова частота поділяється тригером на 2. Ця властивість поділу частоти T-тригерами широко використовується на практиці.

Звернемо увагу, що в ланцюги зворотного зв'язку

T -тригера ввімкнені лінії затримки, які затримують сигнали попереднього стану тригера на його входах до того часу, поки в ньому проходять перехідні процеси, і відповідно на його виходах змінюються сигнали. Якщо цього не зробити, то тоді тригер може переходити у невизначені стани, що призводить до його неправильної роботи.

Якщо на схемі D -тригера з'єднати інверсний вихід \bar{Q} з входом D ланцюгом зворотного зв'язку (див. рис. 10.6), то буде одержаний T -тригер, який за кожним тактовим сигналом буде змінювати свій стан на зворотний. У цій схемі з тієї причини, що і в T -тригері, у ланцюгу зворотного зв'язку вводиться лінія затримки.

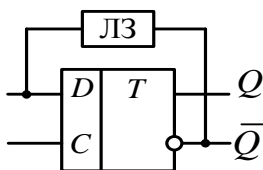


Рисунок. 10.6 – Побудова T -тригера на D -тригері

Звернемо увагу на те, що відповідно до часової діаграми рис. 3.17 T -тригер працює за переднім фронтом тактового імпульсу за наявності на цей час сигналу на вході D .

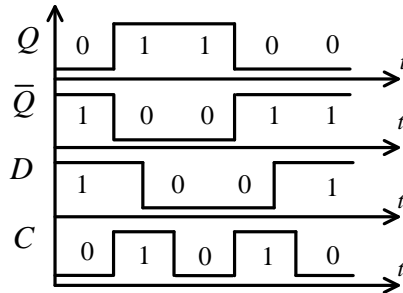


Рисунок 10.7 – Часова діаграма роботи D -тригера як T -тригера

Контрольні запитання і завдання

1. Який тригер називається T -тригером і як він працює?
2. Складіть часову діаграму роботи T -тригера.
3. Поясніть роботу T -тригера на основі його часової діаграми.
3. Як реалізувати T -тригер на основі D -тригера?
4. Чому T -тригер ділить тактову частоту?
5. Для чого у T -тригер введені лінії затримки?

ТЕМА 11. Двоступінчасті тригери

11.1. Узагальнена схема

Двоступінчастим тригером називається тригер, в якому є два з'єднані між собою елементарні тригери, один з яких основний, а інший – допоміжний.

Для двоступінчастих тригерів характерно те, що в них вищезазначена прозорість одноступінчастих тригерів відсутня. Спочатку в них проходять перехідні процеси в тригері першого ступеня, а другий ступінь на цей час відімкнений від першого ступеня, і на його виході сигнал не змінюється. Через певний час, коли пройдуть перехідні процеси в тригері першого ступеня, неспотворена інформація з нього передається до тригера другого ступеня. У результаті такого способу передавання інформації перший ступінь двоступінчастого тригера стає непрозорим, але при цьому для своєї реалізації двоступінчастий тригер потребує удвічі більшої кількості апаратури порівняно з одноступінчастим тригером. Однак тому що надійність роботи цифрової апаратури здебільшого значно важливіша, ніж її ціна, то двоступінчасті тригери широко використовують на практиці.

Структура двоступінчастих тригерів містить два послідовно ввімкнених синхронних елементарних тригери – основний *M* (master) і допоміжний *S* (slave). Тому вони називаються *MS*-тригерами. Крім елементарних тригерів, *MS*-тригери мають у своїй

структурі ще схеми керування ними. Залежно від їх типу *MS*-тригери мають різні назви. Ми будемо розглядати в основному двоступінчасті тригери.

Стандартне позначення двоступінчастого тригера подане на рис. 11.1.

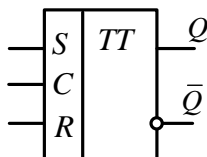


Рисунок 11.1 – Стандартне позначення *MS*-тригера

Структурна схема *MS*-тригера містить два послідовно ввімкнених елементарних тригери, перший з яких основний (*M*-тригер), а другий – допоміжний (*S*-тригер). На рис. 11.1 подана відповідна структура.

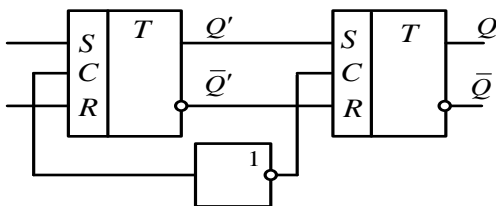


Рисунок 11.2 – Структурна схема двоступінчастого *MS*-тригера

Зі структурної схеми MS -тригера бачимо, що інформація в ньому подається за синхронізувальним сигналом $C = 1$ до основного M -тригера. Другий допоміжний S -тригер на цей час вимкнений інверсним сигналом \bar{C} . При $C = 0$ основний тригер вимикається від входів, а інформація з нього переписується до допоміжного тригера.

Контрольні запитання і завдання

1. Який тригер називається двоступінчастим?
2. Чим відрізняється одноступінчастий тригер від двоступінчастого?
3. Опишіть структуру і роботу двоступінчастого тригера.
4. Наведіть стандартне позначення двоступінчастого тригера.
5. Нарисуйте часову діаграму його роботи.

11.2. Двоступінчасті RS -тригери

Двоступінчастим RS -тригером називається MS -тригер елементарними тригерами якого є двоступінчасті синхронні RS -тригери.

Двоступінчастий RS -тригер з інвертором

З наведеної нижче на рис. 11.3 функціональної схеми

RS-тригера впливає, що інформація за допомогою синхронізувального сигналу *C* через схеми 1 і 2 передається до основного тригера. На цей час схеми 5 і 6 вимкнені за допомогою інвертора 1. Після того як пройдуть перехідні процеси в основному тригері, сигнал *C* знімається і відповідно на виході інвертора з'являється 1. У результаті інформація з основного тригера без перешкод переписується до допоміжного тригера. Очевидно, що основний тригер є непрозорим, у цьому і полягає основна перевага двоступінчастих тригерів.

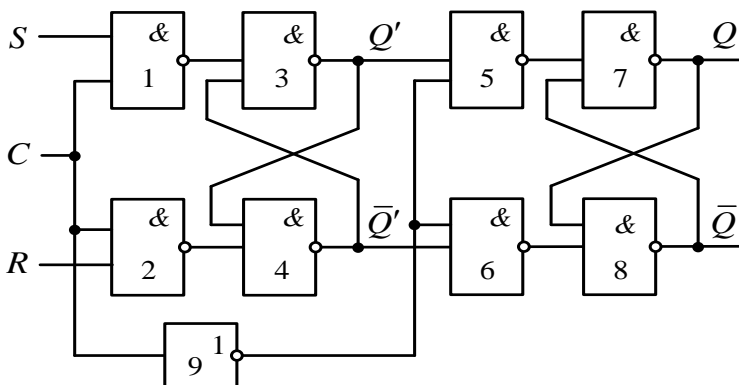


Рисунок 11.3 – RS-тригер із двома ступенями пам'яті та інвертором

Двоступінчастий *RS*-тригер без інвертора

Іншим варіантом *RS*-тригера буде тригер, показаний на рис. 11.4. Його перевагою є те, що у ньому відсутній допоміжний інвертор між елементарними тригерами. Це

зменшує апаратні витрати на реалізацію тригера і підвищує надійність його функціонування.

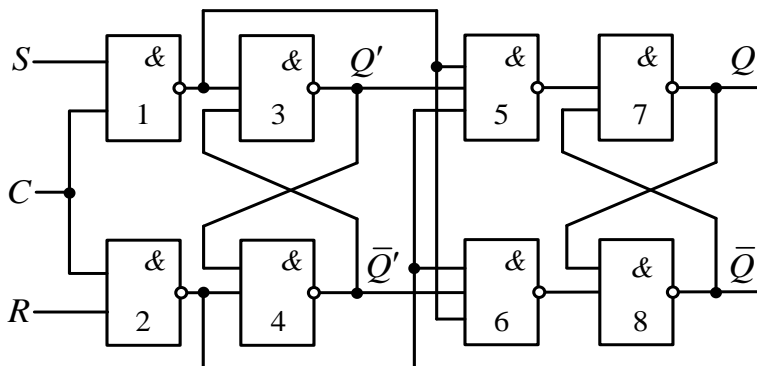


Рисунок 11.4 -- RS-тригер із двома ступенями пам'яті та більш надійним перемиканням каскадів

Якщо $C = 1$ і S або R мають значення 1, то на виході одного з елементів 1 або 2 з'явиться значення 0, яке послугує сигналом установа основного тригера у відповідний стан і одночасно заблокує елементи 5, 6, створюючи на їх виходах значення одиниці, тим самим відімкнувши допоміжний тригер від основного.

Таким чином, при $C = 1$ допоміжний тригер зберігає старий стан, у той самий час як основний тригер набуває нового значення свого стану. Після того як в основному тригері пройдуть перехідні процеси, сигнал синхронізації зникає, тобто C буде дорівнювати 0. На цей час на схемах 1, 2 з'являться одиниці, які дозволять перезапис сталого сигналу з основного тригера до допоміжного через схеми 5, 6. Ці самі сигнали заборонять запис

інформації до основного тригера, тим самим відімкнувши його від зовнішнього середовища. У результаті основний і допоміжний тригери будуть знаходитися в одному й тому самому стані.

Контрольні запитання і завдання

1. Який тригер називається двоступінчастим *RS*-тригером?
2. Опишіть структуру і роботу двоступінчастого *RS*-тригера з інвертором.
3. Опишіть структуру і роботу двоступінчастого *RS*-тригера без інвертора.
4. Складіть часову діаграму роботи *RS*-тригера без інвертора.
5. Чим двоступінчастий *RS*-тригер з інвертором відрізняється від двоступінчастого *RS*-тригера без інвертора?

11.3. Двоступінчасті *JK*-тригери

*Двоступінчастим *JK*-тригером називається *MS*-тригер, в якому є *J* і *K* входи.*

Структура *JK*-тригера

JK-тригер є універсальним двоступінчастим непрозорим тригером, тобто тригером, на основі якого можна будувати різні типи непрозорих тригерів. Позначається так, як показано на рис 11.5.

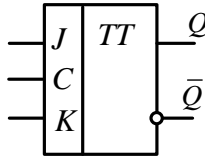


Рисунок 11.5 – Універсальний JK-тригер

Працює він при $C = 1$ відповідно до табл. 11.1 його функціонування, а при $C = 0$ його стан залишається без змін за будь-яких значень вхідних сигналів J і K .

Таблиця 11.1 – Робота JK-тригера

C	J	K	$Q_{(t+1)}$
1	0	0	Q_t
1	1	0	1
1	0	1	0
1	1	1	\bar{Q}_t

Головна відмінність структури JK-тригера від двоступінчастого RS-тригера в тому, що у ньому виходи перехресно пов'язані зі своїми входами (див. рис. 11.6). Як випливає з табл. 11.1, це дозволяє JK-тригеру на відміну від RS- тригера при $C = 1$ змінювати стан під час надходження кожного наступного синхроімпульсу, що надає йому додаткової функції. При цьому входи $J = K = 1$.

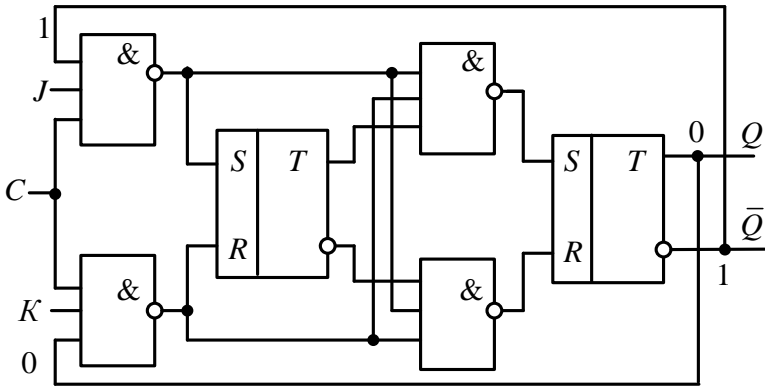


Рисунок 11.6 – Структурна схема JK -тригера

Функціональна схема JK -тригера, що реалізує структурну схему на рис. 11.6, наведена на рис. 11.7. Вона працює таким чином. При $J = K = 0$ C -сигнал не може відкрити входні елементи I , і тригер перебуває в режимі зберігання свого попереднього стану.

При $J = 1, K = 0, C = 1$ тригер встановлюється в одиничний стан. При $J = 0, K = 1, C = 1$ тригер встановлюється в нульовий стан. При $C = 0$ відбувається перезапис стану основного M -тригера до S -тригера. При $J = 1, K = 1, C = 1$ основний тригер змінює свій стан на протилежний, який при $C = 0$ переписується до допоміжного тригера.

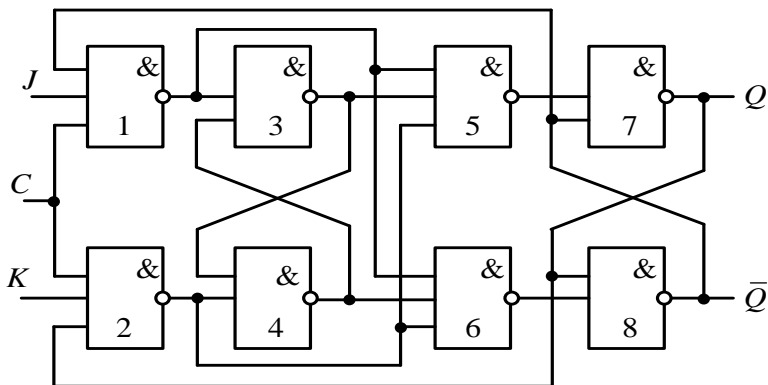


Рисунок 11.7 – Функціональна схема JK -тригера

Контрольні запитання і завдання

1. Які тригери називаються JK -тригерами? Опишіть їх роботу. У чому полягають їх переваги порівняно з одноступінчастими тригерами?
2. Чим JK -тригер відрізняється від одноступінчастого RS -тригера?
3. Складіть часову діаграму роботи JK -тригера.
4. Як позначається JK -тригер?
5. Опишіть структурну схему JK -тригера і складіть її часову діаграму.
6. Опишіть функціональну схему JK -тригера і складіть її часову діаграму.

11.4. Двоступінчасті T - і D -тригери

JK -тригер шляхом певного з'єднання його входів легко перетворити на тригери інших типів. За будь-яких комбінацій вхідних сигналів J і K , окрім однієї, за якою

сигнали $J = K = 1$, він діє подібно до RS -тригера, причому вхід J відіграє роль входу S , а вхід K – відповідно входу R . На відміну від RS -тригера для JK -тригера значення $J = K = 1$ не є забороненим. Таким чином, за різних значень J і K JK -тригер поводить себе як синхронний непрозорий RS -тригер.

За вхідної комбінації $J = K = 1$ цей тригер перетворюється на T -тригер (див. рис. 11.8), тобто у кожному такті роботи схеми відбувається зміна стану тригера, і відповідно вихідні сигнали змінюють своє значення на зворотне.

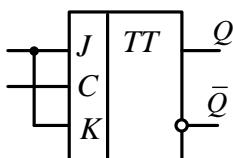


Рисунок 11.8 – Двоступінчастий T -тригер

Під час подавання сигналу з входу J через інвертор на вхід K JK -тригер перетворюється на непрозорий двоступінчастий D -тригер (див. рис. 11.9).

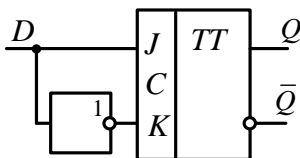


Рисунок 11.9 – Двоступінчастий D -тригер

Контрольні запитання і завдання

1. Яким чином можна *JK*-тригер перетворити на *D*- і *T*- тригери?
2. Побудуйте часові діаграми роботи *D*- і *T*-тригерів.
3. У чому полягає відмінність двоступінчастого *JK*-тригера від двоступінчастих *D*- і *T*-тригерів?
4. Складіть часову діаграму роботи двоступінчастого *D*-тригера, реалізованого на *JK*-тригері.
5. Одержіть двоступінчастий *T*-тригер на *JK*-тригері та опишіть його роботу.
6. Поясніть роботу двоступінчастого *T*-тригера на основі його часової діаграми.

ТЕМА 12. Регістри

12.1. Нагромаджувальні регістри

Регістром називається функціональний пристрій, призначений для короткочасного зберігання та перетворення цифрової інформації.

Регістр складається з окремих тригерів, кожен з яких служить для запису та зберігання одного розряду двійкового числа.

За призначенням регістри поділяють на *нагромаджувальні, зсуву і перетворення.*

Нагромаджувальні регістри з установленням нуля

Нагромаджувальні регістри використовують для запису, зберігання й читання двійкових комбінацій, причому всі ці дії здійснюються паралельно і одночасно за всіма їх розрядами. Вони можуть бути з попереднім установленням нуля та парафазним входом. На рисунку 12.1 наведений трирозрядний нагромаджувальний регістр із попереднім установленням його тригерів у нулі.

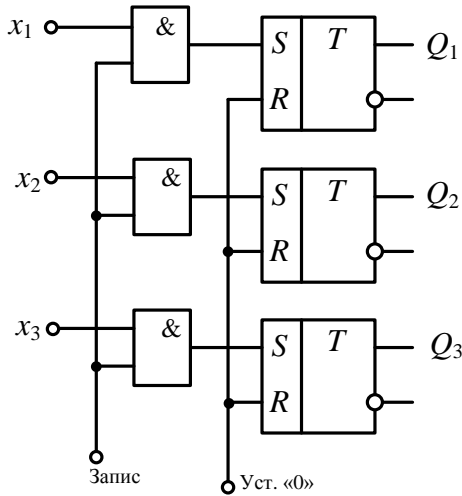


Рисунок 12.1 – Регістр із установленням нуля

Регістр складається із трьох RS -тригерів і трьох схем I , кожна з яких має два входи. На один із цих входів подається сигнал запису інформації – «Запис», а на інший – інформаційний сигнал $x_i - 0$ або 1 . При появі сигналу «Запис» вхідна інформація записується в тригери регістра. Замість RS -тригерів можна використовувати інші типи тригерів, наприклад D -тригери. Цей регістр досить простий і може бути побудований як на одноступінчастих, так і двоступінчастих тригерах.

Регістри із парафазним входом

Недоліком регістра з установленням нуля є те, що для запису сигналів у нього потрібно мати два такти – попереднє установлення регістра в нуль запис інформації до регістра. Щоб уникнути цього недоліку, була запропонована схема на рис. 12.2 нагромаджувального регістра із парафазним входом на двоступінчастих тригерах.

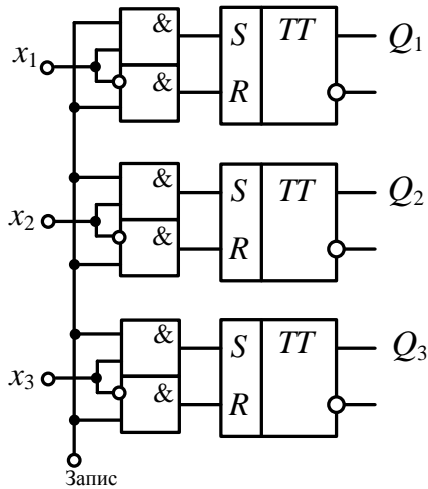


Рисунок 12.2 – Регістр із парафазним входом

У цьому регістрі перед кожним тригером установлені дві двовходові схеми I , одна з яких має інверсний вхід. Їх виходи з'єднані з установлювальними входами тригера в 1 і 0. Тому, коли надходить сигнал «Запис» разом із

синхросигналом (його входи на тригерах не показані), вхідна інформація записується до тригерів регістра першого ступеня, а потім, після зникнення синхросигналу, переписується до допоміжного тригера і залишається там. Недолік такої схеми регістра полягає у тому, що він потребує додаткових апаратурних витрат у вигляді двовходових схем *I*. Описана схема може бути реалізована і на одноступінчастих тригерах.

Контрольні запитання і завдання

1. Який цифровий пристрій називається регістром і яку функцію він виконує?

2. На які типи поділяють регістри?

4. Що таке нагромаджувальний регістр і в чому полягає його відмінність від інших типів регістрів?

5. Які є види нагромаджувальних регістрів? Опишіть їх переваги і недоліки.

6. Наведіть структуру нагромаджувального регістра з попереднім устанавленням нуля і поясніть її роботу.

7. Наведіть структуру і поясніть роботу регістра з парафазними входами.

12.2. Регістри зсуву

Регістрами зсуву двійкових чисел називаються цифрові пристрої, які організують зрушення двійкового числа, що в них зберігається, вліво чи вправо одночасно за всіма розрядами.

Уведення чисел до регістра зсуву проводиться у послідовному або паралельному коді. На рисунку 12.3 показаний регістр зсуву, в якому розміщене двійкове число, що зсувається вправо. Воно вводиться послідовно за тактами, починаючи з першого тригера зліва, поданням двійкових цифр на його вхід D . Перед цим тригери встановлюються в нулі сигналом, поданим на входи R .

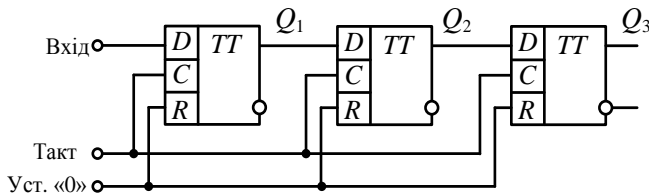


Рисунок 12.3 – Зсув двійкових чисел вправо

Вочевидь воно буде зсуватися за кожним синхросигналом вправо до того часу, поки регістр не очиститься, і всі тригери не будуть встановлені в нуль. Після цього потрібно буде внести до регістра нове двійкове число.

Контрольні запитання і завдання

1. Який пристрій називається регістром зсуву?
2. Наведіть схему регістра зсуву і опишіть його роботу.
3. Побудуйте часову діаграму роботи регістра зсуву і опишіть її роботу.
4. У яких кодах може вноситись двійкове число до регістра?

12.3. Розподільники імпульсів

Регістр зсуву, на виходах якого забезпечується почергове виникнення одного імпульсу, називається розподільником імпульсів.

Будь-який регістр зсуву, до якого внесене двійкове число лише з однією 1, буде розподільником імпульсів. Однак якщо ця одиниця переходить з останнього тригера на перший, то такий розподільник імпульсів буде називатися кільцевим розподільником імпульсів.

На рис. 12.4 наведений кільцевий розподільник імпульсів, складений із 3 тригерів. У ньому прямі виходи з'єднані з входами D наступних тригерів, а вихід останнього тригера – з входом D першого.

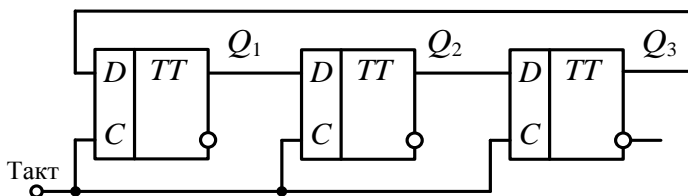


Рисунок 12.4 – Кільцевий розподільник імпульсів

Часова діаграма роботи цього пристрою наведена на рис. 12.5. Із неї бачимо, що спочатку всі тригери регістра встановлюються в нульовий стан. Потім при появі сигналу на вході D до першого тригера зліва вноситься 1 (на діаграмі цього не показано), яка за кожним синхросигналом переноситься до сусіднього справа тригера вже без подання сигналів на входи D -тригерів.

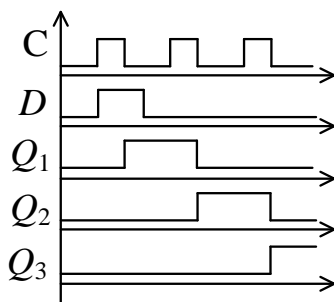


Рисунок 12.5 – Діаграма роботи розподільника імпульсів

Контрольні запитання і завдання

1. Який пристрій називається розподільником імпульсів?
2. Накресліть схему розподільника імпульсів та опишіть його роботу.
3. Побудуйте часову діаграму роботи розподільника імпульсів та опишіть її роботу.
4. Який пристрій називається кільцевим розподільником імпульсів?

ТЕМА 13. Лічильники імпульсів

13.1. Загальні поняття

Лічильником називається функціональний вузол, призначений для підрахунку кількості сигналів (імпульсів) що надходять на його вхід, і фіксації цієї кількості у вигляді кодової комбінації, запам'ятовуваної тригерами.

Лічильники можуть як підраховувати імпульси, так і віднімати. Перші з них називають підсумовувальними лічильниками, або прямої лічби, другі – віднімальними, або зворотної лічби. Також відомі реверсивні лічильники, які залежно від керувальних сигналів або підсумовують сигнали, що подаються на їх входи, або віднімають. Ці та інші види лічильників широко використовують на практиці у таймерах, електронних часах, далекомірах, перетворювачах кодів, мікропроцесорах, комп'ютерах, частотомірах тощо. На сьогодні відомо більше 30 практичних застосувань лічильників і їх кількість продовжує зростати. Лічильники бувають двійкові, багатозначні, завадостійкі, швидкодіючі.

Стандартне позначення підсумовувального лічильника наведено на рис. 13.1.

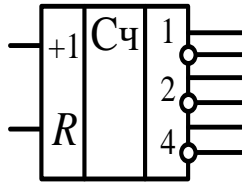


Рисунок 13.1 – Підсумовувальний лічильник

Число розрядів лічильника визначає кількість його різних стійких станів, яка називається коефіцієнтом перерахунку. Коефіцієнт перерахунку двійкового лічильника дорівнює 2^n , де n – число розрядів лічильника. Лічильники поділяють на лічильники прямого рахунку (підсумовувальні), зворотного рахунку (віднімальні) і реверсивні, тобто вони відповідно до керувальних сигналів можуть як підсумовувати, так і віднімати.

Контрольні запитання і завдання

1. Дайте визначення лічильника.
2. Наведіть стандартне позначення лічильника і опишіть його роботу.
3. Що таке коефіцієнт перерахунку? Як він визначається залежно від кількості розрядів?
4. Які є типи лічильників?
5. Де на практиці застосовують лічильники?

13.2. Підсумовувальні лічильники

Підсумовувальним лічильником називається цифровий пристрій, який підсумовує імпульси, що подаються на його вхід.

Підсумовувальний лічильник на 3 розряди на JK -тригерах наведений на рис. 13.2, а часова діаграма його роботи – на рис. 13.3.

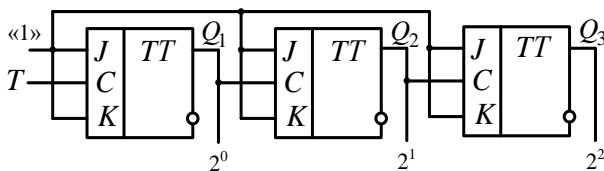


Рисунок 13.2 – Підсумовувальний лічильник

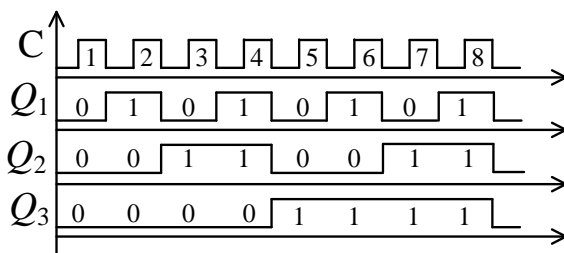


Рисунок 13.3 – Часова діаграма

У лічильнику, що розглядається на рис. 13.1, входи J - і K -тригерів перемкнені між собою. Тим самим JK -тригери перетворюються на лічильні тригери. Це означає, що за кожним тактовим імпульсом стан першого

зліва тригера змінюється на зворотний. У цьому разі зміна стану тригера проходить за заднім фронтом тактового імпульсу. При цьому на вхід тригера повинні надійти два тактові імпульси, щоб він перейшов із попереднього стану до іншого. Це впливає з часової діаграми роботи лічильника на рис. 13.2.

У свою чергу, зміна стану першого тригера приведе за його заднім фронтом до зміни стану другого тригера на зворотний, а потім за заднім фронтом другого тригера зміниться стан третього. У результаті послідовно після нульового стану лічильника 000 будуть одержані стани 001, потім – 010 і так далі до 111. Усіх станів, очевидно, буде 8. При цьому початкова частота поділяється на кожному тригері на 2, так що в кінцевому підсумку вона буде поділена на 8. На цій властивості двійкових лічильників будують подільники частот, які широко використовують у цифровій техніці.

Контрольні запитання і завдання

1. Дайте визначення підсумовувального лічильника.
2. Накресліть схему підсумовувального лічильника і опишіть її роботу.
3. Поясніть часову діаграму роботи підсумовувального лічильника та опишіть його роботу.
4. Чому в підсумовувальному лічильнику сигнал до наступного тригера знімається з прямого виходу попереднього тригера?

13.3. Віднімальні лічильники

Віднімальним лічильником називається цифровий пристрій, який віднімає імпульси, що подаються на його вхід, від наявного в ньому числа.

Як бачимо зі схеми віднімального лічильника на трьох JK-тригерах, наведеного на рис. 13.4, сигнал на синхронізувальний вхід С наступного тригера у ньому надходить не з прямого входу, як це було у підсумовувальному лічильнику, а з інверсного. Тоді сигнали будуть не підсумовуватись, а відніматися. Це відбувається тому, що коли зліва перший тригер під дією заднього фронту тактового імпульсу переходить з нульового стану в одиничний, на його інверсному виході сигнал переходить з 1 на 0, що відповідає задньому фронту імпульсу. А якраз за цим переходом імпульсу спрацьовує другий тригер, коли переходить в інший стан, у цьому разі в одиничний. У свою чергу, він вмикає третій тригер. У результаті заднім фронтом тактового імпульсу всі тригери, які до цього були в нулях, перейдуть в одиничні стани. Наступний тактовий імпульс своїм заднім фронтом може перекинути в нульовий стан тільки перший тригер. У ньому на цей час перепад фронту на інверсному виході буде з 0 на 1, що не дозволяє спрацювати другому тригеру, на тактовий вхід якого був заведений інверсний вихід першого тригера. Третій тригер взагалі не буде змінювати свій стан, тому що з другого тригера на його синхронний вхід ніякого

сигналу подаватися не буде. Відповідна часова діаграма, яка ілюструє цю роботу віднімального тригера, наведена на рис. 13.5.

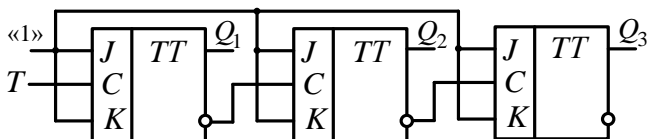


Рисунок 13.4 – Схема віднімального лічильника

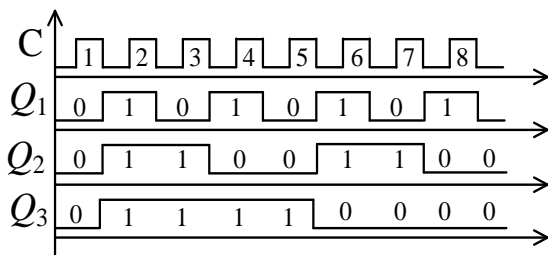


Рисунок 13.5 – Часова діаграма

Контрольні запитання і завдання

1. Дайте визначення віднімального лічильника.
2. У чому полягає принципова різниця між підсумовувальним і віднімальним лічильником?
3. Наведіть схему віднімального лічильника та опишіть її роботу.
4. Побудуйте часову діаграму роботи віднімального лічильника.

13.4. Реверсивні лічильники

Реверсивним лічильником називається цифровий пристрій, який або додає, або віднімає імпульси, що подаються на його вхід, від наявного в ньому числа.

Реверсивні лічильники дозволяють виконувати як операції підсумовування, так і віднімання залежно від сигналів, що подаються на керувальні входи (рис. 13.5). Цих входів два – один вмикає функцію додавання, а інший – віднімання. Відповідно проходить перемикання лічильника з режиму підсумовування на режим віднімання. У першому режимі прямі виходи тригерів з'єднуються із синхронізуювальними (тактовими) входами, в другому режимі з ними з'єднуються інверсні виходи. Тобто в одному лічильнику об'єднуються два лічильники – підсумовувальний і віднімальний, які вмикаються відповідно до керувальних сигналів.

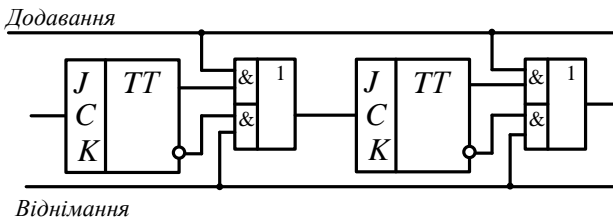


Рисунок 13.6 – Реверсивний лічильник

Працює реверсивний лічильник таким чином. Сигнал «1» подається на шину «додавання», і схема підсумовує. При появі сигналу «1» на шині віднімання» схема починає віднімати.

Контрольні запитання і завдання

1. Дайте визначення реверсивного лічильника.
2. Наведіть схему реверсивного лічильника та його часову діаграму. Опишіть його роботу.
3. Якими перевагами і недоліками характеризуються реверсивні лічильники щодо інших типів лічильників?
4. Де на практиці можуть використовуватися реверсивні лічильники?

13.5. Лічильники з довільним перерахунком

Лічильником із довільним перерахунком називається цифровий пристрій, що має довільний коефіцієнт перерахунку.

Зазвичай лічильники підсумовують до стану, який задається коефіцієнтом перерахунку 2^n . Однак бувають випадки, коли потрібно перераховувати за довільним коефіцієнтом. Тоді підрахунок проходить до потрібного стану, встановленого коефіцієнтом перерахунку. На цей момент спрацьовує схема I , налаштована на вибраний

стан лічильника, і сигнал, який вона виробить, встановить тригери лічильника в нулі. Після цього підрахунок вважається закінченим.

Можна також реалізувати схему лічильника з довільним коефіцієнтом підрахунку, в якій попередньо до нього вноситься потрібне число. Тоді лічильник підраховує до числа 2^n , але кількість імпульсів, які він при цьому сприйме, буде меншою на величину попередньо внесеного числа. Ця кількість імпульсів і буде потрібним коефіцієнтом перерахунку.

Контрольні запитання і завдання

1. Дайте визначення лічильника з довільним перерахунком.

2. Наведіть схему лічильника з довільним перерахунком і подайте його часову діаграму. Опишіть його роботу.

3. Опишіть сферу застосування лічильників із довільним перерахунком.

4. Що показує в лічильниках коефіцієнт перерахунку?

Післямова

Автор звертає увагу читачів на те, що все, що написано в підручнику, може бути використано в реальних схемах. Побудовою електронних цифрових пристроїв і систем він займається більше 40 років. За цей час за його участі було одержано близько 40 винаходів і серед них – більше 20 винаходів на цифрові лічильники.

Важливим при написанні книги було те, що більшість цих винаходів були перевірені в діючій електронній апаратурі. На її основі були виготовлені надійні та швидкодіючі мікросхеми особливого призначення на базових матричних кристалах, які працюють і зараз. Тому автор упевнений в корисності матеріалу, поданого в підручнику. Вибрав він його як на основі своєї багаторічної практичної діяльності в галузі електронної цифрової техніки, так і з десятків книг із цифрової електроніки як вітчизняних, так і зарубіжних авторів.

Має надію, що ця досить невелика і за змістом, і за обсягом книга буде добре сприйнята читачами та одержить позитивні відгуки. Хотілося б майбутнім поколінням схемотехніків, які захочуть займатися цією справою, побажати успіхів для досягнення нових висот у цій хоча й нелегкій, але дуже цікавій, на погляд автора, науці. Від автора.

Додаток А

(рекомендований)

Модульний і сесійний контроль

1. Комбінаційні автомати та автомати з пам'яттю.
2. Позитивна і негативна логіка.
3. Багатозначна і двозначна логіка. Переваги двозначної логіки.
4. Функціонально повні логічні базиси. Дати визначення, навести приклади.
5. Повний і мінімальний базис. Булевий базис.
6. Інвертори. Стандартне зображення. Логіка функціонування. Таблиця істинності. Часова діаграма роботи. Релейна схема.
7. Кон'юнктор. Стандартне зображення. Логіка функціонування. Таблиця істинності. Часова діаграма роботи. Релейна схема. Реалізація на основі диз'юнктора і інвертора за допомогою правила де Моргана.
8. Диз'юнктор. Його стандартне зображення. Логіка функціонування. Таблиця істинності. Часова діаграма роботи. Релейна схема. Реалізація на основі кон'юнктора і інвертора за допомогою правила де Моргана.

Продовження додатка А

9. Елементи «штрих Шеффера». Таблиця істинності. Реалізація логічних схем I , $АБО$, інвертора на основі елементів «штрих Шеффера».

10. Елементи «стрілка Пірса». Таблиця істинності. Реалізація логічних схем I , $АБО$, інвертора на основі елементів «стрілка Пірса».

11. Елемент «сума за модулем 2». Таблиця істинності. Логіка функціонування. Схеми реалізації. Часова діаграма роботи.

12. Елемент «виключне $АБО$ ». Логіка функціонування. Таблиця істинності. Схеми реалізації. Часова діаграма роботи.

13. «Мажоритарний елемент». Логіка функціонування. Схеми реалізації. Часова діаграма роботи. Таблиця істинності.

14. Синтез логічних схем у булевому базисі. Навести приклад.

15. Синтез логічних схем у базисі «штрих Шеффера». Навести приклад.

16. Синтез логічних схем в базисі «стрілка Пірса». Навести приклад.

17. Синтез логічних схем на основі негативної логіки. Навести приклад.

18. Дешифратори. Визначення і призначення. Умове позначення дешифратора. Повний і неповний дешифратор. Таблиці істинності.

Продовження додатка А

19. Лінійні дешифратори. Таблиця істинності. Функціональна схема лінійного дешифратора. Синтез на основі булевих функцій.

20. Реалізація логічних функцій дешифраторами. Навести приклад.

21. Каскадні дешифратори. Їх переваги і недоліки щодо лінійних дешифраторів. Функціональна схема двокаскадного дешифратора та її робота.

22. Пірамідальні дешифратори. Функціональна схема. Їх переваги і недоліки.

23. Матричні дешифратори. Логіка роботи. Функціональна схема. Опис логічними функціями.

24. Двоступінчастий дешифратор. Логіка роботи. Функціональна схема.

25. Дешифратор із економією апаратурних витрат. Алгоритм функціонування. Функціональна схема. Опис логічними функціями.

26. Мультиплексор. Загальна інформація. Призначення. Позначення. Інформаційні та адресні входи. Опис логічними функціями.

27. Функціональна схема мультиплексора. Часова діаграма його роботи. Логіка роботи. Опис логічними функціями.

28. Мультиплексор 2 в 1. Функціональна схема. Логіка роботи. Опис логічними функціями. Часова діаграма його роботи.

Продовження додатка А

29. Реалізація логічних функцій за допомогою мультиплексора. Навести приклад.

30. Каскадне вмикання мультиплексорів. Функціональна схема. Логіка роботи.

31. Демультиплексори. Дати визначення, призначення. Стандартне позначення. Логіка роботи. Опис логічними функціями.

32. Структурна і функціональна схема демультиплексора. Описати її роботу.

33. Шифратор. Дати визначення. Стандартне позначення. Логіка роботи. Таблиця істинності для шифратора. Опис логічними функціями.

34. Функціональна схема шифратора. Логіка її роботи. Таблиця істинності. Логічний синтез шифратора.

35. Пріоритетний шифратор. Функціональна схема. Логіка роботи. Часова діаграма роботи.

36. Пріоритетний шифратор із формуванням двійкового номера. Функціональна схема. Логіка роботи.

37. Перетворювачі кодів. Стандартне позначення. Визначення. Логіка роботи. Таблиця істинності. Таблиця функціонування перетворювача кодів. Навести приклад.

38. Реалізація перетворювача кодів за схемою дешифратор-шифратор. Навести приклад. Структурна і функціональна схема.

Продовження додатка А

39. Реалізація перетворювача кодів на основі синтезу системи логічних функцій. Приклад такого синтезу і реалізації на основі цих функцій схеми перетворювача.

40. Загальне визначення однорозрядного і багаторозрядного суматора. Логіка роботи. Приклад підсумовування чисел. Структурна схема багаторозрядного суматора.

41. Однорозрядний півсуматор. Стандартне позначення. Таблиця істинності півсуматора. Логічні функції суми і перенесення. Функціональна схема.

42. Економічний півсуматор. Логічне перетворення вихідних функцій. Функціональна схема.

43. Однорозрядний суматор. Стандартне позначення. Таблиця істинності функціонування. Синтез логічних функцій суматора. Функціональна схема.

44. Функціонування однорозрядного суматора на двох півсуматорах. Таблиці істинності. Структурна схема.

45. Суматор послідовної дії. Логіка роботи. Структурна схема.

46. Суматор паралельної дії. Логіка роботи. Структурна схема.

47. Схеми порівняння. Визначення. Стандартне позначення. Логіка роботи. Порівняння на більше і менше. Навести приклади.

48. Порівняння на рівність цифр двох чисел одного розряду. Логіка роботи. Таблиця істинності. Логічні функції. Схемна реалізація.

Продовження додатка А

49. Порівняння двох двійкових чисел на рівність. Логічні функції. Схемна реалізація.

50. Порівняння на нерівність в одному розряді. Логіка роботи. Таблиця істинності. Логічні функції. Схемна реалізація.

51. Спрощення логічної функції порівняння на нерівність в одному розряді. Її схемна реалізація. Логіка роботи.

52. Порівняння двох двійкових чисел на нерівність. Логічна функція. Реалізація відповідної схеми. Логіка її роботи.

53. Тригери. Визначення. Стандартне позначення. Прямий та інверсний вихід. Нульовий і одиничний стани тригера. *RS*-тригер. Алгоритм роботи.

54. Функціональна схема *RS*-тригера на елементах *АБО*. Логіка його роботи. Часова діаграма роботи. Таблиця істинності роботи тригера.

55. Функціональна схема *RS*-тригера на елементах *I*. Логіка його роботи. Часова діаграма роботи. Таблиця істинності роботи тригера.

56. Асинхронні та синхронні *RS*-тригери. Логіка роботи. Функціональна схема асинхронного тригера. Стандартне позначення асинхронного тригера.

57. *D*-тригер. Стандартне позначення. Логіка роботи. Таблиця істинності. Часова діаграма його роботи.

58. Прозорі та непрозорі схеми тригерів. Їх недоліки. Навести приклади.

Продовження додатка А

59. Тригери з двома ступенями пам'яті. *RS*-тригер. Стандартне позначення. Логіка роботи. Функціональна схема двоступінчастого тригера з інвертором для зв'язку з другим ступенем.

60. Тригери з двома ступенями пам'яті. *RS*-тригер. Стандартне позначення. Логіка роботи. Функціональна схема двоступінчастого тригера з перехресними зв'язками з першим ступенем.

61. Регістри. Визначення. Загальна характеристика.

62. Нагромаджувальні регістри з установленням нуля і з парафазним входом. Пояснити часову діаграму їх роботи.

63. Регістри зсуву. Визначення. Схема і часова діаграма роботи.

64. Розподільники імпульсів. Схема і часова діаграма роботи. Кільцеві розподільники імпульсів.

65. Лічильники імпульсів. Визначення. Загальний опис. Класифікація.

66. Підсумовувальні лічильники. Визначення. Схема і часова діаграма.

67. Віднімальні лічильники. Визначення. Схема і часова діаграма.

68. Реверсивні лічильники. Схема і часова діаграма. Сфери застосування.

69. Лічильники з довільним коефіцієнтом перерахунку. Схема та часова діаграма їх роботи.

Додаток Б

(рекомендований)

Контрольні задачі

1. Реалізувати після мінімізації в булевому базисі з позитивною логікою функцію, що набуває значення 1 на наборах 1, 2, 5, 6, 11, 12, 17. На наборах 3, 4, 9, 10 її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

2. Реалізувати після мінімізації в булевому базисі з позитивною логікою функцію, що набуває значення 1 на наборах 1, 2, 8, 6, 12, 13, 17. На наборах 3, 4, 7, 9, 10 її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

3. Реалізувати після мінімізації в булевому базисі з позитивною логікою функцію, що набуває значення 1 на наборах 1, 3, 5, 6, 11, 12, 17. На наборах 8, 4, 9, 10 її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

4. Реалізувати після мінімізації в булевому базисі з негативною логікою функцію, що набуває значення 1 на наборах 1, 2, 5, 6, 11, 12, 17. На наборах 3, 4, 9, 10 її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

Продовження додатка Б

5. Реалізувати після мінімізації в булевому базисі з негативною логікою функцію, що набуває значення 1 на наборах 1, 2, 8, 6, 11, 12, 17. На наборах 3, 4, 7, 9, 10 її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

6. Реалізувати після мінімізації в булевому базисі з негативною логікою функцію, що набуває значення 1 на наборах 1, 3, 5, 6, 11, 12, 17. На наборах 8, 4, 9, 10 її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

7. Реалізувати після мінімізації в базисі «штрих Шеффера» функцію, що набуває значення 1 на наборах 1, 2, 5, 6, 11, 12, 17. На наборах 3, 4, 9, 10 її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

8. Реалізувати після мінімізації в базисі «штрих Шеффера» функцію, що набуває значення 1 на наборах 1, 5, 6, 11, 12, 16. На наборах 3, 4, 7, 9, 10 її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

9. Реалізувати після мінімізації в базисі «штрих Шеффера» функцію, що набуває значення 1 на наборах 1, 5, 6, 11, 12, 17. На 8, 4, 9, 10 наборах її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

Продовження додатка Б

10. Реалізувати після мінімізації в базисі «стрілка Пірса» функцію, що набуває значення 1 на наборах 1, 2, 5, 6, 11, 12, 17. На 3, 4, 9, 10 наборах її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

11. Реалізувати після мінімізації в базисі «стрілка Пірса» функцію, що набуває значення 1 на наборах 1, 2, 8, 6, 12, 13, 17. На 3, 4, 7, 9, 10 наборах її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

12. Реалізувати після мінімізації в базисі «стрілка Пірса» функцію, що набуває значення 1 на наборах 1, 3, 5, 6, 11, 12, 17. На 8, 4, 9, 10 наборах її значення невизначене, тобто може бути як 1, так і 0. На інших наборах її значення дорівнює 0.

13. Реалізувати в позитивній логіці неповний лінійний дешифратор, що дешифрує набори 1, 2, 5, 6, 11, 12, 17.

14. Реалізувати в негативній логіці неповний лінійний дешифратор, що дешифрує набори 1, 2, 3, 6, 11, 12, 19.

15. Реалізувати в позитивній логіці неповний лінійний дешифратор, що дешифрує набори 1, 2, 6, 8, 12, 13, 17.

16. Реалізувати в негативній логіці неповний лінійний дешифратор, що дешифрує набори 1, 2, 5, 6, 12, 14, 17.

Продовження додатка Б

17. Реалізувати в позитивній логіці неповний лінійний дешифратор, що дешифрує набори 1, 3, 5, 7, 11, 12, 19.

18. Реалізувати в позитивній логіці неповний лінійний дешифратор, що дешифрує набори 1, 3, 5, 6, 11, 12, 17.

19. Реалізувати в позитивній логіці неповний шифратор, що шифрує набори 1, 2, 5, 6, 11, 12, 20.

20. Реалізувати в негативній логіці неповний шифратор, що шифрує набори 1, 2, 5, 6, 11, 12, 17.

21. Реалізувати в позитивній логіці неповний шифратор, що шифрує набори 1, 2, 6, 9, 12, 13, 17.

22. Реалізувати в негативній логіці неповний шифратор, що шифрує набори 1, 2, 5, 6, 10, 13, 17.

23. Реалізувати в позитивній логіці неповний шифратор, що шифрує набори 1, 3, 5, 6, 11, 12, 19.

24. Реалізувати в негативній логіці неповний шифратор, що шифрує набори 1, 4, 5, 6, 11, 12, 20.

25. Реалізувати за схемою дешифратор-шифратор перетворювач кодів комбінацій 1, 3, 5, 6, 11, 12, 17 на комбінації 2, 4, 5, 6, 7, 18, 19.

26. Реалізувати за схемою дешифратор-шифратор перетворювач кодів комбінацій 2, 4, 5, 6, 7, 18, 19 на комбінації 1, 3, 5, 6, 11, 12, 17.

Продовження додатка Б

27. Реалізувати за схемою дешифратор-шифратор перетворювач кодів 1, 4, 6, 8, 11, 12, 19 на комбінації 2, 4, 5, 6, 7, 19, 29.

28. Реалізувати за схемою дешифратор-шифратор перетворювач кодів комбінацій 2, 4, 5, 6, 7, 19, 29. на комбінації 1, 4, 6, 8, 11, 12, 19.

29. Реалізувати шляхом синтезу логічних функцій перетворювач кодів комбінацій 1, 3, 5, 7, 11, 12, 17 на 2, 4, 5, 6, 7, 18, 19.

30. Реалізувати шляхом синтезу логічних функцій перетворювач кодів комбінацій 2, 4, 5, 6, 7, 18, 19 на комбінації 1, 3, 5, 6, 11, 12, 17.

31. Реалізувати шляхом синтезу логічних функцій перетворювач кодів комбінацій 1, 4, 6, 8, 11, 12, 19 на комбінації 2, 3, 5, 6, 7, 19, 29.

32. Реалізувати шляхом синтезу логічних функцій перетворювач кодів комбінацій 2, 4, 5, 6, 9, 19, 29 на комбінації 1, 4, 6, 8, 11, 12, 19.

Додаток В (рекомендований)

Розрахунково-графічна робота

1. Розробіть схему цифрового автомата, який би перебирав числа відповідно до варіанта, поданого в табл. 14.1. Побудуйте часову діаграму його роботи.

Таблиця 14.1 – Набори, які потрібно перебрати

<i>Варіант</i>		<i>Варіант</i>	
1	1, 4, 10, 11, 12, 15	16	3, 6, 7, 9, 10, 11, 15
2	2, 3, 6, 8, 9, 13, 14	17	2, 3, 5, 7, 11, 14, 15
3	0, 2, 5, 8, 13, 14, 15	18	2, 3, 5, 7, 10, 11, 15
4	0, 3, 5, 7, 8, 11, 12	19	3, 5, 6, 7, 10, 11, 15
5	2, 3, 5, 6, 10, 12, 13	20	1, 2, 3, 7, 11, 12, 15
6	0, 1, 3, 5, 6, 7, 11	21	0, 1, 2, 3, 6, 8, 5
7	2, 3, 4, 7, 8, 9, 12	22	1, 2, 3, 4, 6, 7, 11
8	1, 4, 5, 9, 10, 11, 12	23	0, 2, 3, 5, 6, 7, 11
9	0, 3, 6, 7, 8, 10, 12,	24	1, 2, 3, 4, 6, 7, 15
10	6, 9, 10, 11, 12, 14,	25	0, 2, 3, 5, 6, 7, 15
11	7, 8, 10, 11, 13, 14,	26	1, 2, 3, 5, 8, 9, 11
12	2, 7, 9, 10, 11, 14, 15	27	0, 1, 3, 9, 10, 11, 15
13	3, 6, 10, 11, 13, 14,	28	3, 4, 5, 7, 10, 13, 15
14	0, 1, 2, 3, 4, 7, 8	29	4, 5, 7, 11, 12, 13
15	1, 3, 6, 7, 10, 11, 15	30	1, 5, 6, 7, 11, 13, 15

Продовження додатка В

2. Побудуйте в булевому базисі комбінаційну схему відповідно до варіанта функції F , поданої в табл. 14.2.

Таблиця 14.2 – Варіанти функції F

	Варіант функції F		Варіант функції F
1	$(A \sim B) \vee C \rightarrow A$	16	$A \sim C \downarrow B \rightarrow A$
2	$(A \sim B) \vee C \rightarrow B$	17	$A \sim (C \downarrow B \vee \bar{C})$
3	$(A \sim B) \rightarrow C \vee A$	18	$(A \downarrow B) \downarrow (A \downarrow C)$
4	$(A \sim B) \rightarrow C \vee B$	19	$(A \rightarrow B) \sim C \wedge \bar{B}$
5	$A \sim (B / C) \rightarrow B$	20	$C \vee \bar{A} \rightarrow B \sim C$
6	$(A \sim B) / C \vee A$	21	$A \rightarrow B \sim (C \downarrow A)$
7	$(A \rightarrow B) \wedge C \sim A$	22	$(A \sim B) \wedge C \rightarrow \bar{B}$
8	$(A \rightarrow B) \rightarrow C \vee A$	23	$(A \sim B) \vee \bar{A} \rightarrow C$
9	$A \rightarrow B \sim C / B$	24	$A \rightarrow B \wedge C \sim A$
10	$A \sim B \wedge C / A$	25	$A \rightarrow B / (C \rightarrow A)$
11	$A \rightarrow B \sim C \vee \bar{A}$	26	$(A \sim B) \wedge (\bar{A} \rightarrow C)$
12	$(\bar{A} \sim B) \wedge (A \sim \bar{C})$	27	$(A \sim B) \sim C \vee \bar{A}$
13	$(C \vee \bar{A}) \rightarrow (A \sim B)$	28	$C \vee B \sim A \vee \bar{B}$
14	$A \sim B \rightarrow C \vee \bar{A}$	29	$A \rightarrow B \sim C \wedge \bar{B}$
15	$A \sim C \vee B / A$		

Список літератури

1. Электронные вычислительные машины. Лабораторный практикум : учеб. пособие / И. Ф. Бабалова, А. Г. Баданов, Л. А. Батанов и др. ; под ред. Г. Н. Соловьева. – Москва : Высш. шк., 1987. – 319 с.
2. Бабич Н. П. Комп'ютерна схемотехніка : навч. посіб. / Н. П. Бабич, І. А. Жуков. – Київ : МК-Прес, 2004. – 412 с.
3. Схемотехніка електронних систем : у 3 кн. Кн. 2. Цифрова схемотехніка : підручник / В. І. Бойко, А. М. Гуржій, В. Я. Жуйков та ін. – 2-ге вид. допов. і переробл. – Київ : Вища шк., 2004. – 423 с.
4. Инфраструктура мозгоподобных вычислительных процессов / М. Ф. Бондаренко, В. И. Хаханов, О. А. Гузь, Ю. П. Шабанов-Кушнаренко. – Харьков : ХНУРЭ, 2010. – 160 с.
5. Борисенко О. А. Лекції з дискретної математики (множини і логіка) : навч. посіб. / О. А. Борисенко. – Суми : Університетська книга, 2002. – 176 с.
6. Борисенко О. А. Дискретна математика : підручник / О. А. Борисенко. – Суми : Університетська книга, 2007. – 255 с.
7. Борисенко О. А. Біноміальні автомати : навчальний посібник / О. А. Борисенко. – Суми : Вид-во СумДУ, 2005. – 152 с.

8. Борисенко О. А. Біноміальна лічба і лічильники : монографія / О. А. Борисенко. – Суми : Вид-во СумДУ, 2005. – 121 с.

9. Букреев И. Н. Микроэлектронные схемы цифровых устройств / И. Н. Букреев, Б. М. Мансуров, В. И. Горячев. – 2-е изд. – Москва : Сов. радио, 1975. – 368 с.

10. Елисеев В. В. Программно-технические комплексы АСУТП : учеб. пособие / В. В. Елисеев, В. А. Ларгин, Г. Ю. Пивоваров. – Киев : Киевский университет, 2003. – 429 с.

11. Вавилов Е. Н. Синтез схем электронных цифровых машин / Е. Н. Вавилов, Г. П. Портной. – Москва : Советское радио, 1963. – 440 с.

12. Электронные промышленные устройства : учебник / В. И. Васильев, Ю. М. Гусев, В. Н. Миронов и др. – Москва : Высш. шк., 1988. – 303 с.

13. Зубчук В. И. Справочник по цифровой схемотехнике / В. И. Зубчук, В. П. Сигорский, А. Н. Шкуро. – Киев : Техника, 1990. – 448 с.

14. Калбертсон Дж. Т. Математика и логика цифровых устройств / Дж. Т. Калбертсон ; пер. с англ. – Москва : Просвещение, 1965. – 267 с.

15. Кривуля Г. Ф. Схемотехніка : навч. посібник / Г. Ф. Кривуля, В. М. Рябенський, В. С. Буряк. – Харків : ТОВ «Компанія СМІТ», 2007. – 250 с.

16. Мураховский В. И. Устройство компьютера / В. И. Мураховский ; под ред. С. В. Симоновича. – Москва : Аст-ПРЕСС КНИГА, 2003. – 640 с.

17. Потемкин И. С. Функциональные узлы цифровой автоматики / И. С. Потемкин. – Москва : Энергоатомиздат, 1988. – 320 с.

18. Цифровые вычислительные машины (элементы, узлы и устройства, машины). Лабораторный практикум : учеб. пособие / Г. Н. Соловьев, Б. И. Кальнин, А. А. Рыбаков и др. ; под ред. Г. Н. Соловьева. – Москва : Атомиздат, 1977. – 312 с.

19. Столингс Уильям. Структурная организация и архитектура компьютерных систем / Уильям Столингс; пер. с англ. – 5-е изд. – Москва : Изд. дом «Вильямс», 2002. – 896 с.

20. Точи, Рональд Дж. Цифровые системы. Теория и практика / Дж. Рональд, Точи, Нил С. Уидмер ; пер. с англ. – 8-е изд. – Москва : Изд. дом «Вильямс», 2004. – 1024 с.

21. Угрюмов Е. П. Цифровая схемотехника : учеб. пособие / Е. П. Угрюмов. – 3-е изд. – Санкт-Петербург : БХВ – Петербург, 2010. – 816 с.

22. Хаханов В. И. Проектирование и тестирование цифровых схем на кристаллах / В. И. Хаханов, Е. И. Литвинова, О. А. Гузь. – Харьков : ХНУРЭ, 2009. – 484 с.

23. Проектирование и верификация цифровых систем на кристаллах. Verilog & System Verilog / В. И. Хаханов, И. В. Хаханова, Е. И. Литвинова, О. А. Гузь. – Харьков : ХНУРЭ, 2010. – 528 с.

24. Шоломов Л. А. Основы теории дискретных, логических и вычислительных устройств / Л. А. Шоломов. – Москва : Наука, 1980. – 400 с.

Навчальне видання

Борисенко Олексій Андрійович

ЦИФРОВА СХЕМОТЕХНІКА

Підручник

Художнє оформлення обкладинки О. А. Борисенка
Редактор М. Я. Сагун
Комп'ютерне верстання О. А. Борисенка, І. Є. Бражник

Формат 60×84/16. Ум. друк. арк. 11,63. Обл.-вид. арк. 7,56. Тираж 300 пр. Зам. №.

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



Борисенко

Олексій Андрійович

Фахівець у галузі цифрової електроніки. Доктор технічних наук (1991), професор (1992). Закінчив Харківський інститут радіоелектроніки (1970). З 1980 року працює в Сумському державному університеті: старший викладач, доцент, професор, завідувач кафедри.

Має понад 200 наукових праць, серед яких близько 40 винаходів, 20 монографій і навчальних посібників, один підручник. Основний науковий напрям – підвищення надійності та швидкодії електронних цифрових систем і пристроїв на базі завадостійких систем числення, серед яких детально досліджені біноміальні системи.

Нагороджений знаком «Відмінник освіти України». присвоєне звання «Заслужений професор Сумського державного університету».

