

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**Івано-Франківський національний технічний
університет нафти і газу**

**Кафедра інформаційно-телекомунікаційних технологій та
систем**

**О. В. Євчук
І. Т. Левицький**

ОСНОВИ ЦИФРОВОЇ ТЕХНІКИ

КОНСПЕКТ ЛЕКЦІЙ

**Івано-Франківськ
2021**

УДК 621.4
Є-17

Рецензент:

Николайчук М.Я. кандидат технічних наук, доцент кафедри інформаційно-телекомунікаційних технологій та систем Івано-Франківського національного технічного університету нафти і газу

Євчук О. В., Левицький І. Т.
Є-17 Основи цифрової техніки
Конспект лекцій. Івано-Франківськ
ІФНТУНГ, 2021. 131 с.

Конспект лекцій містить матеріал для проведення лекційних занять з дисципліни «Цифрова та аналогова схемотехніка». Розроблений відповідно до робочої програми навчальної дисципліни, чинного навчального плану. Може бути використаний студентами денної та заочної форм навчання.

Призначено для підготовки бакалаврів за спеціальністю 151 – «Автоматизація та комп'ютерно-інтегровані технології».

УДК 621.4

© Євчук О.В.
© ІФНТУНГ, 2021

0. ВСТУП

Цифрова електроніка, це розділ електроніки, що ввійшов в повсякденне життя починаючи з 80-х років XX століття. Цифрові CD та DVD програвачі, мобільні телефони, комп'ютери, цифрові фото- та кіно камери – ось далеко не повний перелік всіх приладів, в яких основне ядро процесів виконане з використанням цифрової техніки. Розвиток технологій виробництва інтегральних схем (ІС) призвів до значного збільшення швидкості їх роботи, що дозволило використати теоретичну базу цифрової техніки там де раніше панували винятково аналогові системи.

Крім таких переваг, як економічність, конструктивна технологічність, і т.ін., є ще й принципи цифрової обробки сигналів, які дозволяють досить просто і точно реалізувати складні алгоритми, які не під силу аналоговій техніці. Останні покоління ЕОМ реалізовані на основі мікропроцесорних: інтегральних схем, що містять на напівпровідниковій пластині площею не більше 1 см² окремі блоки, а то і цілі пристрої. Малі габаритні розміри і низьке енергоспоживання та вартість таких машин дозволяють вбудовувати їх в різні пристрої та системи радіоелектроніки, причому продуктивність сучасних мікроЕОМ призначених для таких цілей співрозмірна з продуктивністю персональних ЕОМ кінця 90-х рр. минулого століття. Це дозволяє, наприклад, вбудувати в сучасну пральну машину надвичайно складний алгоритм роботи, який враховуватиме велику кількість параметрів процесу, що в кінцевому випадку призводить до зменшення енерговитрат та підвищення якості роботи та збільшення надійності.

"Основи цифрової техніки" є базовим курсом для вивчення таких дисциплін як "Мікропроцесорні пристрої", "Мікроконтролери", тощо. Тому доскональне вивчення основ цифрової техніки, є важливим завданням, до якого майбутньому спеціалісту слід віднестись з особливою відповідальністю. В свою чергу маю сподівання що пропонований конспект лекцій зможе допомогти йому в цьому нелегкому процесі. Разом з тим слід відзначити, що конспект в силу обмеженості його об'єму не може повністю відображати всієї різноманітності сучасної цифрової техніки, тому для більш ґрунтовного розуміння предмету слід ознайомитись із літературою наведеною в кінці конспекту. Крім того, дуже корисним є вивчення технічної документації на цифрові ІС провідних світових виробників *Texas Instruments, Motorola, Philips, Toshiba, National Semiconductor* яка вільно розповсюджується за допомогою мережі Internet. Інший шлях самовдосконалення – це набуття практичного досвіду в цифровій схемотехніці. Зараз цей процес теж є значно простіший ніж ще 20 років тому, внаслідок наявності значної кількості "комп'ютерних тренажерів" – симуляторів роботи електронних схем. Для роботи студентів під час вивчення курсу "Основи цифрової техніки" з цією метою зручно користуватись такими симуляторами як *Spectrum MicroCAP*, та *Proteus*. Проведене "дослідження" роботи цифрових пристроїв за їх допомогою принесе користь при вивченні та зекономить значну кількість часу.

1. МАТЕМАТИЧНІ ОСНОВИ ЦИФРОВОЇ ТЕХНІКИ

1.1 Відображення інформації у цифровій техніці

Інформація (від лат *informatio* – роз'яснення, виклад, обізнаність) як об'єкт передання, поширення, перетворення, зберігання чи безпосереднього виконання потребує матеріального носія. Таким носієм інформації у радіоелектроніці є електричний сигнал - напруга або струм. Як функція часу електричний сигнал може бути неперервним або дискретним. Неперервний (або аналоговий) сигнал у заданому діапазоні зміни аргументу t набуває довільної множини значень, дискретний сигнал - лише при певних значеннях аргументу nt (де τ - інтервал дискретності, $n = 0,1,2,\dots$). Тому дискретний сигнал може мати тільки обмежене (скінченне) число рівнів напруги або струму. Ці рівні зафіксовані на однакових інтервалах часу nt , що називаються тактами (в музиці, до речі, тактами також називають інтервали часу, хоча там вони можуть бути не обов'язково однакової розмірності). У ЦТ основним носієм інформації є дискретний сигнал. Щоб він адекватно відображав інформацію, яка може бути подана у найрізноманітніших формах (числами, буквами, сигналами різної природи чи якимись символами), число рівнів і значення їх величин мають підпорядковуватися певній системі відліку, тобто системі числення, що характеризується певною сукупністю символів (алфавітом) і правил. Людина у повсякденному житті користується десятковою (арабською) системою числення, що складається з 10 цифр (0...9). Здавалося б, що й для відображення інформації з допомогою дискретного сигналу найкраще підходить десяткова система числення - досить лише мати шкалу з десяти однакових за величиною рівнів (квантів) напруг. Проте вона не стала основною в ЦТ з багатьох причин. Цифрові сигнали, створені на основі десяткової системи числення, технічно реалізувати досить важко і тому економічно невигідно, бо такого типу цифрові пристрої повинні мати десять рівнів напруги.

Найпростішою з усіх позиційних систем числення є двійкова, або бінарна система, що складається з двох цифр: 0 і 1. Звідси й походження слова-терміна "біт", що означає "двійкова цифра" тобто 0 або 1. Двійкова система дозволяє найефективніше відобразити інформацію з допомогою електричних цифрових сигналів, якщо, наприклад, низький рівень напруги або відсутність імпульсу позначити як логічний нуль (лог. 0), а високий рівень напруги або наявність імпульсу - як логічну одиницю (лог. 1).

Всю інформацію, таким чином, можна зобразити у вигляді ланцюжка 0 і 1, тобто набору чи комбінації бітів, що називають словом. Отже, послідовність 0 і 1 певної довжини (слово) може зображати будь-яке число, кожний розряд (біт) якого з одиницею інформації.

Задана кількість двійкових розрядів - бітів може не відповідати обсягу інформації, яка також вимірюється у бітах. Це дуже часто є причиною непорозумінь (через плутання цих двох різних понять). У цифровій та

обчислювальній техніці 1 біт означає один двійковий розряд - символ, що може набувати лише ціле число, тобто двійкове n - розрядне число, що дорівнює n біт. Але щодо обсягу інформації n -бітне число не обов'язково може мати n біт інформації, воно цілком може дорівнювати, наприклад, 0.76 біт або 0.1 біт. Як правило, чим менше число бітів (як двійкових розрядів) використовується для відображення певної інформації, тим більша густина інформації. Наприклад (див.таб.1.1):

Таблиця 1.1 Однакова кількість інформації в різнопозиційних числах

число з 4 позиціями	число з 1 позицією	число з 2 позиціями
0000	0	00
0001	1	01
0010	2	10
0100	3	11

З допомогою двійкових чисел у цифровій та мікропроцесорній техніці зображаються дані - відомості, факти тощо, все те, що необхідно для виконання певної дії.

Таблиця 1.2 Кодування в поширених системах числення

"2"	"8"	"10"	"16"
0	0	0	0
1	1	1	1
10	2	2	2
11	3	3	3
100	4	4	4
101	5	5	5
110	6	6	6
111	7	7	7
1000	10	8	8
1001	11	9	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F
10000	20	16	10
10001	21	17	11
.....
11001	31	25	19
11010	32	26	1A
11011	33	27	1B

Дані подаються у такому (формалізованому) вигляді, що після перетворення у фізичний носій інформації вже у вигляді цифрового сигналу

їх можна передавати, обробляти і зберігати за допомогою технічних носіїв чи засобів. Тому дуже часто замість слова "дані" вживається слово "інформація". Набір правил, що розкриває спосіб зображення даних з допомогою умовних знаків чи символів, називається кодом. Носіями інформації в електронних обчислювальних пристроях (ЕОП) є електричні сигнали, найчастіше — напруги. Інформація обробляється в цифровій формі. Кожній цифрі ставиться у відповідність визначений рівень сигналу.

Одиницею зображення даних, а також обміну даними, якою як єдиним цілим оперують між собою окремі цифрові пристрої чи вузли, є восьмирозрядне (восьмибітне) двійкове число - байт. Отже, 1 байт = 8 біт, і, очевидно, що з допомогою такого слова можна передати одне з $2^8 = 256$ різних повідомлень, тобто можна, наприклад, закодувати текст якоюсь мовою, де кожному символу (букві, розділовому знаку, цифрі тощо) має відповідати одна кодова комбінація з 8 біт, або 1 байт. На практиці доцільно знати і вміти користуватися (наприклад, для визначення обсягу пам'яті ЕОМ) такими одиницями інформації.

Систем числення може бути безліч. (Поряд із згаданими системами у пристроях ЦТ можна зустріти й такі, як, наприклад, трійкова ($P=3$). Очевидно, що вибір тої чи іншої системи числення залежить від певних критеріїв, які мають виконуватись для забезпечення оптимального результату. В загальному довільне число A у позиційній системі числення з постійною основою P може бути виражене числовим рядом:

$$A_p = a_n p^n + \dots + a_0 p^0 + a_{-1} p^{-1} + \dots + a_{-m} p^{-m} = \sum_{i=-m}^n a_i p^i \quad (1.1)$$

де n – число знаків до коми; m – число знаків після коми; a_i – число в i -му розряді; p – вага i -го розряду;

Двійкова система має набір цифр $\{0,1\}$, тому $p = 2$. Довільне число в A_2 двійковій системі можна зобразити виразом (1.1), результат обчислення якого – представлення цього числа в десятковій системі числення. Наприклад:

$$A_2 = 101,101_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = 5,625_{10}$$

Для практичних розрахунків слід запам'ятати значення перших 13-ти розрядів цифр двійкової системи числення:

$$2^0=1, 2^1=2, 2^2=4, 2^3=8, 2^4=16, 2^5=32, 2^6=64, 2^7=128, \\ 2^8=256, 2^9=512, 2^{10}=1024, 2^{11}=2048, 2^{12}=4096;$$

Шістнадцяткова система має набір цифр $\{0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F\}$, тому $p = 16$. При цьому великими буквами латинського алфавіту зображені цифри 10,11,12,13,14,15 ($_{10}$). Наприклад:

$$A_{16} = 504F_{16} = 5 \cdot 16^3 + 0 \cdot 16^2 + 4 \cdot 16^1 + 15 \cdot 16^0 = 20559_{10}$$

Для практичних розрахунків слід запам'ятати значення перших 6-ти розрядів цифр шістнадцяткової системи числення:

$$16^0=1, 16^1=16, 16^2=256, 16^3=4096, 16^4=65536, 16^5=1048576$$

Десяткова система числення, якою користуємось у повсякденному житті, не є найкращою з точки зору її технічної реалізації у цифрових пристроях. Відомі на сьогодні елементи, що мають десять стійких станів, щоб розрізнити всі десять символів (рівнів), наприклад, декатрони, мають невисоку швидкодію та складні для виготовлення. Отже, виникає закономірне питання: яка система числення найкраща? Вчені довели, що раціональною з точки зору мінімальних витрат умовного обладнання є система числення з основою $R \approx 2.72 \approx 3$. Отже, трійкова система краща за двійкову. Однак з'ясувалося, що за такими характеристиками цифрових пристроїв, як швидкодія, об'єм пам'яті, техніка виконання арифметичних та логічних операцій, найкращою є двійкова система числення.

Розглянемо, чому крім десяткової та двійкової систем числення в цифровій схемотехніці використовуються ще й такі системи числення, як вісімкова, шістнадцяткова і т. ін. Справа в тому, що якщо нас цілком влаштовує десяткова система, а для цифрового пристрою (мікропроцесорів, ЕОМ тощо) - виключно двійкова, бо там інформація обробляється тільки за допомогою нулів і одиниць, то проміжною, сполучною ланкою між нами і цифровим пристроєм має бути така система числення, яка була б найзручнішою як для взаємного переведення обох систем, так і для забезпечення простої технічної реалізації процедури. Для цього власне використовують проміжні системи числення - коди, за допомогою яких можна уникнути довгих ланцюжків з нулів та одиниць, які властиві двійковій системі. До таких систем кодування належить двійково-десятковий код (ДЦК).

Особливість ДЦК полягає в тому, що кожній десятковій цифрі даного розряду відповідає чотирибітове (група з чотирьох бітів) двійкове число - тетрада. Наприклад, об'єднання двох тетрад дає можливість зобразити двозначне десяткове число, а з точки зору інформативності - реалізувати $16 \times 16 = 256$ двійкових комбінацій. При такому числі комбінацій можна закодувати не тільки 80 різних друкованих знаків клавіатури друкарської машинки, але й усі букви латинського алфавіту, розділові знаки, різні символи тощо.

Але при перетворенні десяткових чисел ДЦК має надлишковість, бо при групуванні тетрадами шість комбінацій відпадає. Водночас при групуванні тріадами (трима бітами), навпаки, інформативність падає. Цей факт і пояснює зручність шістнадцяткової системи числення, бо для ДЦК як разі потрібно 16 бітових комбінацій, які повністю забезпечує одна тетрада ($1111_2 = 15_{10}$). Зрозуміло, що при групуванні тріадами найзручнішою буде вісімкова система числення - там як раз вистачає 8 бітових комбінацій ($111_2 = 7_{10}$).

Треба мати на увазі, що обидва коди - "16" і "8" - це тільки спосіб зображення двійкових чисел, якими оперують цифрові пристрої та мікропроцесори. Однак з точки зору інформатики код "16" має більші можливості, ніж код "8". Перевага коду "16" над кодом "8" ще в тому, що

мікропроцесори маніпулюють словами, які залежно від функціональних можливостей можуть мати набори з 4, 8, 16 або 32 біт.

ДДК називають ще кодом 8-4-2-1, бо ці цифри відповідають вагам розрядів однієї тетради відповідно 23 (старшого розряду), 22, 21 і 20 (молодшого розряду). До двійково-кодованої десяткової системи числення належать й інші системи кодування.

При синтезі таких комбінаційних цифрових пристроїв, як перетворювачі кодів, що призначені для перетворення одного коду в інший, необхідно мати дані про різновиди цих систем кодування.

У табл. 1.3 зведені найбільш поширені в ЦТ різновиди позиційних кодів. До них належать ДДК 8-4-2-1, код 2-4-2-1 (або код Айкена), код з лишком 3 (або код $[8-4-2-1]+3$), код Грея та циклічний код Джонсона.

ДДК 8-4-2-1, який утворений зображеннями кожної десяткової цифри вагами 23-22-21-20, в найбільш поширеним, тому його найчастіше застосовують для переводу в інший код.

ДДК з лишком 3($[8-4-2-1]+3$) застосовують для кодування декад при двійково-десятковому зображенні чисел. Особливістю цього коду є те, що всі тетради мають значення на три одиниці більше від тетради коду 8-4-2-1.

Таку властивість самодоповнення має код Айкена (2-4-2-1). Для цього коду ваги розрядів тетради відповідно дорівнюють 21-22-21-20. Таблиця кодування, як видно з табл. 2.2, поділяється на дві частини: 0-4 - це тетради, що повторюють еквіваленти; 5-9 - кожна тетрада у ДДК має лишок +0110. Така властивість коду Айкена дозволяє довільну цифру однієї частини таблиці перетворити на її доповнення до 9 простим інвертуванням.

Особливе місце серед позиційних систем кодування посідають код Грея і циклічний код Джонсона. Це спеціальні коди, які мають непостійні ваги розрядів. Якщо у двійковому коді при переході від зображення одного числа до зображення наступного може відбуватись одночасна зміна цифр у кількох його розрядах, то в кодах Грея і Джонсона сусідні числа відрізняються цифрою (1 або 0) тільки в одному розряді. Така особливість цих кодів запобігає появі джерела помилки у роботі апаратури у тих випадках, коли здійснюється послідовний перехід (зміна) числа з одного такту в наступний (наприклад, від 6 до 5),

У циклічному коді Джонсона перехід до наступного числа здійснюється шляхом послідовної заміни 0 на 1 до заповнення всіх розрядів одиницями, а потім заміною 1 на 0 до заповнення нулями. Помилки за допомогою кода Джонсона виявляються тоді, коли у "хвилі нулів" цього коду з'являється одна або кілька одиниць, а в "хвилі одиниць" - відповідно один або кілька нулів.

Окреме місце у ЦТ займає кодування алфавітно-цифрової інформації. Сюди належить текстова інформація, що являє собою послідовність алфавітно-цифрових символів, кожний з яких підлягає кодуванню. Щоб не було подвійного тлумачення, тобто щоб кожному символу відповідав тільки один код, застосовують спеціально призначені для цього стандартні коди. Зокрема, найчастіше в ЕОМ типу IBM застосовують код ASCII (American

Standard Code of Information Interchange-американський стандартний код обміну інформацією), де, наприклад, десятковому числу 1 відповідає код 00010000(25) числу 2 - код 00010001, числу 3 - код 00010010 тощо, а букві А - код 00011000, букві В код 00011001, букві С - код 00011010 і т. ін.

Таблиця 1.3 Приклади двійкових кодів

ДЕС. ЦИФР А	ДДК 8-4-2-1	ДДК 2-4-2-1	ДДК /8-4-2-1/+3	КОД ГРЕЯ	КОД ДЖОН СОНА	ОБЧИСЛ. КОД ДВІЙКОВ . ЧИСЛА	ДОПОВНЯ ЛЬНИЙ КОД ДВІЙКОВ ОГО ЧИСЛА
0	0000	0000	0011	0000	000001111100000
1	0001	0001	0100	0001	10000	11110	11111
2	0010	0010	0101	0011	11000	11000	11110
3	0011	0011	0110	0010	11100	11100	11101
4	0100	0100	0111	0110	11110	11110	11100
5	0101	1011	1000	0111	11111	11111	11011
6	0110	1100	1001	0101	01111	01111	11010
7	0111	1101	1010	0100	00111	00111	11001
8	1000	1110	1011	1100	00011	00011	11000
9	1001	1111	1100	1101	00001	00001	10111
10	0001 0000	0001 0000	0001 0011	1111	00000	00000	10110
11	0001 0001	0001 0001	0001 0100	1110	10000	10000	10101
12	0001 0010	0001 0010	0001 0101	1010	11000	11000	10100
13	0001 0011	0001 0011	0001 0110	1011	11100	11100	10011
14	0001 0100	0001 0100	0001 0111	1001	11110	11110	10010
15	0001 0101	0001 1011	0001 1000	1000	11111	11111	10001
16	0001 0110	0001 1100	0001 1001	11000	01111	01111	10000
17	0001 0111	0001 1101	0001 1010	11001	00111	00111	01111
.....

1.2 Перетворення числової інформації

У процесі перетворення інформації, яку несуть цифрові сигнали, у цифровій системі (пристрої) виникає необхідність переведення чисел з однієї системи числення в іншу еквівалентну систему. Це означає що, наприклад, десятковому числу A_{10} має відповідати двійкове число A_2

$$A_{10} = \sum_{i=-m}^{n-1} a_i (10)^i = \sum_{j=-k}^{l-1} a_j (2)^j = A_2 \quad (1.2)$$

де $a_j = 0, 1, 2, \dots, 9$ - цифра в i -му розряді $(m+n)$ -розрядного числа A_{10} ; $a_j = 0$ або $a_j = 1$ - цифр в j -му розряді $(k+1)$ -розрядного двійкового числа A_2 .

Перевід чисел з будь-якої позиційної системи числення у код "10" вже розглядався у підрозд.1.1. Для цього досить скористатися виразом (1.1), з

допомогою якого розрахована сума ряду дає шуканий результат.

Розглянемо тепер перевід чисел у різних системах числення. Загальним правилом переводу числа з однієї системи числення до іншої є виконання таких послідовних кроків:

1) ділення в цілих числах заданого числа AP на основу P тої системи, в яку переводиться дане число AP ;

2) якщо частка не дорівнює нулю, її слід взяти за нове число і повторити операцію п.1; якщо частка дорівнює нулю, перейти до п.3;

3) вписати всі остачі у зворотному порядку, починаючи з останньої, і взяти їх за цифри шуканого числа $a_{n-1} a_{n-2} \dots a_1 a_0$

За такими самими правилами виконуються взаємні перетворення (переводи) в інших системах числення. Тільки при цьому ділити або множити потрібно на число, яке дорівнює основі тієї системи числення, до якої треба перейти. Розглянемо приклад переводу числа 118_{10} з десяткової системи числення в двійкову:

$$\begin{array}{r}
 118 \mid 2 \\
 \hline
 59 \mid 2 \\
 \hline
 29 \mid 2 \\
 \hline
 14 \mid 2 \\
 \hline
 7 \mid 2 \\
 \hline
 3 \mid 2 \\
 \hline
 1 \mid 1 \\
 \hline
 0 \mid 1
 \end{array}$$

\uparrow
 0 1 1 1 0 1 1 0 2

Записуючи неподільну частку і залишки в порядку, зворотному їхній появі, знаходимо що $118_{10} = 1110110_2$

1.3 Двійкова арифметика

У цифрових та мікропроцесорних пристроях над двійковими числами виконуються як логічні, так і арифметичні операції. Арифметичні операції (додавання, віднімання, множення, ділення) над двійковими числами здійснюються (реалізуються) з допомогою спеціальних алгоритмів, які не використовуються у десятковій системі числення. У цих алгоритмах переважає операція додавання як додатних, так і від'ємних чисел, яку досить просто реалізувати.

Цифрові пристрої оперують тільки цілими і дійсними числами. Останні можуть бути подані з фіксованою або з плаваючою комою (точкою) (2;4). Незалежно від форм зображення знаки дійсних чисел "+" і "-" в ЦТ кодують цифрами - відповідно нулем (0) і одиницею (1).

Основною арифметичною операцією, яка використовується в цифрових пристроях для виконання різних обчислень, є операція алгебраїчного додавання двійкових чисел. Операція віднімання легко виконується через додавання, якщо змінити знак від'ємника на протилежний, а саме: $A-B = A+(-B)$. Операції множення і ділення також виконуються з допомогою операції

додавання та деяких логічних дій при застосуванні зсуву часткових результатів ліворуч або праворуч

Зупинимось на операції додавання двійкових чисел.

Операція додавання в цифрових пристроях виконується порозрядно, починаючи з молодших розрядів доданків. При цьому в кожному одноіменному розряді доданків підсумовуються відповідні цифри та переноситься попередній розряд суми. Додавання молодших розрядів двійкових чисел здійснюється лише з двома доданками:

- $0+0=0$; $0+1=1$; $1+0=1$; $1+1=10$,

де у числі 10 цифра 1 - перенос у наступний (старший) розряд.

Пристрій, що виконує операцію додавання двійкових чисел, називається суматором, а пристрій, що реалізує підсумовування молодших розрядів двійкового числа, - напівсуматором. Порозрядне підсумовування двох чисел як суматор, так і напівсуматор виконують за модулем 2 згідно з правилом $0\oplus 0=0$; $0\oplus 1=1$; $1\oplus 0=1$; $1\oplus 1=0$, яке це застосовують для порівняння двох чисел.

Для виконання операції додавання від'ємних чисел застосовуються спеціальні коди - обернений та доповняльний. Обернений код $\overline{A_2}$ від'ємного числа A_2 отримується при інвертуванні всіх цифр у кожному розряді, тобто заміні нуля одиницею, одиниці нулем, а у знаковому розряді ставиться одиниця. Отже, обернений код числа $A_{10} = -28$ буде $\overline{A_2} = 1/1100011$. Якщо виконувати операцій $A_2 - B_2 = A_2 + \overline{B_2}$, тобто зображати від'ємне число B_2 оберненим кодом $\overline{B_2}$, може виникнути одиниця переносу, яку потрібно додати до молодшого розряду одержаної суми. Однак цю операцію, яка називається циклічним переносом, технічно реалізувати не вигідно, бо вона вимагає ще однієї операції додавання, що істотно збільшує час виконання дії. Тому для додавання від'ємних чисел перевага надається доповняльному коду числа $\overline{A_2}$, який утворюється від оберненого додаванням одиниці до його молодшого розряду. При цьому відпадає необхідність у циклічному переносі одиниці та ще одного додавання. У випадку переповнення у знаковому розряді одиниця переносу просто відкидається і не враховується. Тому всі від'ємні числа, які використовуються для виконання різних арифметичних дій, подають у доповняльному коді. Отже, доповняльний код $\overline{A_2}$ від'ємного n -розрядного числа A_2 одержується з оберненого $\overline{A_2}$ як $\overline{A_2} = \overline{A_2} + 1$ (у молодшому розряді), де $\overline{A_2} = 1, \overline{X_n} \overline{X_{n-1}} \dots \overline{X_1} \overline{X_0}$ - від'ємне число в оберненому коді.

Якщо у знаковому розряді суми отримано одиницю, тобто результат від'ємний, значення отриманого числа є у доповняльному коді, а якщо - нуль, результат додавання отримано у прямому коді. Сума доповняльних кодів двійкових чисел має доповняльний код результату. Отже, віднімання чисел довільного знака можна звести до операції додавання:

$$A_2 - B_2 = A_2 + (-B_2) = A_2 + \overline{B_2} + 1 = A_2 + \overline{B_2}$$

1.4 Основні поняття та закони бульової алгебри

У зв'язку з двійковим зображенням цифрових сигналів, що набувають тільки двох значень /0 і 1/, найзручнішим математичним апаратом для аналізу та синтезу цифрових систем є алгебра логіки (бульова алгебра). У бульовій алгебрі символи 0 і 1 характеризують стани змінних та їх функцій і тому не можна розглядати ці символи як арифметичні числа. Алгебра логіки - це алгебра станів, а не алгебра чисел, і їй властиві, на відміну від звичайної алгебри- логічні дії над станами.

Основне поняття алгебри логіки - висловлення, тобто речення, в якому міститься суть (сенс) твердження істинності або його заперечення (хибності). Будь-яке висловлення можна позначити символом, наприклад, X або Y , і вважати, що, $X = 1$ або $Y = 1$, якщо висловлення істинне, а $X = 0$ або $Y = 0$, якщо висловлення неістинне (хибне). Оскільки будь-яка змінна (або її функція) може мати стан 0 або 1, в алгебрі логіки кожній двійковій змінній ставиться у відповідність обернена до неї (інверсна) змінна, така, що якщо $X=0$, то $\overline{X}=1$, а якщо $X=1$, то $\overline{X}=0$. Подвійне заперечення дає аргумент, тобто $\overline{\overline{X}} = X$. Риска "-" над змінною X означає, що над змінною \overline{X} здійснено операцію логічного заперечення. Ця елементарна і дуже важлива у бульовій алгебрі логічна операція називається інверсією (логічним запереченням). Вона означає, що якщо висловлення ($X=1$) істинне, то висловлення "НЕ X " (\overline{X}) - хибне ($\overline{X}=0$), а якщо висловлення X - хибне ($X=0$), то висловлення \overline{X} - істинне ($\overline{X}=1$).

Логічна функція $Y = f(X_1, X_2, \dots, X_n)$ - це складне висловлення з кількох простих, які пов'язані між собою логічними операціями. Логічна функція Y набуває значення 0 або 1 ($Y \in \{0,1\}$) при наборі логічних двійкових змінних (аргументів) $X_i \in \{0,1\}$.

Якщо кожному значенню аргументу відповідає тільки одне значення логічної функції, така функція називається однозначною, якщо два або більше - багатозначною. Однозначна функція $Y = f(X)$ може бути зображена тільки двома аргументами (0 і 1) і двома своїми значеннями (0 і 1). Отже, число однозначних функцій у цьому випадку може бути тільки $2^2=4$. Особливо цікава одна з них - інверсія $Y = \overline{X}$, решта - тривіальні.

Вхідний набір - це певна комбінація значень двійкових змінних X_i у логічній функції Y . Максимальне число вхідних наборів визначається як $N = 2^n$ (де n - число змінних), максимальне число логічних функцій n змінних - як $M = 2^n = 2^{2^n}$.

Двозначні функції $Y = f(X_1, X_2)$ мають $N=4$ можливих значень аргументів / $X_1, \overline{X_1}, X_2, \overline{X_2}$ / і два різних значення функцій (Y і \overline{Y}), що в результаті дає

24=16 різних двозначних булевих функцій. Число наборів аргументів $2^2=4$. а їх значення такі: $\{X_1X_2\}, \{X_1\bar{X}_2\}, \{\bar{X}_1X_2\}, \{\bar{X}_1\bar{X}_2\}$. У випадку функції трьох змінних $f(X_1, X_2, X_3)$ різних наборів аргументів буде

$$2^3 = 8 : \{X_1X_2X_3\}, \{X_1X_2\bar{X}_3\}, \{X_1\bar{X}_2X_3\}, \{X_1\bar{X}_2\bar{X}_3\}, \{\bar{X}_1X_2X_3\}, \{\bar{X}_1X_2\bar{X}_3\}, \{\bar{X}_1\bar{X}_2X_3\}, \{\bar{X}_1\bar{X}_2\bar{X}_3\}$$

а різних значень тризначної функції буде вже $2^8=256$.

Логічна функція, яка має певні значення 0 або 1 на всіх вхідних наборах, називається повністю визначеною функцією. Якщо логічна функція має значення, які на деяких вхідних наборах не визначені, їх називають байдужими (або непевними). Частково визначену логічну функцію можна зробити повністю визначеною (довизначеною) приписуванням байдужим наборам довільних значень функції.

Множину логічних Функцій n змінних можна утворити з допомогою трьох основних логічних операцій:

- логічного заперечення (“-”) - інверсії;
- логічного додавання (“v”) – диз’юнкції;
- логічного множення (“.”) кон’юнкції.

Для згаданих операцій справедливі такі аксіоми (тотожності або правила):

- універсальна множина - $\{1 \cdot X = X; 1 \vee X = 1\};$
- нульова множина - $\{0 \cdot X = 0; 0 \vee X = X\};$
- повторення (тавтологія) - $\{X \cdot X \dots X = X; X \vee X \vee \dots \vee X = X\};$
- доповнення - **{Ошибка! Объект не может быть создан из кодов полей редактирования.;** $X \vee$ **Ошибка! Объект не может быть создан из кодов полей редактирования.;**
- подвійна інверсія - $\overline{\overline{X}} = X$.

В алгебрі логіки діє принцип дуальності, згідно з яким дві функції рівносильні одна одній, якщо на всіх можливих наборах змінних вони набувають одного і того самого значення, тобто $f_1(X_1, X_2, \dots, X_n) = f_2(\bar{X}_1, \bar{X}_2, \dots, \bar{X}_n)$. Принцип дуальності (двоїстості) є основним принципом булевої алгебри. Він має велике практичне значення, що визначило одну з переваг цифрової техніки. Застосовуючи його, можна будувати нові логічні функції, логічні елементи та пристрої, причому досить просто. Властивість дуальності /двоїстості/ булевої функції є основою наступних двох правил булевої алгебри:

правило К. Шеннона стверджує, що для одержання алгебраїчного виразу інверсної функції необхідно у згаданій функції всі змінні замінити на інверсні їм, всі знаки кон’юнкції - на знаки диз’юнкції, а всі знаки диз’юнкції - на знаки кон’юнкції;

правило де Моргана стверджує, що інверсія кон’юнкції дорівнює диз’юнкції інверсій, а інверсія диз’юнкції – кон’юнкції інверсій.

У булевій алгебрі також діють закони, за якими можна встановити аналітичні зв’язки між різними логічними функціями і виконувати відповідні перетворення для спрощення складних виразів. Основні закони булевої

алгебри зведені в табл.1.4. Справедливість аксіом і законів легко перевірити прямою підстановкою нуля або одиниці на місце конкретного аргументу.

Таблиця 1.4 Основні закони булевої алгебри

Закони комутативності /переміщення/	$X_1 X_2 = X_2 X_1$ $X_1 \vee X_2 = X_2 \vee X_1$
Закони асоціативності /сполучення/	$X_1 (X_2 X_3) = (X_1 X_2) X_3 = X_1 X_2 X_3$ $X_1 \vee (X_2 \vee X_3) = (X_1 \vee X_2) \vee X_3 = X_1 \vee X_2 \vee X_3$
Закони дистрибутивності /розподілу/	$X_1 (X_2 \vee X_3) = X_1 X_2 \vee X_1 X_3$ $X_1 \vee X_2 X_3 = (X_1 \vee X_2)(X_1 \vee X_3)$
Закони склеювання	$(X_1 \vee X_2)(X_1 \vee \bar{X}_2) = X_1$ $X_1 X_2 \vee X_1 \bar{X}_2 = X_1$
Закони поглинання	$X_1 (X_1 \vee X_2) = X_1$ $X_1 \vee X_1 X_2 = X_1$
Закони дуальності /правило де Моргана/	$\overline{X_1 X_2} = \bar{X}_1 \vee \bar{X}_2$ $\overline{X_1 \vee X_2} = \bar{X}_1 \bar{X}_2$

Застосовуючи аксіоми (тотожності) та закони булевої алгебри, можна отримати нові логічні формули, а також довести справедливість того чи іншого закону на основі інших.

Особливої уваги заслуговують закони дуальності (правило де Моргана), з допомогою яких кон'юнкцію можна виразити через диз'юнкцію, і навпаки. Таку операцію часто треба здійснювати при перетвореннях Логічних виразів. Корисними для практики можуть бути також наслідки законів дуальності, зокрема:

$$X_1 \vee X_2 = \overline{\bar{X}_1 \bar{X}_2}; \quad (1.2)$$

$$X_1 X_2 = \overline{\bar{X}_1 \vee \bar{X}_2}; \quad (1.3)$$

Закони дуальності та наслідки з них справедливі для довільного числа змінних:

$$\overline{X_1 \vee X_2 \vee \dots \vee X_n} = \bar{X}_1 \bar{X}_2 \dots \bar{X}_n,$$

$$\overline{X_1 X_2 \dots X_n} = \bar{X}_1 \vee \bar{X}_2 \vee \dots \vee \bar{X}_n,$$

1.5 Властивості логічних функцій

Властивості логічних функцій розглянемо на прикладі двозначної логічної функції $Y = f(X_1, X_2)$ яка на практиці зустрічається найчастіше. Для цього складемо табл. 1.5 всіх можливих наборів аргументів у вигляді двійкових чисел: 00, 01, 10, 11; всі 16 значень функції розміщено у порядку зростання двійкових чисел (від 0000 до 1111).

Таблиця 1.5 Двозначні логічні функції

ФУНКЦІЯ	НАБІР АРГУМЕНТІВ				НАЗВА ЛОГІЧНОЇ ФУНКЦІЇ	ВИЗНАЧЕННЯ ЛОГІЧНОЇ ФУНКЦІЇ
	0	0	1	1		
f_1	0	0	0	0	Константа нуль	0
f_2	0	0	0	1	Кон'юнкція	$X_1 X_2$
f_3	0	0	1	0	Заборона X_2	$X_1 \bar{X}_2$
f_4	0	0	1	1	Повторення X_1	X_1
f_5	0	1	0	0	Заборона X_1	$\bar{X}_1 X_2$
f_6	0	1	0	1	Повторення X_2	X_2
f_7	0	1	1	0	Виняткове АБО	$X_1 \oplus X_2 = X_1 \bar{X}_2 \vee \bar{X}_1 X_2$
f_8	0	1	1	1	Диз'юнкція	$X_1 \vee X_2$
f_9	1	0	0	0	Функція Пірсона	$\bar{X}_1 \vee \bar{X}_2$
f_{10}	1	0	0	1	Рівнозначність	$X_1 \sim X_2 = X_1 X_2 \vee \bar{X}_1 \bar{X}_2$
f_{11}	1	0	1	0	Інверсія X_2	\bar{X}_2
f_{12}	1	0	1	1	Імплікація від X_2 до X_1	$X_1 \vee \bar{X}_2$
f_{13}	1	1	0	0	Інверсія X_1	\bar{X}_1
f_{14}	1	1	0	1	Імплікація від X_1 до X_2	$\bar{X}_1 \vee X_2$
f_{15}	1	1	1	0	Функція Шефера	$\bar{X}_1 X_2$
f_{16}	1	1	1	1	Константа одиниця	1

Всі наведені 16 логічних Функцій на практиці не застосовуються, бо, як видно з табл. 1.5, яка, до речі, має симетрію, функції f_1 і f_{16} тривіальні, f_4 , f_6 , f_{11} , f_{13} - не залежать від одного з аргументів. Хоча решта десять функцій залежать від двох аргументів, однак і серед них є такі, які можна одержати через інші.

Аналогічно арифметичним операціям у бульовій алгебрі також існує

поняття "першості виконання" операції, що визначає, яка з логічних операцій має виконуватися раніше. Отже, для логічних операцій першість визначається у такій послідовності: 1 - заперечення, 2 - кон'юнкція, 3 - диз'юнкція, 4 - імплікація, 5 - рівнозначність. При наявності у виразі логічної функції круглих дужок ступінь першості збільшується на одиницю.

Зауважимо, що деякі з наведених у табл. 1.5 функцій одержують методом перенумерації (перейменування або декомпозиції) аргументів логічних функцій. Наприклад, функція f_{12} отримується з функції f_{14} , якщо X_1 перенумерувати на X_2 і навпаки, беручи набір аргументів справа наліво або зліва направо. Функцію f_{15} можна отримати з функції f_{13} підстановкою замість X_1 іншою функції f_2 (тобто проінвертувавши набір аргументів). Така операція називається суперпозицією. Отже, застосовуючи метод суперпозицій, можна одержати більш складні логічні функції. При цьому виникає питання, чи можливий набір більш простих функцій, за допомогою яких можна було б отримати як завгодно складну логічну функцію. З практичної точки зору це дуже важливе питання, бо воно стосується технології виготовлення мікросхем і т. ін. З теоретичної точки зору воно пов'язане з основним поняттям бульової алгебри - функціональною повнотою системи логічних функцій.

Набір (система) булевих функцій вважається функціонально повним, якщо на його основі або на базисі можна отримати довільну бульову Функцію, застосовуючи лише метод суперпозиції.

Функціонально повних наборів-базисів можна отримати досить багато. Найбільш поширені серед них наведені в табл. 1.6.

Найпростішим /елементарним/ базисом, що є основою бульової алгебри, є набір трьох основних логічних функцій (або операцій): f_2 і f_8, f_{11} або f_{13} , на яких зупинимось більш детально (табл. 1.5, 1.6):

інверсія - логічне заперечення або функція НЕ; ця функція згадувалася раніше як однозначна, а тепер розглядається як двозначна (f_{11} і f_{13}), хоча залежить тільки від одної з двох змінних,

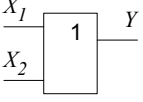
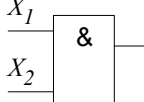
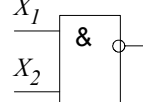
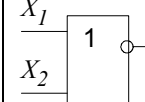
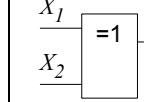
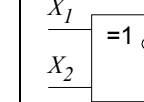
диз'юнкція - логічне додавання або функція АБО, яка істинна тоді, коли істинні або X_1 , або X_2 , або обидві змінні;

кон'юнкція - логічне множення або функція І, яка істинна тільки тоді, коли і X_1 і X_2 істинні.

До більш спрощених базисів, з допомогою яких можна побудувати будь-яку складну цифрову систему обробки інформації, належить, наприклад, набір з двох елементарних логічних функцій f_8 і f_{13} або f_2 і f_{13} і навіть набір лише з одної логічної функції f_9 або f_{15} . Решту логічних функцій, які відсутні у цих базисах, можна одержати на основі правил (законів) алгебри логіки.

Елементи, що реалізують бульові (логічні) операції, називаються логічними елементами (ЛЕ). Якщо логічні операції і прийнято зображати у вигляді формул, то ЛЕ - графічно у вигляді схем. Умовне графічне позначення ЛЕ прийнято зображати прямокутником, у якого лінії зліва - входи аргументів X_i , справа - функція Y . Тип логічної операції задається спеціальною позначкою: інверсія - кружком на вході або виході (ЛЕ - інвертор), диз'юнкція - 1, кон'юнкція - & (табл. 1.6).

Таблиця 1.6 Основні логічні функції.

Вхідні змінні		Диз'юнкція	Кон'юнкція	Штрих Шефера	Стрілка Пірса	Виняткове АБО:	Виняткове АБО-НЕ
X_1	X_2	АБО: $Y = X_1 \vee X_2$ $= X_1 + X_2$	І: $Y = X_1 X_2 =$ $= X_1 \wedge X_2$	І-НЕ: $Y = \overline{X_1 X_2} =$ $= X_1 / X_2$	АБО-НЕ $Y = \overline{X_1 \vee X_2}$ $= X_1 \downarrow X_2$	$Y = X_1 \oplus X_2 =$ $= X_1 \overline{X_2} \vee \overline{X_1} X_2$	$Y = X_1 \sim X_2 =$ $X_1 X_2 \vee \overline{X_1} \overline{X_2}$
0	0	0	0	1	1	0	1
0	1	1	0	1	0	1	0
1	0	1	0	1	0	1	0
1	1	1	1	0	0	0	1
Назва і умовне /схемне/ позначення		Диз'юнктор 	Кон'юнктор 	Елемент Шефера 	Елемент Пірса 	Суматор 	Суматор інвертор 

Логічні функції багатьох змінних одержують аналогічно розглянутому випадку $f(X_1, X_2)$ застосуванням методу суперпозиції та аксіом і законів алгебри логіки. Слід зауважити, що базисні функції обов'язково містять у собі операцію інверсії. Побудовані на їх основі логічні елементи (наприклад, на елементах Шефера або Пірса) дозволяють будувати функціональні вузли цифрових систем будь-якої складності.

Особливий інтерес для практики має функція f_7 сума за модулем 2, яку ще називають "виняткове АБО". Логічний елемент-суматор за модулем 2, що реалізує цю функцію, широко застосовують у різних цифрових функціональних пристроях комбінаційного типу.

Особливість функції "виключне АБО" в тому, що вона збігається з функцією АБО в усіх випадках, за винятком, одного, коли всі змінні набувають значення одиниці, а саме при $X_1 = X_2 = 1$. З цієї причини, очевидно, ця функція й називається "виняткове АБО" Символ \oplus псевдоплюс означає, що змінні (аргументи) X_1 і X_2 пов'язані логічною функцією "виняткове АБО" яка істинна тоді, коли одна із змінних (X_1 або X_2) є істинною:

$$X_1 \oplus X_2 = X_1 \overline{X_2} \vee \overline{X_1} X_2 = (X_1 \vee X_2)(\overline{X_1} \vee \overline{X_2})$$

Виняткове АБО (цю функцію ще називають нерівнозначністю або нееквівалентністю) має властивості:

$$\text{комутативності } X_1 \oplus X_2 = X_2 \oplus X_1;$$

$$\text{асоціативності } X_1 \oplus (X_2 \oplus X_3) = (X_1 \oplus X_2) \oplus X_3;$$

$$\text{дистрибутивності відносно кон'юнкції } X_1(X_2 \oplus X_3) = (X_1X_2) \oplus (X_1X_3).$$

Справедливі також аксіоми $X \oplus 0 = X$; $X \oplus 1 = \bar{X}$; $X \oplus \bar{X} = 1$; $X \oplus X = 0$.

На основі властивостей і аксіом функції "виняткове АБО" можна одержати функцію елементарного базису:

$$\text{НЕ - } Y = \bar{X}_1 = X_1 \oplus 1; \quad \text{І - } Y = X_1X_2 = (X_1 \oplus X_2) \oplus X_1 \nu X_2;$$

$$\text{АБО - } Y = X_1 \nu X_2 = X_1 \oplus X_2 \oplus X_1X_2$$

Для функції "виняткове АБО-НЕ" (рівнозначність або еквівалентність) справедлива така рівність:

$$X_1 \sim X_2 = \overline{X_1 \oplus X_2} = X_1X_2 \nu \bar{X}_1\bar{X}_2 = (\bar{X}_1 \nu X_2)(X_1 \nu \bar{X}_2) = \bar{X}_1 \oplus X_2 = X_1 \oplus \bar{X}_2$$

1.6 Форми зображення логічних функцій

Будь-яку логічну функцію можна задати або зобразити у різних формах: словесно, таблично, числового запису, аналітичне і координатної карти чи діаграми. Розглянемо кожен з цих форм окремо.

Словесне зображення - це логічне висловлення, під яким розуміють довільне твердження, щодо якого можна сказати, істинне воно або хибне. Наприклад, функцію $f_{14} = \bar{X}_1 \nu X_2$ - імплікацію від X_1 до X_2 ($X_1 \rightarrow X_2$) словесно можна зобразити (описати) так (див. табл. 1.5): функція f_{14} хибна тоді і тільки тоді, коли X_1 істинне і X_2 хибне.

Табличне зображення характеризується таблицею істинності, яка має 2^n рядків за числом вхідних наборів, n - стовпців за числом змінних X_1, X_2, \dots, X_n т - стовпців за числом функцій (Y_1, Y_2, \dots, Y_m) . У випадку однозначної функції число стовпців таблиці істинності дорівнює $n+1$, а для багатозначної функції $n+ t$. Приклад однозначної функції трьох змінних $Y = f(X_1, X_2, X_3)$, яку зображено таблицею істинності, наведеної у табл. 1.7. Функцію, зображену таблицею істинності, можна також "читати" аналогічно, як при словесному зображенні.

Таблиця 1.7 Таблично задана логічна функція

НОМЕР НАБОРУ	X_1	X_2	X_3	Y
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0

6	1	1	0	0
7	1	1	1	1

Кожний спосіб чи форма зображення має свою специфіку і тому буде зручною там, де даний спосіб є найбільш простим чи компактним. Таблична форма запису хоча й наочна і може бути застосована для довільного числа змінних, однак при аналізі властивостей функції такий запис не є компактним. Для випадку чотирьох і більше змінних табличне зображення логічної функції є суттєво складнішим та громіздким. Простішим буде числовий запис, коли логічна функція задається у вигляді послідовності десяткових чисел, що є еквівалентами тих вхідних наборів, на яких дана функція набуває значення 1 або 0. Наприклад, таблицю істинності (табл. 1.7) легко переписати у числову, форму: для вхідних наборів, на яких $Y = 1$ маємо $Y = Y(0,3,4,7)$, а для наборів, на яких $Y = 0$, маємо $Y = \overline{Y}(1,2,5,6)$ або для $Y = 0$ маємо $Y = \&/1,2,5,6/$, а для $Y = 1$ $Y = \&/0,3,4,7/$, де кожний з записів повністю зображує логічну функцію, подану табл. 1.7, і, як бачимо, а більш компактним.

При аналітичному зображенні функція задається алгебраїчним виразом, який отримують при застосуванні логічних операцій до змінних бульової алгебри. Якщо попередні форми зображення логічної функції лише вказують значення функції на конкретних наборах змінних, то аналітичне зображення, крім того, ще дозволяє виконувати різні аналітичні перетворення, які необхідні як для аналізу, так і для синтезу цифрового пристрою, що реалізує або має реалізувати дану логічну функцію.

Нехай на фіксованому вхідному наборі $\{X_1, X_2, \dots, X_n\}$ задана логічна функція $Y = f(X_1, X_2, \dots, X_n)$. Оскільки кожна змінна $\overline{X}_i \in \{X_i, \overline{X}_i\} = \{1, 0\}$, набір значень змінних, очевидно, є конкретним двійковим числом $\{\overline{X}_1, \overline{X}_2, \dots, \overline{X}_n\}$. За номер вхідного набору можна взяти довільне число

$$i = 2^{n-1} X_1 + 2^{n-2} X_2 + \dots + 2^0 X_n$$

В інших випадках, наприклад, коли набір $\{X_{n-1} X_{n-2} \dots X_1 X_0\}$ зображує двійкове число, більш зручним є запис номера набору як:

$$i = 2^{n-1} X_{n-1} + 2^{n-2} X_{n-2} + \dots + 2^0 X_0$$

Найбільш раціональним є зображення логічних функцій в так званих нормальних (канонічних) формах запису, зокрема, у вигляді диз'юнкції кон'юнкцій або кон'юнкції диз'юнкцій змінних, що взяті з інверсіями або без них. Окремі вирази функції змінних $\overline{X}_1 \overline{X}_2 \dots \overline{X}_n$ називають термами, а самі (однозначні) функції $f_i(\overline{X}_1, \overline{X}_2, \dots, \overline{X}_n)^n$ - конституентами.

Кон'юнктивний терм (мінтерм або конституента одиниці) m_i - це терм, що зв'язує всі змінні, які зображені у прямій або у інверсній формі знаком кон'юнкції, причому

$$m_i = f_i(\overline{X}_1, \overline{X}_2, \dots, \overline{X}_n) = \{1, \text{якщо номер набору дорівнює } i;$$

$$0, \text{якщо номер набору не дорівнює } i \};$$

$$\text{Наприклад: } m_2 = f_2(0,1,0) = \overline{X}_1 X_2 \overline{X}_3$$

$$\text{Або } m_{13} = f_{13}(0,1,1,0,1) = \overline{X}_1 X_2 X_3 \overline{X}_4 X_5$$

Диз'юнктивний терм (макстерм або конституента нуля) M_i - це терм, що зв'язує всі змінні, які зображені у прямій або у інверсній формі знаком диз'юнкції, причому

$$M_i = f_i(\overline{X}_1, \overline{X}_2, \dots, \overline{X}_n) = \{0, \text{якщо номер набору дорівнює } i;$$

1, якщо номер набору не дорівнює i ;

$$\text{Наприклад: } M_7 = f_7(0,1,1,1) = X_1 \nu \overline{X}_2 \nu \overline{X}_3 \nu \overline{X}_4$$

$$\text{Або } M_0 = f_0(0,0) = X_1 \nu X_2$$

Ранг терма r визначається кількістю змінних, які входять у даний терм. Наприклад, мінтерм m_2 має $r = 3$, а m_{13} має $r = 5$, макстерм M_7 має $r = 4$, а M_0 має $r = 2$.

Згідно із законом дуальності будь-яку логічну функцію n змінних можна зобразити як диз'юнкцію кон'юнкцій (мінтерм) або як кон'юнкцію диз'юнкцій (макстерм) змінних. Однозначне зображення логічних функцій одержують тільки при так званих удосконалених нормальних формах (УНФ), тобто таких, при яких мінтерми або макстерми формуються з усіх аргументів логічної функції і є одного (причому максимального) рангу. Якщо логічна функція складається з набору диз'юнкцій елементарних кон'юнкцій одного рангу, її називають удосконаленою диз'юнктивно-нормальною формою (УДНФ), а якщо з набору кон'юнкцій елементарних диз'юнкцій - удосконаленою кон'юнктивно-нормальною формою (УКНФ) зображення. Для n -змінних складається 2^n мінтерм m_1, m_2, \dots, m_n (при УДНФ) або 2 макстерм M_1, M_2, \dots, M_N (при УКНФ).

В загальному випадку алгебраїчний вираз логічної функції в УДНФ набував вигляду

$$Y_{\text{УДНФ}} = \bigcup_{i=1}^{2^{n-1}} f_i m_i \quad (1.5)$$

де f_i, m_i - значення функції (0 або 1) та мінтерма на i -му наборі змінних.

В УКНФ загальний вираз логічної функції

$$Y_{\text{УКНФ}} = \prod_{i=1}^{2^{n-1}} (f_i \nu M_i) \quad (1.6)$$

де f_i, M_i - значення функції (0 або 1) та макстерма на i -му наборі змінних.

Застосовуючи закони алгебри логіки (див. табл.1.4), неважко довести еквівалентність обох форм зображення будь-якої логічної функції. Такі форми зображення логічних функцій необхідні при проектуванні цифрових пристроїв.

Для згаданого прикладу (див. табл.1.7) можна записати два тотожних (дуальних) вирази логічної функції. Беручи лише до уваги конституенти 1, маємо

$$Y_{\text{УДНФ}} = \overline{X}_1 \overline{X}_2 \overline{X}_3 \nu \overline{X}_1 X_2 X_3 \nu X_1 \overline{X}_2 \overline{X}_3 \nu X_1 X_2 X_3$$

а для конституент 0

$$Y_{\text{УКНФ}} = (X_1 \vee X_2 \vee \bar{X}_3)(X_1 \vee \bar{X}_2 \vee X_3)(\bar{X}_1 \vee X_2 \vee \bar{X}_3)(\bar{X}_1 \vee \bar{X}_2 \vee X_3)$$

Отже, якщо логічну функцію задано таблично, то для переходу до її аналітичного зображення в УНФ роблять так:

для зображення в УДНФ описують ті набори змінних, для яких функція дорівнює одиниці, тобто для конституент I; для кожного такого набору складають мінтерм, причому змінні $X_i = 0$ замінюють на \bar{X}_i і одержані мінтерми з'єднують диз'юнкцією;

для зображення в УКНФ виписують набори, для яких функція дорівнює нулю, тобто для констатуент 0 для кожного набору складають макстерм, причому змінні $X_i = 1$ замінюють на \bar{X}_i і одержані макстерми об'єднують кон'юнкцією.

Приклад: Утворити УДНФ логічної функції, що задана в ДНФ:

$$Y = X_1 \vee \bar{X}_1 X_2 \vee X_1 \bar{X}_2 X_3$$

Розв'язання. Для підвищення рангу кон'юнкції молодшого рангу застосовуємо доповнення типу $(X_i \vee \bar{X}_i)$:

$$\begin{aligned} Y &= X_1(X_2 \vee \bar{X}_2)(X_3 \vee \bar{X}_3) \vee \bar{X}_1 X_2(X_3 \vee \bar{X}_3) \vee X_1 \bar{X}_2 X_3 = \\ &= X_1 X_2 X_3 \vee X_1 \bar{X}_2 X_3 \vee X_1 X_2 \bar{X}_3 \vee X_1 \bar{X}_2 \bar{X}_3 \vee \bar{X}_1 X_2 X_3 \vee \bar{X}_1 X_2 \bar{X}_3 \vee X_1 \bar{X}_2 X_3 \end{aligned}$$

Приклад: Таблично-задану логічну функцію $Y = f(X_1, X_2, X_3)$ записати в аналітичній формі (див.табл. 1.8) для тотожних зображень УДНФ та УКНФ.

Розв'язання. В УДНФ:

$$\begin{aligned} Y &= f(X_1, X_2, X_3) = f_1(0,0,1) \vee f_2(0,1,0) \vee f_3(0,1,1) \vee f_7(1,1,1) = \\ &= \bar{X}_1 \bar{X}_2 X_3 \vee \bar{X}_1 X_2 \bar{X}_3 \vee \bar{X}_1 X_2 X_3 \vee X_1 X_2 X_3; \end{aligned}$$

В УКНФ:

$$\begin{aligned} Y &= f(X_1, X_2, X_3) = f_0(0,0,0) \vee f_4(1,0,0) \vee f_5(1,0,1) \vee f_6(1,1,0) = \\ &= X_1 X_2 X_3 \vee \bar{X}_1 X_2 X_3 \vee \bar{X}_1 X_2 \bar{X}_3 \vee \bar{X}_1 \bar{X}_2 X_3; \end{aligned}$$

Таблиця 1.8 Таблично задана логічна функція.

НОМЕР НАБОРУ	X_1	X_2	X_3	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

Слід зауважити, що в даному прикладі вибір форми зображення логічної функції не має принципового значення, бо число мінтерм дорівнює числу макстерм. Проте у випадках, коли число одних переважав над числом інших,

наприклад, кількість $f_i = 1$ більша за кількість $f_i = 0$, або навпаки, тоді менш громіздким буде функція з меншим числом терм.

Приклад: Перейти до УКНФ логічної функції, що задана в УДНФ

$$Y = X_1 \bar{X}_2 \vee \bar{X}_1 X_2 \vee \bar{X}_1 \bar{X}_2$$

Розв'язання. Спочатку спростимо вираз за законом склеювання:

$$Y = X_1 \bar{X}_2 \vee X_1 X_2 \vee \bar{X}_1 X_2 = X_1 \bar{X}_2 \vee \bar{X}_1$$

Застосувавши правило Шеннона, маємо

$$\bar{Y} = (\bar{X}_1 \vee X_2) X_1 = \bar{X}_1 X_1 \vee X_2 X_1 = X_1 X_2$$

За принципом подвійної інверсії та законом дуальності остаточно одержимо

$$\bar{\bar{Y}} = \bar{X_1 X_2} = \bar{X}_1 \vee \bar{X}_2$$

1.7 Мінімізація логічних функцій

Зведення логічної функції до УДНФ або до УКНФ дає однозначне зображення Y . Однак для технічної реалізації такої логічної функції властивість однозначності зображення буде зручним тільки в тому випадку, якщо повним набором логічних елементів є елементарний базис, що складається з окремих елементів І, АБО, НЕ, причому з числом входів елементів І та АБО, що дорівнює рангу термів логічної функції. На практиці часто доводиться будувати цифрові пристрої на різних базисах, і тоді з'ясовується, що удосконалені форми зображення логічних функцій не завжди найекономніші. Їм властива надлишковість, яка підлягає спрощенню, тобто мінімізації. Цій процедурі, до речі, можуть підлягати логічні функції інших, не обов'язково удосконалених, форм зображення.

Мінімізація - це процес зведення логічної функції до такого виду, який припускає більш просту і дешеву її фізичну реалізацію, тобто з меншим числом логічних елементів за рахунок зменшення числа логічних символів, кількості змінних та зв'язків між елементами.

Відомо кілька методів мінімізації, серед яких найбільш поширеними в інженерній практиці є такі:

- безпосередніх перетворень;
- карт Карно;
- Квайна.

Знайти гарантовано мінімальний вираз для довільної логічної функції можна лише методом повного перебору в процесі мінімізації всіх можливих варіантів. Від руки (на папері) це реально здійснити для невеликої кількості змінних. Із збільшенням числа змінних складність розглянутих методів мінімізації зростає за геометричною прогресією і стає під силу лише автоматизованим методам, які призначені для ЕОМ.

Для оцінки складності логічної функції, яка зображена тою чи іншою формою, вводиться поняття ціни реалізації (покриття) логічної функції. За

Квайнном ціна покриття логічної функції визначається числом змінних (букв) в конститuentі, що рівнозначно числу входів логічних елементів, які реалізують задану функцію. Кон'юнкції вищого рангу покриваються відповідними кон'юнкціями нижчого рангу. Наприклад, кон'юнкції $\overline{X_1}X_2\overline{X_3}$ і $X_1X_2\overline{X_3}$ покриваються кон'юнкцією $X_2\overline{X_3}$.

В загальному випадку операцію покриття зображає операція склеювання (див. табл.1.3) $AX_i \vee A\overline{X_i} = A$, де A - елементарна кон'юнкція або імпліканта, тобто така логічна функція, яка отримана в результаті склеювання і яка на будь-якому наборі змінних $\{\overline{X_1}, \overline{X_2}, \dots, \overline{X_n}\}$ набуває такого самого значення, що й сама функція (конститuent), що її утворила. Чим, більше число покриттів, що знижують ранг імпліканти, тим простіший кінцевий вираз. Прості імпліканти – це елементарні кон'юнкції (що мають менше число членів, ніж змінних) найнижчого рангу, які входять у дану логічну функцію. Якщо в днз'юнкції простих імплікант жодну з них шляхом поглинання вилучити (покрити) не вдається, таку диз'юнкцію називають тупиковою, або мінімізованою диз'юнктивно-нормальною формою (МДНФ) заданої функції.

Таким чином, загальною задачею мінімізації логічних функцій є знаходження скороченої ДНФ, тобто знаходження всіх простих імплікант, визначення серед них тупикових ДНФ і, нарешті, вибір з останніх мінімальних, які утворюють МДНФ заданої логічної функції.

Метод безпосередніх перетворень - це аналітичний метод спрощення логічних функцій з допомогою аксіом та законів бульової алгебри (див. табл.1.4). При набутті певних навичок цей метод є досить ефективним для малої кількості змінних (як правило, не більше трьох).

Для мінімізації логічних функцій методом безпосередніх перетворень корисно використовувати такі формули бульової алгебри:

$$X_1 \vee X_1 X_2 = X_1;$$

$$X_1 \vee \overline{X_1} X_2 = X_1 \vee X_2;$$

$$X_1 X_2 \vee X_1 \overline{X_2} = X_1;$$

$$X_1 (X_1 \vee X_2) = X_1;$$

$$X_1 (\overline{X_1} \vee X_2) = X_1 X_2;$$

$$(X_1 \vee X_2)(X_1 \vee \overline{X_2}) = X_1$$

У деяких випадках ефективніше мінімізувати інверсію логічної функції для отримання простішого виразу в МДНФ. Це може бути тоді, коли логічна функція має переважаюче число одиниць, ніж нулів. Тоді, маючи МДНФ інверсної функції, при технічній реалізації цієї функції достатньо на виході побудованої схеми під'єднати інвертор для відновлення прямої функції.

Приклад: Отримати МДНФ для логічної функції, що задана таблицею істинності (див. табл. 1.9).

Таблиця 1.9 Таблично задана логічна функція

X_1	X_2	X_3	Y
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Розв'язання. Оскільки у даній функції число одиниць переважає над числом нулів, виконаємо мінімізацію для її інверсії:

$$Y = \overline{X_1}X_2\overline{X_3} \vee X_1X_2\overline{X_3} = X_2\overline{X_3}(\overline{X_1} \vee X_1) = X_2\overline{X_3}$$

Отже, $Y = X_2\overline{X_3}$. Порівняємо з МДНФ прямої функції:

$$\begin{aligned} Y &= \overline{X_1}\overline{X_2}\overline{X_3} \vee \overline{X_1}\overline{X_2}X_3 \vee \overline{X_1}X_2\overline{X_3} \vee \overline{X_1}X_2X_3 \vee X_1\overline{X_2}\overline{X_3} \vee X_1\overline{X_2}X_3 \vee X_1X_2\overline{X_3} \vee X_1X_2X_3 = \\ &= \overline{X_2}\overline{X_3} \vee \overline{X_2}X_3 \vee X_2X_3 = \overline{X_2} \vee X_3 \end{aligned}$$

Приклад : Методом безпосередніх перетворень мінімізувати логічну функцію

$$Y = \overline{X_1}\overline{X_2}X_3 \vee \overline{X_1}X_2\overline{X_3} \vee \overline{X_1}X_2X_3 \vee \overline{X_1}\overline{X_2}\overline{X_3} \vee X_1X_2\overline{X_3}$$

Розв'язання:

$$\begin{aligned} Y &= \overline{X_1}\overline{X_2}X_3 \vee \overline{X_1}X_2\overline{X_3} \vee \overline{X_1}X_2X_3 \vee \overline{X_1}\overline{X_2}\overline{X_3} \vee X_1X_2\overline{X_3} = \\ &= X_1(\overline{X_2}X_3 \vee X_2\overline{X_3} \vee X_2X_3 \vee \overline{X_2}\overline{X_3}) \vee X_1X_2\overline{X_3} = \\ &= \overline{X_1}(\overline{X_2}(X_3 \vee \overline{X_3}) \vee X_2(\overline{X_3} \vee X_3)) \vee X_2\overline{X_3}(X_1 \vee \overline{X_1}) = \\ &= \overline{X_1}(\overline{X_2} \vee X_2) \vee X_2\overline{X_3} = \overline{X_1} \vee X_2\overline{X_3} \end{aligned}$$

1.8 Структурна реалізація логічних функцій

Наступним кроком після мінімізації логічної функції є побудова її структурної схеми. Цей етап проектування належить до структурного синтезу цифрового пристрою. Початковими даними для виконання структурного синтезу логічної схеми автомата є, як правило, МДНФ або МКНФ логічної функції, а для багатозначної функції - система логічних Функцій, що зображені у МДНФ або у МКНФ.

Задачу структурної реалізації логічної функції (або функцій) сформулюємо так: для заданих вхідних змінних X_1, X_2, \dots, X_n , на наборах яких

визначена (або частково визначена) логічна функція, що зображена у МДМФ або у МКНФ, побудувати структурну логічну схему, яка б реалізувала цю функцію у заданому базисі.

За основний критерій при цьому беремо мінімум апаратурних затрат, під яким слід розуміти мінімальну кількість ЛЕ та мінімальне число зв'язків між ними.

Визначальну роль у забезпеченні критерію за мінімумом апаратурних затрат відіграє елементний базис, тобто певний набір функціонально повних ЛЕ, на яких можна реалізувати довільну логічну функцію. Якщо базис наперед не заданий, то при такому абстрактному синтезі жодних перетворень із заданою логічною функцією робити не потрібно, досить лише структурно реалізувати за допомогою ЛЕ всі її логічні операції.

Однак, якщо базис за умовою завдання наперед заданий, дану логічну функцію необхідно спеціально перетворити. Головна мета цих перетворень - зведення виразу функції до заданого базису. Раціональність і ефективність такого підходу забезпечує швидку побудову структурної логічної схеми синтезованого цифрового пристрою. Наприклад, якщо задамо базис 2І-НЕ (елемент Шефера) або 2АБО-НЕ (елемент Пірса), для реалізації функції змінних відповідними перетвореннями і замінами МДНФ або МКНФ цієї функції мають бути зображені у вигляді $\overline{X_k X_l}$ або $\overline{X_k \vee X_l}$. Очевидно, що при заданому базисі 2І-АБО-НЕ логічна функція має набувати такого остаточного вигляду: $\overline{X_i X_j \vee X_k X_l}$.

При виконанні структурного синтезу цифрового пристрою доцільно користуватися розкладанням логічної функції за принципом, що нагадує дерево, коріння якого вихід (або виходи), а гілки - входи змінних.

Нехай, наприклад, у базисі елемента Шефера потрібно реалізувати логічну функцію

$$Y = X_1 \vee X_2 \vee X_3$$

Оскільки елемент Шефера - це 2І-НЕ, в даній функції необхідно зробити такі перетворення, в яких у явному вигляді проявляться ознаки функції $\overline{X_k X_l}$. Для одержання виразу даної функції у заданому базисі потрібно застосувати подвійну інверсію та закон дуальності (правило де Моргана):

$$Y = \overline{\overline{X_1 \vee X_2 \vee X_3}} = \overline{\overline{X_1} \overline{X_2} \overline{X_3}} = \overline{\overline{abc}} = \overline{dc} = \overline{ec}$$

Отже, для реалізації даної функції у базисі елемента Шефера потрібно 4 ЛЕ НЕ і 2 ЛЕ 2І-НЕ. Інвертор легко реалізується на 2І-НЕ, якщо його входи з'єднати (при $X_1 = X_2 = X$, $Y = \overline{X_1 X_2} = \overline{X}$) або якщо до одного з входів прикласти високий рівень (наприклад, при $X = 1 - Y = \overline{X_1 X_2} = \overline{1 \cdot X_2} = \overline{X}$). Схему, що реалізує диз'юнкцію трьох змінних на елементах Шефера, показано на рис. 1.1.

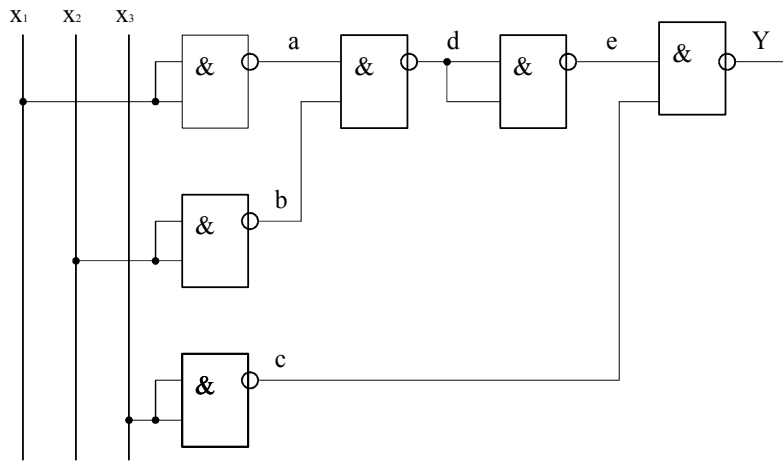


Рис. 1.1 Диз'юнкція трьох змінних в базисі 2І-НЕ

Якщо елементи Шефера замінити елементами Пірса, то згідно з принципом дуальності така логічна схема буде реалізувати кон'юнкцію трьох змінних.

1.9 Загальні відомості про цифрові автомати

Вичерпне визначення поняття "цифровий автомат" дав автор цього терміну В.М.Глушков^[1]: "Електронні цифрові машини з програмним керуванням являють собою приклад одного з найпоширеніших сьогодні типів перетворювачів дискретної інформації, названих дискретними або цифровими автоматами. Тому задача синтезу схем електронних цифрових машин з програмним керуванням входить як частковий випадок в більш загальну задачу синтезу схем цифрових автоматів". Отже, будь-який елемент, вузол, пристрій чи навіть ЕОМ, незалежно від складності їх функціонування, є перетворювачами цифрової інформації – цифровими автоматами.

У загальному випадку на вхід цифрового автомата надходить множина двійкових змінних X_0, X_1, \dots, X_{N-1} , а з виходу знімається множина двійкових функцій Y_0, Y_1, \dots, Y_{N-1} . Відмінна особливість цифрових автоматів полягає в тому, що цей функціональний зв'язок визначається також дискретною множиною внутрішніх станів, причому перехід з одного стану в інший здійснюється стрибкоподібно. Реальні цифрові автомати можуть мати лише скінченну множину внутрішніх станів, а тому - скінченне число станів, входів та виходів. Через це цифрові автомати називають ще скінченними.

Вихідні сигнали цифрового автомата залежать як від вхідних сигналів, що діють у даний (фіксований) момент часу, так і від передісторії, тобто від тих сигналів, які надійшли на його входи раніше і зафіксувались в елементах пам'яті – запам'ятовувачах. Отже, роботу автомата слід розглядати щодо конкретного інтервалу часу T -такту. Такт - це скінченний відрізок часу, який необхідний для передачі одного з розрядів двійкового числа /біта/ - у разі послідовного коду, або всього двійкового коду (слова) одночасно - при паралельному коді. Залежно від того, чим визначається такт T , розрізняють

асинхронні та синхронні автомати. В асинхронних цифрових автоматах $T \neq \text{const}$ і зміна вхідних сигналів зразу викликає певну зміну вихідних сигналів, у синхронних $T = \text{const}$ і тому зміна вхідних сигналів викликає певну зміну вихідних тільки після подачі синхронізуючих (тактових) імпульсів, які керують роботою автомата.

Для опису законів функціонування цифрових автоматів зручно користуватись абстрактним часом, що набуває цілих невід'ємних значень ($t=0,1,2,\dots$), а не тактом T . Наприклад, позначимо такти роботи автомата як t і $t + 1$. Алгебраїчний вираз, який розкриває функціональний зв'язок цифрового автомата між вихідним сигналом у такті $t + 1$ і множиною вхідних сигналів та станів у попередньому такті t називається функцією переходу δ .

Найпростішою математичною моделлю цифрового автомата з одним входом X і одним виходом є абстрактний автомат, що заданий сукупністю таких величин:

скінченою множиною вхідних сигналів (вхідний алфавіт) автомата

$$X = \{x_0(t), x_1(t), \dots, x_N(t)\};$$

скінченою множиною вихідних сигналів /вихідний алфавіт/ автомата

$$Y = \{y_0(t), y_1(t), \dots, y_N(t)\};$$

довільною множиною станів /алфавіт станів/ автомата

$$Q = \{q_0(t), q_1(t), \dots, q_N(t)\}$$

а також початковим станом автомата $q_0(t) \in Q$, функцією переходу автомата з одного стану в інший $\delta(q, x)$ та функцією виходу автомата $\lambda(q, x)$.

1.10 Різновиди цифрових автоматів та особливості їх функціонування

У теорії автоматів найповніше описані синхронні (цифрові) автомати. Закон функціонування будь-якого абстрактного синхронного автомата визначається його вихідним сигналом $y(t+1)$, який залежить від вхідних сигналів $x(t+1)$ та від внутрішніх станів (чи стану) $q(t+1)$ і/або $q(t)$ автомата. Тому існують дві можливості визначення реакції вихідного сигналу на ці дії:

1) коли $y(t+1)$ однозначно залежить від вхідного сигналу $x(t+1)$ і попереднього стану $q(t)$, такий автомат описується системою функцій переходу і виходу

$$\begin{cases} q(t+1) = \delta\{q(t), x(t+1)\}; \\ y(t+1) = \lambda\{q(t), x(t+1)\}, t = 0, 1, 2, \dots \end{cases} \quad (1.4)$$

і називається автоматом 1-го роду;

2) коли $y(t+1)$ однозначно залежить від вхідного сигналу $x(t+1)$ і стану $q(t+1)$ у даний момент часу -

$$\begin{cases} q(t+1) = \delta\{q(t), x(t+1)\}; \\ y(t+1) = \lambda\{q(t+1), x(t+1)\}, t = 0, 1, 2, \dots \end{cases} \quad (1.5)$$

це автомат 2-го роду.

Цифрові автомати 1-го роду, що функціонують за законом (1.4), називаються автоматами Мілі (Mealy). а частковий випадок автоматів 2-го роду, для яких вихідні сигнали залежать тільки від стану $q(t+1)$ автомата і не залежать від значень вхідних сигналів, називаються автоматами Мура (Moore). Для них закон функціонування буде заданий частковим випадком системи рівнянь автоматів 2-го роду, а саме

$$\begin{cases} q(t+1) = \delta\{q(t), x(t+1)\}; \\ y(t+1) = \lambda\{q(t+1)\}, t = 0, 1, 2, \dots \end{cases}$$

Отже, на відміну від автомата Мілі вихідний сигнал у автоматі Мура залежить тільки від біжучого стану автомата і в явному виді не залежить від вхідного сигналу. Автомати Мура і Мілі взаємозамінні – існують прості способи еквівалентного переходу від одного до іншого.

Довільний абстрактний цифровий автомат Мілі або Мура називають ще автоматом з пам'яттю, тобто таким, що здатний запам'ятовувати попередню інформацію, якщо він мав число внутрішніх станів більше за один. Якщо цифровий автомат має лише один внутрішній стан, він називається автоматом без пам'яті. Стан такого автомата в процесі функціонування не змінюється, оскільки він тільки один. Тому вихідний сигнал автомата без пам'яті залежить лише від вхідного в даному такті сигналу й не залежить від попередніх станів. Закон функціонування таких тривіальних цифрових автоматів без пам'яті описується одним рівнянням $y(t) = Y\{x(t)\}$

Оскільки логічний стан виходів цифрового автомата без пам'яті залежить тільки від комбінації логічних сигналів на входах в даний момент часу, його називають комбінаційним пристроєм (чи схемою); КП - це асинхронний цифровий автомат.

У загальному вигляді КП (див. рис1.2) має N входів, на які подаються сигнали X_0, X_1, \dots, X_{N-1} , і M виходів, з яких знімаються сигнали Y_0, Y_1, \dots, Y_{M-1} . Отже, КП описується системою логічних функцій

$$y_l = Y\{X_0, X_1, \dots, X_{N-1}\}, l = 0, 1, M - 1$$

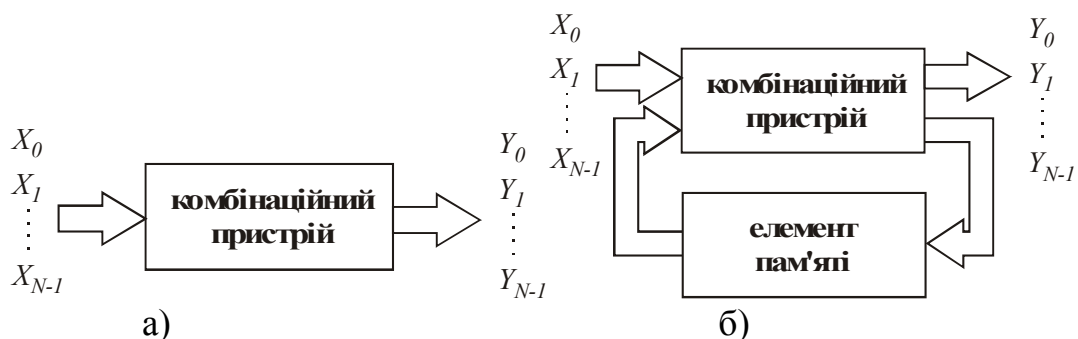


Рис. 1.2- Структурні схеми цифрових автоматів

Реалізацію (синтез) КП здійснюють переважно на ЛЕ. Найпростіший випадок реалізації КП - коли схема має один вихід (L=1) .

На відміну від КП значення вихідних сигналів у цифрових автоматах з

пам'яттю залежать не тільки від значень вхідних сигналів в даний момент часу, але й від їх попередніх значень. Звідси очевидно, що такі пристрої реалізують функціональний зв'язок вже не між окремими значеннями вхідного та вихідного сигналів, а між їх послідовностями. Тому автомати з пам'яттю називають послідовнісними. На відміну від КП роботу послідовнісних пристроїв (схем) слід розглядати у часі. Отже послідовнісний пристрій (ПП) (рис.1.2,б) повинен мати пам'ять, щоб значення вихідного сигналу залежало від попереднього вхідного сигналу. Ця попередня інформація використовується для побудови ПП у вигляді сукупності так званих внутрішніх сигналів q_i , що виробляють елементи пам'яті (ЕП).– запам'ятовувачі.

Основною задачею теорії цифрових автоматів є задача аналізу і синтезу вже розглянутих двох класів цифрових пристроїв. Для цього використовується основний математичний апарат – алгебра логіки. У зв'язку з тим що в основі побудови ПП є КП (див. рис. 1.2,б), задачу синтезу цифрових пристроїв переважно зводять до задачі синтезу КП. Це так званий канонічний метод структурного синтезу ПП.

У табл. 1.7 подані типи КП і ПП, що стоять на різних ступенях інтеграції.

Таблиця 1.7 Класифікація інтегральних схем

СТУПІНЬ ІНТЕГРАЦІЇ	КОМБІНАЦІЙНІ ПРИСТРОЇ (СХЕМИ)	ПОСЛІДОВНІСНІ ПРИСТРОЇ (СХЕМИ)
МІС	Логічні елементи з різними логічними та функціональними можливостями	Тригери
СІС	Перетворювачі кодів, шифратори (дешифратори, мультиплексори), демультимплексори, суматори, цифрові компаратори, драйвери	Регістри, лічильники, генератори числових послідовностей
ВІС	Арифметико-логічні пристрої, програмовані логічні матриці, постійні запам'ятовувальні пристрої	Багаторозрядні регістри, запам'ятовувальні пристрої великих об'ємів пам'яті тощо

1.11 Загальні питання синтезу цифрових автоматів

Синтез цифрового пристрою взагалі полягає у побудові такого автомата, який реалізує наперед задану довільним чином функцію "вхід-вихід". Якщо синтез нескінченних автоматів викликає переважно академічний інтерес (важливо лише реалізувати потрібне відображення "вхід-вихід"), синтез скінченних автоматів виходить з практичних задач, і тому завдає, як правило, багато труднощів через додаткові вимоги.

Труднощі залежать здебільшого від способу задавання умов функціонування автомату.

У багатьох випадках з'ясовується, що єдиного методу синтезу скінченних автоматів не існує. Синтез скінченного автомата довільної складності можна

поділити на кілька етапів:

- по-блоковий синтез – передбачає поділ всього автомата на окремі блоки, для кожного з яких визначене завдання, яке вони повинні розв'язати, а також накреслені функціональні зв'язки (обмін) між ними;
- абстрактний синтез – полягає у побудові абстрактного автомата (наприклад, його таблиці чи функції переходів і виходів) за обраним способом задавання функції "вхід-вихід"; для ПП тут визначається об'єм необхідної пам'яті для кожного блока;
- структурний синтез – передбачає побудову структурної схеми автомата; тут встановлюється структура всього автомата з урахуванням структури його вхідних і вихідних сигналів, а також здійснюється вибір елементів для побудови схеми, якщо вони наперед не задані;
- надійнісний синтез – забезпечує надійність (безвідмовність, довговічність, ремонтпридатність) функціонування автомата за допомогою певного перетворення побудованої схеми;
- технічний синтез – кінцевий етап, на якому виявляються спотворення сигналів, що можуть виникати через неідеальність застосовуваних елементів, і вживаються відповідні заходи щодо усунення цих спотворень.

Поділ на етапи дозволяє побачити лише загальну картину процесу синтезу автоматів і, безперечно, у ряді випадках можливі відхилення від розглянутої послідовності. Зокрема, при синтезі простих автоматів етап по-блокового синтезу звичайно опускають і, навпаки, при синтезі особливо складних автоматів до цього етапу доводиться багаторазово повертатись. У деяких випадках доцільно сусідні етапи об'єднати, не розмежовуючи їх, для більшої ефективності процесу синтезу. В інших ситуаціях, враховуючи міркування щодо надійності, етап надійнісного синтезу виконують вже з самого початку.

Перші три етапи синтезу (по-блоковий, абстрактний і структурний) під час проектування цифрових ВІС об'єднують в один комплексний етап так званого функціонально-логічного проектування. При такому процесі проектування велику роль відіграє програмне забезпечення (банк даних) у структурі наскрізної САПР як основного засобу інтеграції різних етапів синтезу ВІС. Після функціонально-логічного проектування при синтезі цифрових ВІС настає етап топологічного проектування, який є складовою частиною технічного синтезу.

При синтезі цифрових автоматів (пристроїв) можуть застосовуватись як евристичні, так і формальні методи. При синтезі КП найбільш поширеними є евристичні методи, які ґрунтуються на винахідницькій діяльності розробника. Це творчі методи, які залежать від професійного рівня спеціаліста. Однак вони не забезпечують позитивної ролі формальних методів синтезу, які втілені в САПР і дозволяють стандартизувати процес синтезу. Разом з тим, не слід перебільшувати їх можливості, тому що, як свідчить практика, якість отриманих формальних розв'язків часто

поступається перед якістю аналогічних евристичних розв'язків.

Синтез цифрових автоматів (пристроїв) залежно від функціонального базису можна виконувати на різних рівнях (див. табл. 1.7). Зокрема, на найнижчому рівні (МІС) - на ЛЕ або на тригерах, на рівні СІС - на різних функціональних вузлах, на рівні ВІС - на функціональних блоках і пристроях. Ці рівні синтезу цифрових пристроїв в дзеркальним відображенням сучасного рівня технології виготовлення мікросхем. Найбільш складним і трудомістким в процес виготовлення цифрових ВІС - він, фактично, узагальнює всі попередні рівні проектування. Тому проектування цифрових ВІС можна розбити на окремі етапи: логічний, функціональний і архітектурний. На кожному з них розв'язуються певні задачі, які розширюють та поглиблюють знання про ВІС, що підлягає проектуванню.

Основним критерієм ефективності синтезу цифрового автомата є забезпечення мінімуму вартості реалізації логічної функції. При цьому слід мати на увазі, що вартість цифрового пристрою залежить як від часу, так і від рівня технології, на якій він виготовлений.

Поняття вартості цифрового пристрою, синтезованого, наприклад, на рівні МІС, є відмінним від поняття вартості аналогічного пристрою на рівні ВІС чи НВІС.

До питання вартості цифрового пристрою можна підійти з позиції надійності або кількості (числа) елементів, з яких складається даний пристрій, та зв'язків між ними. Безперечно, існує певний взаємозв'язок між обома критеріями, однак в контексті технологічного прогресу спостерігається деяка відмінність. Поняття вартості цифрового пристрою, що побудований на дискретних елементах та ЛЕ, пов'язане з питанням надійності роботи пристрою, однак головну роль відіграє кількість елементів і зв'язків між ними. Власне кількісний фактор тут виступає визначальним при оцінюванні вартості пристрою.

Вартість виготовлення ВІС, зокрема, визначається в основному площею схеми на кристалі і безпосередньо не пов'язана з числом дискретних елементів. Мінімальне число елементів ВІС ще не передбачає мінімальної площі кристалу, а отже, вартості ВІС. Дуже часто схема ВІС з великим числом елементів при високій регулярності структури займає невелику площу і тому є дешевою, бо, крім того, її ще вигідно проектувати. При цьому надійність з'єднань у самому кристалі ВІС є дуже високою порівняно з надійністю з'єднань (тобто точок зовнішніх з'єднань) між самими мікросхемами. Тому поняття вартості ВІС пов'язане в основному з площею кристалу і числом точок зовнішніх виводів (з'єднань). Оскільки розрахунок мінімальної площі схеми ВІС викликає значні труднощі; на практиці традиційно прийнято за основний критерій синтезу цифрового пристрою на ВІС вважати мінімум кількості ЛЕ та їх виводів чи зв'язків між ними. Якщо цифровий пристрій реалізується на ВІС, при використанні схем з високою регулярністю структури типу ПЗП або ПЛМ зменшення числа виразів у логічній функції, як правило означає зменшення площі схеми, а отже,

зниження вартості всього пристрою.

Контрольні запитання

1. В чому відмінність двох класів цифрових пристроїв?
2. Що таке асинхронний та синхронний автомати?
3. Що відображає функція переходу скінченного автомата?
4. Чим визначається абстрактний автомат?
5. Чим відрізняється автомат 1-го роду від автомата 2-го роду?
6. Які цифрові автомати належать до автомата Мілі і автомата Мура?
7. Який автомат називається, комбінаційним?
8. Який автомат називається послідовнісним?
9. В чому полягає синтез КП і ПП?
10. З яких етапів складається синтез скінченного автомата?
11. Особливість синтезу цифрових ВІС.
12. Від чого залежать рівні синтезу скінчених автоматів?
13. Який критерій визначає ефективність синтезу цифрового автомата?
14. Що розуміють під поняттям вартості цифрового пристрою, побудованого на різних рівнях ступенів інтеграції?

2. СХЕМОТЕХНІКА ЦИФРОВИХ ЕЛЕМЕНТІВ

При структурній реалізації логічної функції, що виконується етапі структурного синтезу цифрового автомата, ЛЕ розглядався як абстрактний елемент логічної схеми. Наступним кроком до технічної реалізації цифрового автомата є побудова його принципової схеми, де основними стають питання схемотехніки не лише ЛЕ, а й цифрових елементів взагалі. На цьому етапі структурного синтезу необхідні знання як технічних можливостей вибраних мікросхем цифрового елемента, так і особливостей їх роботи разом з іншими елементами. Питання схемотехніки цифрових елементів тісно пов'язане з фізичним носієм двійкової інформації - цифровим сигналом. Тому насамперед розглянемо різні форми зображення та способи передачі цифрового сигналу по каналах зв'язку між окремими цифровими елементами.

2.1 Види цифрових сигналів, та способи їх передачі

У підрозд. 1.1 були розглянуті сигнали як фізичні носії інформації, а також, природа цифрового сигналу як різновиду дискретних сигналів. Отже, цифровий сигнал - це фізичний носій інформації, що являє собою дискретну послідовність електричного сигналу (найчастіше напруги) низького і високого рівнів, з допомогою яких зображаються двійкові числа 0 і 1. Логічні рівні, яким відповідають логічні змінні лог.0 і лог.1, позначатимемо відповідно U^0 і U^1 , зустрічаються ще позначення з англ. аббревіатурою U^L , де L - від Low (низький), і U^H , де H - від High (високий).

Погоджено, що при так званій позитивній логіці високим рівнем напруги U^1 (відносно нульового рівня напруги) кодується лог.1, а низьким рівнем U^0 - лог.0, причому $U^1 > U^0$, а при негативній логіці навпаки, $U^1 < U^0$. У зв'язку з цим ЛЕ можна розглядати з позиції як позитивної логіки (яка, до речі, є найбільш поширеною), так і негативної. Зрозуміло, що функціонування ЛЕ в обох випадках буде різним. Цю особливість іноді зручно використати на практиці, адже перехід з одної логіки до іншої є еквівалентним, що впливає з принципу дуальності і підтверджується правилом де Моргана. В цьому легко переконатись, якщо розглядати, наприклад, дію ЛЕ АБО у позитивній логіці, а дію ЛЕ І - у негативній; їх функціонування буде еквівалентним, бо $y = X_1 \vee X_2 = \overline{\overline{X_1} \overline{X_2}}$

Єдиним детермінованим цифровим сигналом, що задає темп роботи всього цифрового пристрою чи системи, тобто такти, є сигнал синхронізації (синхросигнал). Як і належить періодичному сигналу, синхро-сигнал характеризується постійним періодом повторення T та коефіцієнтом заповнення $\gamma = t_i / T$, де t_i - тривалість імпульсу; найчастіше $\gamma = 0.5$ (величина, обернена до коефіцієнта заповнення γ , тобто $1/\gamma = q$, називається щільністю). Синхроімпульси (або тактові імпульси) забезпечують синхронізацію роботи цифрового пристрою чи системи, тобто точне часове узгодження роботи їх складових частин - цифрових елементів, для виконання

заданих операцій. Виробляє тактові синхроімпульси генератор тактових імпульсів (ГТІ), який працює незалежно від інших вузлів пристроїв чи системи. Від частоти і стабільності роботи ГТІ залежать швидкодія й стабільність роботи всього цифрового пристрою чи системи.

Іноді термін "синхронізація" замінюється на "стробування", а синхроімпульс називають строб-імпульсом, або сигналом опитування.

Дія синхросигналів забезпечує сталу тривалість циклів виконання тих чи інших операцій незалежно від їх змісту. Такий спосіб керування роботою елементів і вузлів цифрового пристрою називається синхронним. За синхронним принципом працюють синхронні автомати. Дія синхросигналів регламентує роботу цифрового елемента, знижуючи при цьому ймовірність появи збою. При асинхронному способі керування операціями синхроімпульси не потрібні, бо перехід до наступної операції в асинхронних автоматах здійснюється зразу ж після появи інформаційних сигналів. Асинхронний спосіб припускає більшу швидкодію виконання операцій, однак ймовірність збою пристрою при цьому більша.

За способом кодування двійкової інформації цифрові сигнали можуть бути потенційними, імпульсними та імпульсно-потенційними.

Потенційний цифровий сигнал - це високий (лог.1) або низький (лог.0) рівень напруги (або струму), який може залишитися незмінним протягом не менше одного періоду синхроімпульсів (такту). Тому, коли потрібно визначити кількість лог.1 або лог.0, що слідує один за одним, досить перерахувати число синхроімпульсів.

При імпульсному відображенні двійкової інформації значенню лог.1 або лог.0 відповідає наявність або відсутність імпульсу при синфазній дії синхроімпульсу. Отже, ідентифікувати такий цифровий сигнал нескладно.

Імпульсно-потенційний спосіб зображення цифрових сигналів передбачає сумісне використання розглянутих способів. Такий спосіб відображення двійкової інформації називається динамічним. Потенційна складова сигналу використовується переважно для керування (дозволу або заборони роботи цифрового елемента), а імпульсна - для виконання конкретної операції (логічної, арифметичної, опитування, формування тощо).

Цифрові сигнали передають інформацію від джерела (передавача) до споживача (приймача) по каналах зв'язку. Провідники, по яких передаються дані за допомогою цифрових сигналів, називаються лініями, а сукупність ліній - шиною. Каналом зв'язку між передавачем і приймачем може бути як лінія (однопровідна), так і шина. По лінії цифрові сигнали передаються послідовно біт за бітом, починаючи з молодшого біта (МБ), тобто молодшого розряду двійкового числа. Тому першим на вхід приймача надходить МБ двійкового коду (слова), а потім інші біти. Функцію запису кожного біта кодового слова в приймальний пристрій виконують синхроімпульси, такт за тактом. При цьому розряд числа визначається номером такта, починаючи з нульового, що збігається з МБ слова. Отже, для послідовної передачі інформації, що зображена n - бітним словом, потрібно n тактів синхронізації.

Послідовний спосіб раніше застосовувався переважно для передачі інформації на велику відстань, однак на даний час в зв'язку з переходом частот передачі інформації в радіодіапазон, він застосовується і для передачі сигналів між ІС за допомогою спеціальних послідовних протоколів (SPI, I²C, SPDIF, I²S, тощо). Це зумовлено потребою забезпечення відповідних хвилевих характеристик ліній передачі даних, і природно, це легше зробити для двох-трьох провідників ніж для 8, 16 і т.д.

Для передачі багаторозрядних слів використовують послідовно-паралельний код, при якому слова розбивають на "склади", тобто пакети даних. Наприклад, 32-розрядне слово для асинхронної передачі по однобайтній 8-ми розрядній шині розкладають на чотири "склади" (пакети даних).

За таким способом передачі цифрових сигналів окремі "склади" слова, таким чином, передаються, а іноді й обробляються послідовно, а зображаються паралельно.

Стан лінії або шини може бути *активним* або *пасивним*. У першому випадку на лінії або на лініях шини цифрові сигнали (лог.1 або лог.0) присутні. Пасивний стан характеризується відсутністю обох станів, натомість лінія набуває третього, так званого *високоімпедансного стану Z*. У стані Z на окремій лінії або на шині зв'язку "передавач-приймач" має місце умовний "обрив" - нескінченно великий опір (високий імпеданс), який забезпечує спеціально призначений для цього цифровий елемент - тристановий драйвер.

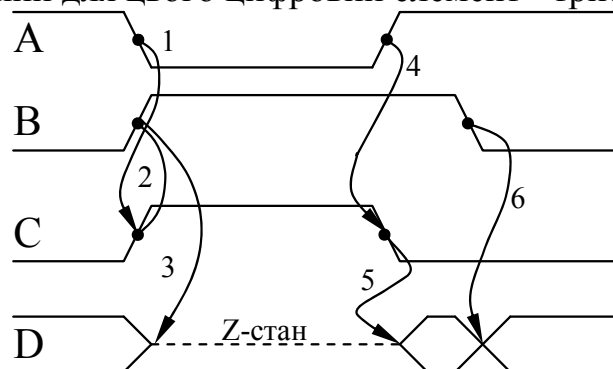


Рис. 2.1 Часові діаграми роботи цифрового пристрою

Для відображення причинно-наслідкових зв'язків між процесами, що протікають у колах взаємодії окремих елементів та вузлів цифрового пристрою чи системи, використовують часові діаграми (рис.2.1). Протікання взаємопов'язаних процесів, що відбуваються на лініях A, B, C і шині D можна легко пояснити, однак зауважимо, що умовне зображення цифрових сигналів на діаграмі, відрізняється від зображення на окремих лініях (стан Z, зображений пунктирною лінією, - це пасивний стан шини, роздвоєні лінії означають активний стан шини, тобто там може бути або лог.0 або лог.1. Стрілки 1,2 показують, що елемент B перейде у стан лог.1 лише після переходу елемента C у стан лог.1, до якого спричиниться елемент A, перейшовши у стан лог.0. Стрілка 3 показує перехід шини D у стан Z після перемикавання B у стан лог.1. Стрілки 4, 5 ілюструють перехід шини D в

активний стан за умови, якщо на A діє перехід $0 \rightarrow 1$, на B рівень лог.1, а на C - перехід $1 \rightarrow 0$. Інверсія активного стану на шині D (стрілка б) буде лише при переході $1 \rightarrow 0$ на B . Рівні напруг або струмів в багатьох випадках не позначають, однак в технічній документації повинен бути вказаний тип логіки пристрою (ТТЛ, КМОН, тощо).

2.2 Класифікація цифрових елементів

Цифровим інтегральним мікросхемам, які є основою елементної бази сучасної ЦТ, властива різноманітність за багатьма ознаками - технологією виготовлення, логікою роботи, будовою електричної схеми, ступенем інтеграції, областю застосування тощо. Проте, незважаючи на таке різноманіття цифрові мікросхеми мають деякі спільні властивості, характеристики та параметри, за якими їх виділяють в окремі групи чи серії.

Елементи ЦТ – це найменші функціональні частини будь-якого цифрового пристрою або системи.

За функціональним призначенням цифрові елементи можна розподілити на логічні елементи (ЛЕ), запам'ятовувачі та допоміжні.

Логічні елементи (ЛЕ) це елементарні комбінаційні пристрої /КП/ з одним виходом, що реалізують, як правило, одну логічну функцію за законами бульової алгебри і характеризуються функціональною повнотою, тобто можуть бути логічним базисом.

Запам'ятовувачі - це елементарні комірки пам'яті однобітної інформації (0 або 1), які можна скласти з ЛЕ.

ЛЕ та запам'ятовувачі утворюють групу цифрових елементів загального призначення - універсальних елементів, на основі яких можна будувати (синтезувати) цифрові пристрої довільного призначення.

ЛЕ загального призначення виготовляють в основному як МІС. Це забезпечує максимальну гнучкість та ефективність при проектуванні на їх основі різних нестандартних і не досить складних функціональних вузлів. До ЛЕ загального призначення належать і такі, що мають розширені логічні та функціональні можливості, тристанові драйвери, ЛЕ з відкритим електродом (колектором або емітером) тощо.

Допоміжні елементи, або елементи спеціального призначення, використовуються в основному для перетворення електричних сигналів в цифрові або навпаки, а також для формування, комутування, індикації, спряження чи узгодження логічних рівнів тощо. Вони практично повністю входять до цифрових пристроїв першої групи – комбінаційних пристроїв, (не враховуючи систем ЦАП і АЦП). Отже, до них належать спеціальні елементи спряження - інтерфейсні мікросхеми, узгоджувачі рівнів сигналів мікросхем різних технологій, приймачі-передавачі як формувачі двійкових сигналів на лініях зв'язку різних типів, елементи цифрової індикації тощо.

Цифрові елементи та пристрої розрізняють також за способом введення та виведення кодових слів, а саме: послідовної, паралельної та змішаної дії.

2.3 Основні характеристики та параметри цифрових мікросхем

До основних характеристик мікросхем ЛЕ, за якими визначаються їх параметри, належать статична, динамічна та навантажувальна. Простіші цифрові елементи характеризуються швидкодією $t_{зам.сер}$, навантажувальною здатністю (коефіцієнтом розгалуження по виходу) $K_{РОЗ}$, коефіцієнтом об'єднання по входу (числом входів логічного елемента) $K_{ОБ}$, завадостійкістю $U_{зав}$, споживаною потужністю $P_{сер}$, напругою живлення U і рівнем сигналу.

Розглянемо ці характеристики разом з параметрами ЛЕ, які з них випливають.

Швидкодія – один із найважливіших параметрів, що характеризується середнім часом затримки розповсюдження сигналу $t_{зам.сер} = \frac{t_{затр}^1 + t_{затр}^0}{2}$, де $t_{затр}^1$ і $t_{затр}^0$ – затримка включення і виключення схеми.

Статична, або передавальна, характеристика мікросхем ЛЕ - це залежність вихідної напруги (потенціалу) U_Y від вхідної напруги U_X на одному з входів ($U_Y = f(U_X)$) при незмінних рівнях напруги на інших входах ЛЕ.

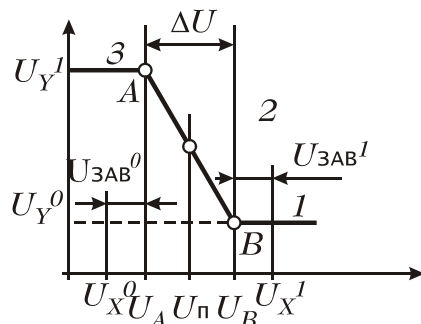


Рис. 2.2 Передавальна характеристика інвертора

На рис. 2.2 зображена типова передавальна характеристика інвертора, на якій можна розрізнити три ділянки: 1 - відповідає стану $U_Y = U_Y^0$; 2 - проміжному логічному стану (перепаду) $U_Y = U_A = U^1 - U^2$ в якому мікросхема знаходиться в активному режимі роботи; 3 – стану $U_Y = U_Y^1$. Зі сторони входу U_X на передавальній характеристиці мікросхеми ЛЕ також можна виділити пороги перемикання U_A для лог.0 і U_B для лог.1 та область між ними - зону логічної невизначеності $\Delta U = U_B - U_A$.

За передавальною характеристикою мікросхем ЛЕ визначаються такі основні статичні параметри, як рівні напруг U^0, U^1 , напруги логічного перепаду $U_A = U^1 - U^2$, порогова напруга U_n і параметри, що належать до завадостійкості.

Завадостійкість (або допустима амплітуда завади) $U_{зав}$ - це максимально допустима величина потенційної завади, яка при появі на вході мікросхеми не викликає хибного перемикання (збою), тобто небажаного переходу цієї мікросхеми із стану 0 у стан 1 або навпаки. Розрізняють

статичну (довготривалу) та динамічну (короткотривалу) завади. У довідниках наводяться дані лише статичної завадостійкості.

Статична завадостійкість ЛЕ вища за динамічну, бо на неї під час перехідного процесу паразитні ємності впливають менше, ніж при короткочасних завадах. Динамічну заваду важче виміряти, оскільки вона залежить більше не від типу мікросхеми, а від зовнішніх факторів.

До статичних характеристик мікросхем ЛЕ належать також вхідна характеристика ЛЕ - $I_X = f(U_X)$, за допомогою якої визначають вхідні струми: $I_X \geq 0$, який витікає із схеми при $U_X = U^1$, і $I_X \leq 0$, який тече у схему при $U_X = U_1^1$, а також вихідні характеристики ЛЕ - $U_Y^0 = f(I_H^0)$ і $U_Y^1 = f(I_H^1)$ де I_H^0 і I_H^1 - струм, що тече у схему, і струм, що витікає із схеми, залежно від числа N ЛЕ, що навантажують даний ЛЕ:

$$I_H^0 = N \cdot I_X^0 ; I_H^1 = N \cdot I_X^1 \quad (2.1)$$

Стабільність режимів роботи ЛЕ залежить від енергетичних параметрів - потужності P і струму I_C споживання. Середня потужність споживання одного ЛЕ при статичному режимі роботи

$$P_{ECT} = 0,5 \cdot (P_{CT}^0 + P_{CT}^1) = 0,5 \cdot E \cdot (I_C^0 + I_C^1), \quad (2.2)$$

де P_{CT}^0, P_{CT}^1 і I_{CT}^0, I_{CT}^1 - потужності і струми, які споживані мікросхемою відповідно при U_Y^0 і U_Y^1 - напруга живлення мікросхеми.

При збільшенні частоти перемикавання в мікросхемах ЛЕ, як правило, зростає струм споживання I_C . Внаслідок цього мікросхема споживає додаткову динамічну потужність P_D , що прямо пропорційна частоті перемикавання. Отже, повна середня потужність $P_C = P_{CCT} + P_D$; значення її можна знайти у довідниках для конкретної максимальної частоти перемикавання.

До статичних характеристик мікросхем ЛЕ належить також коефіцієнт об'єднання входів K_{OB} що визначає максимальне число входів ЛЕ і тим самим число незалежних вхідних змінних X_i . Розрізняють $K_{OB\&}$ входів І та $K_{OB\vee}$ входів АБО Із збільшенням K_{OB} розширюються логічні та функціональні можливості ЛЕ. Але при цьому, як правило, погіршуються такі параметри ЛЕ, як швидкодія та завадостійкість. В окремих серіях мікросхем передбачені спеціальні входи для вмикання так званих логічних розширювачів, що забезпечують збільшення K_{OB} . В існуючих серіях звичайні мікросхеми ЛЕ мають $K_{OB} \leq 8$. За допомогою розширювачів у цих ЛЕ можна збільшити K_{OB} до 10 і більше.

Навантажувальна здатність - це параметр мікросхеми ЛЕ, який

залежно від схемотехнічних особливостей може характеризуватись або значенням вихідних струмів I_Y^0 і I_Y^1 або так званим коефіцієнтом розгалуження K_{PO3} виходу, або за значенням ємності навантаження, тобто сумарною ємністю зовнішніх кіл, що ввімкнених до виходу мікросхеми.

Коефіцієнт розгалуження виходу або коефіцієнт навантаження, K_{PO3} дорівнює максимальному числу входів аналогічних ЛЕ, які можна ввімкнути одночасно до виходу даної мікросхеми при збереженні її основних параметрів. Для різних типів (серій) мікросхем K_{PO3} може дорівнювати від одиниці до кількох десятків. Для мікросхем, наприклад ТТЛ різних серій, $K_{PO3}=5...20$. Типове значення $K_{PO3}=10$. Спеціальні ЛЕ з потужним виходом мають $K_{PO3} \geq 30$

Чим більший K_{PO3} тим ширші логічні можливості мікросхеми і тим менше число ЛЕ потрібно для побудови складного цифрового пристрою. Однак збільшення K_{PO3} обмежене, бо при зростанні навантаження погіршуються такі важливі параметри, як статична завадостійкість та середній час затримки сигналу. У деяких довідниках крім K_{PO3} наводяться значення максимально припустимого вихідного струму I_Y^0 та максимально припустимої ємності навантаження, які треба знати стикуванні (спряженні) ЛЕ різних серій.

Розглянуті параметри залежать від значення напруги живлення E мікросхеми ЛЕ. У довідниках для кожної серії мікросхем наводиться номінальне значення E та його допустимий розкид. Зниження E хоч і зменшує споживану потужність P_C , проте при цьому завжди погіршуються завадостійкість, навантажувальна здатність та швидкодія мікросхеми. Перевищення E призводить до перегрівання мікросхеми, а отже, до дрейфу статичної характеристики мікросхеми (зсуву кривої вправо вверх згідно з рис. 2.2). Вихід з робочого діапазону температур $T_{MIN}...T_{MAX}$, який можна знайти в довідниках, призводить до порушення практично всіх параметрів мікросхеми.

До другорядних параметрів мікросхем ЛЕ належать кількість напруг живлення та їх полярності; *у тих випадках, коли при розробці цифрового пристрою швидкодія та споживана потужність не є найважливішими, при виборі серії основну роль відіграє вартість мікросхеми.*

2.4 Порівняльні характеристики цифрових мікросхем

На характеристики та параметри цифрових мікросхем значно впливає технологія їх виготовлення. Групу мікросхем, що виконані за однаковою або близькою технологією і мають схожі схемотехнічні рішення та деякі характеристики, називають серією мікросхем. Різновидів технологій виготовлення цифрових мікросхем досить багато. Але незважаючи на це всіх

їх об'єднує логічна спільність, а саме – однотипність "цеглинок" цифрової техніки ЛЕ, тобто базис, у якому можна виконувати логічне проектування (синтез) цифрових пристроїв, абстрагуючись при цьому від будь-якої конкретної серії чи технології. Це і є головною перевагою ЦТ.

Прив'язка до конкретної серії мікросхем необхідна на етапі схемотехнічного проектування. Тут вже потрібні знання про характеристики та параметри мікросхеми, виходячи з вимог щодо роботи цифрового пристрою та критеріїв, які поставлені перед розробником.

Залежно від схемотехнічної реалізації ЛЕ поділяють на такі типи мікросхем:

- ТТЛ (Т²Л) – транзисторно-транзисторної логіки на біполярних транзисторах;
- МОН або МДН – на польових транзисторах (МОН - метал-оксид-напівпровідники; МДН - метал-діелектрик-напівпровідник);
- КМОН – із симетричною комплементарною структурою на польових транзисторах *n*- та *P*-типів;
- ЕЗЛ – емітерно-зв'язаної логіки на біполярних транзисторах;
- ІІЛ (І²Л) - інтегрально-інжекційної логіки.

Кожна технологія виготовлення мікросхем безперервно удосконалюється у напрямку збільшення швидкодії, зменшення споживаної потужності та зростання ступеня інтеграції.

У табл. 2.1 зведені усереднені характеристики мікросхем, що виготовлені за найбільш поширеними в практиці технологіями.

Таблиця 2.1 Характеристики мікросхем виконаних в деяких технологіях

Тип Техно-логії	Серія мікросхем	Параметри одного ЛЕ				Зарубіжний аналог
		$t_{зс}$, нс	$P_{сст}$, мВА	$t_{зс}P_{сст}$, нДж	E , В	
ТТЛ	K155, KM155	10	10	100	5	SN74
	K131	11	23	263	5	SN74
	K134, KP134	66	1	66	5	SN74
ТТЛШ	K551	3	20	60	5	SN74 S
	K1531, KP1531	3	4	12	5	SN74 F
	K555, KM555, 533	10	2	20	5	SN74 LS
	K1533, KP1533	4	2	8	5	SN74 ALS
ЕЗЛ	K100, K500, 700	2	25	50	-5,2	MC10K
	K1500	0,75	40	30	-4,5	F00K
	K1800	1,5	20	30	-5,2	MC1080
КМОН	K176	100	10 ⁻³	0,1	9	CD4000B
	K561, 564, 1561	15-50	10 ⁻³	3	3-15	CD4000
І ² Л	K583, KP 584	5	0,2	1	5-9	-

З метою якісного та кількісного порівняння на рис. 2.3,а показані логічні

рівні для входів мікросхем КМОН та ТТЛ у процентному співвідношенні щодо напруги живлення, а на рис. 2.3,б - характеристики рівнів напруг на вході X та виході Y ЛЕ ТТЛ. Різниці між гарантованими входними та вихідними, високим та низьким рівнями напруг визначають можливості ЛЕ не реагувати на завади. Отже, згідно із зображеними на рис. 2.3,б даними характерними гарантованими для ТТЛ рівнями ϵ : на вході - для $U_X^0 = 0...0,8$ В, для $U_X^1 = 2,0...5,0$ В; на виході - для $U_Y^0 = 0...0,4$ В; для $U_Y^1 = 2,4...5,0$ В.

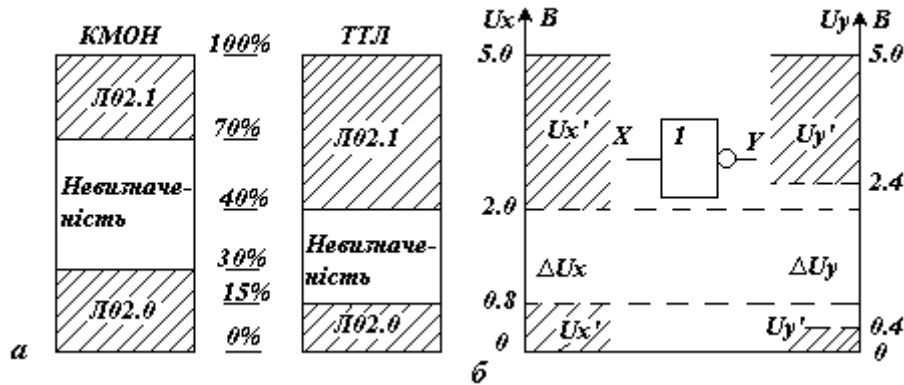


Рис. 2.3 Рівні напруг для входів ЛЕ ТТЛ та КМОН

Мікросхеми ЕЗЛ належать до найбільш швидкодіючих ЛЕ (частота перемикання ЛЕ серій К100 і К500 становить 150 мГц). Висока швидкодія забезпечується за рахунок глибоких зворотних зв'язків і ненасиченого режиму роботи всіх транзисторів, який виключає процес розсмоктування неосновних носіїв заряду.

Мікросхеми ЛЕ І²Л належать до перспективних компактних і швидкодіючих ЛЕ. Вони забезпечують високу щільність розміщення елементів на кристалі (приблизно у 100 разів більшу від ТТЛ). ЛЕ І²Л споживають на порядок меншу потужність порівняно з ЛЕ на КМОН. Висока швидкодія цих ЛЕ досягається за рахунок як власної технології, так і застосування діодів Шоткі.

2.5 Схеми логічних елементів

Тип логіки (ТТЛ, КМОН, тощо) визначається внутрішньою структурою ЛЕ. Логічні елементи можуть бути виконані різноманітними

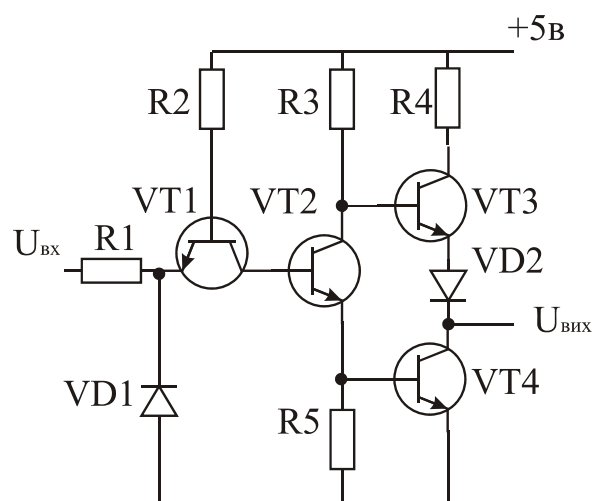


Рис. 2.4 Логічний елемент "НЕ" ТТЛ

способами, у вигляді ІС або дискретних схем. Особливості будови визначають основні характеристики ЛЕ. На рис.2.4 наведена базова схема ТТЛ інвертора Діоди VD1,VD2 підвищують завадостійкість ЛЕ не допускаючи від'ємних викидів напруг, які можуть виникати при перехідних процесах. Збільшуючи кількість емітерів на транзисторі VT1 наприклад до двох, можна одержати схему "2I-HE". Транзистор VT2 виконує функцію фазорозщеплювача, подаючи на транзистори VT3, VT4 протифазні сигнали. Таким чином, коли транзистор VT3 відкритий, то VT4 закритий і навпаки. Це значно зменшує розсіювану потужність у порівнянні із тим випадком, коли б на схемі були відсутні елементи VT3, VD2. Типові робочі напруги ТТЛ (0,+5в) набули широкого розповсюдження в промисловості і стали фактичним стандартом, якого дотримуються навіть тоді, коли використовують при розробці пристроїв інші типи логіки. Елементи ТТЛШ відрізняється від ТТЛ наявністю в схемі діодів Шотки, які не дозволяють біполярним транзисторам заходити в область глибокого насичення, цим самим підвищуючи загальну швидкодію ЛЕ (30-80МГц ТТЛШ у порівнянні із 15-20МГц ТТЛ).

Широкого розповсюдження набули типи ЛЕ виконані на польових структурах (метал-діелектрик-напівпровідник, МДН). Основна особливість мікросхем цих елементів є незначне споживання вхідного струму в статичному режимі ($\leq 100 \mu\text{A}$). Основним недоліком до останнього часу була низька швидкодія ($< 3 \text{ МГц}$), однак на сучасному рівні розвитку наявні КМОН серії ІС з швидкістю перемикання сигналу до 80МГц.

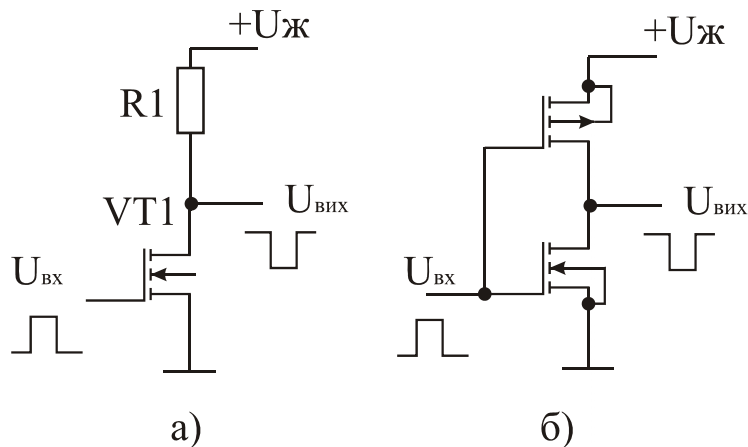


Рис. 2.5 Базові схема логічного елемента "НЕ" КМОН

На рис.2.5 а) показана базова схема МОН інвертора з резистором навантаження, а на рис.2.5 б) інвертор на парі МОН транзисторів різних каналів, тобто комплементарних МОН транзисторах (КМОН). Особливість роботи таких транзисторів дозволяє обійтися у схемі без фазорозщеплюючого транзистора, і тим самим спрощує схему. Основним недоліком таких схем є їхня чутливість до статичної електрики, що спричиняє їх вихід з ладу. Тому при роботі з КМОН ІС слід користуватись антистатичним браслетом, та заземлити жало паяльника. Самі ж мікросхеми (як, до речі і польові транзистори) слід транспортувати загорнутими у фольгу

або в спеціальних антистатичних футлярах.

До елементної бази надвисокої швидкодії належать ЛЕ емітерно-зв'язаної логіки (ЕЗЛ). Базовою схемою ЕЗЛ є струмовий ключ, побудований за схемою диференційного підсилювача, наведений на рис. 2.6. Така схема може виконувати роль інвертора та буферного повторювача вхідного логічного сигналу.

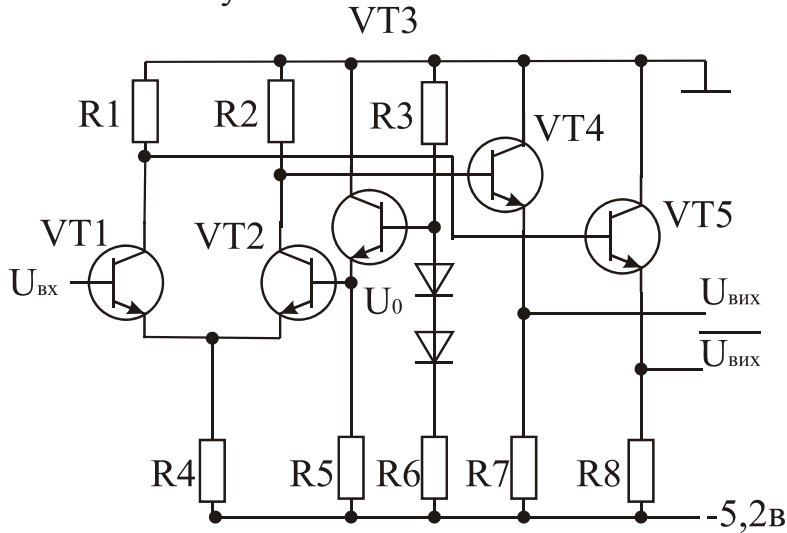


Рис. 2.6 Струмовий ключ ЕЗЛ

Поріг перемикання струмового ключа задається зовнішньою опорною напругою U_0 , яка формується спеціальною схемою R3,R5,R6,VT3. Вихідні емітерні повторювачі забезпечують високу навантажувальну здатність по струму (до $K_{роз}=15$). В серійних ЛЕ ЕЗЛ колекторні кола заземляють, а емітерні під'єднують до від'ємної напруги живлення, що підвищує завадостійкість схеми. Залежно від способу кодування вхідного сигналу, тобто від того, який рівень напруги вважати лог.0 або лог.1 ЕЗЛ схема може виконувати різні операції. Для збільшення кількості входів струмового ключа до VT1 паралельно (колектор до колектора, емітер до емітера) під'єднують додаткові біполярні транзистори, при чому кількість базових входів схеми збільшується. Належність до тої чи іншої логіки визначається взятим рівнем U_x відносно рівня U_0 . Якщо за лог.0 взяти рівень U_{ex}^0 , що лежить нижче від рівня U_0 , а за лог.1 рівень U_{ex}^1 , що лежить вище U_0 - одержимо функціонування схеми в позитивній логіці. Двовходовий варіант такої схеми реалізує функції АБО та АБО-НЕ в залежності від вибору виходу, з якого знімається сигнал. Інший вибір напруг, призводить до функціонування схеми в негативній логіці, тоді схема реалізує функції І та І-НЕ. Для того, щоб глибше розібратись з особливістю функціонування такої схеми, слід дослідити роботу двовходового варіанту такої схеми в MicroCAP.

2.6 Елементи з розширеними функціональними можливостями

На практиці виникають ситуації, коли потрібно розширити логічні або

інші функції ЛЕ, не порушуючи при цьому властиве кожному ЛЕ функціонування і не виходячи разом з тим за межі оптимальних технічних параметрів. До характеристик та параметрів ЛЕ, які досить часто доводиться розширювати, належать навантажувальна здатність K_{PO3} , кількість логічних змінних K_{OB} , логічне функціонування, діапазони логічних рівнів тощо. Ці та інші можливості цифрових елементів можна розширити на основі аналітичного підходу шляхом застосування аксіом та законів бульової алгебри або з допомогою ЛЕ спеціального призначення.

2.6.1 Синтезовані логічні елементи

Збільшення кількості входів ЛЕ, можна досягнути за допомогою каскадування аналогічних елементів, користуючись законом дуальності (правилом де Моргана) або способом подвійної інверсії.

Нехай у базисі 2І-НЕ потрібно побудувати шестивходовий елемент І. Для цього, застосовуючи спосіб подвійної інверсії, задану логічну функцію розбиваємо на групи по 2І-НЕ:

$$Y = X_1 X_2 X_3 X_4 X_5 X_6 \equiv \overline{\overline{X_1 X_2} \overline{X_3 X_4} \overline{X_5 X_6}}$$

Як видно з одержаного виразу, синтез схеми вимагає значних апаратних затрат. При наявності одного ЛЕ типу 4АБО-НЕ достатньо лише 3 ЛЕ 2І-НЕ. Це легко довести за правилом де Моргана:

$$Y = X_1 X_2 X_3 X_4 X_5 X_6 = \overline{X_1 X_2} \vee \overline{X_3 X_4} \vee \overline{X_5 X_6}$$

Розширення по І для цього випадку побудовано на рис.2.7. Багатовходові елементи І можна застосувати для виявлення лог.0 на виході, наприклад, якогось функціонального пристрою.

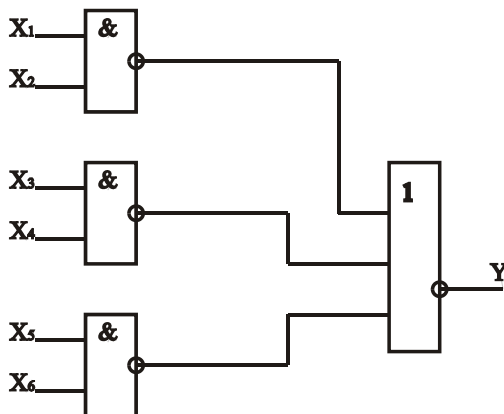


Рис. 2.7 Реалізація функції 6І

пірамідальним. У пірамідальній структурі підлягають обробці одночасно всі операнди, бо логічні операції у такій схемі розпаралелені. Тому пірамідальна структура забезпечує мінімально можливу затримку сигналу.

Побудова багатовходового АБО виконується аналогічно - за принципом дуальності. Розширення по АБО використовують, наприклад, для виявлення хоча однієї лог.1 на виході якогось цифрового пристрою.

Структурний метод об'єднання маловходових елементів у багатовходову схему називають

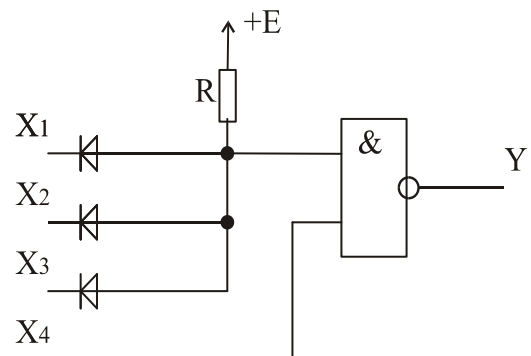


Рис. 2.8 Діодне розширення ЛЕ

Кількість входів ЛЕ можна збільшити також під'єднанням додаткових зовнішніх діодів і резистора так, як показано на рис.2.8. Така "діодна" логіка, (в закордонній літературі існує ще один термін: *"логіка Мікі-Мауса"*) [3] може бути корисною в деяких випадках, наприклад при експрес-ремонті цифрової апаратури, у разі виходу з ладу одного з входів ЛЕ. Резистор необхідно розраховувати відповідно до вимог за швидкодією (найчастіше $R = 1..2$ кОм). При більшому значенні R час затримки t_{sc} зростає. Треба мати на увазі, що при такому способі збільшення $K_{об}$ завадостійкість для нульового рівня сигналу U_x^0 буде нижчою, бо на безпосередньому вході ЛЕ (у точці з'єднання діодів) напруга стане дещо вищою від напруги на вході схеми, причому на величину спаду напруги на діодах. Тому зловживати використанням діодної логіки не слід.

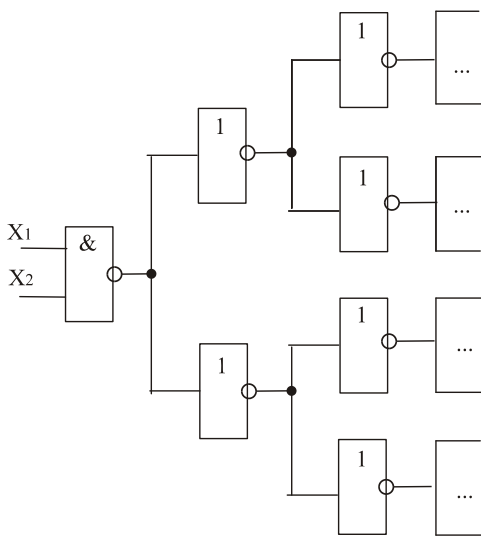


Рис. 2.9 Розгалуження вихідного сигналу

Для збільшення кількості виходів ЛЕ, тобто коефіцієнту $K_{роз}$ і відповідно навантажувальної здатності ЛЕ, більш ефективним є спосіб каскадного розгалуження сигналу. За цим способом можна одержати досить великий коефіцієнт $K_{роз}$ навіть до 100.

Схема каскадного розгалуження ЛЕ 2І-НЕ на інверторах показана на рис.2.9.

2.6.2 Логічні елементи з відкритим колектором

Більші логічні можливості без особливих апаратурних затрат мають ЛЕ ТТЛ, які відрізняються від базової схеми тим, що у них точки колектора або емітера (або обох електродів) транзистора-фазорозщеплювача $VT2$ виведені назовні мікросхеми. Це спеціальні ІС, що мають допоміжні виводи колектора (К) і емітера (Е) (наприклад, К155ЛР1, К155ЛР3 і К155ЛР4). Приєднавши додатково зовнішні транзистори, можна одержати підсилені прямий та інверсний сигнали. При цьому

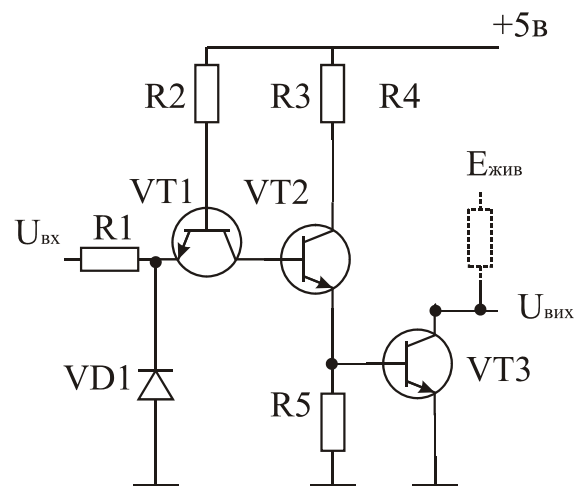


Рис. 2.10 Інвертор з відкритим колектором

потужність зовнішніх транзисторів може бути на порядок більша від аналогів, виконаних в ІС. Однак такі мікросхеми не є поширеними в практиці, тому коли потрібно мати $K_{роз} > 10$, більш ефективні спеціальні мікросхеми з відкритим колектором, вихідного транзистора. Зображена на рис. 2.10 схема ТТЛ з відкритим колектором має ту особливість, що вона порівняно з базовою схемою дозволяє більш гнучко реалізувати різноманітні задачі проектування. Це, зокрема, використання ЛЕ з відкритим колектором як інтерфейсу для нетипового навантаження, що має своє живлення, наприклад, для спряження або керування зовнішніми пристроями виведення інформації, індикації, сигналізації тощо. Мікросхеми ЛЕ з відкритим колектором (наприклад, К155Л11, І55ЛН3) дозволяють підводити до вихідного транзистора $E = 30$ В. Таким ЛЕ можна задати лінійний режим роботи і використовувати їх як підсилювачі. За допомогою ЛЕ з відкритим колектором можна нарощувати кількість входів ЛЕ (тобто збільшувати $K_{об}$).

Як видно з рис. 2.10, у принциповій схемі ЛЕ з відкритим колектором відсутній верхній вихідний транзистор та його опір навантаження. На виході Y цього елемента, високий рівень може формуватись лише при під'єднанні зовнішнього навантаження (резистора, реле, світлодіода тощо). Крім схемотехнічних можливостей ЛЕ з відкритим колектором здатні реалізувати додатково ще й логічні операції завдяки тому, що вони допускають паралельне під'єднання аналогічних виходів на одне спільне навантаження. Таке об'єднання виходів називається монтажною логікою,

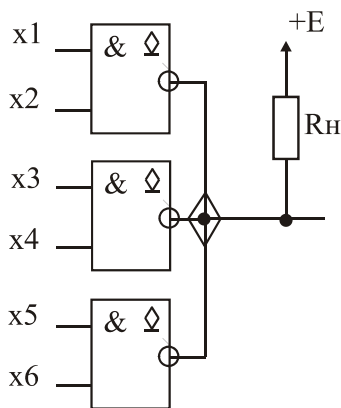


Рис. 2.11 Схема 2І-НЕ

бо вона утворюється у результаті вищезгаданого з'єднання (монтажу).

На рис. 2.11 показана схема на ЛЕ 2І-НЕ з відкритими колекторами, з'єднання яких на виході Y , разом з R_H утворюють монтажне І;

На об'єднаному виході низький рівень ($Y = 0$) буде тоді, коли хоча б один з елементів має на своєму виході $Y_i = 0$ тобто відкритий вихідний транзистор. Це можливе тільки при лог.1 на входах X_i хоча б на одному з ЛЕ. З другого боку,

тільки при наявності лог.1 одночасно на всіх виходах Y_i можна отримати лог.1 на загальному виході Y . Це можливе тільки при наявності лог.0 хоча б на одному з входів X_i ЛЕ схеми. Отже, виходи Y_i ЛЕ разом з R_H утворюють монтажне І:

$$Y = Y_1 Y_2 Y_3 = \overline{X_1 X_2 X_3} \overline{X_4 X_5 X_6}$$

Якщо застосовувати правило де Моргана, цей вираз можна переписати:

$$Y = \overline{X_1 X_2 \vee X_3 X_4 \vee X_5 X_6}$$

Як видно з останнього виразу, щодо вхідних змінних дана схема реалізує операцію 2І-ЗАБО-НЕ.

ЛЕ з відкритим колектором використовуються також для підвищення рівня лог.1, зокрема, для узгодження виходів ТТЛ із входами КМОН при напрузі живлення останніх $E = +5\text{В}$. Їх також успішно застосовують як драйвери цифрових сигналів.

У всіх розглянутих випадках застосування ЛЕ з відкритим колектором потрібно вміти вибирати опір навантаження R_H . Від правильного вибору резистора R_H залежать практично майже всі параметри ЛЕ, а саме такі, як потужність розсіювання, час затримки сигналів, навантажувальна здатність та завадостійкість. Від значення R_H залежать вихідні струми I_Y^0 та I_Y^1 схеми, а отже, й вихідні напруги U_Y^0 та U_Y^1 . Щоб не виходити за межі їх допустимої зони нечутливості статичної характеристики ЛЕ, запас за завадостійкістю логічних рівнів U_Y^0 та U_Y^1 для мікросхем ТТЛ з відкритим колектором має бути таким самим, як і для ЛЕ із складним виходом (див. табл. 3.2), тобто $U_Y^0 \leq 0.4\text{В}$ і $U_Y^1 \geq 2.4\text{В}$. Отже, оптимальне значення R_H не повинно виходити за гранично допустимі межі, тобто $R_{H\min} \leq R_H \leq R_{H\max}$

Значення $R_{H\min}$ і $R_{H\max}$ для конкретного випадку залежать від мікросхем вибраної серії від числа під'єднаних до спільної точки входів n та виходів m мікросхем.

ЛЕ з відкритим емітером відрізняються від ЛЕ з відкритим колектором тим, що емітер вихідного транзистора не під'єднаний до корпусу ("землі") мікросхеми, а виведений назовні, а колектор під'єднаний, до виводу, на який подається напруга живлення E . Навантаження до цих ЛЕ під'єднується між виводами відкритого емітера і корпусу. Отже, вихідний каскад ЛЕ являє собою емітерний повторювач, який на відміну від ЛЕ з відкритим колектором забезпечує кращу навантажувальну здатність. Зокрема, ЛЕ з відкритим емітером буде забезпечувати більший вихідний струм лог.1 I_Y^1 , що витікає зі схеми, ніж I_Y^0 , що втікає у схему ЛЕ, як це має місце у ЛЕ з відкритим колектором. ЛЕ з відкритим емітером (зокрема, мікросхеми 559ИП4, 1102АП2 і т.ін.) використовуються також як інтерфейсні мікросхеми мікропроцесорної техніки.

2.6.3 Тристановий драйвер

Шини передачі інформації, це як правило довгі провідники, що мають значні паразитні ємності і тому чутливі до завад. Крім того, активний вихід ЛЕ незручний у тих ситуаціях, коли потрібно вести обмін даними одночасно з кількома функціональними блоками або вузлами, як це має місце у мікропроцесорній техніці.

Якщо різні цифрові вузли під'єднані до інформаційної шини за допомогою ЛЕ, які можуть виводити на свої вихідні лінії тільки лог.0 або лог.1, то може виникнути ситуація, коли одна частина пристрою "вимагає" один логічний рівень, інша – інший. За таких умов фактичний стан лінії стає невизначеним, бо виникає конфлікт на шині, коли кожний з пристроїв намагається встановити на лінії свій рівень.

Сумісну роботу кількох блоків на одній лінії інформаційної шини забезпечує ЛЕ з трьома вихідними станами. Це так званий тристановий драйвер, або Z-буфер, у якого на виході крім звичайних двох станів (0 і 1) є ще третій - "обрив", тобто високоомний вихід. Вихідний опір ЛЕ в третьому (так званому стані Z) становить сотні кілоом. ЛЕ з трьома станами поєднує всі переваги базового ЛЕ - високу швидкодію та завадостійкість, а також здатність працювати на загальну шину, що властиво для ЛЕ з відкритим колектором.

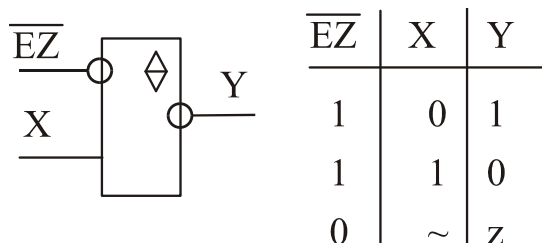


Рис. 2.12 Тристановий драйвер

На рис. 2.12 наведені таблиця істинності та умовне позначення тристанового драйвера. У цьому випадку активним сигналом, тобто таким, який змінює стан ЛЕ згідно із своїм призначенням (а саме у стан Z), в лог.0, а не лог.1, як у позитивній логіці.

Це характерно для мікропроцесорної техніки, де керуючі сигнали, як правило, підлягають стандарту негативної логіки, що позначається ризикою заперечення над позначенням сигналу. Тому дозволом на виконання потрібної функції (в даному випадку переходу ЛЕ в стан Z є стан лог.0 на вході ЛЕ при $\overline{EZ} = 0$).

Тристанові КМОН-драйвери відрізняються від ТТЛ-драйверів меншою швидкодією, у стані Z вони забезпечують практично ідеальну розв'язку (вихідний опір $> 10 \text{ МОм}$).

Тристанові КМОН-драйвери успішно використовують як буферні підсилювачі при роботі на значне ємнісне навантаження. Зокрема, мікросхеми К561ЛН1, 564ЛН1, 564ЛН2 можна ефективно використовувати для побудови оперативних запам'ятовувальних пристроїв очікувальних мультівібраторів, при роботі на довгі лінії тощо.

Час затримки t_{3C} поширення сигналу у цих ЛЕ залежить від напруги живлення, яку для МОН-структур можна змінювати у широких межах. Наприклад, для мікросхеми К561ЛН1 $t_{3C} = 65 \text{ нс}$ при $E = 15 \text{ В}$, а час переходу від стану Z у U_Y^1 становить 40 нс.

При $E = 5 \text{ В}$, що потрібно для узгодження з ЛЕ ТТЛ, *всі перехідні процеси збільшуються у 3 рази*.

Тристанові драйвери широко застосовують у цифрових мікропроцесорних системах як шинні формувачі, для двонапрямлених ліній.

2.7 Інтерфейсні мікросхеми

Завдяки розвитку мікропроцесорної техніки з'явився новий клас цифрових мікросхем - інтерфейсних. Вони, як правило, побудовані на СІС і ВІС і не призначені для виконання логічних операцій їх основне призначення— організація міжблокових з'єднань та інтерфейсів цифрових

систем (мікроЕОМ).

Кожний блок цифрової системи зв'язаний з іншим блоком шиною. Шини, що передають по своїх лініях двійкову інформацію у вигляді слів, бувають одно- та двонаправленими. *Однонаправлені* шини забезпечують передання цифрових сигналів лише в одному напрямі - від джерела (передавача) інформації до приймача. До них належать адресні та шини керування. *Двонаправлені* шини - це, як правило, шини даних, з допомогою яких сигнали передаються в обох напрямках (хоча при потребі можуть використовуватись як однонаправлені).

Оптимальне функціонування блоків цифрової системи значною мірою залежить від впливу шини на параметри та характеристики сигналу. Адже сигнали, що виробляє передавач, пройшовши певну відстань по шині до приймача, практично завжди втрачають свою форму (особливо при переданні сигналів по кабелю, який має велику ємність) і погіршують свої характеристики та параметри. Саме тому блоки чи вузли цифрової системи під'єднують до шини не безпосередньо, а через інтерфейсні мікросхеми - модулі, які забезпечують оптимальне міжблокове спряження.

Можна виділити інтерфейсні мікросхеми, що утворюють групу так званих *шинних передавачів-приймачів* (ШПП). Основне призначення ШПП полягає у підсиленні електричних сигналів, що передаються по двонаправлених лініях зв'язку (шинах), а також для узгодження рівнів цих сигналів і реалізації функцій часової та логічної буферизації даних. Найпростіші серед інтерфейсних мікросхем *шинні формувачі* (ШФ), або шинні драйвери. ШФ орієнтовані головним чином для ретрансляції сигналів та електричного узгодження однонаправлених шин.

Логічний стан (0 або 1) лінії шини у будь-який момент часу визначається рівнем сигналу, який у вигляді струму проходить з одного блока в інший. Чим більше блоків під'єднано до лінії, тим більша сила струму навантаження; і якщо вона перевищує припустиме значення, це може призвести до невідповідності логічних рівнів на шині. Тому надлишковий струм, що при цьому виникає, необхідно якимось чином зняти з лінії. Ця процедура носить назву *буферування* і реалізується з допомогою ШПП, що в даному випадку відіграє роль звичайного повторювача (буфера). Отже, буферування служить для підтримання параметрів сигналу у припустимих межах. Ця процедура особливо необхідна при значному ємнісному навантаженні. Наприклад, для передання сигналу по довгому кабелю, що має велику ємність, буфер потрібно встановлювати зразу на виході передавача до кабеля. Для буферування шин застосовують інтерфейсні мікросхеми, що побудовані на ЛЕ а відкритим колектором або на тристанових драйверах. Ці ЛЕ, крім того, дозволяють реалізувати на кожній лінії шини монтажу логіку з різними блоками системи, зокрема "монтажне АБО" для об'єднання їх виходів.

Залежно від числа блоків, що під'єднані до шини, ступінь її навантаження буде різним. Тому системні (магістральні) шини мають вищу

навантажувальну здатність, бо до них під'єднується велика кількість блоків та вузлів системи. Значно меншу навантажувальну здатність мають локальні шини даних, бо вони під'єднуються до окремих блоків системи.

Схеми ШПП і ШФ виконують (в основному на ТТЛШ) чотирирозрядними (К589АП6, К589АП26, 555ИП6), восьмирозрядними (К580ВА86/87, 555АП6) і як здвоєні чотирирозрядні (КІ533АП3).

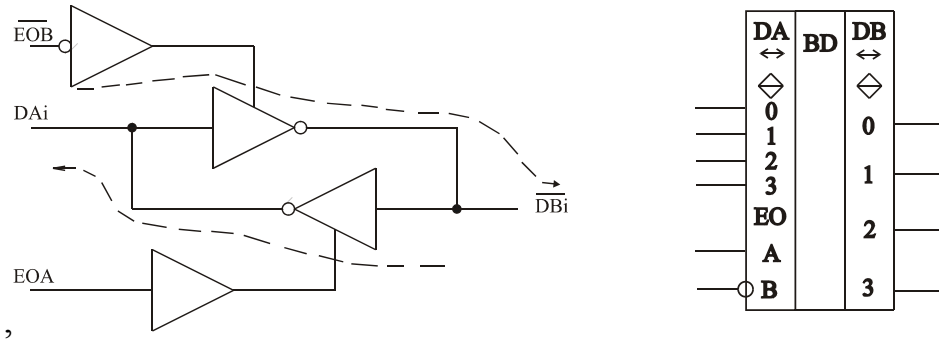


Рис. 2.13 Один канал і схемне позначення двонапрявленого ШПП

Принцип дії двонапрявленого ШПП пояснює рис 2.13 та табл. 3.4. У колі кожного каналу (чи розряду) схеми буфера ввімкнені зустрічне два тристанові драйвери, роботою кожного з яких (через повторювачі) керують сигнали дозволу EOA і \overline{EOB} .

Таблиця 2.2 Режими роботи ШПП

ЗНАЧЕННЯ КЕРУЮЧИХ СИГНАЛІВ		НАПРЯМ ПЕРЕДАЧІ ІНФОРМАЦІЇ
EOA	\overline{EOB}	
0	0	Від DA_i до \overline{DB}_i
0	1	Передача відсутня
1	0	Заборона
1	1	Від \overline{DB}_i до DA_i

Для сигналу EOA активним є високий рівень (лог.1), а для сигналу \overline{EOB} - низький рівень (лог.0). Отже, при $EOA = 0$ ($\overline{EOB} = 1$) відбувається передача від \overline{DB}_i до DA_i , бо працює нижній драйвер. При $\overline{EOB} = 0$ ($EOA = 0$) працює верхній драйвер, і сигнал передається від DA_i до \overline{DB}_i . Передача сигналу буде відсутня при $EOA = 0$ $\overline{EOB} = 1$, бо тоді обидва драйвери переходять у високоімпедансний стан Z . Зрозуміло, що одночасна дія двох активних сигналів $EOA = 1$ і $\overline{EOB} = 0$ недопустима.

Слід зауважити, що в деяких випадках замість ШПП або ШФ залежно від навантаження на шину можна застосувати К155ЛН1, К555ЛН1, К561ЛН1 або К555ЛН2, К155ЛН3, К155ЛН5, які забезпечують підвищену навантажувальну здатність у порівнянні з іншими представниками своїх серій ІС.

2.8 Узгоджувачі рівнів

При проектуванні цифрових пристроїв часто доводиться використовувати мікросхеми різних серій, які відрізняються за різними ознаками. Сумісне застосування різноманітних мікросхем викликає значні труднощі. На практиці найчастіше доводиться узгоджувати мікросхеми ЛЕ різних технологій і між собою, і з такими навантаженнями, як цифрові індикатори, дисплеї, лампи розжарювання тощо. Узгодження ЛЕ з такими елементами у загальному випадку можуть бути: логічними - для формування керуючих кодів; за струмом, залежно від сили і напрямку на вході та виході, і за напругою - для узгодження вхідних та вихідних рівнів. Очевидно, що залежно від типу навантаження може виникати необхідність у поєднанні різних типів узгодження.

Спряження типів мікросхем пов'язане в першу чергу з необхідністю узгодження вхідних та вихідних рівнів напруг та струмів сигналів. Якщо, одним вузлом схеми за вимогою щодо швидкодії відповідають мікросхеми ЕЗЛ, то для інших є неприпустимою досить значна потужність, яку споживають ЛЕ ЕЗЛ, і тому для цих ЛЕ необхідні ЛЕ малопотужних серій. Крім електричного, також необхідно забезпечити й часове спряження різнотипних ЛЕ.

На практиці особливий інтерес викликають узгоджувачі рівнів типу ТТЛ ↔ КМОН.

Узгодження ТТЛ → КМОН при однаковій напрузі живлення +5 В здійснюється безпосереднім під'єднанням виходу ЛЕ ТТЛ до входу ЛЕ КМОН. Оскільки на вході ЛЕ КМОН струм мізерно малий, узгодження за струмом забезпечується автоматично. Якщо потрібно збільшити швидкодію, можна використати більш потужне джерело вихідного струму. Однак узгодження за напругою вимагає додаткових заходів, які викликані тим, що мінімальний рівень $U_{Y \min}^1$ ТТЛ (=2.4В) нижчий за мінімально припустимий рівень $U_{X \min}^1$ КМОН (=3.5В). Тому для підвищення запасу завадостійкості за виходом ЛЕ ТТЛ між його виходом і +Е слід увімкнути узгоджувальний резистор 2...5 кОм залежно від серії ТТЛ.

Якщо джерела живлення обох ЛЕ різні, причому $E_{ТТЛ} < E_{КМОН}$, що часто зустрічається на практиці, то при застосуванні ЛЕ ТТЛ з відкритим колектором, припустиме значення узгоджувального резистора визначається за формулами (2.3) і (2.4).

$$R_{H \max} = \frac{U_H}{I_H^1} = \frac{E - U_Y^0}{m \cdot I_Y^1}$$

$$R_{H \min} = \frac{U_H}{I_H^0} = \frac{E - U_Y^1}{I_X^1 + I_X^0 + n}$$

Тут: E- напруга живлення
 I_Y^0, U_Y^0 - вихідні струм та напруга при лог.0, I_Y^1, U_Y^1 - вихідні струм та напруга при лог.1
 I_X^1, I_X^0 - вхідні струми

Для узгодження типу ТТЛ→КМОН при напрузі живлення останнього $E_{\text{КМОН}} \approx 10\text{В}$ застосовують мікросхеми узгоджувачів рівнів К564ПУ6, К564ПУ7 і К564ПУ8, які містять у своєму складі по шість елементів. Відмінною особливістю мікросхеми К564ПУ6 є наявність окремих для кожного каналу узгодження сигналів дозволу, з допомогою яких відповідний вихід елемента схеми може перейти у Z -стан.

Узгодження КМОН \rightarrow ТТЛ, тобто коли джерелом сигналу є вихід ЛЕ КМОН, а навантаженням - вхід ЛЕ ТТЛ, має певні особливості. Узгодження за напругою забезпечується автоматично внаслідок того, що при $E_{ТТЛ} = E_{КМОН}$ у ЛЕ КМОН для лог.1 $U^1_{Y\min} > U^1_{X\min}$ ЛЕ ТТЛ і для лог.0 у ЛЕ КМОН $U^0_{Y\max} < U^0_{X\max}$ ЛЕ ТТЛ. Значення струму при лог.1 I_X^1 ЛЕ ТТЛ (який тече у схему) дорівнює приблизно 0,1 мА, і якщо число n входів ЛЕ ТТЛ, що навантажують ЛЕ КМОН, не перевищує $K_{OB} = 5..10$, узгодження за струмом при лог.1 забезпечується без додаткового збільшення навантажувальної здатності ЛЕ КМОН. Однак узгодження за струмом при лог.1 забезпечити автоматично не вдається через те, що струм I_X^0 ЛЕ ТТЛ майже дорівнює струму I_Y^0 ЛЕ КМОН. Струм I_X^0 навіть одного ЛЕ ТТЛ (наприклад, серії 155 $I_X^0 \approx 1...2$ мА) перевищує струм I_Y^0 ЛЕ КМОН, а отже, знижує навантажувальну здатність останнього. Узгоджувачами за струмом у даному випадку можуть бути мікросхеми КМОН 564ЛЕ5, 564ЛЕ6 і 564ЛЕ10, в яких всі входи для цього об'єднуються. Тоді навантажувальна здатність ЛЕ визначатиметься числом паралельно ввімкнених n -канальних транзисторів, і для розглянутих мікросхем узгоджувачів вона зростає у 2-4 рази.

У випадку, коли напруги живлення $E_{ТТЛ} < E_{КМОН}$, необхідно виконувати узгодження як за струмом, так і за напругою. Для цього використовують узгоджувачі рівнів (див. рис. 2.15-типу КМОН \rightarrow ТТЛ) – 564ЛН2 або 564ПУ4, які мають підвищену навантажувальну здатність за струмом. Ці мікросхеми можуть живитися від одного джерела живлення ЛЕ ТТЛ ($E = +5$ В). Оскільки $E_{КМОН} > +5$ В, та для запобігання руйнування узгоджувача застосована, схема захисту на діодах, яка допускає на своєму вході напругу до +15 В. Відповідно до типу ЛЕ ТТЛ обидва узгоджувачі

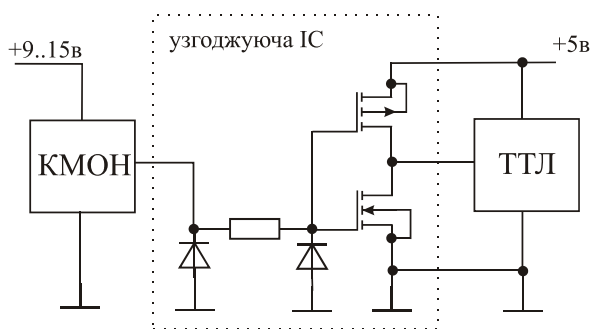


Рис. 2.14 Узгодження КМОН-ТТЛ

можуть мати при $E = +5$ В ($\pm 5\%$) $K_{PO3} = 2$ (для універсальних ЛЕ), $K_{PO3} = 9$ (для малопотужних ЛЕ) і $K_{PO3} = 16$ (для мікропотужних ЛЕ). Останній, наприклад, має вихідний каскад з відкритим стоком, що забезпечує струм I_Y^0 , достатній для під'єднання десяти входів ТТЛ серії.

Більшість біполярних мікросхем різних типів можуть сумісно використовуватись без застосування складних узгоджувачів (в окремих випадках з використанням узгоджувачів резисторів або подільників

напруги). Цьому сприяє те, що за стандартні рівні вхідних і вихідних напруг у цих мікросхемах вибрані рівні ТТЛ. До них належать мікросхеми ТТЛШ, І²Л та комбіновані типу ТТЛ-І²Л і ТТЛ-ЕЗЛ. Виняток становлять мікросхеми ЕЗЛ-технології, як мають такі рівні напруг: живлення $E = -5.2\text{В}$ і $-2,0\text{В}$, $U_Y^0 = -1.85... -1.65\text{В}$ і $U_Y^1 = -0.96... -0.81\text{В}$. Електричне спряження ЛЕ ЕЗЛ із стандартними ЛЕ ТТЛ, ТТЛШ і І²Л здійснюється в основному за допомогою мікросхем узгоджувачів, які за своєю будовою є значно складнішими від узгоджувачів інших типів.

Мікросхеми К500ПУ124 і К500ПУ125 – це узгоджувачі типу ТТЛ → ЕЗЛ і ЕЗЛ → ТТЛ відповідно, а мікросхема К1800ВА4 - типу ТТЛ ↔ ЕЗЛ,

2.9 Завадостійкість цифрових пристроїв

До найважливіших проблем схемотехніки цифрових елементів належать питання захисту цифрових сигналів і електричних кіл від внутрішніх і зовнішніх завад та шумів. Весь процес проектування цифрової системи може бути зведений нанівець, якщо на етапах синтезу не будуть вжиті відповідні схемотехнічні заходи щодо забезпечення високої завадостійкості та електромагнітної сумісності окремих вузлів в реальних умовах її експлуатації. Актуальність цієї проблеми зумовлена, насамперед, підвищенням швидкодії цифрових мікросхем при зниженні амплітуд робочих сигналів, і, як наслідок, підвищення рівня зовнішніх завад. Ці фактори відповідно створюють передумови неминучої появи у будь-якій електронній схемі додаткових паразитних параметрів ємнісного або індуктивного характеру, які в реальних умовах можуть погіршити швидкодію та завадостійкість, а іноді навіть призвести до повної втрати працездатності всієї системи. З огляду на те, що цифрові елементи можуть бути як джерелами, так і приймачами завад і шумів, проблему захисту цифрових елементів (у тому числі цифрових сигналів, електричних кіл і та ін.) можна розділити на дві групи:

- захист від внутрішніх завад і шумів, які можуть виникати в окремих вузлах;
- захист від зовнішніх завад і шумів, що мають місце при передачі цифрових сигналів по шинах ліній зв'язку між окремими вузлами і на відстань між системами.

Основними джерелами виникнення завад шумів у електричних колах є кола живлення і заземлення, а також вхідні й вихідні кола сигналу. Внутрішній власний шум (окрім теплового шуму) елемента можна зменшити за рахунок правильного розміщення елементів на платі, фільтрації, або екрануванням. Механізм передання по внутрішніх колах схеми небажаних сигналів може бути *електростатичний* (внаслідок ємнісного зв'язку між провідниками), *магнітний* (через магнітний зв'язок, що утворений із замкнутих контурів схеми) і *електромагнітний* (через електромагнітні хвилі, що утворені провідниками - "антенами").

Струми завад і шумів поширюються в електричних колах внаслідок

спадів напруг, що виникають на їх шляху в колах живлення і заземлення. Основною причиною появи завад у цифрових схемах є процеси перемикання логічних рівнів напруг цифрових сигналів, які внаслідок швидкої зміни $\Delta U / \Delta t$ (В/нс) породжують цілий спектр частот в області від нуля до сотень мегагерц. Верхня межа цього спектра, безперечно, залежить від швидкодії ЛЕ, тобто від t_{ze} конкретного типу мікросхеми.

У ЛЕ ТТЛ і КМОН, зокрема, що мають двотактну вихідну схему, При перемиканнях виникає ситуація, коли обидва транзистори протягом короткого часу знаходяться у відкритих станах. Цей момент викликає короткочасне коротке замикання джерела живлення, що супроводжується хоч короткочасним, але значним струмом, що прямує від $+E$ до "землі", який у деяких ЛЕ може досягати сотень міліампер. Цим струмом при наявності паразитних індуктивностей провідників утворюються від'ємний викид струму з боку $+E$ і додатний - з боку загальної шини ("землі"), які стають небажаними для інших ЛЕ. Наприклад, додатний викид напруги з виходу ЛЕ, пройшовши на вхід наступного ЛЕ, може бути ним сприйнятий як лог.1. Якщо для останнього це активний сигнал, він переведе його у інший логічний стан, що рівнозначно збою, і, отже, до порушення всієї системи.

Поряд з максимальним використанням на платі провідної площі під шину "землі", якою можна значно зменшити паразитні індуктивності й опір, традиційним способом захисту розглянутого роду завад є застосування шунтуючих конденсаторів різних типів. Керамічні конденсатори під'єднують до виводів живлення кожної мікросхеми, причому для ЛЕ малої і середньої швидкодії - ємністю 0,1 мкФ, а для швидкодіючих - ємністю 100...1000 пФ. Оксидні або танталові конденсатори як стабілізуючі і розв'язуючі ємністю ≥ 1 мкФ розміщують на шинах джерела живлення у кількох місцях плати.

Найважче боротися із завадами, що викликані ємнісним навантаженням, під яким слід розуміти сумарну ємність, що утворена монтажем провідників і вхідною ємністю вхідного кола наступної мікросхеми. Це ємність, яка може досягати десятків пікофарад, в момент дуже короткочасного перемикання утворює досить значний струм, величину викиду якого можна підрахувати за формулою $I_c = C \cdot \Delta U / \Delta t$, де ΔU - перепад напруги перемикання; Δt - час перемикання. Ситуація ускладнюється особливо тоді, коли сигнали потрібно приймати або передавати за межі плати, а також тоді, коли сама плата має велику площу. У цих випадках найкращим захистом є зменшення паразитних індуктивностей, що утворені довгими провідниками, за допомогою масивної спільної шини "землі" і мінімально можливого монтажу.

Можливість виникнення завад зростає при передачі цифрових сигналів на відстань, бо крім внутрішніх завад і шумів тут виникають ще й зовнішні. Вони стають більш інтенсивними внаслідок збільшення погонної ємності провідників, кабелю або дротів та наявності роз'ємів. У цих випадках

інтенсивність завад визначається як амплітудою цифрового сигналу і ємністю навантаження, так і типом лінії зв'язку. Лінія передачі може бути узгоджена або неузгоджена, симетрична або несиметрична, одно- або двонапрявлена.

У випадку неузгодженої лінії зв'язку (коли опір навантаження не дорівнює хвильовому, тобто характеристичному, опорі лінії) запобігти викидів струму на ньому можна за умови, якщо довжина лінії $l \leq \frac{0.5 \cdot t_{\phi}}{t_{nz}}$,

де t_{ϕ} - тривалість фронту імпульсу; t_{nz} - погонна затримка поширення сигналу в лінії зв'язку, с/м. Лінія зв'язку тут може бути несиметричною, наприклад коаксіальний кабель, який за рахунок повного екранування забезпечує високу завадостійкість. Парою ШПП, що розраховані на роботу з коаксіальним кабелем, в інтерфейсні мікросхеми ТТЛ 559ИП4 (передавач) з відкритим емітером і 589ИП5 або 559ИП7 (приймач) із входом на основі тригера Шмітта. У цьому випадку на виході передавача і на вході приймача досить під'єднати резистори, однакові за характеристичним опором кабеля Z . (див.рис.2.15)

Вищі завадостійкість і швидкодію передачі цифрових сигналів на значну відстань (до кількох кілометрів) можна досягнути при застосуванні симетричної лінії зв'язку, якою

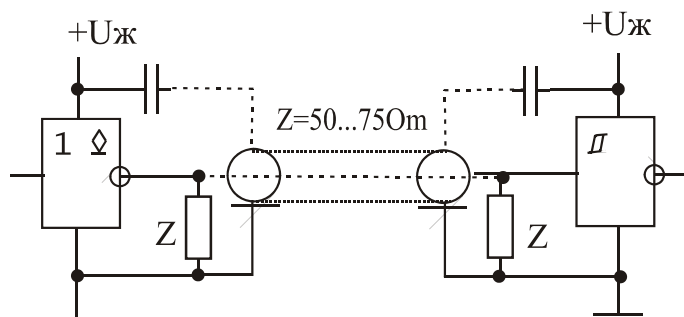


Рис. 2.15 Передача по коаксіальній лінії

є звита пара проводів однакових розмірів і електричних параметрів. Для симетричної лінії зв'язку передавач і приймач повинні мати відповідно диференційні вихід і вхід. За цих умов забезпечується високий ступінь придушення

синфазних завад, що може досягати >70 дБ (3000:1) у діапазоні $0 \dots 1000$ кГц, а також хороше відновлення логічних рівнів цифрових сигналів, які завжди спотворюють свою форму на лінії передачі.

З метою послаблення високочастотних шумів (≥ 100 МГц) на друкованій платі з боку живлення доцільно застосувати П-подібний LC-фільтр, що утворений з шунтуючих керамічних конденсаторів ємністю 1000 пФ і феритового кільця з двома дротяними витками. Енергія електромагнітної хвилі, що проходить крізь ферит зменшується частково перетворюючись в теплову. Крім того, до кожного ЛЕ передавача і приймача цифрових сигналів необхідно під'єднати керамічні конденсатори ємністю $0,1$ мкФ.

Виходи цифрових елементів, які передають сигнали за межі плати чи системи, слід буферувати ЛЕ або ШПП з метою уникнення явища інтерференції завади і корисного сигналу, а також для придушення викидів струму на неузгодженій лінії зв'язку. При необхідності передачі цифрового сигналу на велику відстань, причому на неузгоджену лінію (кабель), вихідні провідники плати системи слід пропускати через ферритове кільце. Це

збільшує послідовну індуктивність кабеля в цілому і, отже, підвищує повний опір для синфазного сигналу на високій частоті.

Якщо синтез КП відбувається тільки на рівні функціонально-логічного проектування, тобто з врахуванням тільки законів булевої алгебри, може скластися думка, що всі ЛЕ у КП діють без затримок і на їх виходах сигнали з'являються у той самий час, що й вхідні сигнали. Більше того, сигнали на виходах зберігаються доти, поки є сигнали на входах. Таке проектування, безумовно, носить формальний характер і не може вважатися остаточним для побудови працездатного пристрою.

При проектуванні реальних цифрових пристроїв необхідно враховувати, що кожний ЛЕ або функціональний вузол КП має власну скінченну затримку сигналу і тому сигнали на виходах схеми будуть з'являтися тільки через певний (для кожного елемента свій) інтервал часу після подачі вхідних сигналів. Це, безперечно; негативно впливає на функціонування всього пристрою, бо затримка у колах ЛЕ не тільки зменшує швидкодію, а й створює хибні сигнали. Отже, одночасна поява двох сигналів на вході ЛЕ або КП, що мають внаслідок однакових причин неоднакові часові затримки поширення, утворюють на цьому вході "логічне змагання" або "гонку". Таке негативне явище стає причиною появи на виході даного ЛЕ або КП нерегламентованих сигналів, які важко навіть зареєструвати.

Хоча ці сигнали, як правило, короточасні (всього кілька наносекунд), поява їх на виході ЛЕ або КП може призвести до зміни стану наступного ЛЕ, наприклад запам'ятовувача (типу тригера), який зафіксує новий стан, що може бути небажаним для всього цифрового пристрою чи системи. Короточасні хибні сигнали-викиди, причиною виникнення яких є "гонки", називають "голками".

Залежно від ситуації на вході КП, яка спричиняє виникнення на його виході "голки", розрізняють статичний та динамічний ризики збою КП.

Статичний ризик збою КП виникає тоді, коли замість постійного рівня (0 або 1), який має бути за логікою роботи на виході схеми, на останньому відбуваються короточасні зміни, тобто з'являється "голка". Причиною появи небажаних змін на виході схеми є часова різниця ("гонка") логічних переходів $0 \rightarrow 1$ або $1 \rightarrow 0$ одного вхідного сигналу щодо другого.

Залежно від типу ЛЕ завада типу "голки", як показано на рис. 2.16 а-б, може бути відповідно нульовою типу (1-0-1) або одиничною типу (0-1-0). Позбутися цієї завади можна, очевидно, введенням у "незатримане" коло, у даному випадку у коло сигналу X_1 , штучної лінії затримки, яку легко виконати з ланцюга потрібного числа буферів-повторювачів або парного числа інверторів.

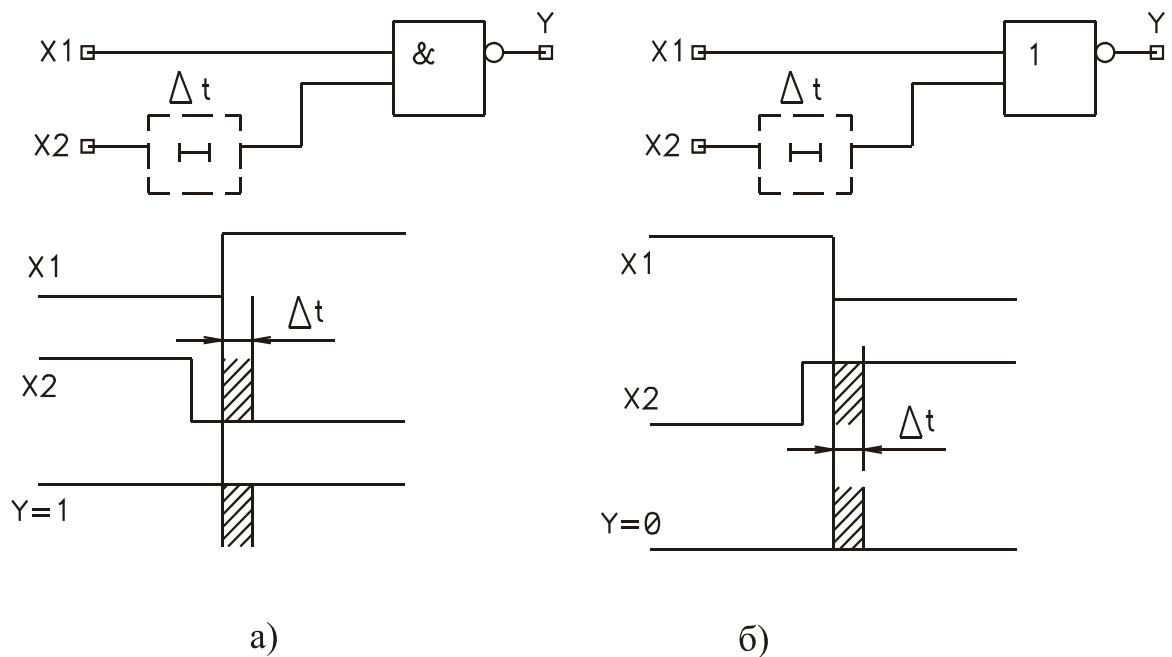


Рис. 2.16 Виникнення збоїв при роботі пристрою

Як показує практика, при розробці нескладних одноплатних цифрових пристроїв (до 10 корпусів МІС та СІС) в діапазоні частот ≤ 15 МГц особливих труднощів не виникає, і немає потреби проводити спеціальні розрахунки електромагнітної сумісності пристроїв. У випадку розробки більш високочастотних друкованих плат слід застосовувати мікросмужкову технологію і враховувати напрям руху зворотніх струмів по шині „землі”, причому ширина доріжки повинна вибиратися за врахування частоти робочого сигналу, робочого струму та хвилевого опору виходів та входів ЛЕ. Те саме особливою мірою стосується плат аналогово-цифрових перетворювачів (АЦП), де неправильне проектування топології друкованих провідників призводить до втрати розрядності А/Ц перетворення, оскільки молодші розряди одержаного внаслідок такого перетворення коду „шумлять”. Слід зауважити, що вирішення проблем завадозахищеності потребує значного досвіду та інтуїції та вимагає здатності передбачити всі можливі джерела негативних впливів. Детальніше про особливості проектування та захисту від шумів і завад можна дізнатися в [2]

2.10 Імпульсні схеми на цифрових елементах

На базі ЛЕ різних технологій можна будувати практично всі відомі на сьогодні (малої та середньої потужності) пристрої імпульсної та цифрової техніки. Серед найбільш використовуваних розглянемо лише такі цифрові пристрої, які за такими властивостями, як простота реалізації, економічність, надійність тощо, ефективно будувати на ЛЕ.

2.10.1 Формувачі

Основне призначення формувачів імпульсних сигналів - перетворення вхідних сигналів довільної форми у цифрові імпульси.

Схеми електрично-керованих ключів, або схеми збігу на ЛЕ належать до найпростіших формувачів імпульсних сигналів.

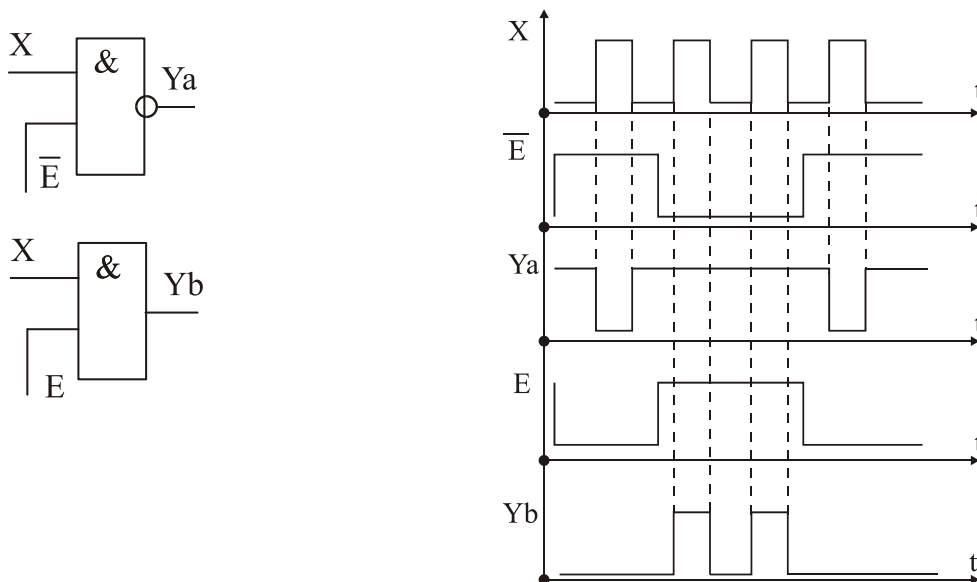


Рис. 2.17 Схеми керування ключами та часові діаграми їх роботи

На рис. 2.17 показано схеми для двох випадків дозволяючих рівнів E та \bar{E} , а також часові діаграми для імпульсних сигналів при однакових тривалостях E і \bar{E} .

Подібні схеми застосовують там, де потрібно за заданим сигналом дозволу E, здійснити передачу по прямому каналу X потенціальних або імпульсних сигналів..

Формувачі коротких імпульсів (ФКІ) можна будувати за допомогою самих лише ЛЕ. Один із варіантів такого формувача і часові діаграми його роботи зображені на рис.2.18.

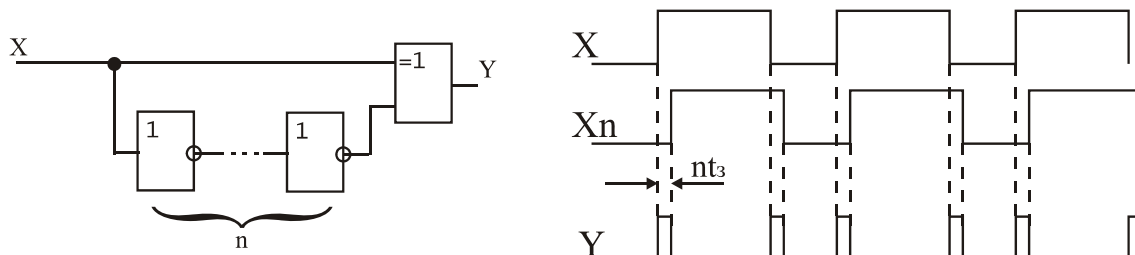


Рис. 2.18.

Дана схема формувача реалізує логічний вираз $Y = X \oplus X$, який в випадку ідеальних ЛЕ завжди дорівнює нулю незалежно від того, що є на вході. Проте для реальних ЛЕ логічні операції завжди виконуються з певним часом затримки t_3 , що використовується для формування коротких імпульсів. Оскільки час затримки t_3 одного ЛЕ відомий, ввімкнення до одного з входів схеми „виключного АБО”, парного числа інверторів дає змогу сформувати на виході імпульс тривалості $t_i = n \cdot t_3$.

В якості лінії затримки можна використовувати також інтегруюче RC-коло, а замість ЛЕ в “виключне АБО” – 2І або 2АБО-НЕ. При цьому вихідні сигнали таких схем будуть співпадати з фронтом або зрізом вхідного сигналу X, а тривалість їх значно зросте, у порівнянні із наносекундними "голками" пристрою рис.2.18.

У RC-схемах вихідний імпульс формується за рахунок поступового наростання фронту та поступового спадання вхідного імпульсу після дії інтегруючого RC-кола. Схема збігу закінчує формувати зріз вихідного імпульсу в момент часу, коли напруга U_c розряду конденсатора С досягне порогового рівня U_n перемикання ЛЕ при переході з U^1 в U^0 (у ТТЛ $U_n \cong 2\text{В}$). Тривалість t_i отриманого імпульсу залежить від сталої часу RC-кола ($\tau = RC$) і величини U_n , яка визначається як розкидом параметрів ЛЕ, так і рівнем завади. Тому значну точність t_i вихідного імпульсу в таких формувачах досягнути важко.

Щоб не порушити номінальних параметрів ЛЕ у випадку ТТЛ, допускаються такі значення елементів RC-ланки: $R \leq 250 \text{ Ом}$, $C = 1 \dots 3 \text{ нФ}$. Якщо $R = 200 \text{ Ом}$, $C = 1 \text{ нФ}$, то тривалість одержаного імпульсу t_i визначають, як

$$t_i = RC \cdot \ln\left(\frac{U^1}{U_n}\right) = 200 \cdot 10^{-9} \cdot \ln\left(\frac{5}{2}\right) = 2 \cdot 10^{-7} \cdot 0.92 \cong 0.2 \text{ мкс.}$$

Ще простіше побудувати аналогічні формувачі коротких імпульсів з допомогою диференціюючого CR-кола, діода та інвертора. На рис. 2.19 зображено відповідно схему формувача при присутності чи відсутності (пунктирні лінії) напруги $E_{зм}$. Тривалість отриманого на виході імпульсного сигналу в обох випадках залежить в основному від сталої часу $\tau = CR$.

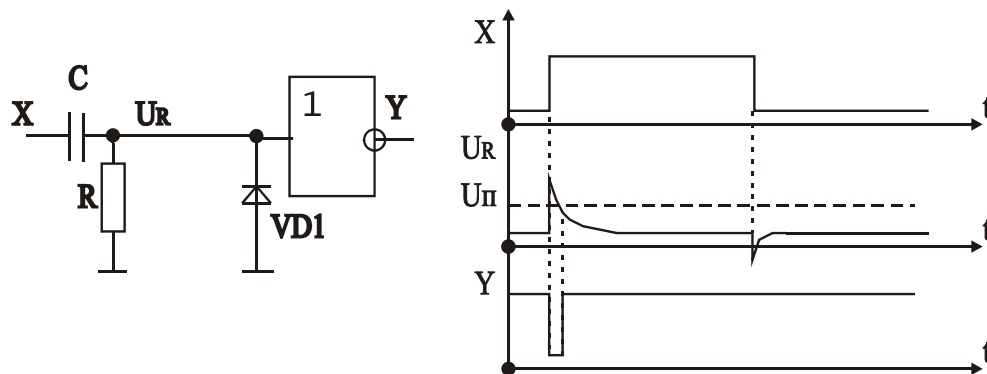


Рис. 2.19 ФКІ на основі CR-кола та його часові діаграми

Опір R слід вибирати з умови забезпечення нормальної роботи ЛЕ - з урахуванням спаду напруги U_R за рахунок струму, що витікає із вхідного кола ЛЕ. Напруга U_R не повинна перевищувати допустимої напруги U_R , і тому для ТТЛ опір R слід вибирати в межах 100...500 Ом.

Для детального ознайомлення з роботою вищенаведених схем, доцільно дослідити їх за допомогою програмного симулятора MicroCAP.

2.10.2 Генератори

Генератори імпульсних сигналів (ГІС) займають визначальне місце у сучасних цифрових та мікропроцесорних пристроях. Це задаючі генератори тактової частоти, стробуючі або строб-генератори (з періодичною установкою початкової фази), синхронізуючі генератори, генератори серії з п-прямокутних імпульсів, генератори поодиноких імпульсів (очікувальний генератор або обдновібратор), тощо.

Одновібратор, або очікувальний генератор, переходить із стійкого стану рівноваги у квазістійкий (тобто у другий тимчасовий стійкий стан) тільки після подачі на його вхід запускаючого короткочасного імпульсу. Тривалість сформованого на виході сигналу залежить від величини сталої часу часозадаючого кола, тобто добутку RC і не залежить від тривалості запускаючого імпульсу, за умови якщо запускаючий імпульс коротший за сформований.

На рис. 2.20 показано схему одновібратора та часові діаграми напруг.

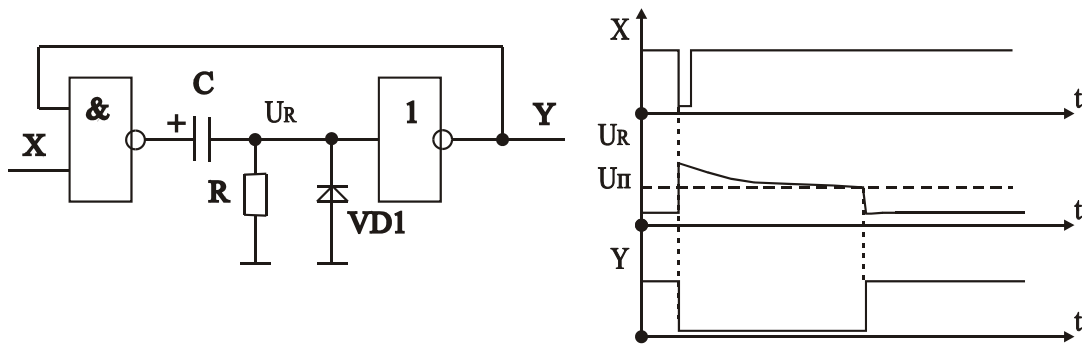
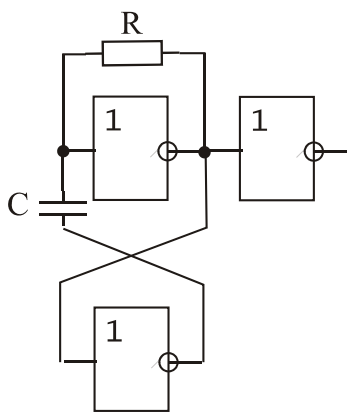


Рис. 2.20 Схема одновібратора та його часові характеристики

Схема має два ЛЕ-2І-НЕ та інвертор НЕ. Останній забезпечує додатний зворотний зв'язок і служить як буферний підсилювач щодо навантаження. У початковому стані $X=1$ і тому на вході інвертора напруга U_R низького рівня. При цьому конденсатор C не заряджений. Як тільки $X=0$, виникає додатний перепад напруги на виході ЛЕ, який у міру заряду конденсатора приводить до зменшення U_R за експоненціальним законом. При досягненні U_R рівня порогової напруги U_n Інвертора на виході $Y=1$. Це викличе на виході 2І-НЕ низький (нульовий) рівень напруги, який швидко розрядить конденсатор C через відкритий діод VD і вихідний транзистор ЛЕ 2І-НЕ. Отже, одновібратор повернеться у свій початковий стійкий стан. Тривалість сформованого імпульсу визначається величиною RC як $t_f \approx RC$ при обмеженому виборі номіналу R для випадку ТТЛ - межах 100...500 Ом. Одновібратори не забезпечують високої точності формування часових імпульсів, на що треба зважати при проектуванні. Надмірне зловживання одновібраторами вважається „поганим смаком” проектувальника.

Мультивібратор - це автоколивальний генератор прямокутних імпульсів заданої тривалості, частоти, амплітуди та полярності. Принцип побудови мультивібратора на ЛЕ полягає в тому, що його робочими ЛЕ є два інвертори, які ввімкнені у коло перехресного додатного зворотного зв'язку (вихід першого з'єднаний із входом другого, а вихід другого - із входом першого). Ці інвертори утворюють схему, що здатна самозбуджуватись, тобто *регенерувати*. За рахунок ввімкнення часозадаючих ланок в колі прямого та зворотного зв'язку поряд з регенеративними процесами у схемі



відбуваються також і релаксаційні процеси. За таких умов на виході мультивібратора виникають релаксаційні коливання певної амплітуди та частоти.

На рис. 2.21 зображена схема мультивібратора на двох інверторах. Регенеративний процес у схемі виникає внаслідок неперервного перезаряду конденсатора C через резистор R . Якщо, наприклад, $Y=1$, то за рахунок зворотного зв'язку $\bar{Y}=0$ і конденсатор C буде

заряджатись (або перезаряджатись) через резистор R до напруги високого рівня.

Як тільки напруга на вході верхнього інвертора досягне порогового рівня спрацювання (для ДЕ ТТЛ $U_n=1,3...1,5$ В), стан цього інвертора зміниться на протилежний ($Y=0$), що зразу приведе до стану $\bar{Y}=1$ на виході нижнього інвертора. Тепер до верхньої обкладки конденсатора через резистор R буде прикладено низький, а до нижньої - високий рівень напруги. Отже, конденсатор C почне перезаряджатися і при досягненні порогової напруги U_n верхній інвертор знову змінить свій стан. Таким чином, релаксаційний процес протягом одного періоду колювання регенераторно змінює свій напрям два рази.

Оптимальний режим роботи мультівібратора забезпечується при виборі R для ТТЛ в межах 200...470 Ом. При цьому період генерованих імпульсів визначається як $T=3RC$. Щоб поліпшити форму, вихідних імпульсів мультівібратора, до його виходів (або входів) під'єднують інвертори. Якщо вибрати $R=220$ Ом, $C=1600$ пф, та частота генерації на виході даного мультівібратора буде $f=1$ МГц.

У тих випадках, коли потрібно окремо регулювати тривалість імпульсу і паузи, тобто регулювати коефіцієнт заповнення γ або щільність генерованих імпульсів, замість резистора R можна застосувати регульований двополюсник, який зображено на рис. 2.22. За його допомогою утворюються імпульси типу "меандр" при середньому положенні повзунка потенціометра R2, а при зміні його положення можна задати потрібний коефіцієнт γ для фіксованого періоду вихідних імпульсів.

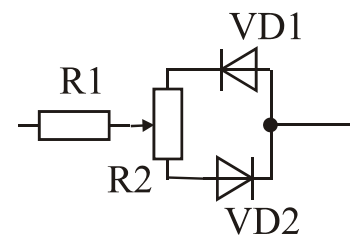


Рис. 2.22 Регулятор щільності

Кварцовий генератор забезпечує високу стабільність частоти $\Delta f/f \cong 10^6...10^{-10}$ за рахунок таких особливих властивостей кварцового резонатора (BQ) як його висока добротність та температурна стабільність на власній резонансній частоті. Такий генератор широко застосовується там, де потрібно мати точне значення частоти протягом тривалого часу.

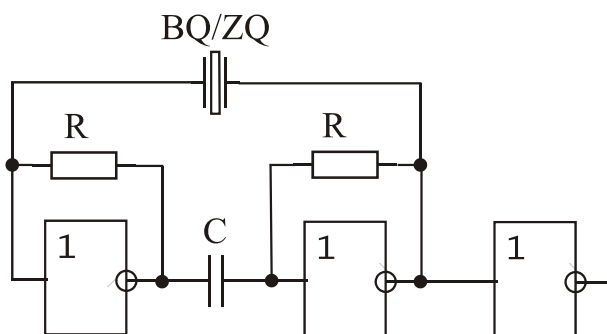


Рис. 2.23 Кварцовий автогенератор

На рис. 2.23 зображено схему кварцового генератора імпульсів на двох інверторах. Її часто застосовують для роботи мікропроцесорів як генератор послідовності тактових імпульсів. Резистори в колах зворотнього зв'язку інвертора утворюють від'ємний зворотній зв'язок, переводячи ЛЕ в лінійний (підсилювальний) режим роботи. Таким чином, кварцовий резонатор

ВQ включений в коло позитивного зворотнього зв'язку (ПЗЗ) двокаскадного лінійного підсилювача. ПЗЗ уможливорює виникнення автоколивань, які *виникають на частоті кварцового резонатора*. Кварцевий резонатор в даному випадку слід розглядати як, фільтр, що пропускає тільки одну частоту (точніше надзвичайно вузьку смугу частот), наприклад 1МГц. Проте одна частота – це гармонічне коливання типу $f(t) = A \sin(\omega t + \varphi)$, яке аж ніяк не входить до визначення цифрового сигналу, тому цю частоту додатково слід перетворити до цифрового сигналу, пропустивши її через вихідний на схемі інвертор, після чого, одержаний прямокутний сигнал можна використовувати. *Подавати сигнал одразу після кварцового резонатора на вхід декількох ЛЕ не слід, оскільки в такому разі сформовані вихідні сигнали будуть несинфазними і це може призвести до помилок в інших частинах схеми.* Частота кварцового ГІС визначається власною частотою кварцового резонатора. Наприклад, кварцовий ГІС на ТТЛ буде забезпечувати на виході Y частоту меандра 1 МГц якщо $R=330$ Ом, а $C=0,003$ мкФ. Дана схема генератора покладена в основу побудови мікросхеми генерації двофазної синхросерії для мікропроцесорів КР580ГФ24. До виводів цих мікросхем досить під'єднати кварцовий резонатор на потрібну частоту та кілька резисторів і конденсаторів.

Генератор лінійно-змінної напруги, найпростіша схема якого зображена на рис. 2.24,а, також належить до релаксаційних ГІС, бо має часозадаюче RC-коло, яке кероване розрядним ключем-інвертором. Функцію останнього у даній схемі виконує ЛЕ з відкритим-колектором.

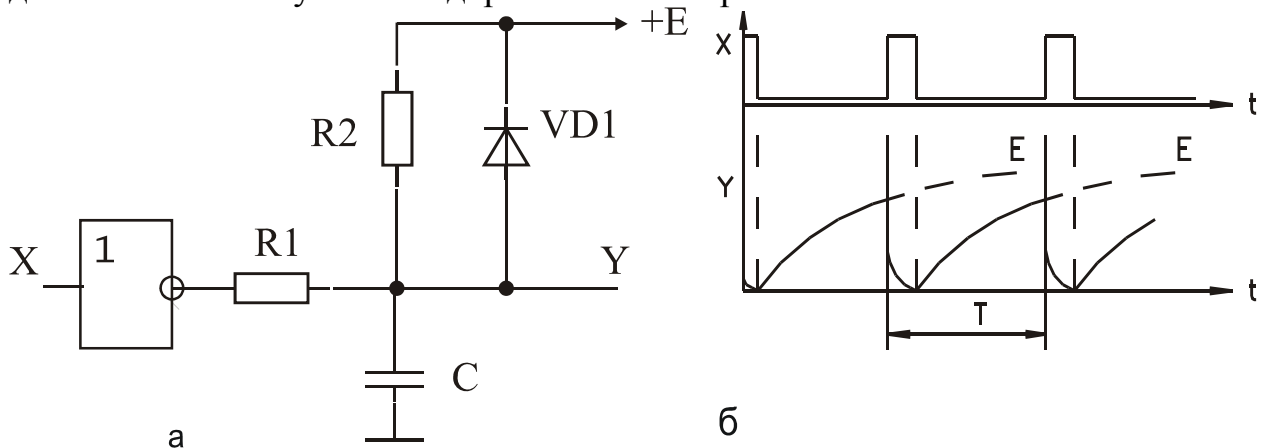


Рис. 2.24 Генератор лінійно-змінної напруги

Як видно з рис. 2.24, при $X=0$ вихідний транзистор ЛЕ закритий і конденсатор С заряджається струмом джерела напруги Е через опір R2. Вихідна напруга припиняє зростати у момент подання вхідного імпульсу, тобто при $X=1$. Отже, період генерування лінійно-змінної напруги задається періодом запускаючих імпульсів. Швидкий спад до нульового рівня вихідної напруги під час дії імпульсу зумовлений значним струмом розряду конденсатора С через відкритий вихідний транзистор ЛЕ. Щоб цей струм (відкритого колектора) не перевищив допустимого значення, його обмежено

опором R1. Якщо потрібно збільшити вихідний струм, застосовують паралельне ввімкнення однакових ЛЕ. Захист вихідного транзистора ЛЕ від пробоя, який може виникнути при відсутності вхідних імпульсів, забезпечує обмежувальний кремнієвий діод VD1.

3. ПРИСТРОЇ ДЛЯ ПЕРЕТВОРЕННЯ ЦИФРОВОЇ ІНФОРМАЦІЇ

На основі досвіду побудови цифрових пристроїв еволюційно виокремились деякі функціональні блоки складені з простих ЛЕ, за допомогою яких здійснюється проектування переважної більшості КП. Такі функціональні блоки почали випускатись у вигляді СІС, зменшуючи тим самим кількість корпусів ІС та знижуючи собівартість готового пристрою. Тому є доцільність розглянути детальніше особливості їх функціонування.

Пристрої комутування цифрових сигналів можуть мати або n інформаційних входів та один вихід для реалізації функції мультиплексування (під'єднаних двох чи більше ліній на одну спільну) або один вхід і m виходів для реалізації оберненої процедури – демультиплексування. Таке функціонування забезпечують відповідно мультиплексор і демультиплексор.

Мікросхеми функціональних КП перетворення та комутування цифрових сигналів, крім інформаційних входів, можуть мати ще додаткові входи керування - ЕІ (дозвіл по входу) і ЕО (дозвіл по виходу). За допомогою входів дозволу ЕІ і ЕО можна реалізувати також нарощення розрядності (збільшення числа входів або виходів) каскадуванням відповідних ІС.

3.1 Шифратори та дешифратори

Шифратор призначений для перетворення цифрової інформації, що подана унітарним n -розрядним кодом, у еквівалентний двійковий m -розрядний код.

Унітарний код це код для якого може існувати тільки один активний стан змінної X_i із множини входних сигналів $\{X_{n-1} \dots X_1, X_0\}$. Отже, шифратор - це перетворювач унітарного коду " 1 з n " у двійковий паралельний код, у якого число виходів m однозначно зв'язане з числом входів n як 2^m . Якщо $n=2^m$, що означає використання повного набору вихідних двійкових комбінацій Y_i , такий шифратор називають повним. Наприклад, шифратор 8-3 є повним, бо він реалізує повний набір можливих комбінацій змінних X_i ($n=8$) у повний вихідний набір Y_i ($m=3$) як $2^3=8$.

У неповному шифраторі число входів n не відповідає числу всіх можливих вихідних комбінацій 2^m , причому завжди $n < 2^m$, що відповідно утворює певне число невикористаних вихідних наборів. Прикладом неповного шифратора, який найчастіше зустрічається на практиці, є шифратор 10-4, що використовується для кодування десяткових чисел у двійково-десятковий код ДДК /8-4-2-1/. Такий шифратор можна застосовувати для кодування десяткових символів (0...9), наприклад, з клавіатури пульта керування. Схематичне позначення шифратора наведено на рис.3.1а, а схема внутрішньої будови на рис.3.1б. Здійснити синтез повного або неповного шифратора можна на базі ЛЕ. Як приклад, здійснимо синтез повного шифратора 8-1.

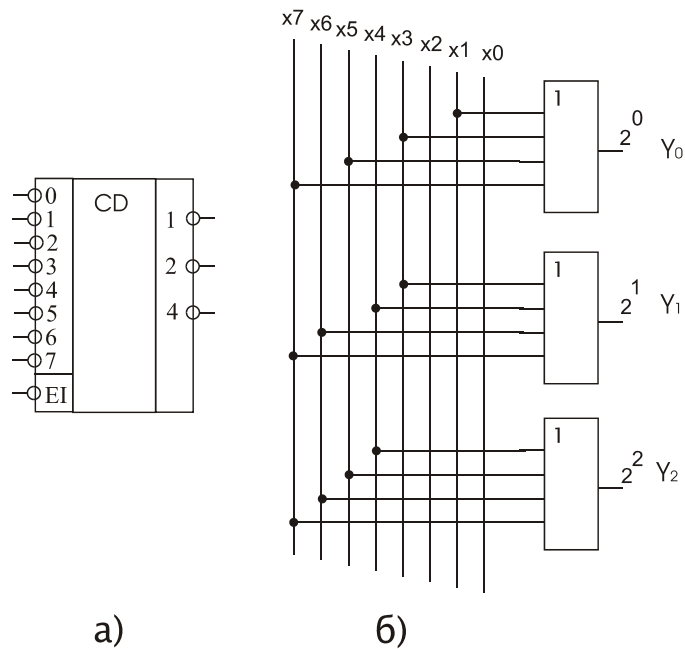


Рис. 3.1 Умовне позначення та внутрішня будова шифратора 8-3

Такий шифратор може бути описаний наступною системою логічних функцій:

$$\begin{cases} y_0 = x_1 \vee x_3 \vee x_5 \vee x_7 = \overline{\overline{x_1 x_3 x_5 x_7}} \\ y_1 = x_2 \vee x_3 \vee x_6 \vee x_7 = \overline{\overline{x_2 x_3 x_6 x_7}} \\ y_2 = x_4 \vee x_5 \vee x_6 \vee x_7 = \overline{\overline{x_4 x_5 x_6 x_7}} \end{cases} \quad (3.1)$$

або таблицею істинності:

Таблиця 3.1 Таблиця істинності шифратора 8-3

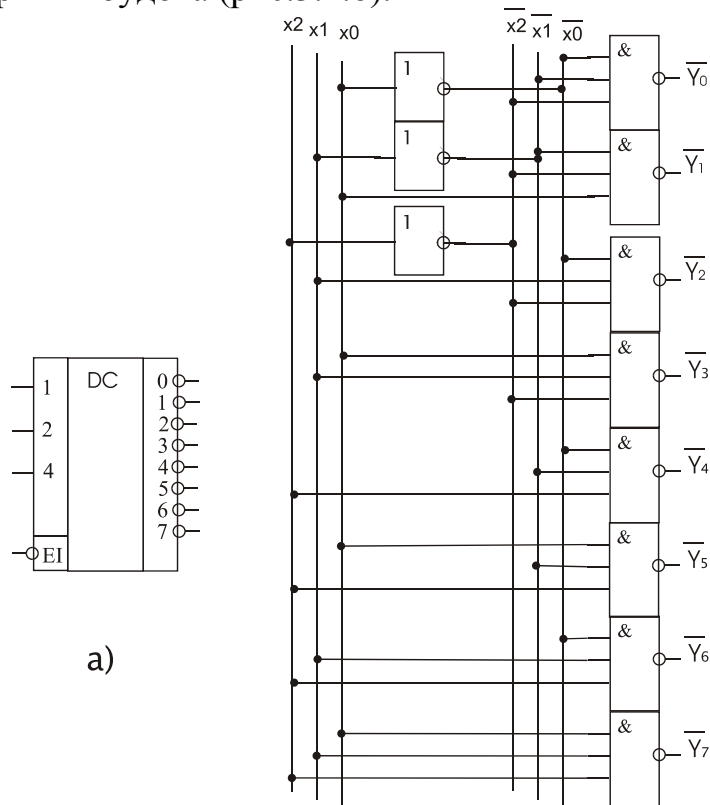
N	X ₀	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	Y ₂	Y ₁	Y ₀
0	1	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	1
2	0	0	1	0	0	0	0	0	0	1	0
3	0	0	0	1	0	0	0	0	0	1	1
4	0	0	0	0	1	0	0	0	1	0	0
5	0	0	0	0	0	1	0	0	1	0	1
6	0	0	0	0	0	0	1	0	1	1	0
7	0	0	0	0	0	0	0	1	1	1	1

Окремі ІС практично зустрічаються рідко, як правило існують мікросхеми, що виконують комбіновані функції, наприклад "шифратор/дешифратор". Крім того, практично використовують шифратори, у яких вихідний код завжди має найбільший номер активного вхідного сигналу $\{X_0..X_n\}$ – так звані пріоритетні шифратори. Приклад роботи пріоритетного шифратора: Вхід $\{x_0=1, x_1=1, x_2=0, x_3=0, x_4=1\}$, тоді вихід $Y=4$.

Дешифратор призначений для розпізнавання (дешифрації) числа, яке подане позиційним n -розрядним двійковим кодом. Найчастіше дешифратор $n - m$ виконує функцію перетворення двійкового кода в унітарний код "1 з m ", тобто виконує функцію, що обернена дії шифратора, і тому для повного дешифратора справедливе співвідношення $m = 2^n$ де m - порядковий номер виходу Y_i дешифратора. Аналітичний опис дешифратора розглянемо на прикладі повного дешифратора 3-8, який описується системою логічних функцій:

$$\left\{ \begin{array}{l} y_0 = \overline{x_2 x_1 x_0} = \overline{x_2 \vee x_1 \vee x_0} \\ y_1 = \overline{x_2 x_1 x_0} = \overline{x_2 \vee x_1 \vee x_0} \\ y_2 = \overline{x_2 x_1 x_0} = \overline{x_2 \vee x_1 \vee x_0} \\ y_3 = \overline{x_2 x_1 x_0} = \overline{x_2 \vee x_1 \vee x_0} \\ y_4 = \overline{x_2 x_1 x_0} = \overline{x_2 \vee x_1 \vee x_0} \\ y_5 = \overline{x_2 x_1 x_0} = \overline{x_2 \vee x_1 \vee x_0} \\ y_6 = \overline{x_2 x_1 x_0} = \overline{x_2 \vee x_1 \vee x_0} \\ y_7 = \overline{x_2 x_1 x_0} = \overline{x_2 \vee x_1 \vee x_0} \end{array} \right. \quad (3.2)$$

Такому опису відповідає схематичне представлення дешифратора на рис. 3.2 а, та його внутрішня будова (рис.3.2.б).



б)

Рис. 3.2 Схематичне позначення та внутрішня структура дешифратора 3-8

У неповного дешифратора число виходів m не відповідає значенню 2^n , причому $m < 2^n$. В інтегральному виконанні зустрічаються як повні (К155ИД3, К155ИД7), так і неповні (К555ИД5, К555ИД10, К561ИД1), а також здвоєні (К155ИД4) дешифратори.

У випадках, коли потрібно побудувати дешифратор на велику кількість виходів на базі дешифраторів з меншим числом виходів, застосовують принцип каскадування. Він полягає у тому, що дані входи дешифраторів розбивають довільним чином на групи, кожна з яких реалізує свою групу логічних функцій. При цьому всі дешифратори повинні бути керованими, тобто мати дозволяючі входи ЕІ. На рис. 3.3 показано двокаскадне з'єднання двох дешифраторів 3-8 для побудови повного дешифратора 4-16, що має входи дозволу \overline{EI} . Каскад 1 працює при активних входах x_0, x_1, x_2 за умови $x_3 = 0$. Як тільки на входах дешифратора з'явиться код $\{x_3 \overline{x_2} \overline{x_1} \overline{x_0}\} = \{1000\}$, верхній каскад закриється, оскільки $x_3 = \overline{EI} = 1$, а відкриється нижній каскад, який через інвертор отримує дозволяючий рівень $x_3 = \overline{EI} = 0$.

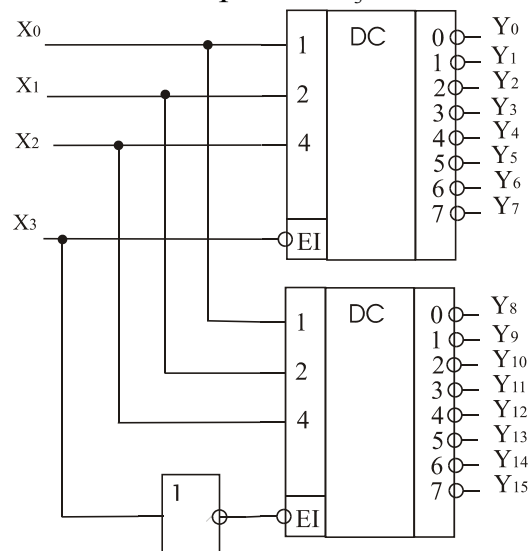


Рис. 3.3 Каскадний синтез дешифратора 4-16

3.2 Мультиплексори та демультимлексори

Це КП, що призначені для комутації цифрових каналів під дією двійкового коду керуючих сигналів.

Мультиплексор (Multiplexor: MUX) призначений для передачі (комутації) сигналів від одного з кількох інформаційних входів X_i (шини даних) на один вихід. Крім інформаційних входів мультиплексор має адресні входи a_j , двійковий код на яких визначає номер активного інформаційного входу, який треба під'єднати до виходу схеми. Отже, мультиплексор має $2^n + n$ входів і один вихід (2^n —число інформаційних входів; n —число адресних входів). Керований мультиплексор має ще один вхід дозволу мультиплексування Е.

Для побудови мультиплектора $2^n \rightarrow 1$ потрібно мати багатовходовий ЛЕ типу І-АБО, що забезпечує передачу з інформаційної шини даних одного з 2^n сигналів, а для керування комутацією – дешифратор.

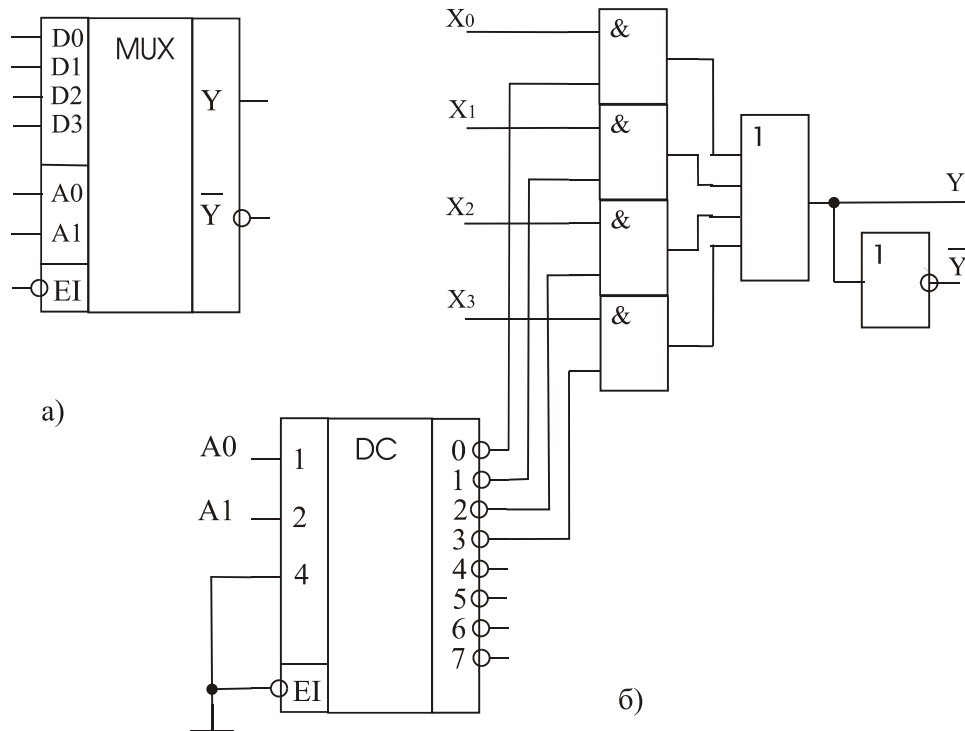


Рис. 3.4 Схематичне позначення та структура мультиплектора 4-1

На рис. 3.4.а показана схема мультиплектора 4-1, що з допомогою $n=2$ адресних сигналів a_1 і a_2 забезпечує вибір одного з $2^2 = 4$ даних X_i . Отже, логічна функція мультиплектора 4-1 має вигляд

$$Y = x_0 \bar{a}_1 \bar{a}_0 \vee x_1 \bar{a}_1 a_0 \vee x_2 a_1 \bar{a}_0 \vee x_3 a_1 a_0 \quad (3.3)$$

Таку логічну залежність неважко абстрагувати для довільної кількості адрес:

$$Y = E \bigvee_{i=0}^{2^n-1} x_i m_i(a) \quad (3.4)$$

де x_i – вхідні інформаційні сигнали, $m_i(a)$ – мінтерми n адресних змінних a_i .

У серіях ЦТ зустрічаються мікросхеми мультиплексорів з різним числом адресних входів, найчастіше $n=2,3,4,\dots$. При конструюванні пристроїв на мультиплексорах, слід мати на увазі, що в переважній більшості своїй мультиплексори комутують цифрові сигнали, хоча існують деякі мікросхеми мультиплексорів (найчастіше КМОН), які дозволяють комутувати й аналогові сигнали. Детальніше про особливості кожної конкретної мікросхеми слід ознайомитись з технічної документації на неї.

Для комутування (мультиплексування) великої кількості сигналів застосовують принцип каскадування. Приклад мультиплексування 8-ми

розрядної шини даних каскадуванням двох мультиплексорів 4-1 показано на рис. 3.5.

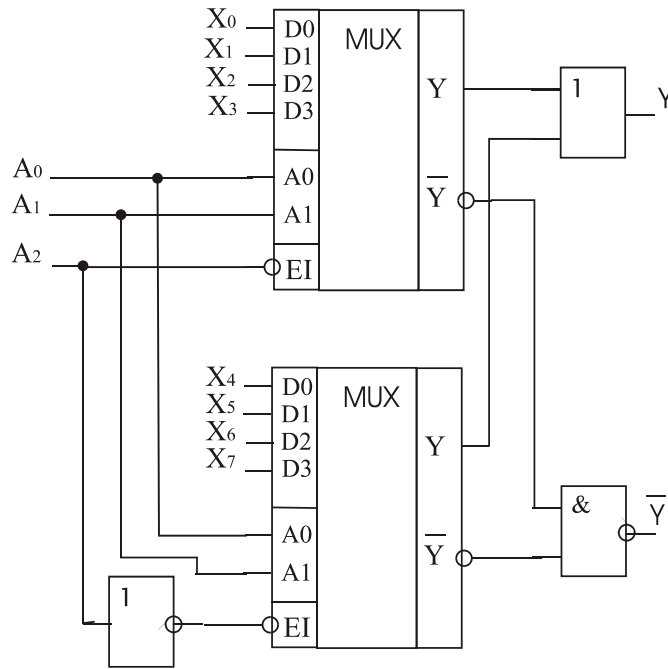


Рис. 3.5 Синтез мультиплексора 8-1 за допомогою мультиплексорів 4-1

Розподіл шини адрес здійснено аналогічно до принципу каскадування дешифраторів (рис.3.3)

За допомогою мультиплексорів можна реалізувати безліч найрізноманітніших цифрових, а у деяких випадках цифро-аналогових схем. Наприклад, на базі мультиплексорів реалізуються: КП багатьох змінних, багатоканальні комутатори цифрових та аналогових сигналів, запам'ятовувальні пристрої, генератори послідовностей двійкових чисел, тощо. Крім згаданих прикладів застосування мультиплексор можна використовувати також і для перетворення паралельного коду, який подано на інформаційні входи, в послідовний, якщо з допомогою лічильника імпульсів потактно змінювати стан адресних входів мультиплексора. Цим способом можна також виконувати послідовне опитування сигналів на 2^n інформаційних входах мультиплексора.

Демультимплексор (Demultiplexor: DMX) призначений для виконання оберненої функції мультиплексора, а саме – передачу (комутацію) сигналу з єдиного інформаційного входу на один з 2^n виходів залежно від коду на n -адресних входах. Демультимплексор можна реалізувати на дешифраторі з n -входами, у якого вхід дозволу Е використовується як інформаційний. Отже, демультимплексор – це розподільувач цифрових сигналів, керований двійковим входом.

Можливий варіант побудови демультимплексора 1-8 зображено на рис. 4.15. Паралельний код, що керує роботою демультимплексора, подають на входи дешифратора, які стають адресними, а виходи останнього під'єднують до входів кон'юнкторів.

У загальному випадку демультимплексор реалізує логічну функцію виду

$$Y_i = X m_i(a). \quad (3.5)$$

З'єднані між собою, другі входи кон'юнкторів утворюють інформаційний вхід X демультимплексора. Отже, на вихід Y_i демультимплексор пропустить вхідний сигнал X тільки через той кон'юнктор, на другому вході якого буде лог. 1, що з'явиться з відповідного виходу дешифратора.

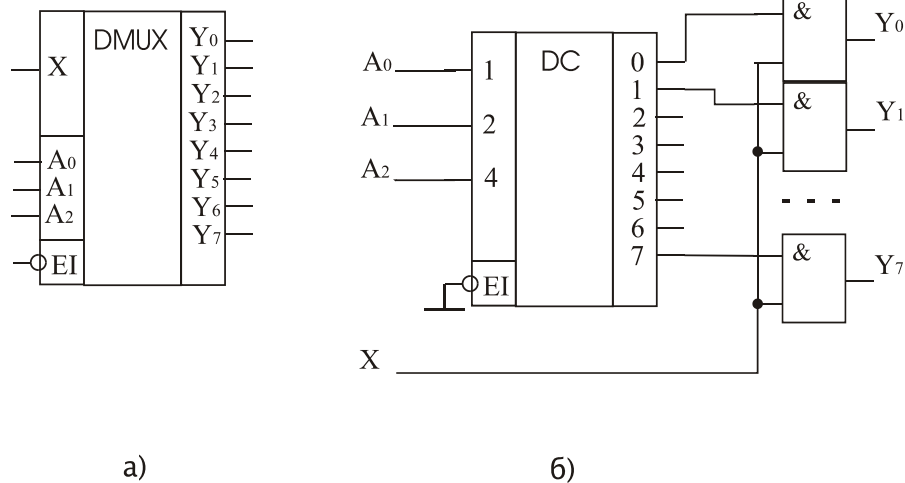


Рис. 3.6 Схематичне позначення та структура демультимплексора 1-8

Деякі мікросхеми ТТЛ, зокрема дешифратори К155ИД3, К155ИД4, К155ИД7, залежно від способу ввімкнення можуть працювати як демультимплексори. Функціональну дуальність мультиплексора і демультимплексора зручно використати для передачі інформаційних двійкових сигналів на відстань, наприклад, по телефонних лініях зв'язку та кабелях. В такій системі зв'язку функцію передавача-перетворювача паралельного коду в послідовний виконує мультиплексор, а функцію приймача-перетворювача послідовного коду у паралельний— демультимплексор. При наявності лінії керування обох перетворювачів забезпечується синхронна робота системи. Перевага такої системи передачі даних на відстань, незважаючи на її низьку швидкодію, полягає в економії затрат, які неминучі при паралельній передачі інформації.

3.3 Синтез комбінаційних пристроїв на дешифраторах

Дешифратори зручно використовувати в тих випадках, коли потрібно сформувати деяку кількість взаємно синхронізованих сигналів, або в загальному реалізувати деяку систему логічних функцій. При цьому дешифратор використовується у вигляді постійного запам'ятовувального пристрою (ПЗП), в якому містяться задані логічні функції. Для того, щоб детальніше зрозуміти вищесказане, розглянемо приклад. Нехай слід розробити цифрових пристрій, на вхід якого надходить циклічна двійкова послідовність $X: (0,1,2,3,4,5,0,1,2,3,4,5... \text{ і т.д.})$. У відповідь на це,

розроблюваний пристрій повинен генерувати три синхронні послідовності, які в часовій формі виглядатимуть так (див. рис.3.7):

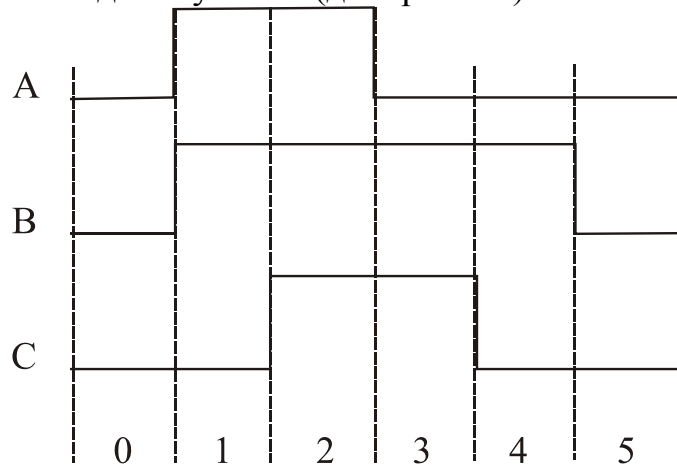


Рис. 3.7 Генеровані цифровим пристроєм сигнали
Система логічних функцій роботи даного пристрою матиме вигляд:

$$\begin{cases} A = \overline{\overline{x_2 x_1 x_0}} \vee \overline{x_2 x_1 x_0} \\ B = \overline{x_2 x_1 x_0} \vee \overline{x_2 x_1 x_0} \vee \overline{x_2 x_1 x_0} \vee \overline{x_2 x_1 x_0} \\ C = \overline{x_2 x_1 x_0} \vee \overline{x_2 x_1 x_0} \end{cases} \quad (3.6)$$

Якщо подивитись на систему логічних функцій дешифратора (3.2), то стає зрозумілим, що там зустрічаються всі можливі терми системи (3.6). Залишається тільки об'єднати необхідні з них за допомогою зовнішніх диз'юнкторів. Схематично це буде мати наступний вигляд (див.рис.3.8):

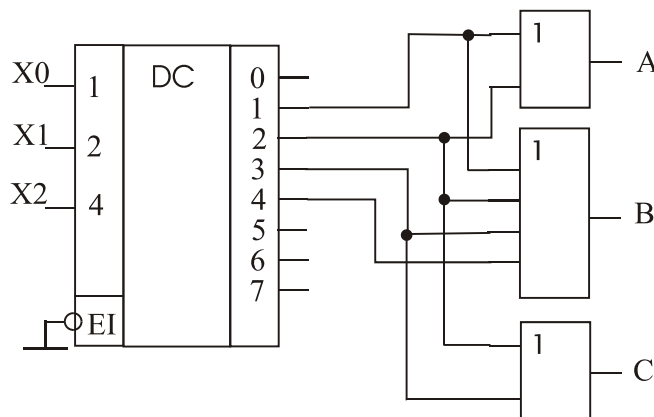


Рис. 3.8 Схема генератора сигналів представлених на рис.3.7

В іншій інтерпретації, робота пристрою може бути представлена так: сигнали X надходять на вхід дешифратора послідовно, тому моменти часу 0,1,2,3,4,5 мають свої відповідні кодові послідовності, які на виході дешифратора перетворюються на сигнал "біжучої" в часі одиниці. Цей сигнал надходить на входи ЛЕ "АБО", які і формують відповідні вихідні сигнали.

3.4 Синтез комбінаційних пристроїв на мультиплексорах

Комбінаційні пристрої на мультиплексорах дозволяють реалізувати довільні логічні функції, і подібно до КП на дешифраторах, забезпечують зменшення кількості корпусів ІС на платі цифрового пристрою. Синтез КП на мультиплексорах полягає в тому, що задавши інформаційним входом лог.0 або лог.1 можна виключити або включити в логічну функцію той чи інший мінтерм, який визначається адресним кодом. Тому мультиплексор зручно застосовувати для реалізації довільної функції n змінних, що подана в УДНФ. Оскільки вихід в мультиплексора один, то й логічну функцію можна реалізувати на ньому одну. Для реалізації системи логічних функцій слід використати каскадування декількох мультиплексорів. Методику синтезу розглянемо на прикладі.

- Нехай слід розробити пристрій, на вхід якого надходить циклічна двійкова послідовність X : (0,1,2,3,4,5,6,7... і т.д.). У відповідь на це, пристрій повинен генерувати кодову послідовність (01110010). Для цього зручно застосувати мультиплексор 8-1, інформаційні входи якого потрібно з'єднати з однойменними розрядами даного коду, як це показано на рис.3.9

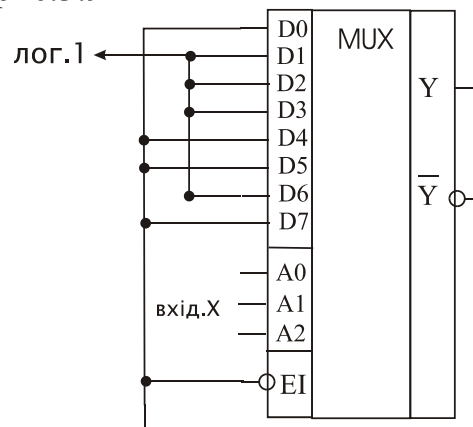


Рис. 3.9 Генератор кодової послідовності на мультиплексорі 8-1

Робота даного пристрою відбувається наступним чином: на адресні входи надходить двійкова послідовність X : (0,1,2,3,4,5,6,7...), що спричиняє послідовне під'єднання до виходу Y входів $D0, D1, D2, \dots$, і як наслідок, на виході з'являється кодова послідовність "запрограмована" за допомогою подачі на ці входи лог.0 та лог.1.

3.5 Перетворювачі кодів

Вони призначені для перетворення одного різновиду цифрового коду в інший. Необхідність у таких перетворювачах для цифрових пристроїв пояснюється тим, що в деяких випадках технічно вигідніше і навіть точніше виконувати певні операції з допомогою інших кодів, а не лише одним двійковим кодом. Різновидів цифрових кодів є дуже багато, і кожен з них має свої переваги при застосуванні в своїх галузях.

Найчастіше виникає необхідність у перетворенні двійкового коду в інший, і навпаки, наприклад, у перетворенні прямого двійкового коду в обернений чи доповняльний код. Перетворювачі кодів необхідні насамперед для технічної реалізації різних арифметичних операцій над двійковими числами, а також для вводу та виводу числової інформації з однієї системи числення в іншу.

На відміну від дешифратора перетворювач кодів може формувати довільні двійкові числа. Кожному вхідному набору (слову) ставиться у відповідність вихідний набір (слово), а не унітарний код, як це має місце у дешифратора. Прикладом може бути перетворювач двійково-десятькового коду (8-4-2-1) у семирозрядний код, що призначений спеціально для керування семисегментним індикатором.

На рис. 4.10 показано спрощену схему ввімкнення двійково-десятькового перетворювача коду в семирозрядний код типу К514ИД1 для цифрового світлодіодного десятикового семисегментного індикатора типу АЛС304А (АЛС324). Мікросхеми серії К514 допускають роботу з вихідними рівнями ЛЕ ТТЛ. Залежно від двійково-десятькового коду (8-4-2-1) на інформаційних входах дешифратора 4-7 та при $E1 = 1$ на відповідних виходах з'являються активні високі рівні, які викликають світіння відповідних сегментів індикатора, тобто зображення десятикових цифр від 0 до 9 згідно з поданою таблицею істинності (табл. 3.1).

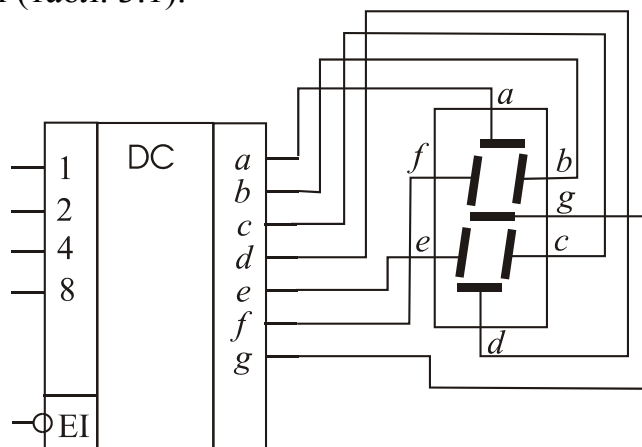


Рис. 3.10 Ввімкнення перетворювача коду 514ИД1 та семисегментного індикатора АЛС304А

Як видно з табл. 3.2, у даного перетворювача-дешифратора 4-7 є шість невикористаних вхідних кодових комбінацій - 1010, 1011, 1100, 1101, 1110, 1111. Для них функції можуть набувати значення 0 або 1. Їх можна, наприклад, використати для індикації малих букв латинського алфавіту, однак для цього слід сконструювати свій власний перетворювач двійкового коду в візуальний код семисегментного індикатора. Тому використання ІС типу К514ИД1 є зручним у випадку потреби відображення тільки цифрової інформації.

Таблиця 3.2 Коди для зображення десяткових цифр від 0 до 9

Десяткова Цифра	ДДК (8-4-2-1)				Семирозрядний код							
	X3	X2	X1	X0	a	b	c	d	e	f	g	
0	0	0	0	0	1	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	0	1
3	0	0	1	1	1	1	1	1	0	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1	1
5	0	1	0	1	1	0	1	1	0	1	1	1
6	0	1	1	0	1	0	1	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1	1

До перетворювачів ДДК у семирозрядний код належать також мікросхеми дешифраторів, що мають також виходи з відкритими колекторами – КІ76ИД2(3), К155ПП5. Перетворювачі на мікросхемах К(М)155ИД8А(Б) і К(М)155ИД9 – це дешифратори ДДК для керування індикаторами, які складаються відповідно з 20 і 27 окремих світлодіодів, а дешифратор К(М)155ИД9 може працювати з напівпровідниковими індикаторами, що мають спільний анод. Все більшого застосування набувають перетворювачі ДДК у коди латинської (К155РЕ22) та російської (К155РЕ21) абетки, а також у код додаткових знаків-символів (К155РЕ23). Ці мікросхеми побудовані за принципом маскових ПЗП об'ємом (256х4)біт.

Розглянуті перетворювачі-дешифратори мають спільну особливість, що об'єднує їх в окремий клас так званих перетворювачів з неваговим (тобто без системи вагових коефіцієнтів у кожному розряді) перетворенням. Другий клас становлять перетворювачі кодів з ваговим перетворенням кодів. Сюди належать, зокрема, перетворювачі ДДК у двійковий код (К155ПР6), і навпаки (К155ПР7), що також побудовані за принципом маскових ПЗП, взаємні перетворювачі двійкового коду і коду Грея тощо.

3.6 Арифметичні пристрої

Розглянуті раніше КП виконують за законами булевої алгебри логічні операції над логічними змінними 0 і 1. Поряд з логічними операціями (заперечення, кон'юнкція, диз'юнкція) у цифровій техніці розглядаються також арифметичні дії над двійковими числами (додавання, віднімання, множення, ділення). Ці операції виконуються за допомогою арифметичних пристроїв, які є складовою частиною, функціональними вузлами мікропроцесорної та обчислювальної техніки. До них належать: суматори різних типів, віднімачі (субтрактори), перемножувачі, подільники, пристрої порівняння і порогові схеми (цифрові компаратори), пристрої виявлення парності (паритету) заданих чисел, арифметично-логічні пристрої тощо.

3.6.1 Арифметичні суматори

Це функціональні вузли, що здійснюють арифметичне додавання чисел. У цифровій техніці підсумовування виконується в основному над двійковими (рідше двійково-десятковими) числами. Додавання багаторозрядного слова за допомогою суматора здійснюється порозрядно з урахуванням переносу в сусідній старший рпзряд. Тому при побудові суматора необхідно враховувати не лише появу переносу в даному розряді, але й можливість одержання аналогічного переносу від сусіднього молодшого розряду.

За принципом побудови і типом використаних елементів розрізняють комбінаційні та накопичуючі суматори. Результати проміжного порозрядного додавання у накопичуючих суматорах зберігаються в елементарних комірках пам'яті, функцію яких виконують тригери. Комбінаційні суматори не мають запам'ятовувачів. У них додавання двійкових чисел здійснюється позиційним кодом одночасно, як і у звичайних КП, результат на виході у комбінаційних суматорах зникає зразу після припинення дії вхідних сигналів. Тому до складу комбінаційних суматорів, як правило, входять вхідні та вихідні регістри, тобто пристрої, що здатні записувати чи перезаписувати проміжний результат підсумовування у послідовному або у паралельному коді.

Для додавання двійкових чисел можуть застосовуватись як одно-, так і багаторозрядні суматори, а сама процедура підсумовування може здійснюватись або послідовно, починаючи з молодшого розряду, або паралельно, коли всі розряди чисел додаються одночасно. Важливою ознакою паралельного суматора є спосіб організації переносу при підсумовуванні. Розрізняють суматори з послідовним, паралельним (наскрізним) та груповим переносом.

Як послідовні, так і паралельні суматори будуються на основі комбінаційного однорозрядного суматора, що складається з напівсуматора.

Напівсуматор – це пристрій (рис. 3.11), що має два входи (для доданків a і b) і два виходи (суми S і переносу P), і призначений для виконання арифметичних дій за правилами, що наведені у табл. 3.3. З

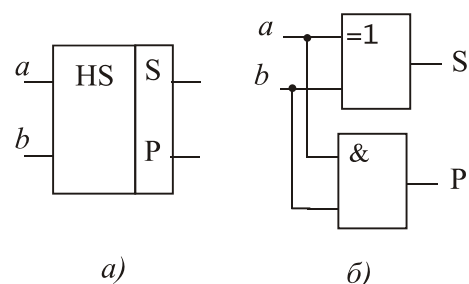


Рис. 3.11 Напівсуматор

таблиці істинності (табл. 4.5) видно, що напівсуматор виконує елементарне додавання двох однорозрядних двійкових чисел та сумовування отриманого результату з переносом у наступний старший розряд. Тому логічна структура напівсуматора має відображати стан обох виходів згідно з виразами:

$$\begin{cases} S = a\bar{b} \vee \bar{a}b = a \oplus b; \\ P = ab \end{cases} \quad (3.7)$$

Відповідно до наведених виразів логічна структура напівсуматора має містити два ЛЕ: суматор за модулем 2 і кон'юнктор, що зображені на рис. 3.11.б.

Таблиця 3.3 Виконання арифметичних дій за правилами напівсуматора

<i>a</i>	<i>b</i>	P	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Однак у логіці роботи напівсуматора не передбачено переносу з сусіднього молодшого розряду, тому напівсуматор може здійснювати додавання тільки у молодшому розряді двійкових чисел. Поява одиниці переносу при додаванні двох розрядів (числа і переносу) дещо змінює правила підсумовування двійкових чисел. Такий однорозрядний суматор потребує ще один (третій) вхід переносу з сусіднього молодшого розряду. Для цього служить так званий повний суматор.

Повний суматор (рис. 3.12) реалізує процедуру додавання двох однорозрядних двійкових чисел з урахуванням переносу з молодшого розряду. Тому він має три входи (a_i, b_i, P_i) і два виходи (S_i і P_{i+1}). Логіка роботи повного суматора наведена у табл. 4.6, де a_i, b_i – доданки двійкових чисел в i -му розряді; P_i, P_{i+1} – переноси, відповідно з молодшого розряду i в сусідній старший розряд $i+1$; S – утворена сума в i -му розряді.

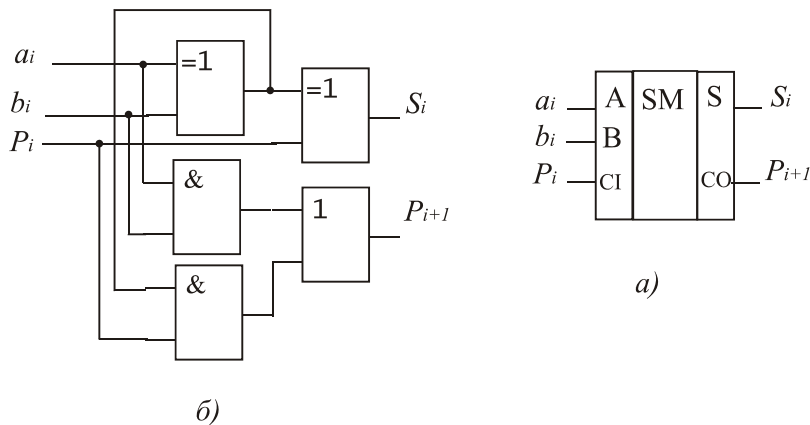


Рис. 3.12 Повний суматор

Таблиця 3.4 Виконання арифметичних дій для повного суматора

a_i	b_i	P_i	P_{i+1}	S_i
0	0	0	0	0
0	1	0	0	1
1	0	0	0	1
1	1	0	1	0
0	0	1	0	1
0	1	1	1	0
1	0	1	1	0
1	1	1	1	1

Згідно з таблицею істинності (табл. 3.4) робота повного суматора двійкових чисел описується такими логічними виразами:

$$S_i = P_i(a_i \oplus b_i) \vee \overline{(a_i \oplus b_i)} = P_i \oplus a_i \oplus b_i;$$

$$P_{i+1} = \overline{P_i} a_i b_i \vee P_i \overline{a_i} b_i \vee P_i a_i \overline{b_i} = a_i b_i \vee P_i (a_i \oplus b_i) = \overline{\overline{a_i b_i} P_i (a_i \oplus b_i)} \quad (3.8)$$

За виразами (3.8) тепер можна побудувати повний суматор, структурна схема та умовне позначення якого зображені на рис. 3.12 (CI –вхід переносу, від *англ. Carry Input*, а CO –вихід переносу, від *англ. Carry Output*).

Для додавання двох n -розрядних двійкових чисел A і B потрібно, очевидно, використати n однорозрядних повних суматорів. При цьому можуть бути два способи підсумовування – послідовне і паралельне. Застосування того чи іншого принципу підсумовування залежить від характеру вводу/виводу чисел та організації переносів багаторозрядного суматора.

3.6.2 Цифрові компаратори

Це арифметичні пристрої, що призначені для порівняння величин двох чисел, що подані у двійковому (двійково-десятковому) коді*. Найпростіший компаратор виявляє лише факт рівності або нерівності двох поданих на його входи n -розрядних чисел (операндів) A і B і формує на виході однобітовий сигнал. рівності (1) або нерівності (0) цих чисел. Слід зауважити відмінність цих пристроїв від аналогових компараторів, які порівнюють рівні напруг на їх входах, і за принципом дії є модифікованими операційними підсилювачами. Тому надалі розглядатимемо тільки цифрові компаратори.

Рівність, зокрема, двох однорозрядних операндів a і b визначається логічною операцією рівнозначності (див.у табл. 1.5 функцію $f9$):

$$Y = \begin{cases} 1 & \text{при } a = b; \\ 0 & \text{при } a \neq b, \end{cases}$$

тобто логічною операцією виняткове АБО-НЕ, яка реалізується суматором-інвертором за модулем 2. Синтез однорозрядного компаратора рівності досить просто здійснюється у довільному базисі.

Порозрядну рівність n -розрядних операндів найпростіше реалізувати за допомогою суматорів-інверторів за модулем 2 і кон'юнкторів, як це показано на рис. 3.13. Такий компаратор рівності порівнює окремі розряди n -розрядних чисел за формулою:

$$Y^{(n)} = y_0 y_1 \dots y_{n-1} = (a_0 \oplus b_0)(a_1 \oplus b_1) \dots (a_{n-1} \oplus b_{n-1}). \quad (3.9)$$

Аналогічний результат одержується й при синтезі компаратора у базисі суматорів за модулем 2 та диз'юнктора. Якщо застосувати закони дуальності, вихідна функція такого компаратора описуватиметься виразом

$$Y = \overline{\overline{y_0} \vee \overline{y_1} \vee \dots \vee \overline{y_{n-1}}} = \overline{(a_0 \oplus b_0) \vee (a_1 \oplus b_1) \vee \dots \vee (a_{n-1} \oplus b_{n-1})}. \quad (3.10)$$

Компаратори порівняння n -розрядних чисел можна будувати за двома принципами – логічним та арифметичним. Перший принцип (логічний) базується на синтезі за таблицею істинності, другий (арифметичний) – на синтезі схеми, що виконує компарування дії віднімання і порівняння їх знаків. На рис.3.14.а показана схема чотирирозрядного компаратора, що реалізує арифметичний принцип порівняння. Даний цифровий компаратор створений на базі чотирирозрядного суматора.

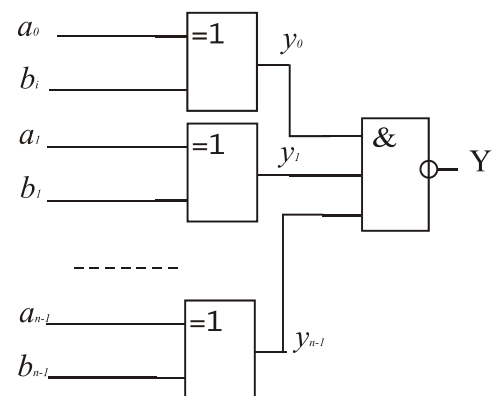


Рис. 3.13 Визначник $A=B$

* Компаратор – з англ. *compare* – порівнювати (схемне позначення функції – CMP).

Функцію дешифратора нуля тут виконує ЛЕ 4АБО. Тільки при рівності всіх розрядів суми $S_0=S_1=S_2=S_3=0$ на виході 4АБО з'явиться 0, а після інвертора 1, яка підтверджує рівність $A=B$. В інших випадках на виході 4АБО одиниця, яка стає дозволом для схеми збігу 2І.

Промисловість випускає цифрові багаторозрядні компаратори як окремі вироби. Це в основному мікросхеми чотирирозрядних компараторів ТТЛШ – К555СП1, 531СП1 і КМОП – 564ИП2, 561ИП2, які мають, вбудовані інвертори для операнда В і додаткові три входи аналізу $I>(A>B)$, $I=(A=B)$, $I<(A<B)$ (див. рис. 3.14.б). Останні призначені для утворення схеми нарощування розрядності операндів, тобто для каскадування послідовним або пірамідальним способом під'єднання однотипних мікросхем.

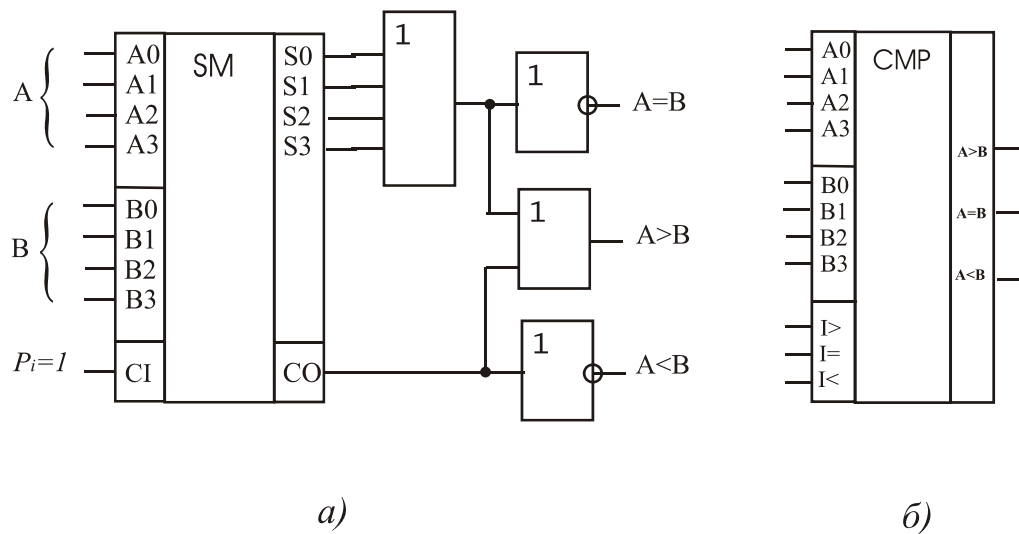


Рис. 3.14 Арифметичний принцип порівняння та позначення цифрового компаратора

3.6.3 Арифметико-логічні пристрої

Для виконання над операндами А і В як арифметичних, так і логічних операцій доцільно застосовувати мікросхеми універсальних арифметико-логічних пристроїв (АЛП), які можна знайти у складі серій ТТЛ і КМОП. Найчастіше вони мають чотирирозрядні входи операндів А і В і придатні для нарощування їх розрядності. Для виконання тих чи інших логічних або арифметичних операцій служать спеціальні керуючі входи. Подаючи на них чотирирозрядний код і задавши тип операції (арифметичної чи логічної), можна виконати за допомогою мікросхеми одну з 16 арифметичних або одну з 16 логічних операцій. Через широкі функціональні можливості мікросхеми АЛП входять до складу мікропроцесорів ЕОМ. У комплекті з АЛП випускають ще мікросхему, яка являє собою тракт групового переносу для виконання прискореного підсумовування багаторозрядних чисел. Самі АЛП при цьому відіграють роль чотирирозрядних груп, оскільки в основному промисловістю випускаються чотирирозрядні АЛП. До таких мікросхем АЛП

належать із серій ТТЛ К155ИП3 (із схемою прискореного переносу К155ИП4), із серій КМОН 564ИП3 (відповідно 564ИП4). Для виконання більш складних за алгоритмом арифметичних та логічних операцій застосовують мікропроцесорні ВІС.

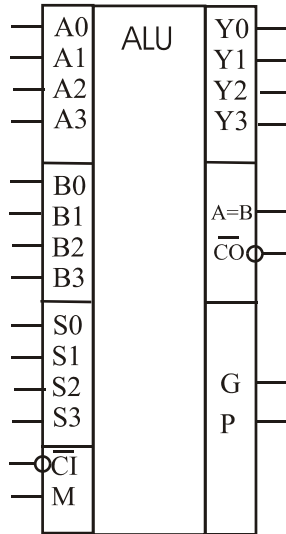


Рис. 3.15 Позначення ІС арифметико-логічного пристрою

Мікросхеми АЛП (К155ИП3 і 564ИП3) – це комбінаційні логічні схеми, що можуть виконувати арифметичні та логічні операції над чотирирозрядними операндами А і В з урахуванням або без урахування переносу P_i . На рис.3.15 зображене умовне схемне позначення мікросхеми АЛП* К564ИП3, яка функціонує згідно з табл. 3.5 (для випадку активних високих рівнів ([34]. АЛП для слів мав чотири пари входів А0-А3 і В0-В3 і чотири виходи $Y_0 - Y_3$, на яких фіксується результат арифметичної (при $M=0$) або логічної (при $M=1$) операції. Потрібну операцію (арифметичну чи логічну) вибирають за допомогою коду на входах $S_0 - S_3$.

Таблиця 3.5 Режими функціонування АЛУ типу К564ИП3

S3	S2	S1	S0	M=1	M=0
0	0	0	0	\bar{A}	$A + \bar{P}_i$
0	0	0	1	$\overline{A \vee B}$	$A \vee B + \bar{P}_i$
0	0	1	0	$\bar{A}B$	$A + \bar{B} + \bar{P}_i$
0	0	1	1	0000	$\bar{P}_i - 1$
0	1	0	0	\overline{AB}	$A + \overline{AB} + \bar{P}_i$
0	1	0	1	\bar{B}	$(A \vee B) + \overline{AB} + \bar{P}_i$
0	1	1	0	$A \oplus B$	$A - B - P_i$
0	1	1	1	$A\bar{B}$	$A\bar{B} - P_i$

* Позначення – АЛУ від англ. Arithmetic and Logic Unit (арифметично-логічний пристрій).

1	0	0	0	$\bar{A} \vee B$	$A + AB + \bar{P}_i$
1	0	0	1	$\bar{A} \oplus \bar{B}$	$A + B + \bar{P}_i$
1	0	1	0	B	$(A \vee \bar{B}) + AB + \bar{P}_i$
1	0	1	1	$A \cdot B$	$AB - P_i$
1	1	0	0	1111	$A + A + \bar{P}_i$
1	1	0	1	$A \vee \bar{B}$	$(A \vee B) + A + \bar{P}_i$
1	1	1	0	$A \vee B$	$(A + \bar{B}) + A - \bar{P}_i$
1	1	1	1	A	$A - \bar{P}_i$

3.6.4 Програмовані логічні матриці

Програмовані логічні матриці (ПЛМ) в загальному розрізняють двох типів – комбінаційні і послідовнісні. Послідовнісні ПЛМ мають властивості комбінаційних + містять вбудовані регістри пам'яті. Останнім часом ця галузь цифрової техніки невинно розвивається, кількість типів послідовнісних ПЛМ зростає мало не кожного року, структура їх щоразу ускладнюється, тому вони є предметом вивчення окремого курсу. Для програмування ПЛМ розроблені спеціальні вхідні мови, на зразок VHDL, AHDL, і т.д., тому ми розглянемо тільки елементарні положення ПЛМ.

Комбінаційна логічна матриця – це пристрій, який може бути представлений мережею взаємоперпендикулярних провідників, в місцях перетину яких знаходяться напівпровідникові елементи – діоди або транзистори, що ввімкнені через легкоплавкі перемички до відповідних провідників матриці.

Під час програмування логічної матриці перемички перепалюють імпульсами струму або зберігають у відповідності до необхідної таблиці, яка відображає необхідний рисунок програмування. Таким чином у комбінаційній ПЛМ може бути реалізована система з m булевих функцій до n вхідних змінних, що містять разом не більше як l терм (складових). В залежності від типу ПЛМ значення m, n, l відрізнятимуться. Структура комбінаційної ПЛМ наведена на рис.3.16.

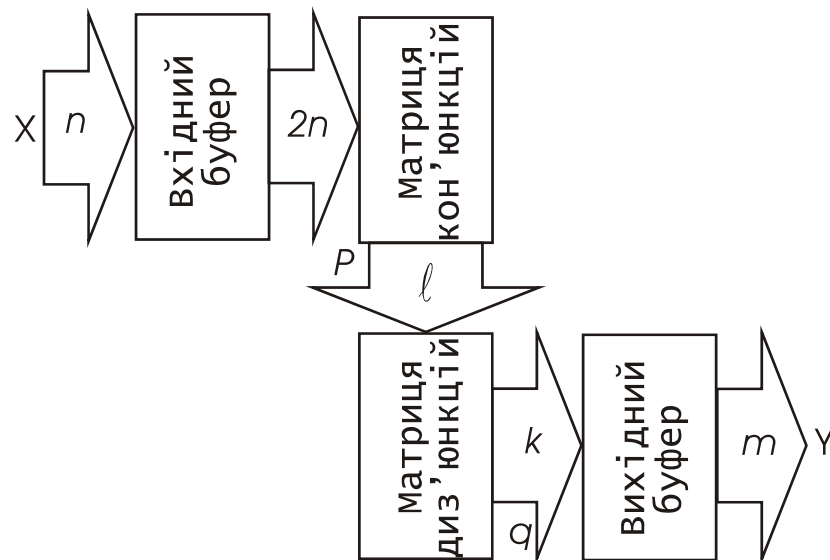


Рис. 3.16 Структура комбінаційної ПЛМ

ПЛМ складається з матриці кон'юнкцій створюють терми виду $x_n \dots x_1 x_0$ та матриці диз'юнкцій які виконують лог. операцію „АБО” над цими термами. Вхідний буфер крім буферування, здійснює ще й іверсію всіх змінних X, тому вихідна шина буфера має $2n$ провідників.

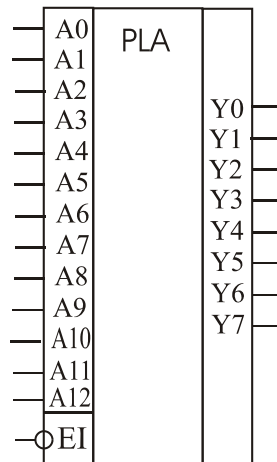


Рис. 3.17 Схематичне позначення ПЛМ

Позначення ПЛМ наведено на рис.3.17. Приклад типів комбінаційних ПЛМ: К556РТ1, К556РТ2. Їх основні параметри $m=8$, $n=16$, $l=48$. На даний час ці ІС слід вважати застарілими, в основному через високе енергоспоживання у порівнянні із сучасними зарубіжними взірцями, тому використовувати їх в сучасних розробках не рекомендовано.

Контрольні запитання по розділу

- Чи може у простого дешифратора бути активними декілька вихідних ліній?
- Будова дешифратора. Основні принципи.
- Будова мультиплексора. Основні принципи.
- Мультиплексори та демультиплексори. Основні відмінності.
- Як здійснюється каскадування дешифраторів?
- В чому полягає синтез комбінаційних пристроїв на основі дешифраторів?
- В чому полягає синтез комбінаційних пристроїв на основі мультиплексорів?
- Принцип дії цифрових компараторів.
- Будова напівсуматора.
- Будова повного суматора.
- Які функції може виконувати арифметико-логічний пристрій?
- Що таке ПЛМ?
- Відмінність між комбінаційними та послідовнісними ПЛМ.

4. ПОСЛІДОВНІСНІ ПРИСТРОЇ

4.1 Особливості функціонування послідовнісних пристроїв

Переважає більшість сучасних цифрових пристроїв є послідовнісними (ПП) або цифровими автоматами з пам'яттю, тобто такими, що складаються з комбінаційної частини (або КП) і елементів пам'яті – запам'ятовувачів інформації. Сама наявність запам'ятовувачів у ПП хоч і ускладнює його роботу, але й значно розширює функціональні можливості ПП. Разом з тим вони є причиною появи додаткової змінної – часу, який потрібно враховувати як при аналізі, так і при синтезі ПП.

Таблиця 4.1 Табличне представлення автомата Мура.

Вхідний набір X	Стан і вихід			
	q_0	q_1	...	q_{k-1}
	y_1	y_0	...	y_j
X_0	q_2	q_0	...	q_i
X_1	q_0	q_1	...	~
...
X_{n-1}	~	q_2	...	q_j

Як бачимо, автомат Мура, який задається табл.4.1, має виходи, які однозначно визначені його станами, і тому вони можуть бути вказані біля вершин графа. Для автомата Мілі, однак, потрібно вказувати пару X_i/Y_j , бо їх виходи залежать як від станів, так і від вхідних наборів.

Порівняно з іншими способами зображення логічного функціонування ПП графічний спосіб є найбільш наочний.

Приклад: Побудувати граф абстрактного автомата Мілі, що заданий сумісною таблицею переходів (табл. 4.2).

Таблиця 4.2 Сумісна таблиця переходів автомата Мілі

$\{X^t\}$	q_0^t	q_1^t	q_2^t	q_3^t
00	$q_1/1$	$q_1/0$	$q_0/0$	$q_3/1$
01	$q_2/0$	$q_1/1$	q_3/\sim	$q_0/0$
10	$q_3/1$	$q_3/0$	$q_0/1$	$q_0/0$
11	q_0/\sim	$q_3/1$	$q_2/1$	$q_2/0$

Розв'язання. Для зручності стани $\{q_i^t\}$ автомата позначимо просто цифрами, що відповідають номеру стану, і розмістимо їх в середині кружечків. Граф таблично заданого автомата показаний на рис. 4.1.

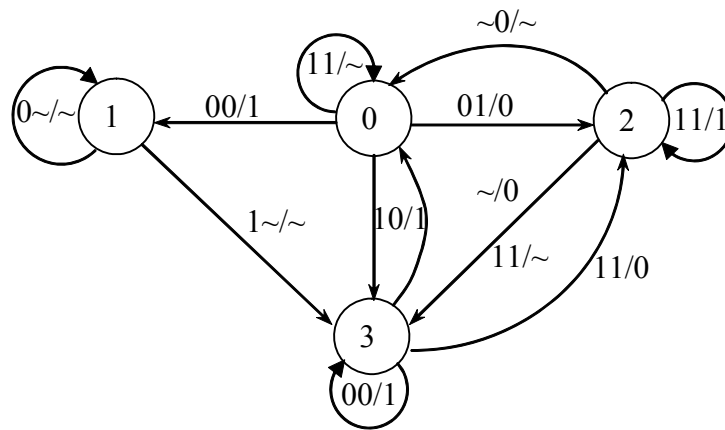


Рис. 4.1 Граф автомата Мілі у відповідності до таблиці 4.2

Зауважимо, що у даному прикладі є кілька однакових переходів станів, які відбуваються під дією різних вихідних сигналів. Це, зокрема, переходи $q_1^t \rightarrow q_1 - 00/0 \vee 01/1$; $q_1^t \rightarrow q_3 - 10/0 \vee 11/1$; $q_2^t \rightarrow q_0 - 00/0 \vee 10/1$; $q_3^t \rightarrow q_0 - 01/0 \vee 10/0$.

Оскільки ці пари вхід/вихід при переході $i \rightarrow j$ мають байдужі розряди (0/1), їх зручніше зобразити так, як це ілюструє граф, а саме: $q_1^t \rightarrow q_1 - 0\sim/\sim$; $q_1^t \rightarrow q_3 - 1\sim/\sim$; $q_2^t \rightarrow q_0 - \sim 0/\sim$; $q_3^t \rightarrow q_0 - \sim\sim/0$.

4.2 Особливості синтезу послідовнісних пристроїв

Всі послідовнісні автомати або пристрої можна поділити на три класи: асинхронні потенціальні автомати, асинхронні імпульсні автомати та синхронні автомати.

До основних функціональних ПП, що виконані, як окремі інтегральні мікросхеми СІС або ВІС, належать регістри, лічильники, тощо.

Структурний синтез ПП зводиться до побудови такої схеми автомата, яка функціонує відповідно до заданих таблиць переходів і виходів автомата. Вихідними даними для структурного синтезу ПП є опис алгоритму його функціонування, згідно з таблицями станів заданий елементарний базис ЛЕ і запам'ятовувачів інформації, а у випадку побудови конструкції ВІС ПП - вимоги до електричних параметрів з урахуванням конструкторсько-топологічних особливостей реалізації вибраного елементарного базису. Процес структурного синтезу ПП можна розкласти на такі етапи:

- 1) оцінка складності алгоритму і при необхідності каскадування ПП;
- 2) формалізований опис алгоритму функціонування пристрою або його каскаду;
- 3) мінімізація станів пристрою і визначення кількості запам'ятовувачів;
- 4) кодування внутрішніх станів пристрою;
- 5) складання таблиць переходів, а також функцій збудження і виходів та мінімізація останніх;
- 6) перетворення мінімізованих функцій збудження і виходів до заданого функціонального базису ЛЕ і запам'ятовувачів;
- 7) побудова структурної логічної схеми ПП;
- 8) тестування.

Критерієм якості економного способу кодування станів є простота схемної реалізації, яка передбачає мінімальне число ЛЕ. Цей спосіб кодування є ефективним особливо для синхронних ПП, бо наявність кіл синхронізації усуває у них такі

недоліки, як змагання (гонки) сигналів, що викликають ризик збою. Економний спосіб кодування полягає у тому, що суміжні кодові комбінації станів автомата відрізняються одна від одної тільки однією цифрою (0 або 1).

Так само, як і у КС, у асинхронних ПП гонки можуть виникати через різні затримки у спрацюванні запам'ятовувачів інформації. Між синтезом тригера на основі ЛЕ і синтезом ПП на основі тригерів є певна різниця, яку треба враховувати у процесі проектування того чи іншого пристрою.

Логіка роботи будь-якого тригера як послідовнісного автомата характеризується таблицями станів - переходів і виходів. Але враховуючи те, що тригер - це автомат Мура, у якого вихідна змінна u^{t+1} повторює значення його внутрішнього стану Q^{t+1} , можна обмежитись таблицею переходів тригера.

На основі таблиці переходів можна отримати функцію переходів тригера, тобто логічну функцію, яка показує зв'язок зміни стану тригера $Q^t \rightarrow Q^{t+1}$ під дією комбінації вхідних сигналів. Якщо під вхідними сигналами X_i^t тригера розуміти як інформаційні, так і керуючі сигнали, функцією переходів тригера є залежність $Q^{t+1} = f(Q^t, X^t)$, за якою, як і за таблицею переходів, можна визначити тип тригера. Як і у КП, функцію переходів тригера потрібно мінімізувати, наприклад, з допомогою карт Карно. Далі у заданому базисі ЛЕ можна будувати структуру тригера. Отже, для синтезу тригера у базисі ЛЕ досить мати мінімізовану функцію переходів тригера.

При синтезі послідовнісних схем на базі тригерів виникає обернена задача - треба визначити, при яких наборах вхідних змінних X_i той чи інший тригер перейде з відомого або заданого стану Q_i^t у наступний стан Q_i^{t+1} . Для цього служать таблиці переходів або матриці переходів тригера, які будуються зв принципом перебору всіх можливих переходів $Q^t \rightarrow Q^{t+1}$ (їх є чотири: $0 \rightarrow 0$, $0 \rightarrow 1$, $1 \rightarrow 0$, $1 \rightarrow 1$). Цим переборам залежно від типу тригера відповідають конкретні комбінації вхідних сигналів, і в тому числі невизначені їх значення, які позначаємо тильдою “~”.

4.3 Тригер – найпростіший запам'ятовувальний пристрій

Спільним елементом всіх ПП є запам'ятовувач бінарної інформації. У ЦТ функцію найпростішого запам'ятовувача, що може зберігати 1 біт двійкової інформації (лог. 0 або лог. 1) виконує *тригер*². Тригер як двостановий** запам'ятовувач інформації характеризується значною функціональною гнучкістю. Поняття "тригер" охоплює безліч пристроїв, які суттєво відрізняються між собою схемним рішенням, функціональними ознаками, способом керування тощо. У цифрових системах радіоелектроніки тригер, крім зберігання інформації, може виконувати також функції перетворювача інформації чи сигналів.

З точки зору теорії автоматів тригер як елементарний скінченний автомат характеризується такими властивостями:

- число вхідних змінних X_i залежить від типу тригера;
- число внутрішніх станів два (0 або 1), що відповідає одній змінній q_i (прийнято позначати як Q);
- число вихідних змінних Y_i одне, значення якого збігається із значенням стану $Q(Y^{t+1} \equiv Q^{t+1})$; отже, тригер – це елементарний автомат Мура.

Тому, коли говорять про стан тригера, то розуміють логічний рівень сигналу на його прямому виході Q /поряд з прямим виходом Q тригер має ще інверсний вихід \bar{Q} /. Якщо стан тригера "1", це означає, що $Q=1, \bar{Q}=0$, якщо "0", відповідно $Q=0, \bar{Q}=1$. Стан тригера може змінюватися під дією різних зовнішніх сигналів. Процес переходу тригера з одного стійкого стану в інший відбувається стрибкоподібно і називається перемиканням тригера. Перемикання тригера складається з двох складових - із встановлення інформації тригера у певний стан (0 або 1) під дією активного вхідного сигналу та із скидання, як правило в стан $Q = 0$, що відбувається під дією спеціального сигналу "скидання" і означає повернення тригера у пасивний (початковий) стан.

4.3.1 Загальна структура та класифікація тригерів

Виходячи із загальних принципів побудови ПП, узагальнюючу структуру тригера можна зобразити з двох частин - комбінаційної схеми (КС) і елементарного двостанового запам'ятовувача, тобто власне тригера (див рис. 4.2). КС, на яку надходять ззовні або подаються по колах додатного зворотного зв'язку різні сигнали, виконує функції керування роботою та формування властивостей всієї схеми тригера. Певна комбінація взаємодії цих входів і визначає відмінність функціонування тригера, а отже, і його тип.

За характером дії вхідних сигналів входи тригера розрізняють: інформаційні і керуючі, які залежно від виконуваної ролі можуть бути дозволяючі та синхронізуючі.

На інформаційні входи тригера подають двійкову інформацію (X_i), яку він має зафіксувати у вигляді нуля або одиниці. На керуючі входи подаються сигнали, які виконують допоміжну функцію – саме з їх допомогою у потрібний момент можна виконати запис чи перезапис та зчитування записаної інформації. Зокрема, на вико-

² Тригер - від англ. *trigger* - спусковий гачок; інша назва *flip-flop*.

навчі входи тригера подаються тактові або синхронізуючі сигнали, які встановлюють у КС момент запису чи зчитування, тобто перемикає тригера

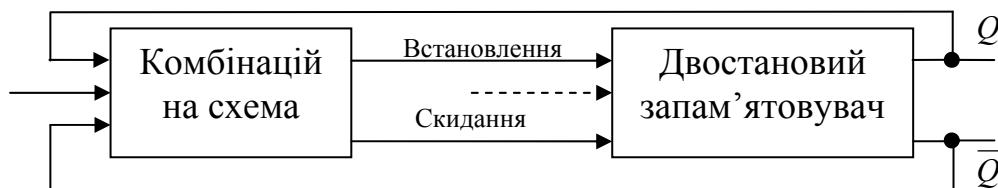


Рис. 4.2 Узагальнююча структурна схема тригера

. Таким чином, встановлення або скидання тригера здійснюється при певній взаємодії вхідних сигналів, які саме роблять його активним або пасивним.

Нижче наведений перелік можливих входів умовного тригера за їх функціональним призначенням.

Інформаційні входи:

R, K – входи окремого встановлення (скидання) тригера в стан 0 (*R* - з англ. *Reset* - скидати, очищувати; *K* - з англ. *Kill* - раптово вимикати), у даному випадку із стану 1 в 0;

S, J – входи окремого встановлення тригера у стан 1 (*S* - з англ. *Set* – установлювати; *J* - з англ. *Jerk* – раптово вмикати), в даному випадку із стану 0 в 1;

D – вхід встановлення тригера у стан 1/0 (*D* - з англ. *Delay* – затримка);

T – вхід перемикає/встановлення /1/ і скидання /0/ тригера, або лічильний вхід (*T* - з англ. *Toggle* – перевертати).

Керуючі входи:

V, E - входи дозволу запису або зчитування інформації (*V* — з англ. *Valve* – клапан; *E* - з англ. *Enable* – дозвіл);

C - виконавчий вхід для тактових чи синхронізуючих імпульсів (*C* - з англ. *Clock* – годинник).

У інтегральному виконанні тригер - це конструктивно єдиний функціональний вузол, тобто мікросхема, яка може бути виготовлена за певною технологією (ТТЛ, КМОН...). З точки зору схемотехніки тригер – це електронна схема, що має два стійких стани (високий або низький рівень), які встановлюються при подачі відповідної комбінації сигналів інформації на керуючі входи схеми і які після закінчення дії цих сигналів зберігаються протягом заданого часу.

Тип тригера визначається функціональною залежністю між сигналом на виході і сигналами на входах. Ця залежність може бути подана різними способами: аналітично, графічно, часовими діаграмами, таблицями станів.

У інтегральному виконанні в основному зустрічаються серед одновходових D- і T-тригери, серед двовходових – RS- і JK -тригери, а серед універсальних - JK(RS) - тригер. Інші типи тригерів, яких досить багато, можуть бути побудовані у базисі різних ЛЕ.

За способом запису інформації, що характеризує часову діаграму роботи тригера і, отже, визначає хід процесу запису інформації у тригер, розрізняють дві групи тригерів: асинхронні та синхронні.

Відмінною особливістю асинхронних тригерів є те, що запис інформації у них

здійснюється безпосередньо з надходженням інформаційного сигналу, що подано на інформаційний вхід тригера. У синхронного тригера, крім інформаційного, є так званий синхронізуючий вхід для тактових імпульсів, які дають дозвіл на запис інформації. Асинхронні тригери застосовуються у колах керування, а також як складова частина більш складних тригерів. Синхронні тригери реагують на інформаційні сигнали тільки при наявності синхронізуючих сигналів. Вони мають найбільше застосування у ЦТ, бо особливість їх роботи забезпечує відсутність негативних явищ у сигнальних колах. В загальному класифікувати тригери можна за трьома найважливішими ознаками: логікою функціонування; способом запису інформації; типом двостанового запам'ятовувача інформації [1]:

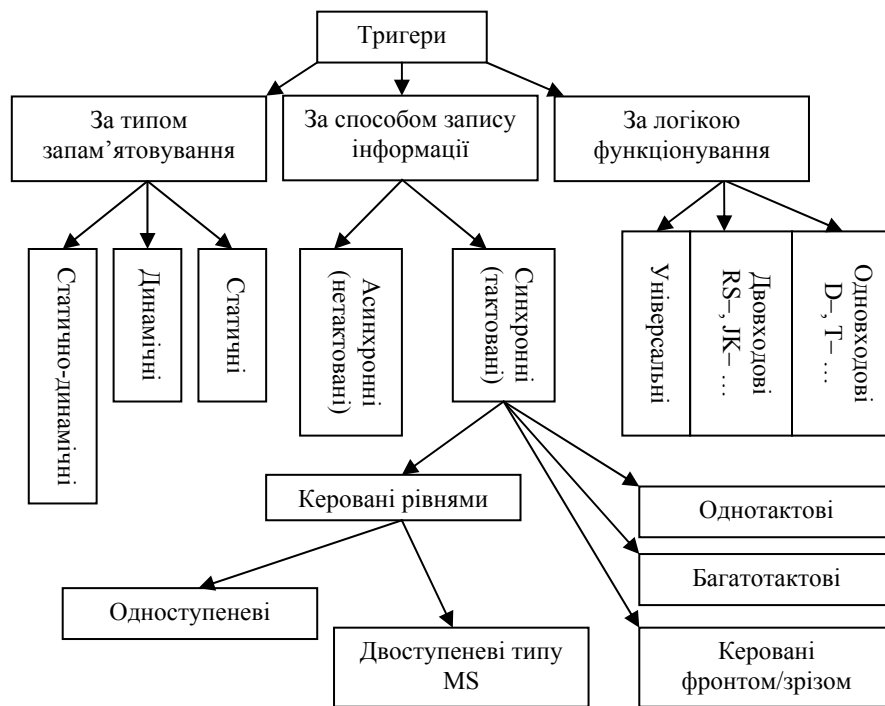


Рис. 4.3 Класифікація тригерів

Розрізняють два типи цифрових сигналів, часові параметри яких здатні змінити стан тригера: *потенціальні* та *імпульсні* (рис. 4.4). *Потенціальні* сигнали X змінюють стан тригера при наявності або відсутності рівня напруги $X = 0$ або $X = 1$. *Імпульсні* - при зміні потенціального сигналу $dX = 1$ тільки при зміні X з одиниці на нуль, $d\bar{X} = 1$, навпаки, при зміні X з нуля на одиницю, $\forall X = 1$ - при будь-якій зміні X . Показані на рис. 4.4 часові параметри цифрових сигналів ідеальні: фронт, зріз та тривалість цих сигналів нескінченно малі.

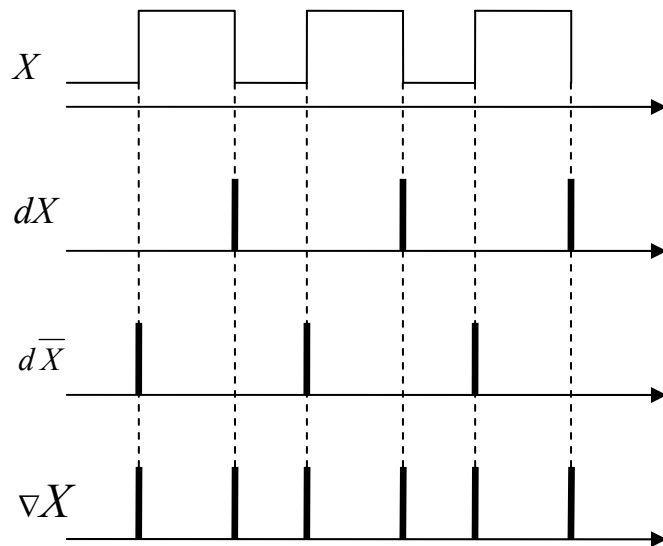


Рис. 4.4 Ілюстрація типів цифрових сигналів

Відповідно до розглянутих типів цифрових сигналів, що застосовуються для зміни стану тригерів, розрізняють входи статичні, які керовані потенціальними сигналами, та динамічні, які керовані імпульсними сигналами, тобто перепадами потенціалів (фронтом чи зрізом).

Залежно від типу застосованого двостанового запам'ятовувача інформації тригери поділяють на класи: статичні, динамічні та статично-динамічні.

Динамічні тригери в основному будуються на МДН-транзисторах. Особливість їх у тому, що вони являють собою замкнене коло, в якому циркулюють імпульси, якщо тригер знаходиться у стані "1". На відміну від динамічних тригерів, параметри яких періодично змінюються, у статичних та статично-динамічних тригерах параметри в одному з двох стійких станів (1 або 0) незмінні.

Тригери позначають так, як показано на рис. 4.5.а,б,* де Т – одноступеневий, а ТТ - двоступеневий тригери. При цьому \overline{RS} -тригер, що на рис. 4.5.а, має статичні входи R і S, а універсальний двоступеневий JK -тригер, крім багатовходових інформаційних входів &J і &K, має динамічний синхронізуючий вхід C типу $dX = 1$, тобто $dC = 1$, при цьому для позначення динамічного входу типу $dC = 1$ застосовують додатний нахил риски.

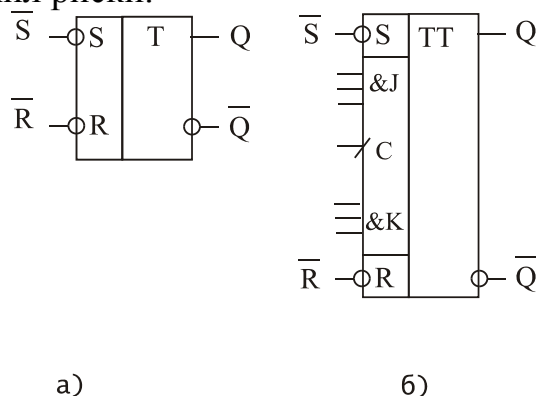


Рис. 4.5 Позначення тригерів

Завдяки своїм значним функціональним можливостям тригери належать до

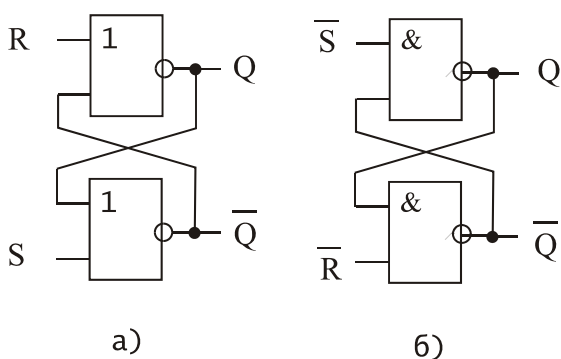
* При оформленні технічної документації, курсових та дипломних робіт, зображення електричних елементів (ЛЕ, тригерів, транзисторів) слід брати з відповідного стандарту. Не можна користуватись іншою літературою, оскільки там ці зображення можуть бути відтворені без дотримання всіх розмірів

найпоширеніших цифрових пристроїв..

4.3.2 Різновиди тригерів

Функціональна структура тригера як двостанового запам'ятовувача бінарних сигналів (див. рис.4.2) досить складна, що закономірно, адже крім запам'ятовування одного біта інформації тригер забезпечує його запис і зчитування у потрібний момент часу. Можливість зміни структури обох частин тригера (комбінаційної і власне тригера) та способів під'єднання зворотних зв'язків зумовлює таку їх різноманітність.

Серед різних структур можна виділити спільну і найпростішу структуру тригера, яку умовно назовемо *базовою*. Такою базовою схемою двостанового запам'ятовувача є асинхронний RS- або \overline{RS} - тригер, бо на його основі можна будувати практично всі відомі типи тригерів. На рис. 4.6.а,б зображені базові схеми RS- і \overline{RS} -тригера, що виконані у базисі ЛЕ відповідно двох елементів Пірса (2АБО-НЕ) і двох елементів Шефера (2І-НЕ). Принцип дії цих тригерів пояснюється спрощеними таблицями станів.



Таблиця 4.3 Стани RS- і \overline{RS} -тригерів

R	S	Q^{T+1}	\overline{O}^{T+1}	\overline{R}	\overline{S}	Q^{T+1}	\overline{O}^{T+1}
0	0	Q^t	\overline{O}^t	0	0	-	-
0	1	1	0	0	1	0	1
1	0	0	1	1	0	1	0
1	1	~	~	1	1	Q^t	\overline{O}^t

Рис. 4.6 Базові схеми RS тригерів

Особливість функціонування асинхронних тригерів полягає у тому, що вони змінюють свій стан тільки в момент дії активних інформаційних сигналів. Зокрема, асинхронний RS-тригер, як видно з таблиці його переходів, при $R = S = 0$ зберігає свій попередній стан Q^t , (0 або 1), а при активних входах $S=1$ або $R = 1$ на його виходах, встановлюється відповідно стан $Q^{t+1} = 1$ ($\overline{O}^{t+1} = 0$) або навпаки. При $R=S=1$, тобто при одночасній дії на обидвох входах сигналів встановлення в "1" і скидання в "0", стан тригера *стає непередбачуваним*. Така комбінація вхідних сигналів для RS-тригерів вважається забороненою ("~" в табл.4.3), а тому її треба уникати, щоб запобігти можливих збоїв. Для асинхронного \overline{RS} -тригера (див. рис. 4.6,б) розглянута характеристика буде справедливою для випадку інверсних ситуацій. Щоб переконатись у справедливості функціонування обох тригерів за їх таблицями переходів, доцільно самостійно прослідкувати всі логічні зміни сигналів на входах і виходах.

З таблиць переходів RS- або \overline{RS} -тригера можна одержати функцію переходів для RS і \overline{RS} -тригера і ця функція переходів матиме однаковий вигляд:

$$Q^{t+1} = \overline{RS}Q^t \vee \overline{RS} = \overline{R}Q^t \vee S = \overline{\overline{R}Q^t \overline{S}} \quad (4.1)$$

Одним з практичних застосувань асинхронного RS -тригера є ліквідація з його допомогою явища електричного "брязкоту" (вібрації) механічних контактів реле,

кнопок, тумблерів, яке викликане ударом під час перемикання. Це негативне явище може викликати небажане спрацювання наступних ЛЕ схеми і привести до збою цифрової системи. Особливо це стосується схем приймання зовнішніх сигналів, наприклад, з клавіатури, кнопок запуску тощо, тобто сигналів, які утворені механічними перемиканнями з високого рівня у низький або навпаки. Вони мають забезпечувати надійний одноразовий перехід пристрою у потрібний стан при першому натисканні механічного перемикача .

На рис.4.7 показана схема придушення явища брязкоту контактів, пояснення роботи якого ілюструють часові діаграми. З допомогою асинхронного $\overline{R}\overline{S}$ -тригера усуваються численні імпульси "деренчання" і виробляється тільки один імпульс з моменту натиснення кнопки чи переведу перемикача.

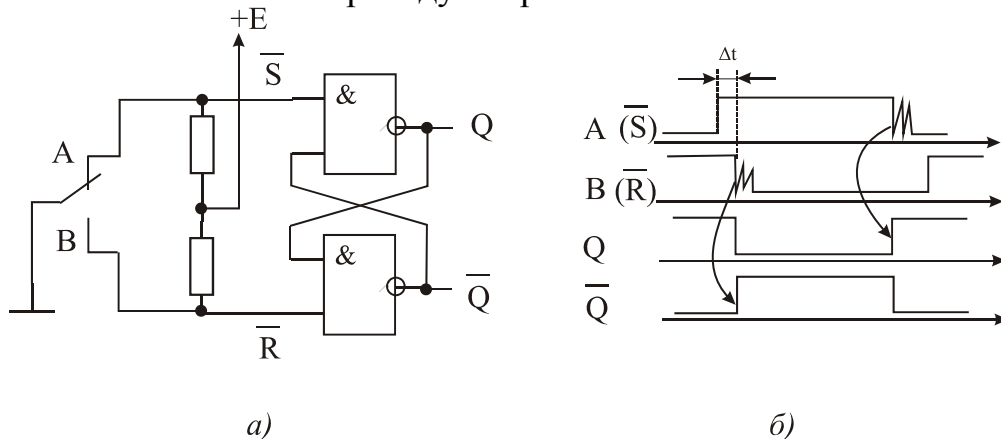
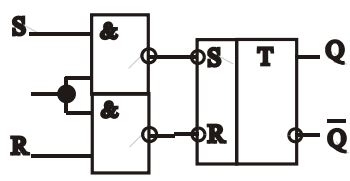
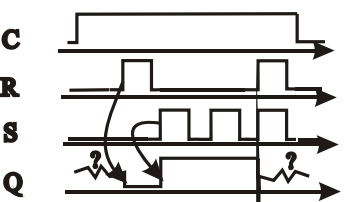
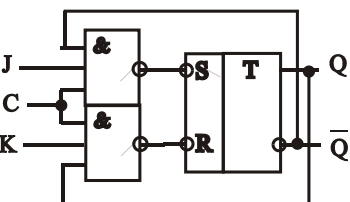
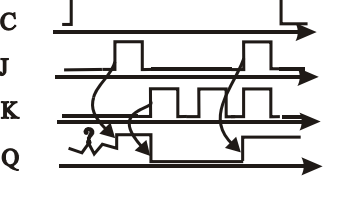


Рис. 4.7 Схема придушення явища брязкоту механічних контактів

Це пояснюється тим (див. рис. 4.7,б), що при зміні положення А перемикача у положення В або навпаки вхід \overline{R} або \overline{S} протягом часу прольоту контакту ще має високий рівень $\approx E$ і тригер зберігає попередній стан. Однак при першому дотику рухомого контакту до нового положення тригер під дією активного сигналу перейде у новий стан, усуваючи таким чином небажаний перехідний процес.

На відміну від асинхронних у синхронних (або тактових) тригерів стан змінюється при наявності разом з інформаційними сигналами синхронізуючих сигналів. Будь-який асинхронний тригер можна перетворити на синхронний, якщо доповнити його виконавчим входом С для синхросигналів. Останній легко утворити об'єднанням незайнятих входів інвертуючих схем збігу 2І-НЕ так, як це подано у табл. 4.4. Тоді лише при $C=1$ тригер реагуватиме не комбінацію інформаційних сигналів, а при $C=0$ залишиться у попередньому стані. Синхронні тригери незалежно від функціональних відмінностей за принципом побудови внутрішньої структури можуть бути одно- або двоступеневі. Одноступеневі тригери можуть мати лише статичне керування, бо приймання і зчитування інформації у них відбуваються одночасно при дії активного сигналу синхронізації, тобто при $C = 1$. Недоліком RS-тригера є наявність заборонених рівнів вхідних сигналів при $R = S = 1$, що може призвести до неоднозначної його роботи. Такої ситуації можна запобігти, якщо виходи Q і \overline{Q} -тригера з'єднати із входами відповідно R і S так, як це показано на рисунку табл. 4.4. Тоді така схема при $R \equiv K$ і $S \equiv J$ перетворюється на схему JK-тригера, у якій відсутня заборонена комбінація вхідних сигналів при $J=K=1$, коли вихідний сигнал JK -тригера стає інверсією попереднього стану.

Таблиця 4.4 Синхронні тригери

Тип тригера	Спрощена таблиця станів			Функція переходів синхронний/асинх.	Схема тригера та часові діаграми
	S	R	Q^{t+1}		
RS-тригер синхрон. C=var	0	0	Q^t	$Q^{t+1} = \overline{\overline{CS} \overline{Q^t} \overline{CR}}$	
	0	1	0		
	1	0	1		
	1	1	~		
Асинхронний C=1				$Q^{t+1} = \overline{\overline{SQ^t} R}$	
JK-тригер синхрон. C=var	J	K	Q^{t+1}	$Q^{t+1} = \overline{\overline{Q^t} C J \overline{Q^t} C K}$	
	0	0	Q^t		
	0	1	0		
	1	0	1		
Асинхронний C=1	1	1	Q^t	$Q^{t+1} = \overline{\overline{Q^t} J \overline{Q^t} K}$	

При інших комбінаціях вхідних сигналів асинхронний JK -тригер працює як RS -тригер. Роботу JK -тригера ілюструє табл. 4.4. Комбінаційна частина у послідовнісних автоматах, що побудована на ЛЕ, має негативну властивість утворювати "гонки", які можуть ініціювати короточасні завади типу "голки". Причиною появи такої завади. Причина цього – різниця затримок поширення сигналів внаслідок різного числа ЛЕ паралельних каналів. Для візуального виявлення цього явища зручно застосовувати двоканальний осцилограф, що дає можливість бачити одночасно зсунуті сигнали і утворену ними заваду.

Усунути явище "гонок" у колах ЛЕ можна за рахунок зовнішнього часового керування (стробування) роботою схеми у такі моменти часу, коли одночасна поява фронтів чи зрізів імпульсів малоімовірна або неможлива. Часове стробування забезпечується синхронізуючими тактовими імпульсами, які подаються на додатковий керуючий вхід тригера. Такий тригер з асинхронного перетворюється у синхронний.

Відсутність заборонених станів у JK-тригера призвело до того, що цей тригер став основою для побудови цілого ряду синхронних тригерів в інтегральному

виконанні. Найбільше практичне застосування в інтегральній схемотехніці дістали синхронні JK-тригери, які можуть бути одноступеневими, тобто із статичним керуванням (або синхронізовані фронтом чи зрізом). Одноступеневий синхронний JK-тригер будується за аналогією з синхронним RS-тригером. Він також має додатковий вхід для синхросигналів, який утворений з'єднанням двох окремих входів схем збігу. При $C = 1$ синхронний JK-тригер перетворюється на асинхронний.

На базі одноступеневих синхронних JK-тригерів можна простою комутацією зовнішніх виводів будувати різні типи тригерів, наприклад, такі, як D і T-тригери.

Розглянемо характерні особливості функціонування та призначення D і T-тригерів, що побудовані на базі одноступеневого JK-тригера (табл. 4.5). Призначення D-тригерів - це затримка на один період (такт) синхронізуючих імпульсів інформації, що надійшли на вхід D. У цифровій схемотехніці застосовують D-тригери, які синхронізовані рівнем C (такий D-тригер називають тригером-заціпкою (від англ. Latch) та D-тригери, які синхронізовані фронтом. В обох випадках D-тригер можна побудувати на базі синхронного або асинхронного JK-тригера, якщо прийняти $J \equiv D$ і $K \equiv \bar{D}$. Як видно з табл. 4.5. для побудови D-тригера на базі JK-тригера досить вхід J з'єднати через інвертор із входом K, що таким чином утворить вхід D. Особливістю D-тригера є зміна стану лише у момент переходу тактового сигналу на вході C з нуля в одиницю, тобто у момент дії переднього фронту синхросигналу D-тригер не сприймає зміну на вході D як при $C = 0$, так і при $C = 1$. Така властивість D-тригера зумовлює його високу завадостійкість за інформаційним входом D. Особливість роботи T-тригера, який ще називають лічильним оскільки він є основним елементом цифрових лічильників, полягає в тому, що стан його змінюється на протилежний при надходженні сигналу $T=1$ і зберігається незмінним при $T = 0$. Згідно з табл. 4.5 стан T-тригера при $J=K=1$ змінюється на протилежний $Q^{t+1}=Q^t$, при надходженні кожного синхроімпульса. У цьому випадку JK-тригер при подачі на вхід C імпульсів функціонує як T-тригер. Функція переходів T-тригера отримується при $J=K=T$ (див. табл. 4.5).

Структурний синтез тригерів можна виконати не лише на базі JK-тригера. Широкі функціональні можливості характерні й синхронному D-тригеру. Зокрема, із синхронного D-тригера легко зробити T-тригер, якщо з'єднати вхід D з виходом \bar{Q} , а вхід C використати як лічильний вхід T, тобто виконати умову $D \equiv \bar{Q}$, $C=T$ (див. табл. 4.3). Для перетворення синхронного D-тригера на асинхронний RS-тригер потрібно виконання умов $CD \equiv S$ і $C\bar{D} \equiv R$.

На відміну від одноступеневих синхронних тригерів, що керовані рівнем або тригерів із статичним керуванням, тригери з динамічним керуванням змінюють свій стан тільки у момент переходу рівня напруги з нуля в одиницю або навпаки. Отже, керуючими сигналами синхронних тригерів з динамічним керуванням є імпульсні сигнали (див.рис.4.4) $dC, d\bar{C}$ або ∇C . Тригери цього типу ще називають тригерами з внутрішньою затримкою або двоступеневими синхронними.

Основною перевагою двоступеневих синхронних тригерів перед одноступеневими є їх внутрішня завадостійкість, яка зумовлена відсутністю порушення стану тригера під час запису інформації, хоча небажана зміна стану тригера можлива також під дією "гонок".

Таблиця 4.5 Синхронні тригери

Тип тригера	Спрощена таблиця станів			Функція переходів синхронний/асинх.	Схема тригера та часові діаграми	
	D	Q ^t	Q ^{t+1}			
D-тригер Синхронний C=var	0	0	0	$Q^{t+1} = \overline{\overline{CDQ^tC}}$		
	0	1	0			
Асинхронний C=1	1	0	1			$Q^{t+1} = D$
	1	1	1			
JK-тригер Синхронний C=var	T		Q ^{t+1}	$Q^{t+1} = \overline{\overline{Q^tCTQ^tCT}}$ $Q^{t+1} = \overline{\overline{Q^tJQ^tK}}$		
	0		Q'			
Асинхронний C=1	1		Q'			

Двоступеневі синхронні тригери (RS-, JK-тригери, тощо) будуються на базі двох послідовно з'єднаних одноступеневих асинхронних або синхронних тригерів, що працюють за принципом "ведучий-ведений", та комбінаційної схеми керування синхронізацією. Такі тригери називають ще MS-тригерами*. Особливість роботи MS-тригерів полягає у тому, що при C=1 здійснюється запис інформації у перший тригер, а при C=0 – передавання інформації з першого тригера в другий та одночасне блокування першого і всіх інших інформаційних входів тригера. Виходами всієї схеми є виходи другого тригера. Отже, для повного циклу роботи двоступеневого тригера потрібно два перепади синхросигналу. Тому і вважається, що MS-тригер керується імпульсом, а не рівнем напруги.

На рис.4.8.а,б показані схема та часові діаграми двоступеневого синхронного JK-тригера, який побудовано на базі каскадного з'єднання двох одноступеневих синхронних RS-тригерів та схеми керування на ЛЕ. Причому для першого M-тригера активним сигналом запису є C=1, а для другого S-тригера C=0. Стан першого тригера змінюється у момент переходу фронту синхросигналу з нуля в одиницю. Отже, по відношенню до вхідних схем JK-тригер має прямий динамічний вхід C і при активних сигналах здійснює запис інформації. По відношенню до схеми, що стоїть на його виході, він має інверсний динамічний вхід синхросигналу, і його вихідний стан визначається станом другого (веденого) тригера, що керований перепадом (зрізом) C з одиниці в нуль, при якому здійснюється зчитування записаної інформації. Для того щоб новий стан тригера не був логічно залежним від вихідних сигналів та своїх станів і водночас щоб синхроімпульси в часі не перекривалися, тобто щоб їх обидві фази (на запис і зчитування) одночасно не

* Від англ. *Master-Slave* – „пан-раб”

набували значень лог.1, у схемі MS-тригера передбачений пороговий інвертор синхроімпульсів. Останній формує обидві фази ϕ_M і ϕ_S синхроімпульсів C і \bar{C} за рахунок спеціально заданих йому логічних порогів спрацювання. В схемі пороговий інвертор має нижчий поріг спрацювання U_s^1 для лог. 1, ніж у М-тригера – отже, він швидше заблокує S-тригер, але вищий U_s^0 для лог.0, ніж у М-тригера, – а тому пізніше розблокує S-тригер. Це дозволяє *по фронту* синхросигналу наперед записати інформацію із входів М -тригера, а разом з тим і MS-тригера при заблокованому М-тригері, а *по зрізу* - читати записану інформацію з S –тригера, тобто з самого MS-тригера, при заблокованому вході MS-тригера. Таким чином, як ілюструє рис. 4.8.б, завдяки пороговому інвертору синхроімпульси на лініях обох фаз (\bar{C}_S і C_M) з'являються у різні моменти часу, які взаємно не перекриваються, з проміжками між імпульсами, що визначаються крутизною фронту і зрізу синхросигналу. Таким чином вдається розділити процеси запису і зчитування інформації, що дуже потрібно для багатьох функціональних пристроїв ЦТ, зокрема для лічильників, у яких одночасно можуть здійснюватись процеси зміни і відліку станів тригерів.

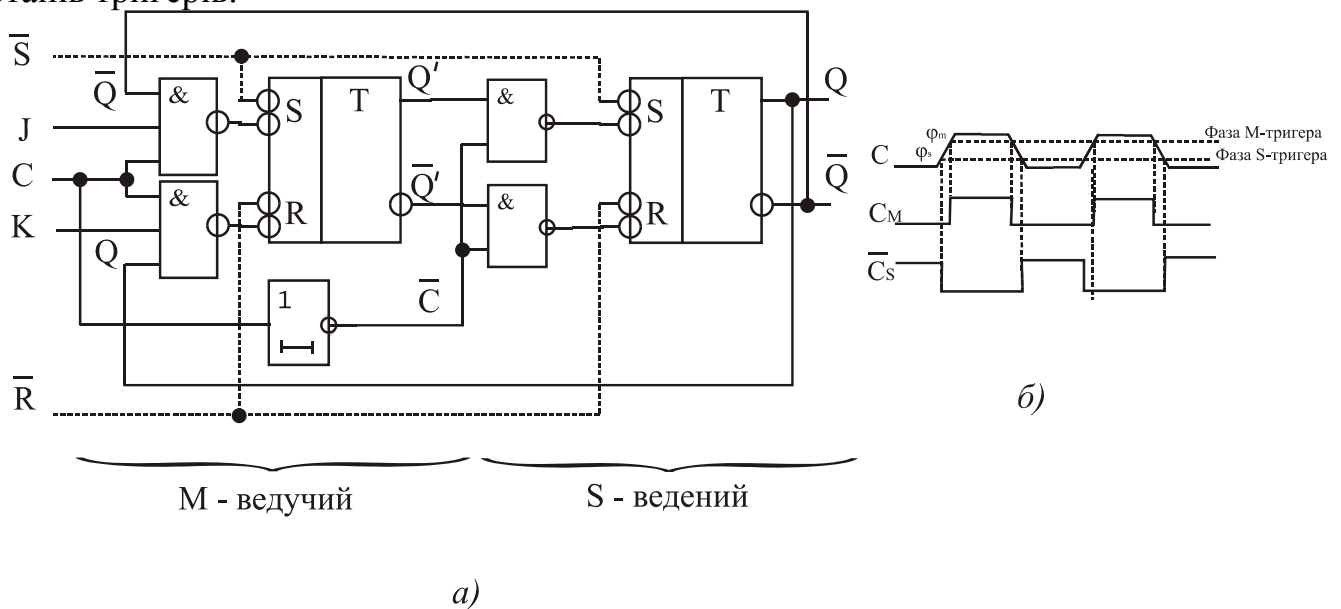


Рис. 4.8 Структура MS тригера

Роботу двоступеневого синхронного; JK-тригера можна пояснити з допомогою часових діаграм, що зображені на рис. 4.9.

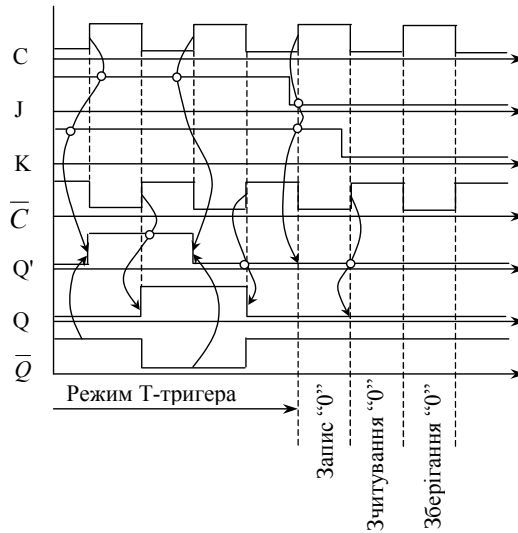


Рис. 4.9 Часові діаграми роботи двоступеневого синхронного JK-тригера

При $J=K=1$ тригер працює у лічильному режимі, тобто як Т-тригер, аналогічно як для випадку одноступеневого JK-тригера (див.табл.4.5). Різниця лише в тому, що у даному випадку зміна стану тригера у лічильному режимі відбувається по зрізу, а не по фронті синхроімпульсів С. Якщо, наприклад, $J=0$ і $K=1$ (тобто згідно з табл.4.5 JK -тригер має перейти в стан $Q^{t+1}=0$), то при надходженні синхроімпульсів стан JK-тригера перейде в нуль тільки при переході синхросигналу на вході С з одиниці в нуль, бо у цей момент відбувається зміна (або збереження) стану другого S-тригера під дією перепаду С з нуля в одиницю. Отже, *запис інформації* в JK-тригер здійснюється на першому (при $C=1$), а *зчитування* - на другому (при $\bar{C}=1$) тактах періода синхроімпульсів. Якщо ситуація на інформаційних входах після другого такту надходження синхроімпульсів не змінилась, то при $J \neq K \neq 1$ даний тригер знаходиться у режимі *зберігання* попереднього біту інформації. Тільки при $J=K=1$ JK-тригер перетворюється на Т-тригер (де $C \equiv T$) і, отже, *здатний частоту надходження синхроімпульсів поділити на 2* (див. рис. 4.9).

Таким чином, двоступеневий синхронний JK-тригер змінює свій стан тільки в момент перепадів синхросигналу, а не під дією рівнів, на вході С. Важливою перевагою двоступеневого JK-тригера є ще те, що інформаційні входи за період синхроімпульсів, тобто під час завантаження 1 біт інформації, функціонально ізольовані від вхідних кіл. Така своєрідна властивість цього тригера забезпечує стійке перемикання тригера навіть тоді, коли частота синхроімпульсів нестабільна. Завдяки таким властивостям на двоступеневих JK-тригерах можна будувати завадостійкі швидкодіючі цифрові пристрої.

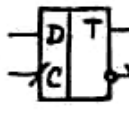
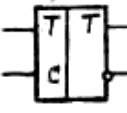
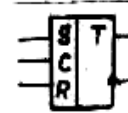
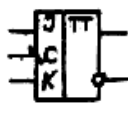
У мікросхемотехніці ТТЛ серії 155 (555) широке застосування має універсальний JK -тригер К155ТВ1. Його відмінність від розглянутого (на. рис. 4.8) лише у наявності потроєних інформаційних входів, що з'єднані за схемою 1: $\&J=J1J2J3$ і $\&K=K1K2K3$ а також несинхронізованих окремих входів \bar{S} і \bar{R} (на рис.4.8 вони показані пунктирними лініями) для асинхронного встановлення тригера у потрібний стан лог.1 або лог. 0. Під час перемикання тригера з одного стану в інший під дією сигналів на інформаційних та синхронізуючому входах на входи \bar{S} і \bar{R} слід прикладати пасивні рівні лог.1. Наявність потроєних входів $\&J$ і $\&K$ дозволяє

розширити функціональні можливості тригера. Зокрема, кон'юнктивне об'єднання входів, J і K спрощує будову лічильників.

Як і у випадку одноступеневого JK-тригера, на базі двоступеневого можна будувати інші схеми тригерів. Шляхом комутації зовнішніх виводів J,K,C,R,S двоступеневий синхронний JK-тригер можна перетворити на тригер, що виконує функції RS-, T-, D-тригерів.

Тригер як двостановий звпам'ятовувач інформації є складовою частиною будь-якого цифрового автомата послідовнісного типу. Тому для синтезу ПП потрібно мати інформацію про функціонування тригера для всієї повноти можливих ситуацій на його входах. Крім таблиці станів та функції переходів необхідно знати, що саме зумовило даний стан тригера, тобто які сигнали на входах можуть викликати зміну стану тригера від Q^t до Q^{t+1} . Таку інформацію відображає так звана таблиця переходів тригера. Кожний тип тригера характеризується своєю таблицею переходів, що наглядно демонструє табл.4.6 для D-,T-,RS-JK-тригерів.

Таблиця 4.6 Таблиці переходів для основних типів тригерів

Код переходу	D-тригер	T-тригер	RS-тригер		JK-тригер	
	D	T	R	S	J	K
0 0	0	0	~	0	0	~
0 1	1	1	0	1	1	~
1 0	0	1	1	0	~	1
1 1	1	0	0	~	~	0
Умовні зображення тригерів						

У інтегральному виконанні можна зустріти різні типи тригерів у серіях відомих технологій:

- ТТЛ(Ш) - JK-тригери (ТВ1, ТВ9, ТВ10, ТВ11) у серіях К/М/155, К/М/555, КР531, /КР/1533; D-тригери (ТМ2, ТМ5) у серіях /К/М/555, /КР/1531, 1533;
- ЕЗЛ - JK-тригери (ТВ135) у серіях К500; D-тригери (ТМ130, ТМ131, ТМ133, ТМ134) у серіях /К/500, К500М, /К/1500;
- КМОН -JK-тригери (ТВ1) у серіях К561, 564, КР1561; D-тригери (ТМ1, ТМ2, ТМ3, ТМ5) у серіях К561, 564, К176. 1564; RS-тригери (ТР2) у серіях К561,564.

4.4 Регістри

Регістром називається послідовнісний пристрій, що призначений для приймання, запам'ятовування, перетворення і передачі двійкової інформації. Регістри можуть використовуватися також для виконання деяких логічних перетворень. У загальному випадку регістр - це блок тригерів і ЛЕ одного типу, що певним чином з'єднані між собою. Введення /запис/ інформації у регістр і зняття /зчитування/ інформації з нього залежать від способу і характеру цих з'єднань. Тому можливих способів приймання і передачі слів може бути чотири:

- з послідовними входом і виходом - регістр типу SISO (S - від англ. Serial);
- з послідовним входом і паралельним виходом - регістр типу SIPO (P - від англ. Parallel);
- з паралельним входом і послідовним виходом - регістр типу PISO ;
- з паралельними входом і виходом - регістр типу PIPO;

Регістр типу SISO призначений для послідовного біт за бітом виконання операцій запису і зчитування n -розрядного слова. Такий регістр ще називають регістром зсуву. Регістр типу SIPO *завантажується послідовно біт за бітом*, а видає записану інформацію одночасно з усіх своїх розрядів за один такт синхросигналу. Приймання інформації у регістрі типу PISO здійснюється *одночасно по всіх розрядах слова* за один такт керування, а зчитування - *послідовно*. Найбільш швидкодіючим очевидно, є регістр PIPO *бо і запис, і зчитування слова у нього відбуваються одночасно і незалежно*. Такий регістр називають регістром пам'яті.

Розрядність будь-якого регістра визначається числом тригерів, кожний з яких як двостановий запам'ятовувач одного розряду слова відповідає за введення, збереження і виведення-1 біт інформації. Регістри можуть відрізнятися між собою за кількістю тактів керування, що необхідні для виконання конкретних операцій. Тому за способом тактування розрізняють одно-, дво- або багатотактні регістри. Для керування однотоктним регістром досить однієї послідовності синхросигналів, багатотактним - кількох. *Операції встановлення регістра у початковий стан, приймання інформації з одного пристрою і передача її в інший пристрій, зсув слова вліво або вправо, реалізуються за допомогою комбінаційної схеми регістра*.

Регістри належать до найбільш поширених функціональних вузлів. Крім зберігання інформації, її зсуву та зчитування, які необхідні для виконання різних арифметичних та логічних операцій над двійковими числами, за допомогою регістрів можна також перетворювати інформацію з одного виду в інший, наприклад, послідовного коду у паралельний або навпаки тощо.

4.4.1 Регістри пам'яті

Це накопичувальні регістри - пристрої в паралельним записом та зчитуванням слова, типу PIPO. Їх основне призначення – зберігання двійкової інформації невеликого об'єму (не більше 2 байт), що подана у паралельному коді. Регістри пам'яті можуть бути синхронізовані рівнем (дозволом $C=1$) або фронтом, чи зрізом синхросигналу залежно від типу застосованих тригерів.

На рис.4.10.а показана схема n -розрядного регістра пам'яті, що побудований на синхронних D-тригерах, та кон'юнкторах на ЛЕ, що виконують функцію збігу. Інформація у вигляді паралельного коду двійкових чисел $\{DI_n, \dots, DI_1, DI_0\}$ заноситься по входній шині і записується у регістрі тільки при наявності дозволяючого рівня $C=1$. Зчитування інформації з регістра можливе тільки при наявності на входах схем збігу дозволяючих рівнів, тобто при $EO=1$.

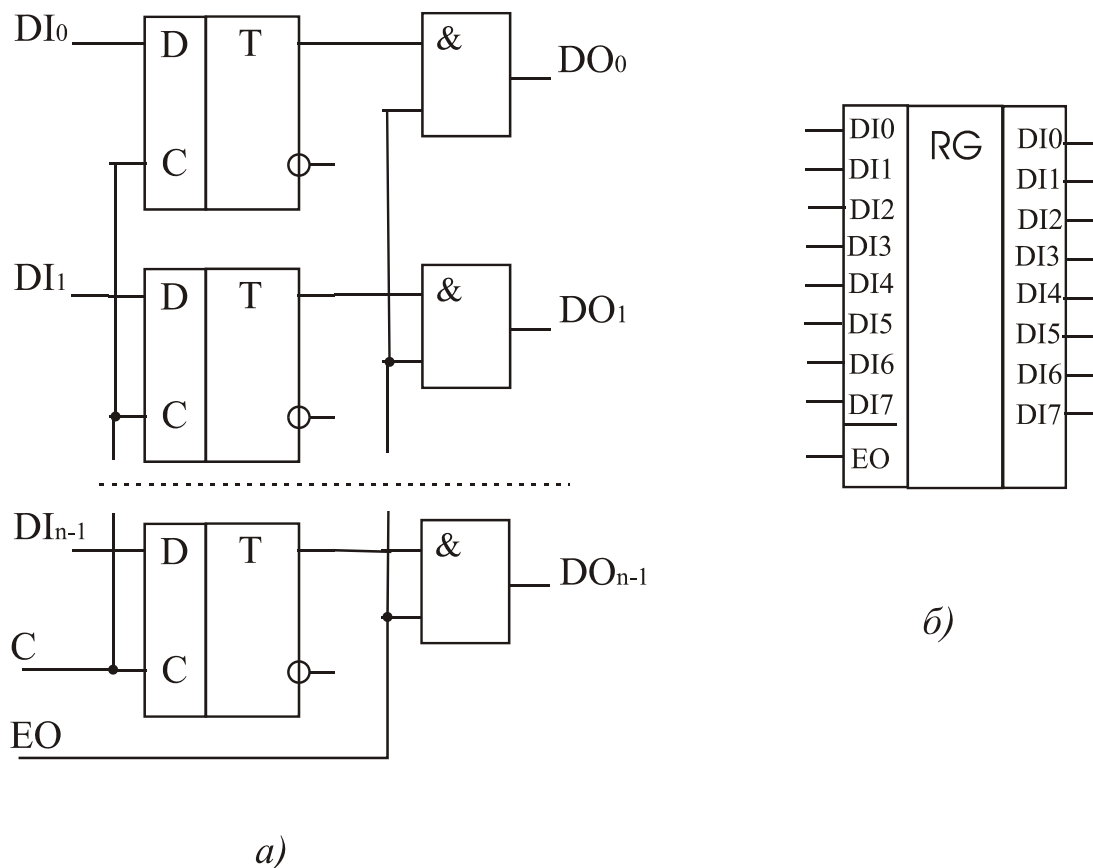


Рис. 4.10 Структура (а) та позначення (б) регістра пам'яті

На рис.4.10.б показане схематичне позначення 8-ми розрядного регістра пам'яті.

У випадку побудови регістра пам'яті на RS -тригерах у схемі потрібно передбачити попереднє "очищення" регістра тобто встановлення всіх тригерів у стан нуль, що здійснюється за допомогою входу скидання R. Попереднє встановлення тригерів в нуль вимагає додаткового часу, то зменшує швидкодію регістра. Для збільшення швидкодії таких регістрів пам'яті застосовують парафазну передачу для перезапису інформації з регістра в регістр. Для цього використовують обидва входи RS -тригерів, по яких одночасно подають сигнали DI_i , та $\overline{DI_i}$. Розрядність регістрів пвм'яті нарощується збільшенням потрібного числа тригерів, тактові входи яких приєднують до шини синхронізації C.

4.4.2 Регістри зсуву

Роботу *регістра зсуву* можна спостерігати в деяких мікрокалькуляторах, де при наборі кожної наступної цифри на клавіатурі числа на індикаторі зсуваються вліво. При цьому виявляються характерні властивості регістрів зсуву:

- регістр зсуву, це пристрій, що має часову пам'ять, завдяки чому числа залишаються на індикаторі навіть при відпусканні клавiш на клавіатурі калькулятора;
- ці пристрої зсувають числа на одну позицію кожного разу як тільки набирається нова цифра.

Таким чином регістри зсуву є виключно корисними приладами, які широко використовуються в цифрових системах. [3]

Отже регістри зсуву – це регістри послідовної дії, що призначені для виконання операції зсуву двійкової інформації, яка подається у послідовному кодi розряд за розрядом. Зсув або переміщення всіх цифр слова може здійснюватись за допомогою регістрів зсуву в напрямі від старших до молодших розрядів (зсув вправо) або від молодших до старших (зсув вліво).

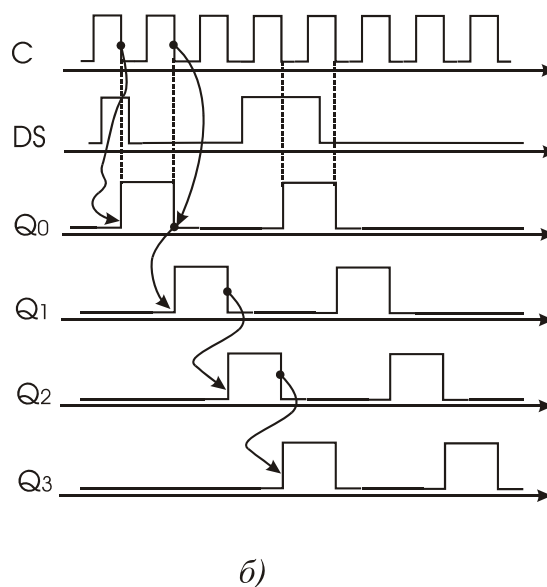
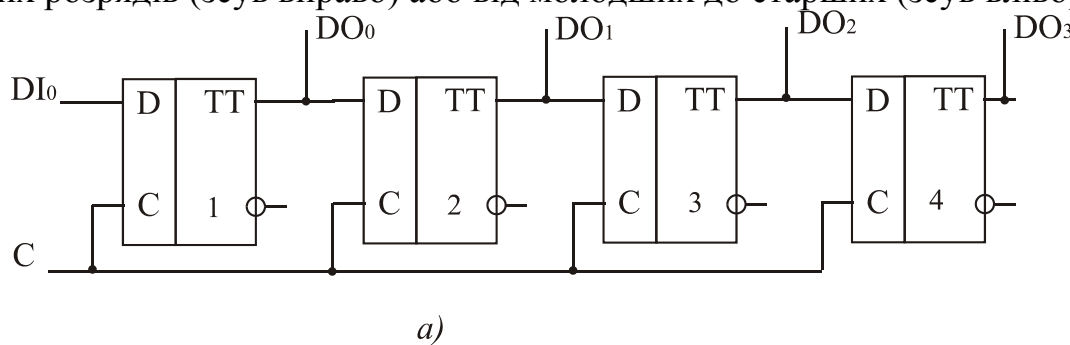


Рис. 4.11 Структура та функціонування регістр зсуву

В обчислювальній техніці регістри зсуву двійкових чисел застосовують для виконання операцій множення та ділення (для виконання множення код

зсувають вліво на один розряд, для ділення – вправо). Операції зсуву на один розряд потрібні лише для позиційної системи числення, при якій вага кожного розряду визначається його позицією в коді.

На основі синхронних RS-, D-, JK-тригерів можна будувати різні структури регістрів зсуву. На рис.4.11.а побудований регістр зсуву типу SIPO на базі чотирьох двоступеневих синхронних D-тригерів. Він має один інформаційний вхід DS (від англ. *Data Serial*), на який інформація наводить у вигляді послідовного коду, і керуючий вхід С, на який подаються тактові синхронізуючі імпульси.

D -тригер (1) відповідає за молодший розряд (MP), в D-тригер (4) -за старший розряд (CP) чотирирозрядного числа, яке потрібно записати у даний регістр зсуву. Тоді з надходженням тактових імпульсів одночасно на динамічні входи С D-тригерів, для яких активним сигналом є зріз тактових імпульсів, вихід кожного тригера набуватиме стану попереднього. Таким чином, одиниця, що надійшла на інформаційний вхід DS регістра, зсунеться (рис.4.11.б вправо) з молодшого розряду до старшого синхронно з періодом надходження тактових імпульсів. Такий регістр належить до регістрів зсуву *вліво*.

Слід зазначити, що правильність запису інформації у регістр зсуву залежить і від того, як на його вхід надходять біти послідовного двійкового коду (починаючи з молодшого чи старшого розряду). У розглянутому на рис. 4.11.а регістрі зсуву вліво послідовний код надходить на вхід DS починаючи зі старшого біта. Для запису чотирирозрядного числа (так само, як і дня зчитування) у даний тригер потрібно чотири такти синхроімпульсів.

Зчитування даних з регістра, що після припинення подачі тактових імпульсів зберігатимуться на його виходах DO_0, DO_1, DO_2, DO_3 , можна виконувати як в послідовному коді, з виходу останнього тригера (4), так і в паралельному коді, відразу з усіх розрядів. За розглянутим принципом роботи та подібною структурною схемою побудовано чотирирозрядний регістр на КМОН-структурі 564ИР2.

Двонаправлені (або реверсивні) регістри зсуву здатні зсувати записану інформацію і вправо, і вліво, тобто завантажене у регістрі слово можна зсувати по лінійці тригерів як вправо, так і вліво. Напрямок зсуву визначається рівнем керуючого сигналу (1 або 0).

Для прикладу розглянемо мікросхему універсального чотирирозрядного регістра зсуву К555ИР1 яка виконана у схемотехніці ТТЛ середнього ступеня інтеграції. Вона має один послідовний DS (D0) та чотири паралельних (D1,D2,D3,D4) інформаційні входи. Умовне позначення регістра зсуву К555ИР1 показано на рис. 4.12. Регістр виконаний на чотирьох RS -тригерах, має два тактових входи С1 і С2, чотири виходи Q0, Q1, Q2 , Q3 і керуючий вхід V, що визначає режим роботи регістра (зсув вправо чи вліво). Можливі чотири режими роботи універсального регістра зсуву К555ИР1 введені у табл. 4.7.

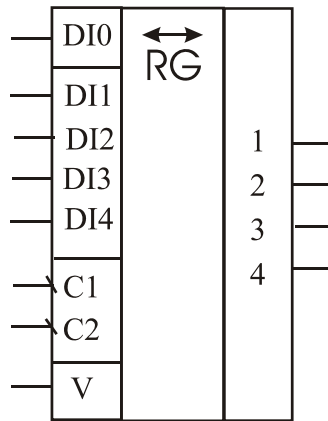


Рис. 4.12 Приклад регістру зсуву. Регістр К555ІР1

Таблиця 4.7 Режими роботи регістра зсуву 555ІР1

Режим роботи	Стан входів				
	DI0	DI1...DI4	V	C1	C2
Зсув вправо	DS	~	0	1→0	~
Зсув вліво	DS	~	1	~	1→0
Паралельний вхід	~	PL	1	~	1→0
Зберігання при зсуві вліво	~	~	0	1	~

Вибір режиму роботи регістра визначається логічним рівнем на вході V (0 або 1). У табл. 4.7 прийняті такі позначення: „~” - невизначений стан (1 або 0), „1→0” - перехід від одиниці до нуля, тобто входи C1 і C2 стають активними під час зрізу тактових імпульсів, що подаються на ці входи.

Застосування регістрів зсуву дуже різноманітне. На їх базі можна будувати як відомі функціональні вузли, так і спеціальні послідовнісні пристрої цифрової техніки. Зокрема, зсув інформації на один розряд вправо або вліво відповідає арифметичній операції ділення або множення на два. Регістр зсуву може виконувати функцію лічильника, якщо на його послідовний вхід подавати одиницю. У системах радіозв'язку та радіолокації регістри зсуву застосовують для побудови радіозаслонів, двійкових кореляторів та різних пристроїв складної обробки радіосигналів. На регістрах зсуву з великим числом розрядів можна побудувати запам'ятовувальний пристрій, якщо тактові імпульси неперервно подавати від синхрогенератора, а вихід регістра замкнути у кільце на його послідовний вхід. У результаті одноразово записаний у регістр код буде нескінченно довго циркулювати у цьому кільці. На базі регістрів зсуву можна також побудувати цифрові лінії затримки.

Розрядність регістра зсуву можна збільшити під'єднанням додаткових тригерів. Так побудовані восьмирозрядні регістри зсуву з великими функціональними можливостями (K531ИР24, K155ИРІЗ, K555ИР8, K555ИР9, 561ИР6, 564ИР6), а також вісімнадцятирозрядні регістри зсуву типу 564ИР1 та КІ76ИР10, що складаються з чотирьох окремих секцій, які зв'язані загальною шиною синхронізації.

4.5 Лічильники

Майже кожна складна цифрова система містить декілька лічильників. Призначення їх очевидне: це підрахунок кількості деяких подій або часових інтервалів, або впорядкування подій в хронологічному порядку. Крім того, лічильники можуть виконувати й інші функції: наприклад їх можна використати для адресації, в якості подільників частоти й елементів пам'яті. Цифрову схему, що виконує відлікову функцію, можна скласти з тригерів, однак в зв'язку з дуже широким застосуванням таких схем, багато виробників випускають лічильники у вигляді монолітних ІС. [3]

Лічильник - це функціональний пристрій, що призначений для підрахування числа імпульсів, які надійшли на його вхід, і фіксації цього числа на його регістрі у вигляді коду.

В обчислювальній та мікропроцесорній техніці лічильники виконують підрахунок кількості циклів послідовно виконаних операцій, які використовуються для утворення адресів команд при звертанні до запам'ятовувальних пристроїв, а також для виконання операцій приймання та передачі кодів тощо.

Лічильник - це скінченний автомат, внутрішній стан якого визначається тільки кількістю лог. 1, які надійшли на його вхід (лог.0 не змінює стану лічильника). Лічильники, як правило, складаються з кількох Т -тригерів, які певним чином з'єднані між собою. Регістри лічильника утворюють виходи тригерів, число яких m - визначає розрядність лічильника.

Основним параметром лічильника є його модуль лічби $M_n = 2^m$, (де m - число розрядів лічильника), що визначає число стійких внутрішніх станів лічильника у тому числі й нульовий стан, тобто максимальну кількість імпульсів, яку він може підрахувати. Отже, у m -розрядному лічильнику може бути 2^m стійких стани і тому його модуль лічби завжди $M_n \leq 2^m$. Якщо кількість імпульсів які надійшли на вхід лічильника, дорівнює M_n -він повертається у початковий стан /у підсумовувальних лічильників - у нульовий/, утворюючи при цьому сигнал (імпульс) переповнення. Це означає, що підрахунок одиниць здійснюється лічильником за модулем лічби M_n , який характеризує місткість лічильника. Для підсумовувального лічильника це максимальне число одиниць, яке він може підрахувати. Дана характеристика лічильника залежить від його розрядності та основи системи числення.

Для заданого модуля лічби M_n кількість тригерів t , що необхідне для побудови лічильника, визначається з умови найближчого більшого цілого числа;

$$m = \text{int}[\log_2 M_n]. \quad (4.2)$$

На виході останнього розряду (тригера) лічильника частота імпульсів у M_n раз менша за частоту вхідних імпульсів. Тому лічильник можна застосовувати як подільник частоти з коефіцієнтом ділення вхідної частоти M_n .

4.5.1 Класифікація лічильників

Залежно від модуля лічби M_n лічильники бувають двійкові (лічильники за модулем 2^m , $M_n = 2^m$) і з довільним модулем (лічильники за модулем M_n у яких $M_n \neq 2^m$ причому число m заокруглюється до більшого цілого числа).

Будь-який лічильник так само, як і регістр, можна виконати у вигляді синхронного або асинхронного, потенціального або імпульсового послідовнісного автомата, в також у вигляді автомата змішаного типу. Спосіб організації лічби (синхронний чи асинхронний) визначається типом Т-тригерів, які реалізують на D-тригерах з динамічним керуванням (синхронних чи асинхронних), а найчастіше на JK-тригерах (у мікросхемному виконанні). Асинхронні лічильники порівняно з синхронними на практиці зустрічаються рідше.

За цільовим функціональним призначенням лічильники поділяють на два типи: проті (підсумовувальні та віднімальні) та реверсивні. У підсумовувальному лічильнику при подачі на вхід імпульсу код числа, що зберігається у лічильнику, зростає на одиницю, а у віднімальному - зменшується на одиницю. Отже, підсумовувальний, лічильник виконує прямий, а віднімальний - обернений підрахунок числа одиниць, що надійшли на його вхід. Реверсивний лічильник може працювати в режимі прямого та оберненого підрахунку.

У зв'язку з тим, що у кожному розряді лічильника, наприклад підсумовувального, виконується операція додавання двійкових чисел, у результаті якої утворюється перенос у старший розряд, лічильники ще розрізняють за способом утворення сигналів переносу. Це, зокрема, лічильники з послідовним, паралельним і послідовно-паралельним переносом. Їх часто називають просто послідовні, паралельні та послідовно-паралельні лічильники. Вони відрізняються способами подачі вхідних тактових імпульсів на входи розрядів. У послідовному лічильнику вхідні імпульси подаються тільки на вхід першого тригера, а у паралельному - одночасно на синхровходи тригерів у всіх розрядах. Різновидом паралельних лічильників є кільцеві лічильники, що будуються на базі регістрів зсуву. Послідовно-паралельний лічильник будують за принципом послідовного з'єднання (каскадування) кількох паралельних лічильників.

Крім модуля лічби M_n лічильники характеризуються ще такими параметрами, як розрізняльна здатність та час реєстрації. Ці параметри характеризують швидкодію лічильника. Розрізняльна здатність лічильника - це мінімальний період проходження вхідних імпульсів, при якому забезпечується надійна робота лічильника. Час реєстрації - максимальний часовий інтервал між моментом закінчення подачі вхідного імпульсу та моментом установаження коду на регістрі лічильника.

4.5.2 Лічильники з послідовним переносом

Послідовні лічильники - це асинхронні лічильники, що мають один вхід x і n виходів (Q) відповідно до числа послідовно з'єднаних тригерів. Найпростіші послідовні двійкові або лічильники за модулем 2^m ($M_{\text{л}}=2^m$) реалізуються на базі кількох Т-тригерів, кожний з яких працює як лічильник за модулем 2. На рис.4.13 показано підсумовувальний лічильник за модулем 8 ($m=3$), який побудований на трьох універсальних двоступеневих JK-тригерах, часові діаграми його роботи зображені на рис. 4.14.

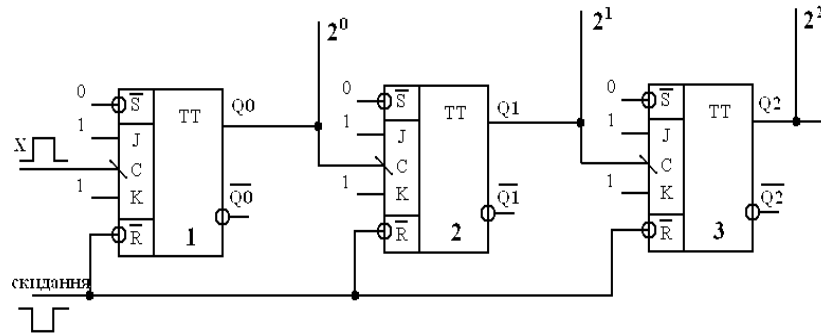


Рис. 4.13 Структура підсумовувального лічильника за модулем 8

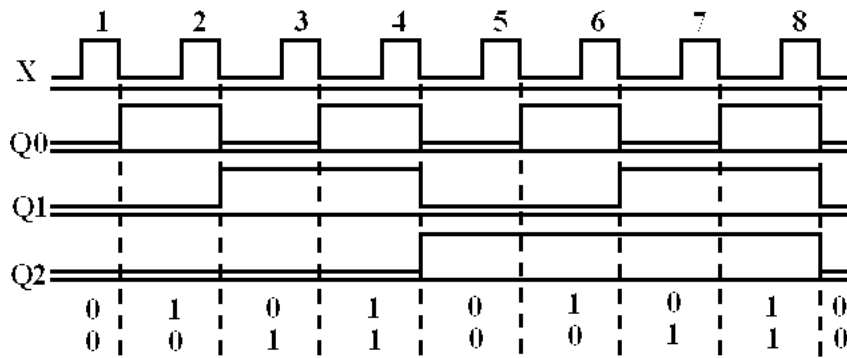


Рис. 4.14 Часові діаграми роботи лічильника за модулем 8

Перед початком роботи лічильника всі його тригери переводяться у нульовий стан імпульсом скиду в нуль. Надходження першого імпульсу своїм зрізом переводить тригер (1), тобто молодший розряд лічильника у стан 1, і на регістрі лічильника фіксується код (001). Другий імпульс, що надходить на вхід X тригера (1), повертає його в стан 0, але при цьому виникає імпульс переносу, що утворений на першому ступені тригера (2), який змінить тепер стан тригера (2) на 1. Отже, на регістрі лічильника з'являється число (010). Заповнення розрядів лічильника а кожним імпульсом буде продовжуватися доти, поки лічильник не відрахує максимальне число на його регістрі (111). Восьмий імпульс на вході лічильника переводить своїм зрізом тригер (1) в стан 0, в імпульс переносу - тригери (2) і (3) в стан 0, повертаючи таким чином лічильник у початковий нульовий стан. Зауважимо, що на інших входах тригерів під час роботи лічильника повинні бути такі рівні: $J=K=1$; $\bar{S}=0$; $\bar{R}=0$. За розглянутим

принципом працюють чотирирозрядні асинхронні лічильники ТТЛ К155ИЕ2 (двійково-десятковий) і К155ИЕ5 (лічильник-подільник).

Розглянутий підсумовувальний двійковий лічильник легко перетворити віднімальний за модулем 8, якщо замість прямих виходів Q_0, Q_1, Q_2 у тригерів використати їх інверсні виходи які при цьому будуть утворювати його регістр вихідного коду. У цьому неважко переконатись, бо при такій комутації виводів тригерів перенос від розряду до розряду буде утворюватися при переводі відповідного тригера у стан 1, а не у стан 0, як це мало місце у підсумовувальному лічильнику. У віднімальному лічильнику кожний прихід імпульсів на його вхід не збільшує, а зменшує вміст лічильника на одиницю. Тому, коли потрібно підрахувати число відданих імпульсів, даний віднімальний лічильник попередньо треба встановити у стан (111).

Таким чином, з кожним надходженням вхідних імпульсів на регістр віднімального лічильника фіксуватиметься обернений код так, як наведено у табл. 4.8 для випадку віднімального/двійкового/ лічильника за модулем 8.

Таблиця 4.8 Робота віднімального лічильника за модулем 8

Регістр віднімального лічильника за модулем 8	Число вхідних імпульсів								
	0	1	2	3	4	5	6	7	8
Q_0	1	0	1	0	1	0	1	0	1
Q_1	1	1	0	0	1	1	0	0	1
Q_2	1	1	1	1	0	0	0	0	1

У розглянутих послідовних двійкових лічильниках зміна стану поширюється послідовно по всій лінійці тригерів. Такі лічильники можна називати по-різному залежно від того, яку характерну властивість підкреслювати. Якщо підкреслити те, що запуск тригерів з лічильника здійснюється не одночасно, тут підходять визначення "з послідовним переносом" і "асинхронний". Назва "лічильник за модулем 2" відбиває максимальне число станів (тобто модуль лічби $M_n = 2^m$), яке фіксується на регістрі лічильника за один повний цикл лічби, а визначення " m -розрядний" показує кількість двійкових розрядів регістра лічильника.

Послідовні лічильники можуть бути й реверсивними, якщо в них передбачити додаткові ЛЕ, які б керували режимом роботи на підсумовування або на віднімання вхідних імпульсів.

Розглянуті асинхронні лічильники з послідовним переносом мають загальний недолік - значний час реєстрації підрахованої кількості вхідних імпульсів, тобто мають малу швидкодію. Цей недолік зумовлений втратою часу на послідовне формування імпульсу переносу в кожному розряді лічильника. У найбільш несприятливому випадку перенос, що виник в молодшому розряді, викличе по черзі переноси у всіх інших розрядах

лічильника і час реєстрації при цьому буде дорівнювати $m \cdot t_{зюп} / \text{де } t_{з.л}$ - затримка переносу/ в одному розряді. Зменшення часу реєстрації лічильника досягається при застосуванні паралельного переносу біта з молодшого у старший розряд, що реалізовано в лічильниках з паралельним переносом. [1]

4.5.3 Реверсивні лічильники

Лічильники цього типу можуть виконувати як додавання, так і віднімання імпульсів. Для їх побудови можуть бути використані принципи роботи та схемні рішення простих лічильників (підсумовувальних та віднімальних). Реверс роботи даного лічильника здійснюється перемиканням кіл передачі переносу так, як це показано на рис.4.15, де зображено синхронний реверсивний лічильник на JK-тригерах.

У схемі лічильника передбачено два режими роботи – додавання ("ДОД"), при якому лічильник підсумовує, тобто збільшує на одиницю число вхідних імпульсів, і віднімання ("ВІД"), при якому він за кожним тактом віднімає одиницю двійкового коду. Як видно з рис. 4.16, реверсування досягається передачею або сигналу переносу з виходів Q_i , або сигналу позички з виходів \bar{Q}_i тригерів. Для організації реверсування зручно застосувати у відповідних розрядах лічильника мультиплексори 2-1 на ЛЕ 2І-НЕ.

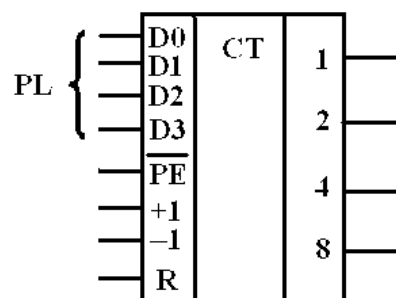


Рис. 4.15 Реверсивний лічильник

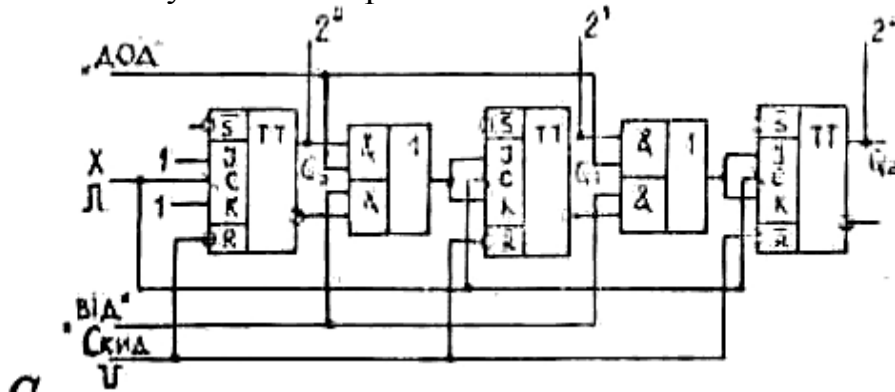


Рис. 4.16 Структура реверсивного лічильника

При подачі рівня 1 на керуючий вхід "ДОД", що є дозволяючим для верхніх кон'юнкторів ЛЕ 2І-АБО, у реверсивному лічильнику прямі виходи тригерів мають зв'язок із входами наступних тригерів і, отже, реалізується режим додавання. Для перемикання лічильника у режим віднімання слід подати одиницю на вхід "ВІД", що під'єднаний до всіх нижніх; кон'юнкторів ЛЕ 2І-АБО. Цього досить для передачі сигналів позичок з інверсних виходів \bar{Q}_i тригерів.

У серіях ТТЛ випускаються мікросхеми реверсивних лічильників різних типів. У серії КІ55, зокрема, це чотирирозрядні реверсивні КІ55ІЕ6 (двійково-десятковий) і КІ55І7 (двійковий) лічильники, які відрізняються лише внутрішньою логікою, яка керована тригерами. Умовне графічне позначення цих лічильників зображено на рис. 4.15, де у середньому полі для КІ55ІЕ6 дається позначка СТ2/10, а для КІ55ІЕ7- позначка СТ2. При подачі послідовності імпульсів на вхід (+1) відбудеться їх підсумовування, в при подачі на вхід (-1) - віднімання. Лічильники встановлюються у довільний стан за допомогою паралельного входу РL а у нульовий стан - рівнем 1 на вході R .

4.5.4 Лічильники з довільним модулем лічби

У розглянутих двійкових лічильників модуль лічби $M_n = 2^m$. Залежно від числа розрядів m такі лічильники здатні підраховувати і реєструвати на своєму регістрі лише 2, 4, 8, 16, 2^m імпульсів. На практиці дуже часто потрібно мати лічильники з довільним модулем лічби $M = M_n \neq 2^m$ що має некратний цілий степінь 2. Найдоцільнішим для практики є десятковий лічильник, що має $M = 10$.

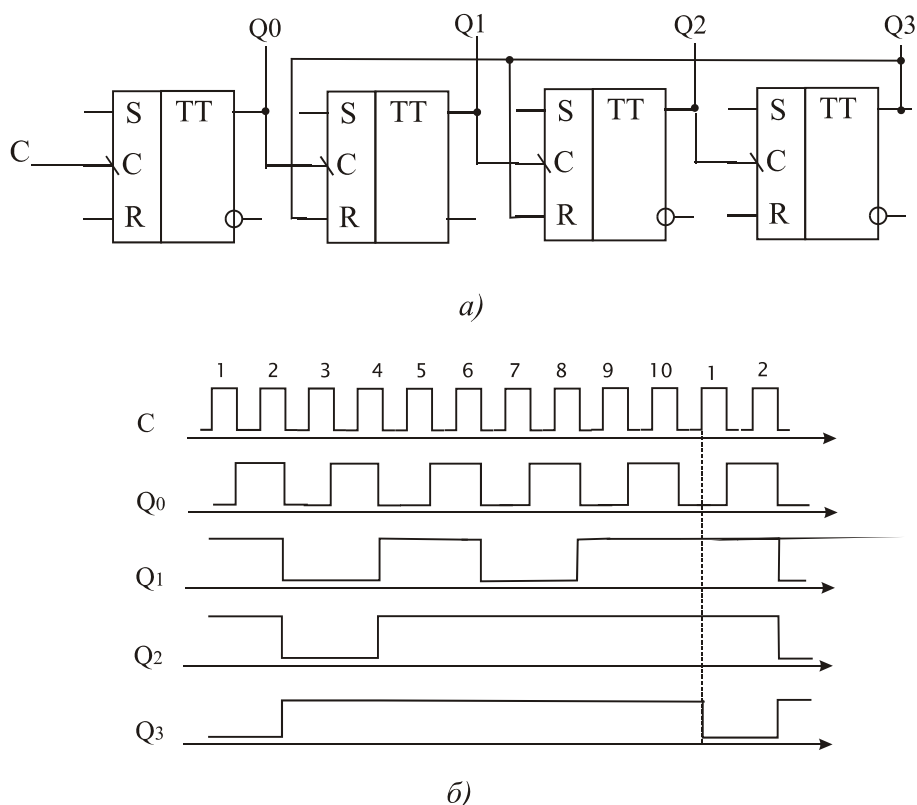


Рис. 4.17 Структура (а) та робота двійково-десятьового лічильника

На рис. 4.17.а зображена функціональна схема асинхронного двійково-десятьового лічильника, що побудований на синхронних RS-тригерах. Таблиця станів тригерів пояснює роботу лічильника при дії послідовності входніх імпульсів X.

Розрядність лічильника за модулем M визначається з умови $2^{m-1} < M < 2^m$,

причому m визначається мінімальним цілим числом згідно з (4.2). Якщо $M=10$, для реалізації (десятькового) лічильника потрібно не менше як $m=4$ тригери, бо $2^3 < 10 < 2^4$. Отже, лічильник за модулем 10 має бути чотирирозрядним і при появі на його регістрі коду $1010_2 = 10_{10}$ повинен скидуватися в нуль. Такий лічильник, очевидно, буде мати невикористані надлишкові стани – їх $\epsilon N=2^4-10=6$, які необхідно вилучити. Щоб позбутися цих станів, застосовують зворотні зв'язки з виходу лічильника на входи тригерів тих розрядів, які у двійковому поданні числа N мають одиницю. Для десятикового лічильника $N=6_{10} = 0110_2$ і, отже, сигнал зворотного зв'язку слід подавати на відповідні входи тригерів другого і третього розрядів. У синхронних лічильників такий принцип побудови внутрішніх зв'язків називається блокуванням переносу. Аналогічно реалізується побудова віднімальних лічильників.

Таблиця 4.9 Стани двійково-десятькового лічильника

Номер стану	Q_3	Q_2	Q_1	Q_0
0	0	1	1	0
1	0	1	1	1
2	1	0	0	0
3	1	0	0	1
4	1	0	1	0
5	1	0	1	1
6	1	1	0	0
7	1	1	0	1
8	1	1	1	0
9	1	1	1	1

Початковий стан лічильника визначається кодом 0110, від якого починається лічба до коду 1111. Після чергового десятикового імпульсу на виході Q_3 лічильника виникає імпульс переносу, який по колах зворотного зв'язку встановлює на розрядах регістра лічильника початковий стан 0110.

Таким чином, за допомогою чотирирозрядного двійкового лічильника можна дуже просто зробити двійково-десятьковий лічильник, виконавши відповідну комутацію виводів даної мікросхеми. Для побудови лічильника за модулем 10 доцільно використовувати, наприклад, мікросхеми ТТЛ К555ИЕ2, К1533ИЕ5, К1533ИЕ6, вводячи кола зворотних зв'язків а виходів на відповідні їх входи.

До лічильників за модулем M належать також лічильники з перериванням циклу. Принцип побудови схем таких лічильників полягає у тому, що скид у нуль здійснюється після досягнення числа M , яке визначається заданим циклом, тобто при $M+1$. Лічильник має схему розпізнавання цього числа, тобто схему збігу, а для скиду в нуль при досягненні M використовується шина скиду R . Лічильники з перериванням циклу можуть бути реалізовані за допомогою довільних вже розглянутих принципів побудови.

Підвищений інтерес до лічильників за модулем М пояснюється можливістю застосування їх як подільників частоти із заданим коефіцієнтом ділення. Такі лічильники - подільники частоти складають основу, наприклад, цифрових годинників, основним вузлом яких є подільник частоти на 60. Для реалізації цього подільника частоти потрібно послідовно з'єднати лічильник-подільник на 10, тобто декадний лічильник (наприклад, КІ533ІЕ1), та лічильник за модулем 6 (лічильник-подільник на 6), наприклад К155ІЕ4. Такі мікросхеми, як КІ76ІЕ3 і КІ76ІЕ4, є лічильниками за модулем 6 і 10 з дешифратором, який придатний для роботи на семисегментний індикатор. Прикладами лічильників-подільників з фіксованим коефіцієнтом ділення частоти кварцових генераторів є п'ятнадцятирозрядний двійковий подільник К176ІЕ5 і розроблені спеціально для цифрових годинників мікросхеми К176ІЕ12, К176ІЕ17 і т.ін.

Спільною властивістю розглянутих лічильників-подільників є фіксований коефіцієнт ділення. Особливий інтерес для практики у функціональному відношенні викликають програмовані лічильники-подільники частоти, модуль лічби яких може змінюватися під дією зовнішніх кодових сигналів. До таких подільників частоти належить мікросхема К155ІЕ8, що являє собою шестирозрядний двійковий лічильник з постійним або із змінним коефіцієнтом ділення K_d максимальне значення якого дорівнює 64. Особливістю функціонування цієї мікросхеми є те, що вона забезпечує будь-який коефіцієнт ділення K_d з одиничним кроком у межах від 64:1 до 64:63. Залежно від кодової комбінації на паралельному шестирозрядному вході лічильника ($D_0, D_1, D_2, D_3, D_4, D_5$) кількість імпульсів на виході $N_{вих}$ схеми за один цикл, що містить $2^6 = 64$ вхідних імпульси $N_{вх}$ на лічильному вході, буде визначатися за формулою

$$N_{вих} = 2^5 D_5 + 2^4 D_4 + 2^3 D_3 + 2^2 D_2 + 2^1 D_1 + 2^0 D_0 \quad (4.3)$$

Якщо коефіцієнт ділення лічильника-подільника

$$K_d = N_{вх} / N_{вих} = 64 / N_{вих}, \quad (4.4)$$

то частота його вихідних імпульсів буде

$$f_{вих} = f_{вх} / K_d = f_{вх} \cdot N_{вих} / 64. \quad (4.5)$$

У мікросхемі К155ІЕ8 передбачена можливість каскадування аналогічних схем, що дає можливість значно збільшити коефіцієнт ділення всього пристрою. Функцію ділення частоти лічильника за модулем М успішно використовують також і в інших цифрових, зокрема вимірювальних, пристроях: частотомірах, таймерах, генераторах, тощо.

4.5.5 Кільцеві лічильники та лічильники Джонсона

Особливість роботи цих лічильників полягає у тому, що кількість імпульсів, які потрібно підрахувати, визначається не кодом, що фіксується на виходах лічильника, а станом останнього тригера. У найпростішому випадку кільцевий лічильник будують на регістрі зсуву, наприклад на чотирирозрядному регістрі КІ55ІР1, що зображений на рис. 4.18, у якого вихід останнього тригера з'єднаний з інформаційним входом першого тригера (молодшого розряду регістра), утворюючи замкнене кільце, по якому циркулює одиниця або нуль. За кожним тактовим імпульсом одиниця або нуль переходить від тригера до тригера з циклом, що дорівнює кількості тригерів n . Отже, щоб запустити цей процес, потрібно насамперед у перший тригер регістра записати одиницю або нуль. Запис, наприклад 1, тобто коду $\{0001\}$ здійснюється при $V=1$ подачею на PL рівнів $D_0=0, D_1=D_2=D_3=0$. Кількість імпульсів підраховується при $V=0$ подачею цих імпульсів на тактовий вхід $C1$. З цього моменту тригери лічильника починають послідовно переходити у стан 1, що реєструється на паралельних виходах регістра як послідовність кодів $\{0001\}$ $\{0010\}$, $\{0100\}$, $\{1000\}$, а далі за рахунок зворотнього зв'язку цикл повторюється знову. Отже, даний лічильник має модуль лічби $M_n=4$, бо може підраховувати не більше чотирьох імпульсів.

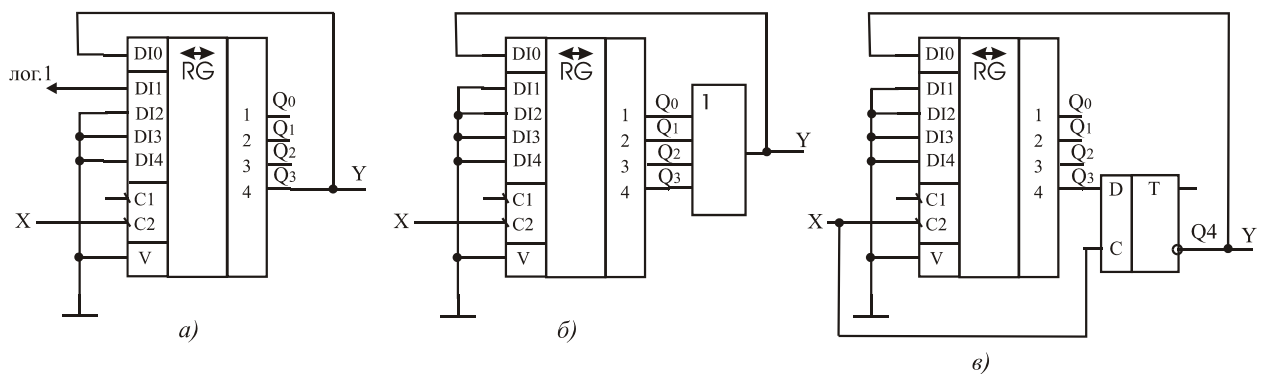


Рис. 4.18 Кільцеві лічильники (а,б), та лічильник Джонсона (в)

Для зменшення ймовірності появи збоїв, що викликані появою або зникненням одиниці або нуля у кільці, замість безпосереднього зворотного зв'язку можна використати комбінаційну схему, яка буде виконувати функцію дешифратора вихідного коду лічильника. У даному випадку (рис. 4.18.б) функцію дешифратора нуля виконує ЛЕ 4АБО-НЕ, на виході якого одиниця з'являється лише тоді, коли на всіх його входах присутні нулі. Такий кільцевий лічильник має $M_n=5$, бо на ньому на відміну від попереднього додатково фіксується ще код $\{0000\}$. Роль п'ятого тригера тут виконує ЛЕ 4АБО-НЕ.

Якщо замість 4АБО-НЕ застосувати ЛЕ 4І-НЕ, у кільці лічильника циркулюватиме не одиниця, а нуль, і 4І-НЕ виконуватиме роль дешифратора одиниці. Іноді такий лічильник зручніше застосовувати як розподілювач одиничних імпульсів, наприклад для генерування багатофазного тактового сигналу.

Щоб збільшити M_n кільцевого лічильника, потрібно мати регістр з більшою розрядністю або застосувати каскадування кількох регістрів.

Модуль лічби M_n кільцевого лічильника можна збільшити вдвічі, якщо між будь-якими двома тригерами застосувати перехресний зворотний зв'язок, з'єднавши інверсний вихід одного з тригерів із входом наступного тригера. Такий кільцевий лічильник перетворюється на так званий лічильник Джонсона. Зображений на рис. 4.18.в лічильник Джонсона на регістрі зсуву К155ІР1 доповнений D-тригером, що за інформаційним входом під'єднаний до виходу старшого розряду регістра, а за тактовим входом С– до тактового входу регістра. Інверсний вихід D-тригера з'єднаний з інформаційним входом регістра, за рахунок чого реалізується перехресний зворотний зв'язок. Тут роль п'ятого тригера лічильника виконує D-тригер.

Таблиця 4.10 Таблиця станів лічильника Джонсона

Q ₄	Q ₃	Q ₂	Q ₁	Q ₀	Такт
0	0	0	0	0	1
0	0	0	0	1	2
0	0	0	1	1	3
0	0	1	1	1	4
0	1	1	1	1	5
1	1	1	1	1	6
1	1	1	1	0	7
1	1	1	0	0	8
1	1	0	0	0	9
1	0	0	0	0	10

На відміну від звичайного кільцевого тригера лічильник Джонсона має модуль лічби $M_n=2n+2=10$ (у ньому додатково фіксуються стани {0000} і {1111}). Таблиця істинності лічильника Джонсона (табл. 4.10) з $M_n=10$ ілюструє потактове поширення спочатку "хвилі" одиниці, а потім "хвилі" нулів, завдяки тому, що спочатку $Q_4=1$, а потім $Q_4=0$. Оскільки на всіх виходах лічильника сигнали мають однаковий період повторення, що дорівнює $10T_0$, де T_0 - період повторення вхідних імпульсів, і до того ж зсунуті один щодо другого на T_0 , їх можна використати для генерування багатофазного сигналу.

На розглянутому чотирирозрядному лічильнику Джонсона можна реалізувати генерування восьмифазного сигналу шляхом подачі тактових імпульсів потрібної частоти. У загальному випадку n -розрядний кільцевий лічильник дозволяє генерувати імпульсні сигнали, що містять $2n$ фаз.

Для лічильника Джонсона також властиві збої у роботі, які проявляються у вигляді обривів "хвиль" одиниць або нулів; Для збільшення надійності роботи лічильника можна застосувати току коректуючу комбінаційну схему як дешифратор вихідного коду, що буде стежити за станом тригерів. Перевага лічильника Джонсона в тому, що стани 01 і 10 (табл. 4.10) двох сусідніх тригерів протягом одного циклу зустрічаються лише один раз незалежно від M_d . Отже, роль коректуючих дешифраторів станів можуть відігравати двохходові ЛЕ 2АБО-НЕ або 2І. Як видно з табл. 4.10, для коректування лічильника Джонсона можуть бути використані всі стани тригерів. Зокрема, на другому такті стан тригерів $\{00001\}$ відповідає мінтерму $\bar{Q}_4, \bar{Q}_3, \bar{Q}_2, \bar{Q}_1, Q_0$ або \bar{Q}_1, Q_0 , а на дев'ятому - $\{11000\}$ - мінтерму $Q_4, Q_3, \bar{Q}_2, \bar{Q}_1, \bar{Q}_0$ або Q_4, Q_3 . До переваг лічильника Джонсона ще можна віднести той факт, що під час лічби вхідних імпульсів тільки один із тригерів змінює свій стан, що запобігає появі хибних імпульсів - "голок".

На мікросхемах 564ИЕ19 і К561ИЕ19 реалізовані п'ятирозрядні лічильники Джонсона. Вони мають п'ять виходів, комутація яких із входом DS дозволяє будувати різні варіанти схем програмованих подільників частоти з коефіцієнтом ділення $K_d = 2 \dots 10$. Зокрема, на лічильнику К561ИЕ19 можна будувати подільник на парне число K_d без додаткових ЛЕ. Для ділення на 2, 4, 6, 8, 10 досить відповідні виходи лічильника $\bar{Q}_4, \bar{Q}_3, \bar{Q}_2, \bar{Q}_1, \bar{Q}_0$ з'єднати із входом D, а для ділення на непарне число потрібно застосувати ЛЕ 2І або 2АБО-НЕ, з'єднавши його з двома виходами лічильника і входом D.

4.6 Контрольні запитання по розділу

- В чому відмінність послідовнісних і комбінаційних пристроїв?
- За якими принципами розрізняють тригери? В чому їх відмінність і подібність?
- Які входи може мати тригер? В чому їх особливість?
- У чому особливість будови двоступеневих тригерів? Чим відрізняються одноступеневі тригери від двоступеневих?
- Для чого призначені регістри і яка між ними відмінність?
- Які є способи здійснення переносів у різних типів регістрів?
- Як здійснюється каскадування регістрів?
- Які параметри характеризують лічильник?
- За якими ознаками розрізняють лічильники? Які існують типи лічильників?
- Як перетворити підсумовувальний лічильник у віднімальний?
- Що таке лічильники з довільним модулем лічби?

5. ІНТЕГРАЛЬНІ ЗАПАМ'ЯТОВУВАЛЬНІ ПРИСТРОЇ

5.1 Загальні відомості

Запам'ятовувальні пристрої (ЗП) належать до одних з найважливіших функціональних пристроїв сучасних цифрових радіоелектронних систем. Вони служать для фіксування, зберігання та обміну певного об'єму інформації. Якщо раніше ЗП застосовувались виключно для ЕОМ, то тепер їх використовують як складові частини пристроїв радіолокації, телебачення, вимірювальної та побутової техніки тощо.

Фіксування цифрового сигналу в ЗП називається записом WR (від англ. Write), а відтворення його - зчитуванням або читанням RD (від англ. Read) інформації. Процеси запису і зчитування називаються процесами звертання до ЗП.

До найважливіших параметрів ЗП належать інформаційна місткість і швидкодія.

Місткість ЗП (або об'єм пам'яті) визначається максимально можливим обсягом збереженої інформації, який задається кількістю або біт, байт, або слів з вказанням їх розрядності. Отже, у першому випадку місткість ЗП вимірюється в бітах або в байтах як добуток числа запам'ятовувачів n на розрядність m слів nm , а в другому випадку у розкритій формі – $n \times m$. Наприклад, позначення ЗП у формі 64x8 біт означає, що ЗП місткістю 512 біт

(або 64 байт) здатна зберігати 64 слова по 8 розрядів. Збільшення об'єму пам'яті розширює функціональні можливості цифрових пристроїв. Одночасно із збільшенням об'єму пам'яті зростає час звертання до ЗП, тобто підвищуються вимоги до швидкодії.

Швидкодія ЗП характеризується періодом звертання (або часом циклу пам'яті) ТЗВ і часом вибірки (або часом доступу до пам'яті) t_B . Період звертання ТЗВ - мінімально припустимий час між двома черговими звертаннями до ЗП. Цей параметр залежить від характеристики і властивостей ЗП відновлюватися після попереднього звертання до нього. Час вибірки t_B - інтервал часу між моментом подачі сигналу вибірки до появи інформації на виході ЗП. Різновидами цього параметру є час доступу при зчитуванні та час доступу при запису. З появою ВІС ЗП високої інтеграції час вибірки t_B наближається до пікосекундного діапазону.

Решта параметрів та характеристик інтегральних ЗП: споживана потужність віднесена до місткості ЗП, навантажувальна здатність, завадостійкість, логічні рівні та надійність визначаються так само, як і для інших типів мікросхем ЦТ.

Основою будь-якого ЗП є запам'ятовувач, що являє собою елементарну комірку пам'яті, яка здатна зберігати, а при звертанні до неї - записувати чи зчитувати 1 біт інформації, тобто один розряд двійкового слова. Таким запам'ятовувачем, як правило, є тригер. Стани 0 і 1 можна зберігати також і за допомогою конденсатора з ключем, роль яких найчастіше відіграють польові транзистори. *Запам'ятовувачі на тригерах належать до статичних, а на конденсаторах - до динамічних.* Такий поділ зумовлений тим, що термін "динамічний" на відміну від терміну "статичний" підтверджує часові зміни, що викликані неминучими струмами витікання конденсатора, який після запису все ж розряджається, а при зчитуванні зарядженого стану частина заряду "розтікається" по шині зчитування. Тому динамічну пам'ять час від часу необхідно регенерувати, тобто періодично підзаряджати. За динамічним принципом будуються ЗП великого об'єму.

Для зберігання невеликого масиву двійкових слів можуть використовуватися регістри і навіть дешифратори. Але вже при необхідності запам'ятовувати десятки слів об'ємом у тисячі біт без спеціально для цього сконструйованих і організованих ЗП обійтись неможливо. Тому ЗП великих об'ємів будують об'єднанням окремих запам'ятовувачів, що нагадують бджолині соти.

Щоб суттєво скоротити число зовнішніх виводів таких ЗП, застосовують матрицевий принцип побудови. *Матриця ЗП* має всього дві шини - запису і зчитування, на перетинах яких знаходяться запам'ятовувачі. Оскільки звертання /запис або зчитування/ має здійснюватися тільки до одного запам'ятовувача, то для цього застосовують адресну вибірку, згідно з якою шукану комірку знаходять за номером стовпчика і рядка, тобто за адресою. Для зображення адрес застосовують комбінації m -розрядного двійкового коду. Отже, число запам'ятовувачів ЗП може складати $n=2^m$.

Таким чином, за способом зберігання інформації ВІС ЗП класифікують:

- на статичні ЗП. запам'ятовувачі яких являють собою тригери, фізичний стан яких під час звертання не руйнується;
- на динамічні ЗП. які побудовані на конденсаторних запам'ятовувачах, що вимагають періодичної регенерації за допомогою спеціальної схеми.

За організаційним принципом розробляють одно- $(n \times 1)$ та багаторозрядні $(n \times m)$ ВІС ЗП, де n, m - відповідно число адрес і розрядів слів, які може зберігати ЗП.

За функціональною ознакою інтегральні ЗП розподіляють на два класи; оперативні та постійні.

Оперативні ЗП (ОЗП, *RAM* від англ. *Random Access Memory* - пам'ять з довільною вибіркою) виконують операції звертання довільної двійкової інформації майже за однаковий час. ОЗП призначені для тимчасового зберігання біжучої інформації і швидкого звертання до запам'ятовувачів. Розрізняють енергозалежні ОЗП, які зберігають інформацію лише при наявності напруги живлення, і енергонезалежні. ОЗП умовно поділяють на три основних типи: середньої місткості помірної швидкодії, високої швидкодії - надоперативні ОЗП і ОЗП великої місткості.

Постійні ЗП (ПЗП, *ROM*- від англ. *Read Only Memory* – пам'ять тільки для читання) призначені для тривалого, зберігання і швидкого зчитування постійно записаної раніше інформації, зміст якої під час експлуатації ЗП не змінюється, а при відсутності живлення не руйнується. Отже, у ПЗП передбачені лише два режими роботи - зберігання і зчитування. На відміну від ПЗП перепрограмовані ПЗП (ППЗП) допускають відновлення занесеної інформації, тобто у них передбачається режим запису. Цей процес реалізують за допомогою спеціально призначених для цього пристроїв (програматорів).

За способом звертання до запам'ятовувачів ВІС ЗП розрізняють:

- на адресні ЗП з довільним звертанням і ЗП з послідовним звертанням, тобто з вибіркою при збільшенні або при зменшенні адресного коду;
- на асоціативні (безадресні) ЗП. пошук інформації у яких здійснюється незалежно від координат запам'ятовувачів, тобто не за адресою, а за певними ознаками самої інформації, що міститься у запам'ятовувачі ЗП.

За технологічним виконанням ВІС ЗП поділяють на біполярні, що використовують схемотехніку ТТЛ і ЕЗЛ, ЗП на основі МОН-структур /р-МОН, n-МОН, КМОН/ та інжекційну І²Л-технологію.

5.2 Оперативні запам'ятовуючі пристрої

Для побудови ОЗП або ЗП з довільною вибіркою інформації використовується практично вся відома на сьогодні схемотехніка - ТТЛ/Ш/ЕЗЛ, /К/МОН, І²Л тощо. ВІС ОЗП виконуються як статичні, так і динамічні.

5.2.1 Статичні запам'ятовувачі ВІС ОЗП

Це двостанові комірки пам'яті - тригери, які крім зберігання одиниці біт інформації (1 або 0), дозволяють здійснювати операцію звертання WR/RD. Вони є найбільш поширеними у практиці цифрової та мікропроцесорної техніки і застосовуються як регістри процесорів надоперативної та буферної пам'яті.

За технологічним виконанням статичні запам'ятовувачі розрізняють на біполярні та польові. Біполярні запам'ятовувачі, хоч і мають серед статичних ОЗП найвищу швидкодію ($t_B=5...50$ нс), однак порівняно з польовими (на ПТ) мають значно меншу інформаційну місткість і досить значну потужність споживання. Це пояснюється тим, що польові запам'ятовувачі займають на кристалі у кілька разів меншу площу, ніж біполярні, та й споживають менший струм.

На рис. 6.1 зображено схему статичного польового запам'ятовувача на базі *n*-МОП-структури. Транзистори VT1 і VT2 утворюють тригер, а транзистори VT3 і VT4 - двонапрямлені ключі вводу/виводу даних, тобто звертання WR/RD. Резистори навантаження R1 і R2 мають високі номінали опорів для забезпечення низького споживання у режимі зберігання.

Схема польового запам'ятовувача має три інформаційні виводи, від логічних станів (0 або 1) на яких залежить той чи інший режим роботи. Зокрема, це два виводи (станів Q і \bar{Q} тригера) шин розрядів (ШР) "1" і ШР "0" і вивід вибірки схеми - шини адреси (ША).

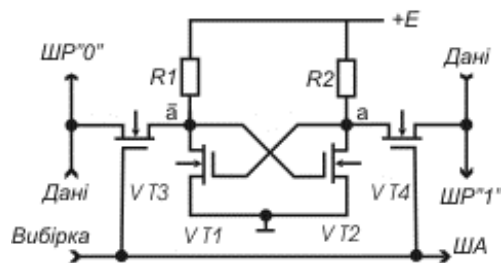


Рис. 5.1 Статична однобітова комірка пам'яті на польових транзисторах

У режимі запам'ятовування транзистори VT3 і VT4 закриті, а у режимі звертання WR/RD - відкриті. Режим запам'ятовування визначає низький потенціал на АШ, а звертання - високий. В останньому випадку, коли транзистори VT3 і VT4 відкриті і стають провідними в обох напрямках, можна виконувати як запис, так і зчитування інформації.

У режимі запису на ШР "1" і ШР "0" встановлюються високий або низький рівні сигналів, які відкривають транзистор VT1 або VT2, і тригер переходить в стан лог. 0 або лог. 1. При зчитуванні інформації із запам'ятовувача на ШР "1" і ШР "0" подається високий потенціал джерела живлення. У результаті цього розряджається паразитна ємність шини (ШР "1" або ШР "0"), яка зв'язана з відкритим транзистором (VT1 або VT2) тригера. У відкритому транзисторі потенціал на виході близький до нуля (наприклад, якщо VT1 відкритий, то

$\bar{Q}=0$, а $Q=1$, бо VT2 – закритий). Отже, при відповідній різниці потенціалів на розрядних шинах (ШР "1"=0, а ШР "0"=1 чи навпаки) дана інформація із запам'ятовувача буде зчитуватися на вихідні буферні підсилювачі шин.

Порівняно з іншими статичні запам'ятовувачі на КМОН-транзисторах споживають найменше потужності. Наприклад, потужність розсіювання у КМОН ВІС ОЗП КР537РУ2А, місткість якого 4К, складає всього 0,5мВт у режимі зберігання і 20мВт у режимі перемикання (звертання). Статичні ВІС ОЗП середньої швидкодії, що виконані на n-МОН-, КМОН- та I²L-технологіях, мають період циклу звертання до запам'ятовувачів від сотень наносекунд до одиниць мікросекунд. Перспективними ВІС ОЗП є, однак, ЗП на базі арсеніду галію. Потужність розсіювання у режимі звертання у таких ВІС складає не більше 10мкВт/біт, а час вибірки не перевищує 10нс.

У структурі ВІС ОЗП статичні запам'ятовувачі компонується у прямокутну матрицю розміром $p \times m$ (де p - число m -розрядних слів), яку називають накопичувачем ЗП. Кожний рядок накопичувача – це адресна шина вибірки, а стовпець – дві розрядні шини звертання ШР "1" і ШР "0" запам'ятовувача.

5.2.2 Динамічні запам'ятовувачі ВІС ОЗП

Їх також виконують на біполярних та польових транзисторах. Відмінність побудови схем динамічних запам'ятовувачів полягає у способі запам'ятовування двійкової інформації, який реалізується зарядом або розрядом паразитної ємності. Схему динамічного запам'ятовувача виконують на базі одного або трьох транзисторів. На відміну від статичного запам'ятовувача динамічний значно простіший і економічніший, бо вимагає вдвічі меншого числа транзисторів, і, отже, на однакових розмірах кристалу дозволяє вдвоє збільшити інформаційну місткість. Тому динамічні ОЗП дають змогу будувати ВІС надвеликих місткостей. Перевага динамічних запам'ятовувачів ще в тому, що їм не потрібне джерело живлення.

Оскільки значення паразитної ємності, як правило, не перевищує 0,1пФ, стала часу розряду накопичувального конденсатора є досить значною - не менше 1нс. Тому, щоб розрізнити одиницю інформації /0 або 1/, яка зберігається у вигляді заряду, необхідно її відновлювати, тобто регенерувати, причому з періодом регенерації не менше від 1нс. Це ускладнює схему запам'ятовувача і організацію структури ВІС ОЗП у цілому.

На рис. 5.2.а,б показано схеми динамічних одно- та тритранзисторного запам'ятовувачів ОЗП, які найбільш поширені у ВІС ОЗП великої місткості /> 16К/. В обох схемах запам'ятовувачів зберігання двійкової інформації здійснюється на конденсаторі С, а транзистор VT відіграє роль перемикача, що передає заряд конденсатора С у розрядну шину при зчитуванні або розряджає його при записі інформації.

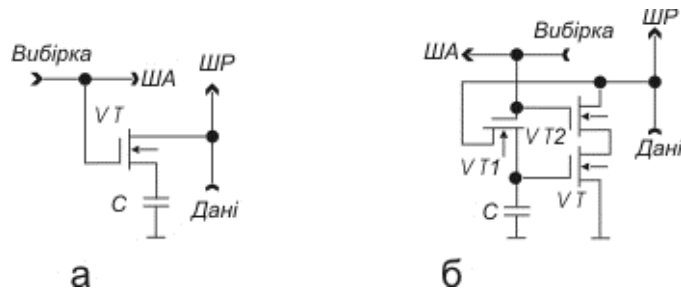


Рис. 5.2 Схеми динамічних запам'ятовувачів

Спільним для наведених тут схем на базі n -МОН-транзисторів динамічних запам'ятовувачів є також те, що для отримання можливості зчитувати інформацію конденсатор C попередньо заряджають, імпульсами високого рівня, які подають на ШР при активізації вибірки, тобто при подачі високого рівня на ША.

Отже, записування інформації в одотранзисторному запам'ятовувачі відбувається подачею високого потенціалу на ШР на відкритий транзистор, а зчитування - струмом розряду конденсатора в ШР з руйнуванням інформації. У тритранзисторного запам'ятовувача накопичувальний конденсатор ізольований від ШР, і тому зчитування інформації відбувається без її руйнування. Проте через струми витікання у конденсаторі потрібна періодична регенерація інформації. Цей процес досягається подачею високого потенціалу на ША і спеціальною схемою регенерації, яку під'єднують до ШР.

Регенерація динамічного запам'ятовувача ОЗП здійснюється за кожним звертанням до нього. Схема регенератора-підсилювача автоматично виконує цей процес при звертанні до стовпця накопичувача з інтервалами близько сотні мікросекунд. Динамічні ОЗП побудовані так, що сам процес звертання до стовпця забезпечує регенерацію інформації в усіх його запам'ятовувачах.

5.2.3 Принцип побудови і структура ВІС ОЗП

Вони визначаються основним вузлом ЗП – $n \times m$ -матрицею запам'ятовувачів, яку називають накопичувачем інформації. Відомо кілька способів організації запам'ятовувачів у накопичувачі. Залежно від способу звертання WR/RD до запам'ятовувачів розрізняють два найбільш поширених типи організації накопичувачів:

- з однокоординатною (або послівною) вибіркою;
- з двокоординатною вибіркою.

При однокоординатній вибірці реалізується пошук групи запам'ятовувачів за одною адресою, тобто однією шиною ША вибирається тільки одне з усіх n m -розрядне слово. Як показано на рис. 5.3.а, такий накопичувач ОЗП має адресні шини ША1, ША2, ..., ША n , що з'єднані з кожним запам'ятовувачем Z_{ij} одноіменного i -го слова, і розрядні шини ШР1, ШР2, ..., ШР m , які з'єднані з одноіменним j - m розрядом усіх її. слів. Крім того, існує ще спільна для всіх

$n \times m$. запам'ятовувачів (Z_{nm}) шина звертання WR/RD, сигнал на якій визначає режим звертання - запису або зчитування. Отже, при появі на одній з адресних шин ША_i сигналу вибірки і -го слова, наприклад при активному високому рівні. Інформацію з кожного запам'ятовувача $Z_{i1}, Z_{i2}, \dots, Z_{im}$ i -го рядка (тобто i -го слова) можна зчитувати шляхом активізації розрядних шин ШР1, ШР2, ..., ШР_m.

Для запису слова, наприклад, у комірку запам'ятовувачів $Z_{21}, Z_{22}, \dots, Z_{2m}$, слід активізувати другий рядок накопичувача, тобто адресну шину ША2, а на всі розрядні шини ШР1, ШР2, ..., ШР_m - прикладати значення розрядів (0 або 1) записуваного слова. При цьому на шину WR/RD треба подати сигнал дозволу на запис інформації. Після зчитування записаної інформації на вході розрядних буферів (вихідних підсилювачів ОЗП) з'являються рівні напруг, що відповідають значенням розрядів (0 або 1) зчитаного слова.

Розглянутий тип організації накопичувача інформації має лише одну координату звертання до запам'ятовувачів, а саме - номер рядка накопичувача. Тому для однокоординатного накопичувача досить одного дешифратора адреси. Накопичувач з однокоординатною вибіркою називають ще двовимірним - типу 2D, бо запам'ятовувачі у ньому розташовані на площині.

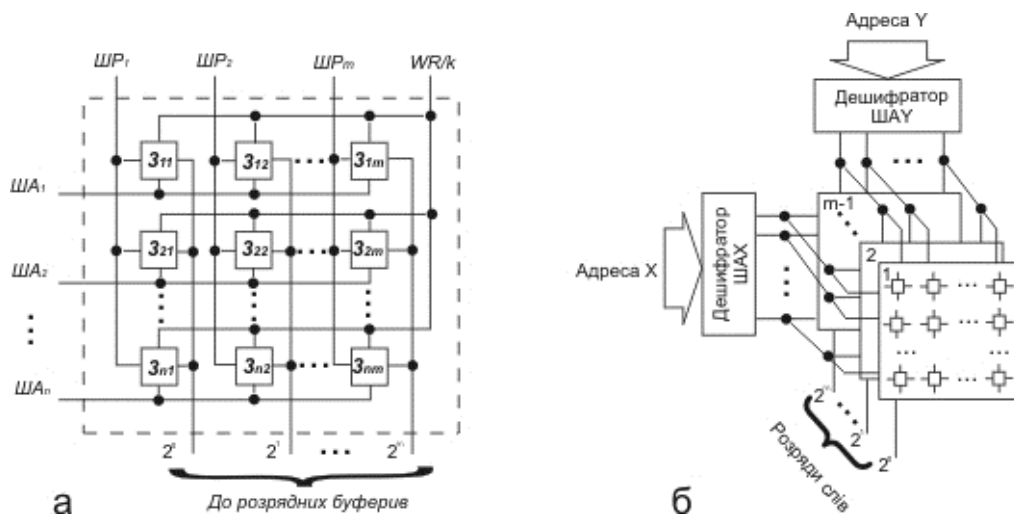


Рис. 5.3 Структурні схеми накопичувачів інформації

У структурній схемі накопичувача з двокоординатною вибіркою (рис. 5.3,б) потрібну комірку запам'ятовувача шукають за допомогою двох адресних шин, - горизонтальних (по рядках) ШАХ і вертикальних (по стовпцях) ШАУ, і розрядної шини ШР. Накопичувач за такою структурою, має тривимірну будову і складається з t накопичувачів розрядів, кожний з яких містить n запам'ятовувачів. Отже, кожний накопичувач відповідає тільки за один одноіменний розряд всіх n слів, а кожне слово записується, зберігається і зчитується в усіх t накопичувачах за ідентичною двокоординатною адресою X_i, Y_i .

Для запису слова у запам'ятовувач за адресою X_i, Y_i . на входи адресних дешифраторів ШАХ і ШАУ подаються відповідні коди X_i, Y_i , а двійкове

значення слова - на спільні розрядні шини ШР1, ШР2, ..., ШРm. Аналогічно здійснюється зчитування записаного слова.

5.3 Принцип побудови і структура ПЗП

Порівняно з ОЗП ВІС пап мають простішу організацію пам'яті та схему керування процесом зчитування інформації. Загальною ознакою всіх ПЗП є також наявність двокоординатної матриці-накопичувача, що являє собою однорідне за технологією "поле" запам'ятовувачів.

ВІС ПЗП дуже нагадують ПЛМ, бо також складаються з матриць; тільки у ПЗП програмують не обидві матриці (кон'юнкцій та диз'юнкцій), а лише матрицю диз'юнкцій. Матриця кон'юнкцій у ній відіграє роль "жорсткого" повного дешифратора всіх 2^n вихідних від n вхідних комбінацій. Тому ПЗП має більше ніж у ПЛМ число входів – рівно стільки, скільки потрібно для забезпечення повної таблиці істинності всіх n адрес. Незважаючи на відмінність у сфері застосування ПЛМ за спільністю ознак часто відносять до класу ЗП як особливий вид ПЗП.

На рис. 5.4 показана узагальнена структурна схема ВІС ПЗП, основою якої є накопичувач інформації, що з'єднаний через ША з дешифратором адрес (ДСА), а через ШР - з блоком мультиплексорів (MUX) Паралельний код адреси A подається на формувач адрес, парафазні сигнали якого надходять на ДСА і блок MUX, які збуджують одну з горизонтальних (адресних) шин ША. Зчитування записаної у запам'ятовувачах інформації відбувається по всіх вертикальних (розрядних) шин ШР через блок мультиплексорів і буфер вводу/виводу. Схема керування синхронізує роботу ДСА і блок MUX, а також служить для забезпечення можливості нарощування об'єму пам'яті об'єднанням виходів (як монтажне АБО для схем з відкритим колектором) кількох ВІС ПЗП або для забезпечення їх роботи на спільну шину (для випадку тристанових виходів).

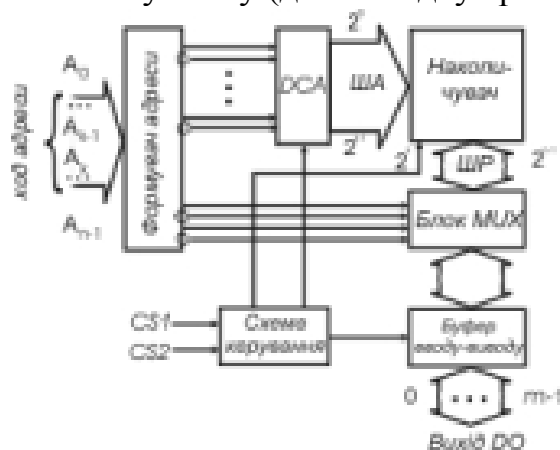


Рис. 5.4 Узагальнена структурна схема ВІС ПЗП

У ВІС РПЗП схема керування, крім того, керує режимами запису, зчитування та стирання інформації. На відміну від ОЗП при зчитуванні

накопичувача ПЗП задається вміст цілого рядка запам'ятовувачів, причому такий рядок може містити навіть кілька слів. З вибраного дешифратором рядка виділяється і передається на вихід потрібне слово за допомогою мультиплексорів.

Для реалізації простого МПЗП досить використати дешифратор і потрібну для цього певну кількість діодів для побудови матриці-накопичувача. Такий МПЗП зображений на рис. 5.5. Він має місткість $2^5 = 32$ біт, які розбиті на $2^2 = 4$ слова по $2^3 = 8$ розрядів у кожному. Отже, це ПЗП здатне запам'ятовувати чотири однобайтних слова ($4 \times 8 = 32$ біт).

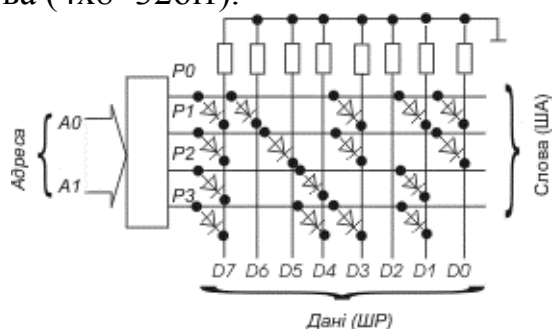


Рис. 5.5 Діодний накопичувач інформації

У координатному полі накопичувача діоди, що відіграють роль запам'ятовувачів, розміщуються у тих “точках”, де повинні зберігатися біти, що мають значення лот. 1. Таким чином, для вибору потрібного слова треба за допомогою дешифратора 2-4 задати адресу $\{A1A0\}$, тобто активізувати один з виходів дешифратора P_i , що рівнозначно появі одиниці на обраній ША. Тоді на тих ШР, на перетинах яких з обраною ША присутній діод, формується одиниця, а на решті ШР - нуль.

Діодні накопичувачі, однак, не знайшли широкого застосування для побудови ВІС МПЗП через їх низьку швидкодію. Найбільшу швидкодію забезпечують біполярні накопичувачі, час звертання у яких < 20 нс. Хоч МОН-накопичувачі мають нижчу швидкодію (час звертання - близько 200...600 нс), однак потужність розсіювання їх значно нижча. Тому у ВІС МПЗП можна зустріти накопичувачі як біполярні, так і польові. Транзистори, що виконують роль запам'ятовувачів, у цих ВІС під'єднують до рядків і стовпців накопичувача так, як це показано на рис. 5.5. Так само, як і у діодного накопичувача, наявність або відсутність транзистора у “точці” перетину ортогональних ліній також відповідає стану 1 або 0 запам'ятовувача.

Структура ВІС ПЗП аналогічна структурі ВІС МПЗП. Подібно до ППЛМ запам'ятовувачі ПЗП найчастіше виготовляються разом з перемичками, які у процесі програмування при вибірці відповідного запам'ятовувача перепалюються електричним струмом достатньої напруги (до 10...12 В). Матеріали для перемичок - титано-вольфрамові, ніхромові та інші сплави. Інший тип запам'ятовувача передбачає закорочування зворотно ввімкнених діодів.

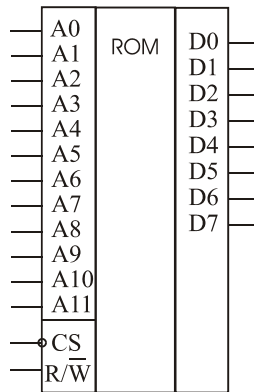


Рис. 5.6 Типове позначення ПЗП

Для програмування ПЗП необхідно насамперед скласти карту програмування, яка по суті є таблицею істинності, де кожній комбінації повного набору n адрес $A_0 \dots A_{(n-1)}$ відповідає код даних $D_0 \dots D_{(n-1)}$, який потрібно “зашифрувати”. Режими програмування у різних мікросхем ПЗП різні, бо визначаються вони не лише технологією виготовлення даної ВІС, але й типом мікросхеми ПЗП (з відкритим колектором чи тристановим виходом тощо) та матеріалом самої перемички. Для перепалювання останніх використовуються спеціально призначений для цього пристрій -програмактор, який має забезпечувати потрібний режим програмування ПЗП (тривалість імпульсу, його амплітуду і струм перепалювання) з допомогою одного імпульсу або пачки імпульсів.

Можливість програмувати робить ПЗП і особливо РПЗП універсальними пристроями ЦТ. У РПЗП можна багаторазово стирати записану інформацію і записувати нову. Тому для виявлення і виправлення помилок РПЗП на відміну від ПЗП є більш ефективними. У накопичувачах РПЗП використовуються спеціальні типи транзисторних структур, що побудовані, зокрема, на основі МОН-транзисторів з плаваючим (ізолюваним) заслоном, які змінюють під час програмування свої характеристики. Ці зміни є ознакою стану збереженої у ПЗП інформації.

Програмування накопичувачів РПЗП з ультрафіолетовим стиранням виконуються у два етапи - спочатку під дією інтенсивного ультрафіолетового випромінювання всі транзистори закриваються, що відповідає запису в усіх запам’ятовувачах лог. 1 (попереднє стирання), а потім за допомогою ША і ШР вибираються ті транзистори запам’ятовувачів, в які потрібно занести лог. 0 (запис інформації). Для цього на адресний вхід РПЗП подається код адреси слова, а на спеціальний вхід (CS-імпульс достатньої для перепалювання перемички напруги і потужності. При записі інформації у РПЗП часто застосовується програмактор, керований мікропроцесором. На рис.5.6 показані умовні графічне позначення ПЗП. Відмінності для різних типів ПЗП полягають в умовних скороченнях: для МПЗП це "ROM" (див. рис.5.6), для ПЗП – "PROM" для РПЗП– "RPROM".

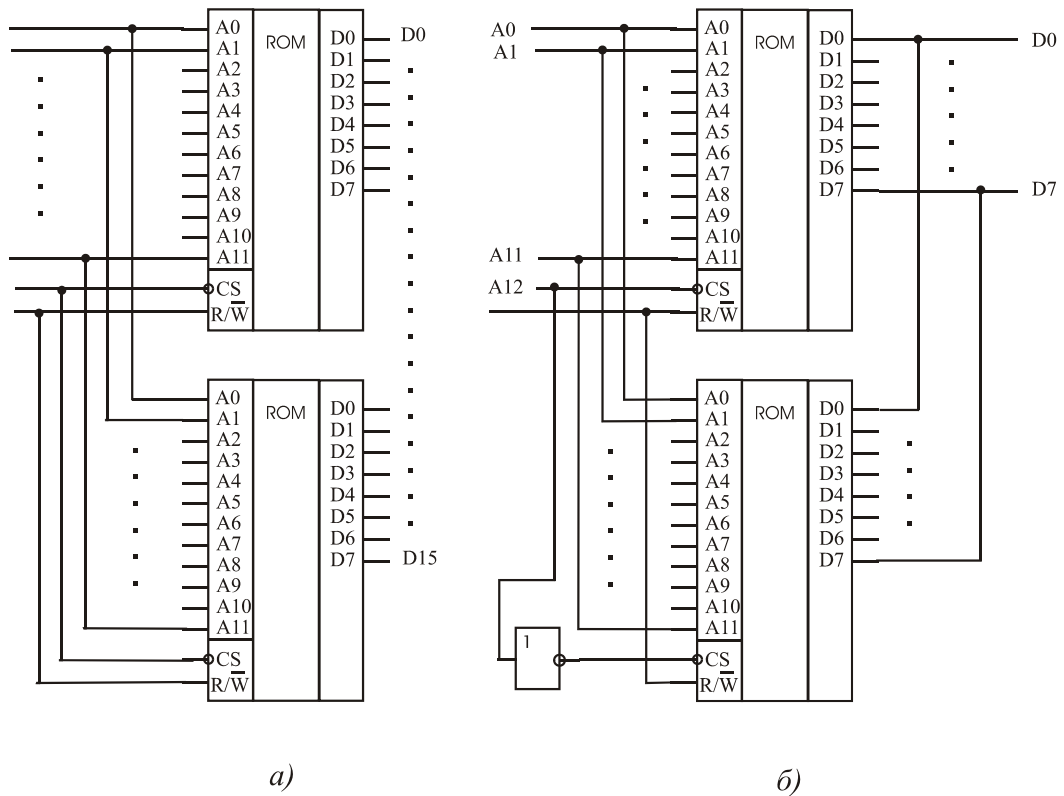


Рис. 5.7 Способи збільшення розрядності даних ПЗП (а) та адрес (б)

Розширити функціональні можливості ПЗП різних типів можна за рахунок каскадування, тобто збільшення, їх інформаційної місткості. Процедуру нарощення ПЗП можна здійснити як для збільшення розрядності m слів, так і для збільшення числа n слів. Це рівнозначно збільшенню певного числа відповідно виходів або входів, яких не вистачає у даному ПЗП для реалізації поставленого завдання. Зрозуміло, що для цієї процедури потрібно мати необхідну кількість однотипних мікросхем ПЗП, а також виконати відповідну зовнішню комутацію виводів даного ПЗП з виводами інших ПЗП. Збільшити розрядності слів (число виходів) ПЗП досить просто - потрібну кількість мікросхем ПЗП вмикають паралельно по одноіменних входах так, як це показано на рад. 5.7.а. При цьому число виходів даних ПЗП від тих самих аргументів (адрес) стане більшим, що рівнозначно збільшенню розрядності. Для збільшення числа збережуваних слів даних застосовують вхід CS (вибір мікросхеми) (рис. 6.12,б). При $A_n=0$ адресується верхній модуль ПЗП, а при $A_n=1$ –нижній. Виходи обох модулів зв'язуються за схемою монтажного АБО.

5.4 Електрично перепрограмовувані ПЗП

В останній час особливого поширення набули електрично-перепрограмовувані мікросхеми EEPROM, які є подальшим розвитком вищеописаних ПЗП. Вони дозволяють здійснювати власний перезапис не тільки за допомогою програматорів, але й внутрішньосистемно – безпосередньо на платі цифрового пристрою. При цьому існують ІС паралельного типу, які функціонально нічим не відрізняються від розглянутих вище ПЗП і способи роботи з ними аналогічні до вищевикладених (за винятком програмування) та послідовного, які мають незначну кількість виводів (як правило 8) і обмін даними з ними відбувається за допомогою одного з послідовних протоколів (найчастіше MicroWire, I²C або SPI). Приклад ІС послідовного типу 2404. Такі ІС мають від 10³ до 10⁵ гарантованих циклів перезапису, що є набагато кращим показником у порівнянні з ПЗП, які стираються за допомогою ультрафіолетового випромінювання (до 100 разів). Послідовні ПЗП найчастіше використовуються для запам'ятовування допоміжної інформації в приладах перед переходом у вимкнений стан (різноманітні коди доступу, позиції регуляторів управління, тощо). Паралельні ПЗП використовуються для запису програм та значних масивів даних в мікропроцесорних пристроях. Оскільки алгоритми програмування таких ІС є доволі громіздкими вони будуть розглядатись в подальшому.
