

**ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КИЇВСЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Кафедра інформаційних систем та технологій

**В.А. ДРУЖИНІН
С.В. ТОЛЮПА
Ю.М. БОЙКО**

ОСНОВИ СХЕМОТЕХНІКИ

Навчальний посібник

**для здобувачів освітнього ступеня «бакалавр» спеціальності
126 «Інформаційні системи та технології»
освітня програма «Програмні технології інтернет речей»**

Київ 2024

Рецензенти:

*д.т.н., доцент, доцент кафедри мережевих та інтернет технологій
Київського національного університету ім. Тараса Шевченка Плющ
Олександр Григорович*

*к.т.н., доцент, доцент кафедри інформаційних систем та технологій КПІ
ім. Ігоря Сікорського Цьопа Наталія Володимирівна;*

*к.т.н., доцент, доцент кафедри радіотехніки та радіоелектронних систем
КНУ ім. Тараса Шевченка Жиров Геннадій Борисович.*

Рекомендовано до публікації кафедрою інформаційних систем та технологій,
протокол №17_23/24 від «29 » квітня 2024р.

Затверджено Вченою радою ФІТ Київського національного університету
імені Т. Шевченка, протокол №3 від 11.11.2024р.

В.А. Дружинін, С.В. Толюпа, Ю.М. Бойко

Основи схемотехніки. Курс лекцій Електронний ресурс]: навчальний посібник для здобувачів освітнього ступеня «бакалавр» спеціальності 126 «Інформаційні системи та технології» освітня програма «Програмні технології інтернет речей» уклад.: В.А. Дружинін, С.В. Толюпа, Ю.М. Бойко – К.: КНУ імені Т. Шевченка, 2024.– 155 с.

Навчальний посібник з дисципліни «**Основи схемотехніки. Курс лекцій**».

Викладено основи схемотехніки цифрових елементів і пристроїв: системи числення, формати та коди чисел, елементи комп'ютерної логіки, синтез цифрових автоматів, принципи роботи базових логічних елементів, цифрових вузлів комбінаційного та накопичувального типу, їх позначення, параметри та характеристики, реалізація на їх основі сучасних мікросхем

Електронна версія цього видання опублікована на сайті кафедри інформаційних систем та технологій факультету інформаційних технологій Київського національного університету імені Тараса Шевченка

© **В.А. Дружинін, С.В. Толюпа, Ю.М. Бойко 2024**

ЗМІСТ

СПИСОК СКОРОЧЕНЬ	5
ВСТУП	6
РОЗДІЛ 1. МАТЕМАТИЧНІ ТА ЛОГІЧНІ ОСНОВИ ПОБУДОВИ ЦИФРОВИХ ПРИСТРОЇВ	7
1.1 Системи числення та кодування чисел.....	7
1.1.1. Системи числення.....	7
1.1.2 Правила переведення чисел у задану систему числення.....	9
1.1.3. Формати цілих та дійсних чисел.....	11
1.1.4. Кодування чисел.....	13
1.1.5. Арифметика операцій з кодами.....	14
1.2 Основи алгебри логіки та логічні функції.....	15
1.2.1 Основні логічні функції алгебри логіки.....	15
1.2.2 Функції перемикання.....	17
1.3 Мінімізація функцій перемикання.....	23
1.3.1. Алгебраїчний метод мінімізації.....	23
1.3.2. Мінімізація булевих функцій за допомогою карт мінтермів.....	24
1.3.3. Інтегральні цифрові мікросхеми логічних елементів.....	27
1.3.4. Синтез схеми ЦП за заданою логічною функцією.....	28
1.3.5 Методи спрощення логічних функцій.....	29
Список питань для самоконтролю	31
РОЗДІЛ 2. ЛОГІЧНІ ЕЛЕМЕНТИ ЦИФРОВОЇ ТЕХНІКИ	31
2.1 Поняття про комбінаційну схему та цифровий автомат.....	31
2.2 Класифікація та характеристики елементної бази цифрових пристроїв.....	33
2.3 Діодні логічні елементи (ДЛ).....	38
2.4 Діодно-транзисторні логічні елементи (ДТЛ).....	39
2.5 Транзисторно-транзисторні логічні елементи (ТТЛ).....	42
2.5.1 Схема та принцип дії ТТЛ-елемента.....	42
2.5.2 ТТЛ-елементи з відкритим колектором.....	44
2.5.3 ТТЛ-елементи на транзисторах Шотткі.....	44
2.5.4. ТТЛ елементи з трьома вихідними станами.....	45
2.5.5. Особливості практичних схем ТТЛ-елементів.....	46
2.6 Емітерно-зв'язані логічні елементи (ЕЗЛ).....	48
2.6.1. Спрощена схема та принцип дії ЕЗЛ-елементів.....	48
2.6.2. Особливості реальних схем ЕЗЛ-елементів.....	50
2.6.3. Поєднання ЕЗЛ-елементів з ТТЛ-елементами.....	51
2.7 Логічні елементи на польових транзисторах.....	52
2.7.1. Логічні елементи на n-каналних МДН-транзисторах.....	52
2.7.2. Логічні елементи на комплементарних парах МДН-транзисторів.....	54
2.7.3. КМДН-логічний елемент з трьома вихідними станами.....	57
2.7.4. Особливості схем ЛЕ на польових транзисторах та узгодження їх з ТТЛ-елементами.....	57
2.7.5. Поняття про логічні елементи з інжекційним живленням.....	59
Список питань для самоконтролю	60
РОЗДІЛ 3. ЦИФРОВІ ІНТЕГРАЛЬНІ ПРИСТРОЇ КОМБІНАЦІЙНОГО ТИПУ	61
3.1 Загальні відомості про комбінаційні пристрої.....	61
3.2 Суматори.....	62
3.2.1 Принцип побудови однорозрядного суматора.....	62
3.2.2. Принцип побудови багаторозрядного суматора.....	64
3.3. Цифрові компаратори.....	66
3.4. Дешифратори.....	69
3.5. Шифратори.....	72
3.6. Перетворювач двійково-десятькового коду у код семисегментного індикатора.....	73
3.7 Мультиплексори та демультиплексори.....	73
3.7.1. Загальна характеристика мультиплексорів.....	73
3.7.2. Реалізація логічних функцій мультиплексорами.....	75
3.7.3. Мультиплексування шин.....	76
3.7.4. Загальна характеристика демультиплексорів.....	76
3.7.5. Демультиплексування шин.....	77
Список питань для самоконтролю	78
РОЗДІЛ 4. ТИПОВІ ЦИФРОВІ ВУЗЛИ З ПАМ'ЯТТЮ	79
4.1. Загальні відомості про цифрові пристрої накопичувального типу.....	79
4.2 Тригери та їхні характеристики.....	80
4.3 Регістри.....	86
4.3.1. Принцип роботи регістрів пам'яті.....	87
4.3.2 Регістри зсуву.....	88
4.4. Лічильники.....	90

4.4.1. Призначення, класифікація та характеристики лічильників.....	90
4.4.2. Лічильники з послідовним переносом.....	91
4.4.3. Лічильники з наскрізним переносом.....	93
4.4.4. Лічильники з паралельним переносом.....	94
4.4.5. Лічильники з коефіцієнтом перерахування, відмінним от 2^n	95
4.5. Подільники.....	96
4.6. Генератори і формувачі імпульсів.....	98
Список питань для самоконтролю.....	100
РОЗДІЛ 5. ЗАПАМ'ЯТОВУВАЛЬНІ ПРИСТРОЇ.....	101
5.1. Типи запам'ятовувальних пристроїв.....	101
5.2. Основні параметри запам'ятовувальних пристроїв.....	102
5.3. Оперативні запам'ятовувальні пристрої.....	103
5.4 Постійні запам'ятовуючі пристрої.....	106
5.5. Перепрограмовані постійні запам'ятовувальні пристрої.....	109
Список питань для самоконтролю.....	110
РОЗДІЛ 6. СИСТЕМИ ВВЕДЕННЯ ТА ВІДОБРАЖЕННЯ ЦИФРОВОЇ ІНФОРМАЦІЇ.....	111
6.1. Пристрої введення цифрової інформації.....	111
6.2. Елементна база світлових індикаторних пристроїв.....	113
6.3. Пристрої індикації.....	115
Список питань для самоконтролю.....	116
РОЗДІЛ 7. ЦИФРО-АНАЛОГОВИ ПЕРЕТВОРЮВАЧІ.....	117
7.1. Загальні відомості.....	117
7.2. Основні параметри та класифікація ЦАП.....	118
7.2.1. Статичні та динамічні параметри ЦАП.....	118
7.3. ПКН на матриці резисторів з ваговими резисторами.....	120
7.4. ПКН на матрице резисторів R-2R.....	121
7.5. Перетворювачі коду в часовий інтервал.....	122
7.5.1 ПКВ на основі двійкового лічильника.....	123
7.5.2. ПКВ на основі лічильника та цифрового компаратора.....	123
7.6 Особливості застосування ЦАП.....	124
Список питань для самоконтролю.....	128
РОЗДІЛ 8. АНАЛОГОВО-ЦИФРОВІ ПЕРЕТВОРЮВАЧІ.....	129
8.1. Основні параметри та класифікація АЦП.....	129
8.2. Перетворювачі напруги в код послідовного рахунку.....	131
8.2.1. ПНК з ступінчастою розгортаючою напругою.....	131
8.2.2. ПНК слідкуючого типу.....	132
8.3. Перетворювачі напруги в код порозрядного кодування.....	133
8.4. АЦП паралельного типу.....	135
Список питань для самоконтролю.....	138
РОЗДІЛ 9. ОСНОВИ МІКРОПРОЦЕСОРНОЇ ТЕХНІКИ.....	139
9.1 Класифікація процесорних пристроїв.....	139
9.2. Мікропроцесори у вимірювальних приладах.....	141
9.3. Поліпшення метрологічних характеристик приладів.....	144
9.4. Процесорні похибки вимірювань.....	146
9.5. Архітектура процесорів.....	147
9.6. Типи пам'яті мікроконтролерів.....	150
9.7. Регістри мікроконтролера. Простір введення-виведення.....	153
9.8. Зовнішня пам'ять.....	154
Список питань для самоконтролю.....	154
РЕКОМЕНДОВАНА ЛІТЕРАТУРА.....	155

СПИСОК СКОРОЧЕНЬ

АЦП - аналого-цифровий перетворювач
ВАХ - вольт-амперна характеристика
ВІС - велика інтегральна схема
ВК - вузол керування
ДДНФ - досконала диз'юнктивна нормальна форма
ДЗКД - дільник зі змінним коефіцієнтом ділення
ДКНФ - досконала кон'юнктивна нормальна форма
ДЛ - діодна логіка
ДНФ - диз'юнктивна нормальна форма
ДТЛ - діодно-транзисторна логіка
ДШ - дешифратор
ЕЗЛ - емітерно-зв'язана логіка
ЕП - елемент пам'яті
ЕРС - електрорушійна сила
ІЖЛ - логіка з інжекційним живленням
ІЛ - інтегральна інжекційна логіка
ІМС - інтегральна мікросхема
ІШ - інформаційна шина
КМДН - комплементарний МДН - транзистор
КНФ - кон'юнктивна нормальна форма
КП - комірка пам'яті
ЛЕ - логічний елемент
МДН - метал-діелектрик-напівпровідник
МДНЛ - логіка на МДН - транзисторах
МР - молодший розряд
МП - мікропроцесор
НЗЗ - негативний зворотний зв'язок
ОП - операційний підсилювач
ОШ - нульова шина
ПВЗ - пристрій вибору та збереження
ПЕОМ - персональна електронно-обчислювальна машина
ПЗЗ - позитивний зворотний зв'язок
ПЗп - підсилювач запису
ПЗч - підсилювач зчитування
ПК - пристрій комутації
ПОП - пристрій оперативної пам'яті
ПП - пристрій пам'яті
ППП - пристрій постійної пам'яті
ПСТЛ - перемикач струму транзисторної логіки
РВВ - регістр уведення-виведення
РПП - репрограмована постійна пам'ять
РШ - розрядна шина
СР - старший розряд
СШ - струмова сигнальна шина
ТІ - тактовий імпульс
ТМП - тонка магнітна плівка
ТТЛ - транзисторно-транзисторна логіка
ТТЛШ - ТТЛ на діодах Шотткі
УГП – умовно графічне позначення
ЦА - цифровий автомат
ЦАП - цифро-аналоговий перетворювач
ЦВ - цифровий вузол

ВСТУП

Інформатизація суспільства – важливе завдання, що потребує інтенсивного розвитку обчислювальної техніки та інших засобів обробки інформації. Усі різноманітні пристрої цифрової техніки: ЕОМ, мікропроцесорні системи вимірювань та автоматизації технологічних процесів, цифровий зв'язок і телебачення, та ін., створюються на загальній елементній базі, до складу якої входять різні за складністю мікросхеми – від логічних елементів, які виконують найпростіші операції, до найскладніших програмованих кристалів, що містять мільйони логічних елементів.

Цифрова техніка і мікроелектроніка є основою сучасної обчислювальної та керуючої техніки, тому широко використовують цілу низку нових класів електронних пристроїв – мікропроцесорів, мікроконтролерів, мікроконвертерів, сигнальних процесорів. Реалізація сучасних технологій передавання інформації неможлива без застосування цифрових систем, робота яких заснована на процесах формування, передавання, приймання та обробці цифрових сигналів.

Комп'ютери як цифрові пристрої входять до складу систем передавання інформації, управління технологічними процесами, автоматизації, медичних технічних комплексів наукових досліджень. Сучасні інформаційно-телекомунікаційні системи, які входять до інтегральних систем передавання інформації, застосовують цифрові сигнали, формування та обробка яких здійснюються цифровою технікою. Схемотехнічні рішення функціональних вузлів апаратури реалізуються на цифрових інтегральних мікросхемах. Вони мають широкі функціональні можливості, які дозволяють проектувати електронну апаратуру з високими технічними показниками.

В даному навчальному посібнику розглянуті теоретичні основи побудови цифрових елементів – системи числення, формати цілих та дійсних чисел, кодування чисел, арифметичні операції з кодами, алгебра логіки та її закони, булеві функції та їх властивості, способи мінімізації булевих функцій, а також принципи аналізу та синтезу цифрових елементів, вузлів та пристроїв на базі логічних елементів. Розглянути цифрових вузлів комбінаційного типу, принципи побудови та схемотехнічні реалізації суматорів, компараторів, дешифраторів, шифраторів, мультиплексорів та демультимплексорів, даються рекомендації по каскадуванню та практичному використанню. Розглянута робота цифрових вузлів послідовнісного типу: тригери, лічильники, подільники, регістри, елементи індикації цифрової інформації, цифрових пристроїв пам'яті, аналого-цифрових та цифро-аналогових перетворювачів.

РОЗДІЛІ. МАТЕМАТИЧНІ ТА ЛОГІЧНІ ОСНОВИ ПОБУДОВИ ЦИФРОВИХ ПРИСТРОЇВ

Під інформацією у широкому розумінні прийнято розуміти різноманітні відомості про події в суспільному житті, явищах природи, про процеси в технічних приладах. Вона знаходиться в нашій мові, в текстах книг і газет, в показаннях вимірювальних приладів і відображає різноманітність, притаманну об'єктам та явищам реального світу. Інформацію, втілену і зафіксовану в деякій матеріальній формі, називають повідомленням і передають за допомогою сигналів. Природа більшості фізичних величин така, що вони можуть приймати будь-які значення в будь-якому діапазоні (температура, тиск, швидкість тощо). Сигнал, що відображає цю інформацію і з'являється на виході відповідного вимірювального приладу, на будь-якому тимчасовому інтервалі може мати нескінченне число значень. Оскільки у даному випадку безперервний сигнал змінюється аналогічно вихідній інформації, його звичайно називають аналоговим, а прилади, в яких діють такі сигнали – аналоговими. Існують також дискретні повідомлення, параметри яких містять фіксований набір окремих значень. А оскільки цей набір кінцевий, то й обсяг інформації в таких повідомленнях кінцевий.

На практиці безперервні повідомлення можна подавати в дискретній формі. Безперервність повідомлень за величиною не може бути реалізована у зв'язку із похибкою джерел і приймачів інформації. Тому до безперервних сигналів, що відображають повідомлення, можна застосовувати квантування за рівнем і за часом. При квантуванні за рівнем сукупність можливих значень напруги або струму замінюють кінцевим набором дискретних значень з цього інтервалу. Квантування за часом передбачає заміну безперервного сигналу послідовністю імпульсів, що слідує через певні проміжки часу які називаються тактовими. Якщо тактові інтервали вибрані відповідним чином, то втрати інформації не відбувається. При одночасному введенні квантування за часом і за рівнем, амплітуда кожної вибірки буде приймати найближче дозволене значення з вибраного кінцевого набору значень. Сукупність всіх вибірок утворить дискретний або цифровий сигнал. Кожне значення дискретного сигналу можна подати числом. В цифровій техніці такий процес називається кодуванням, а сукупність отриманих чисел – кодом сигналу. Замість перетворення або передачі конкретних сигналів ці операції у приладах цифрової техніки можуть бути виконані над їхніми кодами. При цьому можна оперувати і аналоговими сигналами, що перетворюються в цифрові за допомогою аналого-цифрового перетворювача (АЦП).

1.1 Системи числення та кодування чисел

1.1.1. Системи числення

Дискретне повідомлення містить набір чисел і символів. Числа можуть бути представлені в різних системах числення. Форма запису в них чисел суттєво відрізняється між собою, тому, перш ніж перейти до особливостей подання чисел у цифрових пристроях, розглянемо способи їх запису у різних системах числення.

Під системою числення мається на увазі спосіб представлення будь-якого числа за допомогою певного набору цифр (символів), кількість яких дорівнюється основі системи.

Відрізняють *позиційні* і *непозиційні* системи числення. У позиційних системах сутність цифри залежить від позиції, займаної в числі, в непозиційних - така залежність відсутня.

Наприклад, в десятковому числі 129 сутність одиниці (1) в розряді сотень перевершує сутність дев'ятки (9) у розряді одиниць; в непозиційній, наприклад римській системі числення, в числі XXX всі десятки рівнозначні, хоча і розташовані на різних позиціях.

При постановці задач, введення певних числових залежностей у ЦП вигідно використовувати десяткову систему, отже, повинні бути засоби для відображення чисел у цій системі.

Використання в ЦП двійковій системі, що має тільки два символи «0» і «1», пояснюється тим, що будь-яка з двійкових цифр апаратно може бути відображена пристроєм, що має лише два стійких стану, наприклад «ввімкнено», «вимкнено». Такими пристроями можуть бути вимикачі, реле, тригери, електронні ключі. Недолік двійкової системи - значно більша довжина чисел в порівнянні з десятковою системою. Це компенсується простотою пристроїв на яких вони реалізовані, їх однаковістю, більш низьким рівнем затрат на експлуатацію навіть досить складних пристроїв.

У ЦП застосовуються також допоміжні системи - вісімкова і шістнадцяткова, через кратність їх основ двійковій системі ($8 = 2^3$, $16 = 2^4$) і більш короткого запису чисел в порівнянні з двійковою. Десяткова, двійкова, вісімкова і шістнадцяткова системи числення характеризуються наступними даними.

Десяткова: символи $a = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9$; основа $a = 10$.

Двійкова: $a = 0, 1$; $a = 2$.

Вісімкова: $a = 0, 1, 2, 3, 4, 5, 6, 7$; $a = 8$.

Шістнадцяткова: $a = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F$; $a = 16$.

У шістнадцятковій системі, щоб уникнути запису символів понад «9» двома цифрами, використовують букви відповідно для

- 10 - A,
- 11 - B,
- 12 - C,
- 13 - D,
- 14 - E,
- 15 - F.

Утворення чисел у будь-якій системі числення проводять таким чином: фіксують позиції, що називаються розрядами. Кожному розряду присвоюють свою вагу h_i (де i – номер розряду); $h_i = p^i$ (p – основа системи); в розрядах розміщують цифри a_i . Тоді будь-яке число A можна подати у вигляді:

$$A = \sum_{i=-m}^{n-1} a_i h_i \quad (1.1)$$

Послідовність цифр $a_{n-1}, a_{n-2}, \dots, a_1, a_0, a_{-1}, a_{-2}, \dots, a_{-m}$, можна розглядати як код числа в заданій системі числення:

$$N = \underbrace{a_n p^n + a_{n-1} p^{n-1} + \dots + a_0 p^0}_{\text{ціла частина}} + \underbrace{a_{-1} p^{-1} + a_{-2} p^{-2} + \dots + a_{-m} p^{-m}}_{\text{дробова частина числа}} \quad (1.2)$$

- де:
- a_i – будь-який символ з повного набору обраної системи числення;
 - ($i = 1, 2, 3, \dots, n$) або ($i = -1, -2, -3, \dots, -m$);
 - n та m для a_i – порядковий номер розряду цілої (n) та дробової (m) частин числа;
 - n та m для p – показники степеня основи цілої (n) та дробової (m) частин числа.

Приклад. Десяткове число $N_{(10)} = 162,75$, у двійковій, вісімковій і шістнадцятковій системах числення буде представлено відповідно:

$$N_{(2)} = 10100010,11;$$

$$N_{(8)} = 242,6$$

$$N_{(16)} = A2, C.$$

З наведеного прикладу випливає, що запис одного і того ж числа в двійковій системі числення – містить найбільшу кількість розрядів, найкоротший – в шістнадцятковій системі (таблиця 1.1).

Таблиця 1.1 – Форма запису чисел в різних системах числення

Десяткова	Система числення			
	Двійкова	Вісімкова	Двійково-десяткова	Шістнадцяткова
0	0	0	0	0
1	1	1	1	1
2	10	2	10	2
3	11	3	11	3
4	100	4	100	4
5	101	5	101	5
6	110	6	110	6
7	111	7	111	7
8	1000	10	1000	8
9	1001	11	1001	9
10	1010	12	0001 0000	A
11	1011	13	0001 0001	B
12	1100	14	0001 0010	C
13	1101	15	0001 0011	D
14	1110	16	0001 0100	E
15	1111	17	0001 0101	F
16	10000	20	0001 0110	10

1.1.2 Правила переведення чисел у задану систему числення

Для переведення числа з будь-якої системи числення в десяткову необхідно розкласти його в ряд за формулою (1.2) в десятковій системі і члени розкладання скласти, виключивши нульові добутки.

Переклад двійкових чисел у вісімкову і шістнадцяткову системи здійснюється поділом цілої та дробової частин відповідно вліво і вправо від коми на триади при переведенні в вісімкову систему і на тетради - при переведенні в шістнадцяткову систему. При цьому відсутні розряди в останніх триадах (тетрадах) заповнюються нулями.

Під триадами та тетрадами мають на увазі відповідно трьох-і чотирьохрозрядні двійкові числа.

Приклад. Переведемо двійкове число $N_{(2)} = 1110011,0110101$ в вісімкову систему. Ділимо цілу і дробову частини числа на триади:

```
ціла частина - 1  6  3
                001 110 011
дробова частина - 3  2  4
                  011 010 100
```

В останніх триадах цілої та дробової частини дописані нулі. Таким чином число $N_{(8)} = 163,324$ має вигляд $N_{(2)} = 1110011,0110101$.

Це ж число переведемо в шістнадцяткову систему, розділивши цілу і дробову його частини на тетради:

```
7  3  6  A
0111 0011 0110 1010.
```

Тут також в останніх тетрадах дописані нулі. Зауважимо, що величини числа вони не змінюють, тому що дописуються на початку цілої і наприкінці дробової частин числа. Шістнадцяткове число $N_{(16)} = 73,6 A$ має вигляд $N_{(2)} = 1110011,0110101$.

Розглянемо переклад вісімкових і шістнадцяткових чисел у двійкові.

Для перекладу, слід названі числа порозрядно (цілу і дробову частини) представити у вигляді двійкових триад і тетрад, і по ним записати відповідні двійкові числа.

Приклад. Вісімкове число $N_{(8)} = 1675,324$.

Запишемо по порядку триади для кожного розряду цілої та дробової частин -

```
001 110 111 101, 011 010 100.
```

Двійкове число буде $N_{(2)} = 1 110111101,0110101$.

Приклад. Шістнадцяткове число $N_{(16)} = FAB6,3E5$.

Запишемо по порядку тетради для кожного розряду цілої та дробової частин -

```
1111 1010 1011 0110, 0011 1110 0101.
```

Тоді бінарне число $N_{(2)} = 1111101010110110,001111100101$.

Сформулюємо правила перекладу.

1. Для переведення двійкового числа в вісімкове (шістнадцяткове) необхідно поділити цілу і дробову його частини відповідно справа наліво і зліва направо від коми на триади (тетради), заповнивши відсутні розряди в останніх триадах (тетрадах) нулями. Вісімкове (шістнадцяткове) число записується по триадах (тетрадах) зліва направо від цілої частини до дробової.

2. Для перекладу вісімкового (шістнадцяткового) числа в двійкове необхідно кожен розряд цілої та дробової частин числа записати триадами (тетрадами). Двійкове число записується об'єднанням триад (тетрад) цілої та дробової частин числа з виключенням нулів на початку цілою і наприкінці дробової частин.

При складанні програм на алгоритмічних мовах користуються переважно десятковою системою числення. Проте всі арифметичні операції МП проводять у двійковій системі числення, а для переходу до неї використовується проміжний так званий двійково-десятковий код, який представляє собою запис кожного розряду дробової і цілої частин десяткового числа двійковими тетрадами.

Наприклад, число $N_{(10)} = 286,74$ в двійково-десятковому коді записується у вигляді

```
 $N_{(2-10)} = 0010 1000 0110,0111 0100.$ 
```

Переклад цілих і дробових частин десяткових чисел у двійкові, вісімкові і шістнадцяткові числа здійснюється за різними правилами, які можна сформулювати так.

Для перекладу цілого десяткового числа в двійкову (вісімкову, шістнадцяткову) систему необхідно послідовно ділити, це число на основу системи числення 2 (8 або 16), фіксуючи залишки і часту на кожному розподілі до тих пір, доки в останній частці не буде отримано число, менше основи нової системи числення. Число в двійковій (8-ковій, 16-ткової) системі записується за останньою часткою і залишком від поділу на кожному кроці, починаючи з останньої частки, зліва направо.

Приклади перекладу цілого десяткового числа $N_{(10)} = 327$ в двійкову, вісімкову і шістнадцяткову системи показані у вигляді стовпчиків, в яких записані частки і залишки, одержані під час ділення числа 327 відповідно на основу системи числення $a = 2, 8$ і 16 . Порядок запису числа в системі з новою основою (2, 8, 16) зазначений на стовпчиках стрілками (Рис.1.1).

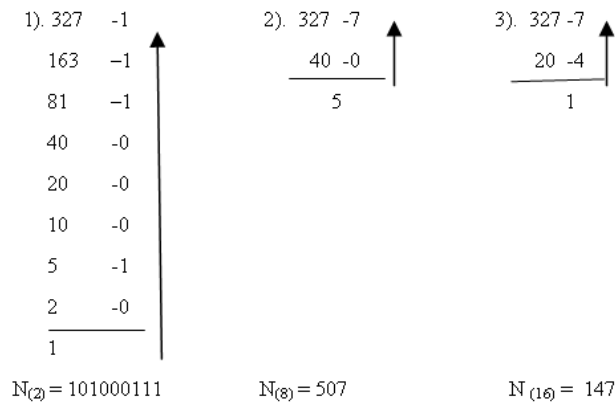


Рис.1.1 Приклади перекладу цілого десяткового числа N у двійкову, вісімкову і шістнадцяткову системи

Для перекладу правильного десяткового дробу або дробової частини змішаного числа у двійкову (вісімкову, шістнадцяткову) систему числення необхідно дробову частину вихідного числа і одержуваних часткових добутків послідовно помножити на 2 (8 або 16) до отримання певної кількості розрядів після коми в новій системі числення. Запис числа в новій системі здійснюється за цілими частинами часткових добутків, починаючи з першого, зліва направо. Зауважимо, що число розрядів в новій системі визначає точність перекладу чисел з однієї системи в іншу. При машинному перекладі число розрядів визначається розрядною сіткою елементів пам'яті.

Приклади перекладу десяткового дробу $N_{(2)} = 0,841$ у двійкову, вісімкову і шістнадцяткову системи показані відповідними стовпчиками послідовних множень на 2, 8 і 16 з зазначенням часткових добутків, отриманих на кожному кроці множення. Стрілками вказано порядок запису чисел в новій системі числення з основами 2, 8 і 16 (Рис.1.2).

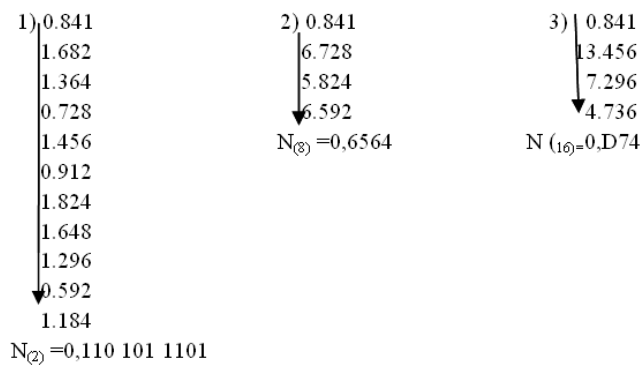


Рис.1.2 Приклади перекладу десяткового дробу у двійкову, вісімкову і шістнадцяткову системи

Перевірка правильності перекладу десяткового дробу $N_{(10)} = 0,841$ в системи числення з основами 2, 8 і 16:

порядок 1 2 3 4 5 6 7 8 9 10
 $N_{(2)} = 0,1 1 0 1 0 1 1 1 0 1 = 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-4} + 1 \cdot 2^{-6} + 1 \cdot 2^{-7} + 1 \cdot 2^{-8} + 1 \cdot 2^{-10} = 0,8408203125;$

порядок 1 2 3 4
 $N_{(8)} = 0,6 5 6 4 = 6 \cdot 8^{-1} + 5 \cdot 8^{-2} + 6 \cdot 8^{-3} + 4 \cdot 8^{-4} = 0,8408203125;$

порядок 1 2 3
 $N_{(16)} = 0,D 7 4 = 13 \cdot 16^{-1} + 7 \cdot 16^{-2} + 4 \cdot 16^{-3} = 0,8408203125.$

Продовживши множення, можна отримати велику точність перекладу.

Рішення алгебраїчних і диференціальних рівнянь різного порядку на ЦП зводиться в кінцевому підсумку до двох елементарних арифметичних операцій - додавання і віднімання. Навіть операції множення і ділення є лише

похідними від додавання і віднімання, будучи доповнені операцією зсуву. Таке спрощення досягнуто завдяки широкому розвитку чисельних методів розв'язання складних математичних залежностей, а також зведенню операцій над числами в різних системах числення до двійкової арифметики.

Правила виконання арифметичних дій над двійковими числами задаються таблицями додавання, віднімання і множення:

додавання	віднімання	множення
$0 + 0 = 0$	$0 - 0 = 0$	$0 * 0 = 0$
$0 + 1 = 1$	$1 - 0 = 1$	$0 * 1 = 0$
$1 + 0 = 1$	$1 - 1 = 0$	$1 * 0 = 0$
$1 + 1 = 10$	$10 - 1 = 1$	$1 * 1 = 1$

Слід зауважити:

- 1) одиниця з результату останньої суми в таблиці складання переходить в старший розряд;
- 2) в останній різниці таблиці розрахунків одиниця запозичується зі старшого розряду, перетворюючись на дві одиниці молодшого розряду.

Особливість операції множення двійкових чисел полягає в тому, що кількість нулів у множнику відповідає числу зрушень добутку ліворуч в стовпчику суми, сумуються ж числа, завжди рівні множнику, так як множаться на одиницю. Операція поділу заснована на операціях віднімання і зсуву.

1.1.3. Формати цілих та дійсних чисел

У ЦП числа представляються у двох формах: з фіксованою крапкою (звичайна форма) та з плаваючою крапкою (напівлогарифмічна форма). При поданні чисел з фіксованою крапкою мається на увазі, що крапка фіксується або перед старшим розрядом дробового числа, або після молодшого розряду цілого числа. На рис. 1.3 надано графічне представлення елементів пам'яті для цілих і дробових чисел з фіксованою крапкою різної розрядності: а) дрібно число у форматі машино слово (4 байту); б) ціле число у форматі машино слово; в) ціле число у форматі машино полуслово (2 байту).

За одиницю подання даних в ПК прийнятий байт - восьмирозрядне двійкове число, тому кількість розрядів осередків пам'яті завжди кратне восьми, а дані, відповідно, мають байтову структуру, тобто складаються з певного числа байтів.

$2^{\text{зн}}$	2^{-1}	2^{-2}	2^{-3}	...	2^{-30}	2^{-31}
a_0	a_1	a_2	a_3	...	a_{30}	a_{31}

а)

$2^{\text{зн}}$	2^{30}	2^{29}	2^{28}	...	2^{-30}	2^{-31}
a_0	a_1	a_2	a_3	...	a_{30}	a_{31}

б)

$2^{\text{зн}}$	2^{14}	2^{13}	2^{-3}	...	2^1	2^0
a_0	a_1	a_2	a_3	...	A_{14}	A_{15}

в)

Рис.1.3 Структура комірок пам'яті при запису двійкових чисел

У знаковий розряд елементу пам'яті a_0 ставиться «0» при позитивному і «1» при негативному числі, записаному в клітинку пам'яті. Найбільше число у формі з фіксованою крапкою відповідає заповненню всіх розрядів клітинок пам'яті одиницями, а найменше - нулями (крім наймолодшого розряду), вони рівні відповідно.

Числа, які за модулем більші наведених максимальних чисел, не можуть бути занесені в комірки пам'яті - відбудеться переповнення розрядної сітки. Попередити переповнення розрядної сітки можна за допомогою масштабування даних перед введенням в ЦП. Формат даних з фіксованою крапкою в ЦП використовується головним чином для подання адрес чисел, констант і даних, які не вимагають високої розрядності. Арифметичні операції над числами, представленими з фіксованою крапкою, виконуються з більш високою швидкістю у порівнянні з числами, представленими у формі з плаваючою крапкою. Проте, у зв'язку з можливим переповненням розрядної сітки використання звичайної форми у розрахунках - обмежується.

Для вирішення широкого кола обчислювальних завдань основним є представлення чисел у формі з плаваючою крапкою, при якій, незважаючи на меншу швидкість виконуваних операцій, не потрібне масштабування даних.

Представлення будь-якого числа N з плаваючою крапкою в загальному вигляді описується формулою

$$N = \pm t a \pm p,$$

де m - мантиса (дробова частина) числа; a - основа системи числення; p - порядок (ціле число).

Наприклад, десяткове число 238,745 і двійкове число 11001,1011, користуючись формулою, можна записати відповідно в декількох варіантах:

$$0,238745 \cdot 10^3; \quad 0,0238745 \cdot 10^4; \quad 0,000238745 \cdot 10^6; \\ 0,110011011 \cdot 10^1; \quad 0,00110011011 \cdot 10^{11}$$

У першому варіанті запису чисел за комою іде значуща цифра (2 в десятковому та 1 в двійковому зображеннях). Це так звана *нормалізована* форма чисел з плаваючою крапкою, останні записи не нормалізовані, так як після коми стоять нулі. Операцію нормалізації чисел МП виконує автоматично. У комірки пам'яті записуються розряди мантиси і порядку, а також їх знаків.

При двійковому поданні чисел для модуля порядку в комірці пам'яті виділяється шість двійкових розрядів (Рис. 1.4), що відповідає $p_{max} = 63$ або $p_{max} = 111111$. Для модуля мантиси виділяється 23 розряди при $m_{max} = 1-2^{-24}$.

Знак мантиси	Знак порядку	Модуль порядку					Модуль мантиси (a=2)					
		2	3	4	...	7	8	9	...	30	31	
0	1											
a ₀	b ₀	b ₁	b ₂	b ₃	...	b ₆	a ₁	a ₂	...	a ₂₃	a ₂₄	

Рис.1.4. Структура комірок пам'яті щодо двійкових чисел з плаваючою крапкою у форматі слова (4 байту)

Отже, діапазон нормалізованих чисел, що представляються в двійковій формі, буде знаходитися в межах

$$N_{min} = (0,100...0) \cdot 2^{-63} = 2^{-64} \\ N_{max} = (0,111...1) \cdot 2^{63} = (1-2^{-24}) \cdot 2^{63}$$

З урахуванням знака мантиси, діапазони негативних і позитивних чисел відповідно рівні

$$\text{від } -(1-2^{-24}) \cdot 2^{63} \text{ до } -2^{-64}, \\ \text{від } +2^{-64} \text{ до } +(1-2^{-24}) \cdot 2^{63}.$$

Зазначені межі в десятковому численні відповідають значенням $N \approx 10^{\pm 19}$, що суттєво більше діапазону чисел з фіксованою крапкою, що представляються в 32 розрядах елементу пам'яті.

Шістнадцяткове представлення чисел (формат слова) в елементі пам'яті має дві модифікації (рис.1.5), що відрізняються способом представлення знака порядку.

Знак мантиси	Знак порядку	Модуль порядку					Модуль мантиси (a=16)				
		2	4	5	...	7	8-11	12-15	16-23	24-27	28-31
0	1										
a ₀	b ₀	b ₁	b ₂	b ₃	...	b ₆	h ₁	h ₂	...	h ₅	h ₆

Рис.1.5 Структура комірок пам'яті щодо 16-ти розрядних чисел з плаваючою крапкою у форматі слова (4 байту)

Для зручності зберігання 16-ти розрядних чисел з плаваючою крапкою у форматі слова вводиться так званий *зміщений порядок*, де розряди мантиси розбиті на полубайти (4 розряди). Кожен з полубайтів відображає одну шістнадцяткову цифру тетрадами (0000 - 1111) - всього шість полубайтів (Рис.1.6). Розряди порядків - двійкові, максимальне значення порядку $p_{max} = 63$. Сутність зміщеного порядку полягає в тому, що остаточне його значення p_{cm} розглядається як сума

$$p_{cm} = p_{cm} + M,$$

де p - порядок введеного в пам'ять числа, який може приймати значення від $p_{max} = 63$ до $p_{min} = -64$; M — зміщення порядку, $M = 2^k$, k - число двійкових розрядів, які використовуються для модуля порядку.

Зміщений порядок зберігається в семи розрядах елементу пам'яті без знакового розряду. Отже, максимальне значення зміщеного порядку $p_{cm\ max}$ може бути рівним $2^7 - 1 = 127$. Діапазон зміни зміщеного порядку при зміні порядку вводиться в пам'ять чисел від $p = -64$ до $p = 63$, буде в межах від $p_{cm} = 0$ до $p_{cm} = 127$. При цьому інформація про знак порядку введеного числа закладена у значенні p_{cm} , без додаткового знакового розряду.

Знак мантиси	Модуль зміщеного порядку					Модуль мантиси (a=16)				
	1	2	3	...	7	8-11	12-15	16-23	24-27	28-31
0										
a ₀	b ₁	b ₂	b ₃	...	b ₆	h ₁	h ₂	...	h ₅	h ₆

Рис.1.6 Структура комірок пам'яті щодо 16-ти розрядних чисел з плаваючою крапкою у форматі слова (4 байту) з зміщеним порядком

Використання зміщеного порядку полегшує дії над порядками при виконанні арифметичних дій над числами, представленими у формі з плаваючою крапкою, зводячи їх до дій над цілими додатними числами. У форматі слова для шістнадцяткового подання отримуємо діапазони нормалізованих чисел з урахуванням знака мантиси:

від $-(1 - 16^{-6}) * 16^{63}$ до -16^{-64}

від 16^{-64} до $(1-16^{-6}) * 16^{63}$.

Це в десятковій системі числення відповідає значенням $N = 10^{-77} \dots 10^{+76}$.

Для зберігання чисел шістнадцяткової системи у форматі двійного слова (8 байт) довжина мантиси становить 14 полубайтів, зміщений порядок -7 полубайтів при загальній довжині 64 розрядів (Рис.1.7).

Знак мантиси	Модуль зміщеного порядку					Модуль мантиси (a=16)				
	1	2	3	7	8-11	12-15	16-55	56-59	60-63
a ₀	b ₁	b ₂	b ₃	b ₆	h ₁	h ₂	h ₁₃	h ₁₄

Рис.1.7 Структура комірок пам'яті щодо 16-ти розрядних чисел з плаваючою крапкою у форматі двійного слова (8 байт) з зміщеним порядком

Діапазон представлення чисел не змінюється а точність представлення збільшується за рахунок мантиси. Застосуванням шістнадцяткової системи у форматі з плаваючою крапкою заощаджується розрядність і створюється зручність запису двійково-кодованих чисел.

1.1.4. Кодування чисел

З метою спрощення арифметичних дій, числа з фіксованою і плаваючою крапкою - представляються спеціальними кодами. Це дозволяє замінити операцію віднімання операцією складання в зворотному і додатковому кодах, підвищити надійність розпізнавання переповнення розрядної сітки, спростити пристрої ЦП, що виконують арифметичні операції.

Для кодування чисел використовуються коди: прямий, зворотній, додатковий, модифікований (зворотній і додатковий).

Числа в кодах зображуються у вигляді правильних дробів, у знакових розрядах яких стоїть нуль - якщо число додатне, і одиниця - якщо число від'ємне.

При цьому позитивні числа в прямому, зворотному і додатковому кодах зображуються однаково, тому можна записати, що при $x > 0$

$$x_{i\bar{0}} = \bar{\bar{0}}_{\bar{c}\bar{a}} = \bar{\bar{0}}_{\bar{a}\bar{a}\bar{a}} = \bar{\bar{0}}$$

Кодовані позитивні числа ($x > 0$) у загальному вигляді можна представити формулою

$$x = 0.x_1x_2x_3\dots x_n$$

де: 0 - ознака позитивності числа (знаковий розряд);

$x_1\dots x_n$ - розряди його мантиси.

Зображення від'ємних чисел у прямому, зворотному і додатковому кодах описуються відповідно загальними формулами

$$x_{np} = 1.x_1x_2x_3\dots x_n,$$

$$\bar{\bar{0}}_{\bar{c}\bar{a}} = 1.\bar{\bar{0}}_1\bar{\bar{0}}_2\bar{\bar{0}}_3\dots\bar{\bar{0}}_i$$

$$\bar{\bar{0}}_{\bar{a}\bar{a}\bar{a}} = 1.\bar{\bar{0}}_1\bar{\bar{0}}_2\bar{\bar{0}}_3\dots(\bar{\bar{0}}_i + 1),$$

де 1 - ознака від'ємності числа, знаковий розряд;

$\bar{\bar{0}}_i$ - перетворення (інверсія) числа в розряді (0 або 1), тобто перехід одиниці в нуль, а нуля в одиницю;

$(\bar{\bar{0}}_i + 1)$ - доповнення числа у зворотному коді одиницею в останньому (наймолодшому) розряді.

На основі загальних виразів можна сформулювати правила перекладу чисел, заданих у вигляді правильного дробу, у прямий, зворотній і додатковий коди.

1. Переклад додатних чисел у прямий, зворотній і додатковий коди не змінює зображення цих чисел.
2. При перекладі від'ємного числа в код, знак «мінус» перед числом опускається в цілій частині, а замість нуля ставиться одиниця. У розрядах мантиси, в прямому коді, послідовність нулів і одиниць не відрізняється від вихідної, у зворотному коді - інверсія нулів та одиниць, в додатковому коді після інверсії нулів та одиниць до наймолодшого розряду мантиси додається одиниця.

Приклад. Задано від'ємне число $x = -0,101101$.

Його зображення в кодах: $x_{i\bar{d}} = 1,101101$ $\tilde{\delta}_{\bar{c}\bar{a}} = 1,010010$ $\tilde{\delta}_{\bar{a}\bar{i}\bar{a}} = 1,010011$

Переклад чисел у модифіковані зворотній і додатковий коди відрізняється від наведеного лише тим, що для додатних і від'ємних чисел замість одного ставляться два знакових розряди відповідно до загальних формул:

При $x > 0$

$$x_{\bar{c}\bar{a}_{-i}} = 00.\tilde{\delta}_1\tilde{\delta}_2\tilde{\delta}_3\dots\tilde{\delta}_i,$$

$$\tilde{\delta}_{\bar{a}\bar{i}\bar{a}_{-i}} = 00.\tilde{\delta}_1\tilde{\delta}_2\tilde{\delta}_3\dots\tilde{\delta}_i.$$

При $x < 0$

$$\tilde{\delta}_i = 11.\tilde{\delta}_{-1}\tilde{\delta}_2\tilde{\delta}_3\dots\tilde{\delta}_i,$$

$$\tilde{\delta}_i = 11.\tilde{\delta}_{-1}\tilde{\delta}_2\tilde{\delta}_3\dots(\tilde{\delta}_i + 1)$$

Представлення чисел у модифікованих кодах полегшує розпізнавання ситуацій переповнення розрядної сітки при виконанні арифметичних операцій. Це пов'язано з появою в знакових розрядах результатів операцій заборонених кодових комбінацій 10 або 01, що не відповідають очікуваним 00 для додатних і 11 для від'ємних результатів.

Приклад. При складанні чисел $A = 00.110101$ і $B = 00.101001$, а також $C = 11.001100$ і $D = 11.100110$, записаних у модифікованому додатковому коді, має місце переповнення розрядної сітки:

$A = 00,110101$	$C = 11,001100$
+	+
$B = 00,101001$	$D = 11,100110$
-----	-----
$A+B=01,011110$	$C+D=1\bar{1},110010$

Як бачимо, результати підсумовування в знакових розрядах містять заборонені комбінації 01 і 10 (одиниця у третьому розряді суми C і D при складанні чисел у модифікованому додатковому коді - відкидається). Це означає, що результати підсумовування більше одиниці і при введенні цих чисел необхідно було провести масштабування додатків. Розпізнавання заборонених комбінацій при переповненні здійснюється спеціальними логічними пристроями.

1.1.5. Арифметика операцій з кодами

Розглянемо порядок виконання операцій при складанні чисел з плаваючою крапкою:

- вирівнювання порядків додатків, менший порядок приводиться до більшого операцією зсуву мантиси вправо;
- мантиси доданків перетворюються в модифікований зворотній (або додатковий) код;
- виконується порозрядне складання кодів мантис, включаючи і знакові розряди,
- при появі додаткової одиниці у знакових розрядах результату в модифікованому додатковому коді ця одиниця відкидається, в модифікованому зворотному коді одиниця додається до наймолодшого розряду мантиси;
- отримана в зворотному (або додатковому) коді мантиса перетворюється в прямий код;
- при отриманні ненормалізованої мантиси суми виконується операція нормалізації (зсув вліво) і зміна порядку.

Приклад. Скласти двійкові числа

$$x_{(2)} = 0.1011101 * 10^{-100} \quad y_{(2)} = -0.1101101 * 10^{-101}$$

Зображення цих чисел у прямому коді з урахуванням порядків буде.

$$x_{i\bar{d}} = 0.1011101 * 10^{-100} \quad \acute{o}_{i\bar{d}} = 1.1101101 * 10^{-101}$$

Зрівняємо порядки:

$$\tilde{\delta}_{i\bar{d}} = 0.1011101 \quad \acute{o}_{i\bar{d}} = 1.01101101$$

Представимо мантиси в модифікованих (зворотному і додатковому) кодах і складемо x і y :

$\tilde{\delta}_{\bar{c}\bar{a}_{-i}} = 00.10111010$	$\tilde{\delta}_{\bar{a}\bar{i}\bar{a}_{-i}} = 00.10111010$
$\acute{o}_{\bar{c}\bar{a}_{-i}} = 11.10010010$	$\acute{o}_{\bar{a}\bar{i}\bar{a}_{-i}} = 11.10010011$
100.01001100	100.01001101
_____ → 1	← _____
цикл. перенос	відкидання
$x + y = 00.01001101$	$x + y = 00.01001101$

Результати підсумовування, отримані в модифікованих кодах, є додатними і ненормалізованими. Для нормалізації необхідно провести зсув мантиси ліворуч на один розряд, а порядок зменшити на одиницю, тоді

остаточно результат буде $x + y = 00.1001101 \cdot 10^{101}$. Перетворювати результат в прямий код - немає необхідності, тому що додатні числа у всіх кодах зображуються однаково.

Сучасні ЦП обробляють не тільки числову, але й текстову, тобто алфавітно-цифрову, інформацію, яка містить цифри, букви, розділові знаки, математичні та інші спеціальні символи.

1.2 Основи алгебри логіки та логічні функції

1.2.1 Основні логічні функції алгебри логіки

В алгебрі логіки розглядаються змінні, які можуть приймати тільки два значення: 0 і 1. В подальшому змінні будемо позначати латинськими буквами x, y, z, \dots . В алгебрі логіки визначено *відношення еквівалентності* ($=$) і три операції:

- диз'юнкція (операція АБО), що позначається знаком \vee ;
- кон'юнкція (операція І), що позначається крапкою, яку можна опускати (наприклад, $x \cdot y = xy$);
- заперечення (інверсія, операція НІ), що позначається рискою над змінними або елементами 0 і 1 (наприклад, $\bar{x}, \bar{0}, \bar{1}$).

Відношення еквівалентності задовольняє такі властивості:

- $x = x$ - *рефлексивність*;
- якщо $x = y$, то $y = x$ - *симетричність*;
- якщо $x = y$ та $y = z$, то $x = z$ - *транзитивність*.

Із відношення еквівалентності слідує *принцип підстановки*: якщо $x = y$, то в будь-якій формулі, що вміщує x , замість x можна підставити y , і буде отримана еквівалентна формула.

Алгебра логіки визначається такою системою аксіом:

$$\left. \begin{array}{l} x = 0, \text{ якщо } x \neq 1, \\ x = 1, \text{ якщо } x \neq 0; \end{array} \right\} \quad (1.3)$$

$$\left. \begin{array}{l} 1 \vee 1 = 1, \\ 0 \cdot 0 = 0; \end{array} \right\} \quad (1.4)$$

$$\left. \begin{array}{l} 0 \vee 0 = 0, \\ 1 \cdot 1 = 1; \end{array} \right\} \quad (1.5)$$

$$\left. \begin{array}{l} 0 \vee 1 = 1 \vee 0 = 1, \\ 1 \cdot 0 = 0 \cdot 1 = 0; \end{array} \right\} \quad (1.6)$$

$$\left. \begin{array}{l} \bar{0} = 1, \\ \bar{1} = 0. \end{array} \right\} \quad (1.7)$$

Аксіома (1.3) стверджує, що в алгебрі логіки розглядаються лише двійкові змінні, аксіоми (1.4) - (1.6) визначають операції диз'юнкції та кон'юнкції, а аксіома (1.7) - операцію заперечення.

Якщо в аксіомах (1.4)-(1.7), заданих парами, провести взаємну заміну операцій диз'юнкції та кон'юнкції, а також елементів 0 і 1, то із одної пари аксіом отримаємо іншу. Ця властивість називається *принципом подвійності*.

За допомогою аксіом алгебри логіки можна довести цілий ряд теорем та тотожностей. Одним із ефективних методів доказу теорем є *метод перебору* всіх значень змінних. Так, методом перебору легко переконатися у справедливості теорем, для зручності зведених у таблицю 1.2.

Якщо в логічний вираз входять операції диз'юнкції та кон'юнкції, то потрібно зберігати порядок виконання операцій: спочатку виконується операція кон'юнкції, а потім операція диз'юнкції. У складних логічних виразах для завдання порядку виконання операцій використовуються дужки.

Деякі теореми та тотожності алгебри логіки мають особливе значення, оскільки дозволяють спрощувати логічні вирази. Особливо часто для перетворення логічних виразів використовуються тотожності (1.17) - (1.20).

Таблиця 1.2 - Закони алгебри логіки

Назва закону	Формули	
Ідемпотентні	$\left. \begin{aligned} x \vee x &= x \\ x \cdot x &= x \end{aligned} \right\}$	(1.8)
Комутативні	$\left. \begin{aligned} x \vee y &= y \vee x \\ x \cdot y &= y \cdot x \end{aligned} \right\}$	(1.9)
Асоціативні	$\left. \begin{aligned} (x \vee y) \vee z &= x \vee (y \vee z) \\ (x \cdot y) \cdot z &= x \cdot (y \cdot z) \end{aligned} \right\}$	(1.10)
Дистрибутивні	$\left. \begin{aligned} x \cdot (y \vee z) &= x \cdot y \vee x \cdot z \\ x \vee y \cdot z &= (x \vee y) \cdot (x \vee z) \end{aligned} \right\}$	(1.11)
Заперечення	$\left. \begin{aligned} x \vee \bar{x} &= 1 \\ x \cdot \bar{x} &= 0 \end{aligned} \right\}, \left. \begin{aligned} 0 \vee x &= x \\ 1 \cdot x &= x \end{aligned} \right\}, \left. \begin{aligned} 1 \vee x &= 1 \\ 0 \cdot x &= 0 \end{aligned} \right\}$	(1.12-14)
Подвійності (де Моргана)	$\left. \begin{aligned} \overline{x \vee y} &= \bar{x} \cdot \bar{y} \\ \overline{x \cdot y} &= \bar{x} \vee \bar{y} \end{aligned} \right\}$	(1.15)
Подвійного заперечення	$\overline{\bar{x}} = x$	(1.16)
Поглинання	$\left. \begin{aligned} x \vee x \cdot y &= x \\ x \cdot (x \vee y) &= x \end{aligned} \right\}$	(1.17)
Операції склеювання	$\left. \begin{aligned} x \cdot y \vee x \cdot \bar{y} &= x \\ (x \vee y) \cdot (x \vee \bar{y}) &= x \end{aligned} \right\}$	(1.18)
Операції узагальненого склеювання	$\left. \begin{aligned} x \cdot y \vee \bar{x} \cdot z \vee y \cdot z &= x \cdot y \vee \bar{x} \cdot z \\ (x \vee y) \cdot (\bar{x} \vee z) \cdot (y \vee z) &= (x \vee y) \cdot (\bar{x} \vee z) \\ x \vee \bar{x} \cdot y &= x \vee y \\ x \cdot (\bar{x} \vee y) &= x \cdot y \end{aligned} \right\}$	(1.19)

(1.20)

Операція «сума за модулем два» (Виключне АБО, логічна нерівнозначність) позначається символом \oplus і визначається співвідношенням

$$x \oplus y = \bar{x}y \vee x\bar{y} = (\bar{x} \vee \bar{y}) \cdot (x \vee y) . \quad (1.21)$$

Таблиця істинності для (1.3) має вигляд:

Таблиця 1.3 - Таблиця істинності для операції «сума за модулем два»

Вираз	Значення
$0 \oplus 0$	0
$0 \oplus 1$	1
$1 \oplus 0$	1
$1 \oplus 1$	0

Операція суми за модулем два комутативна, асоціативна і дистрибутивна відносно операції кон'юнкції, тобто

$$\left. \begin{aligned} x \oplus y &= y \oplus x, \\ x \oplus (y \oplus z) &= (x \oplus y) \oplus z, \\ x \cdot (y \oplus z) &= x \cdot y \oplus x \cdot z \end{aligned} \right\} \quad (1.22)$$

Для неї також вірні наступні тотожності:

$$\left. \begin{aligned} x \oplus 0 &= x; \\ x \oplus 1 &= \bar{x}; \\ x \oplus \bar{x} &= 0; \\ \overline{x \oplus y} &= \bar{x} \cdot \bar{y} \vee x \cdot y = (\bar{x} \vee y) \cdot (x \vee \bar{y}) = \bar{x} \oplus y = x \oplus \bar{y} \end{aligned} \right\} \quad (1.23)$$

1.2.2 Функції перемикання

Будь-який логічний вираз, що складається із n змінних x_1, \dots, x_n , можна розглядати як деяку функцію n змінних. У відповідності з аксіомами (1.3) - (1.7) функція може приймати лише два значення: 0 і 1. Такі функції зручні для опису, аналізу і синтезу схем перемикання, вихідні сигнали яких характеризуються лише двома рівнями напруги: високим (1) і низьким (0). У зв'язку з цим такі функції називаються функціями перемикання.

Властивості функцій перемикання. Для функцій n змінних x_1, \dots, x_n використовується загальне позначення $f(v) = f(x_1, \dots, x_n)$, де $v = (x_1, \dots, x_n)$ - сукупність змінних, що розглядається як n -мірний вектор. Кожна змінна x_p ($p = 1 \dots n$) може приймати лише два значення: 0 і 1.

Конкретне значення змінної x_1, \dots, x_n (0 чи 1) позначається як e_p .

Для завдання функції $f(v)$ слід вказати її значення у всіх точках області визначення, тобто задати значення $f(v_i) = 0$ чи 1, де $i = 0, 1, \dots, 2^n - 1$. Кожній конкретній функції n змінних можна поставити у відповідність 2^n -розрядне число, що складене із значень $f(v_i) = 0$ чи 1 ($i = 0, 1, \dots, 2^n - 1$), які вона приймає в 2^n точках області визначення.

Функції n змінних можуть залежати не від усіх змінних x_1, \dots, x_n . Такі функції називають *виродженими*. Значний інтерес викликають не вироджені функції двох змінних x_1, x_2 , назви яких дані за використовуваними для їх утворення операціями алгебри логіки:

$$\begin{aligned} f(x_1, x_2) &= x_1 \vee x_2 - \text{диз'юнкція (АБО);} \\ f(x_1, x_2) &= x_1 \cdot x_2 - \text{кон'юнкція (І);} \\ f(x_1, x_2) &= \overline{x_1 \cdot x_2} - \text{функція І-НІ;} \\ f(x_1, x_2) &= \overline{x_1 \vee x_2} - \text{функція АБО-НІ;} \\ f(x_1, x_2) &= x_1 \oplus x_2 - \text{сума за модулем 2.} \end{aligned}$$

Область визначення функції n змінних може бути задана таблицею значень, які вона приймає в точках V_i . Такі таблиці називаються *таблицями істинності*.

Таблиця 1.4, яка складена у відповідності до аксіом (1.3)-(1.7) для вказаних функцій, - це таблиця істинності, що задає ці функції.

Таблиця 1.4 - Таблиця істинності для ЛФ двох змінних x_1, x_2

x_1, x_2	$x_1 \vee x_2$	$x_1 \cdot x_2$	$\overline{x_1 \cdot x_2}$	$\overline{x_1 \vee x_2}$	$x_1 \oplus x_2$
0 0	0	0	1	1	0
0 1	1	0	1	0	1
1 0	1	0	1	0	1
1 1	1	1	0	0	0

Функція n змінних $f(v)$ називається *повністю визначеною*, якщо її значення $f(v_i) = 0$ чи 1 задані у всіх точках області визначення. Якщо ж значення функції не задано хоча б в одній точці V_i , то вона називається *не повністю визначеною*.

Принцип і закон подвійності.

Алгебра логіки має властивість, яка називається *принципом подвійності*: якщо має місце тотожність, $f(v, 0, 1/\vee, \&) = g(v, 0, 1/\vee, \&)$ то справедлива також тотожність $f(v, 1, 0/\&, \vee) = g(v, 1, 0/\&, \vee)$, тобто якщо в якій-небудь тотожності виконати взаємну заміну символів 0 і 1 і операцій диз'юнкції та кон'юнкції, то буде отримано також тотожність.

Дві тотожності, які зв'язані між собою таким чином, є подвійними. Істинність самого принципу подвійності не доводиться, оскільки даний принцип є внутрішньою властивістю алгебри логіки (полягає в її аксіомах).

Закони подвійності (теореми де Моргана) (1.13) встановлюють спосіб пошуку інверсних функцій, що є диз'юнкцією і кон'юнкцією двох змінних.

Клод Шеннон запропонував узагальнення цих теорем, що дозволяє відшукати інверсію будь-якої функції $f(v)$.

Закон подвійності Клода Шеннона має вигляд:

$$\overline{f(v/\vee, \&)} = f(\overline{v}/\&, \vee) \quad (1.24)$$

тобто інверсію будь-якої функції можна отримати взаємною заміною змінних x_p ($p=1..n$) і операцій диз'юнкції та кон'юнкції.

Розглянемо кілька прикладів.

Нехай

$$f(v) = \overline{x_2} \cdot x_1 \vee x_2 \cdot \overline{x_1}, \text{ тоді } \overline{f(v)} = (x_2 \vee \overline{x_1}) \cdot (\overline{x_2} \vee x_1)$$

Нехай

$$f(v) = [(\overline{x_2} \cdot x_1 \vee x_3 \cdot x_2) \cdot x_3 \cdot x_1 \vee x_3 \cdot \overline{x_1}] \cdot (x_2 \cdot \overline{x_1} \vee x_3) \vee x_4, \text{ тоді}$$

$$\overline{f(v)} = \{ [(x_2 \cdot \overline{x_1}) \cdot (\overline{x_3} \vee \overline{x_2}) \vee \overline{x_3} \vee \overline{x_1}] \cdot (\overline{x_3} \vee x_1) \vee (\overline{x_2} \vee x_1) \cdot \overline{x_3} \} \cdot x_4$$

На основі закону подвійності легко показати, що

$$\overline{\bigvee_{p=1}^n x_p} = \prod_{p=1}^n \overline{x_p}; \quad \prod_{p=1}^n x_p = \overline{\bigvee_{p=1}^n \overline{x_p}}$$

Теорема розкладу і зв'язані з нею тотожності.

В теорії функцій перемикання особливо важливе значення має *теорема розкладу*: будь-яку функцію $f(v)$ можна розкласти по змінній x_p у формі

$$f(x_1, \dots, x_p, \dots, x_n) = x_p \cdot f(x_1, \dots, 0, \dots, x_n) \vee \overline{x_p} \cdot f(x_1, \dots, 1, \dots, x_n) \quad (1.25)$$

Ця теорема легко доводиться методом перебору:

$$a) x_p = 0 \Rightarrow f(x_1, \dots, \overline{0}, \dots, x_n) = \overline{0} \cdot f(x_1, \dots, 0, \dots, x_n) \vee 0 \cdot f(x_1, \dots, 1, \dots, x_n) = f(x_1, \dots, 0, \dots, x_n)$$

тобто при $x_p = 0$ теорема справедлива незалежно від значень других змінних;

$$b) x_p = 1 \Rightarrow f(x_1, \dots, 1, \dots, x_n) = \overline{1} \cdot f(x_1, \dots, 0, \dots, x_n) \vee 1 \cdot f(x_1, \dots, 1, \dots, x_n) = f(x_1, \dots, 1, \dots, x_n),$$

тобто при $x_p = 1$ теорема справедлива незалежно від значень других змінних, а значить, теорема істинна при будь-яких значеннях всіх змінних, що і вимагалось довести.

За принципом подвійності отримуємо *подвійну теорему розкладу*:

$$f(x_1, \dots, x_p, \dots, x_n) = [\overline{x_p} \vee f(x_1, \dots, 1, \dots, x_n)] \cdot [x_p \vee f(x_1, \dots, 0, \dots, x_n)] \quad (1.26)$$

Теорема розкладу є зручним інструментом для перетворення логічних виразів, що вміщують операцію суми за модулем 2, оскільки в ряді практичних випадків дозволяє звести дану операцію над функціями до простих операцій, наприклад:

$$\begin{aligned} & \overline{x_2 \cdot x_1 \oplus (x_3 \vee \overline{x_1}) \oplus x_3 \cdot x_1 \oplus (x_2 \vee \overline{x_1})} = \\ & = \overline{x_1} \cdot [\overline{x_2 \cdot 0 \oplus (x_3 \vee 0) \oplus x_3 \cdot 0 \oplus (x_2 \vee 0)}] \vee x_1 \cdot [\overline{x_2 \cdot 1 \oplus (x_3 \vee 1) \oplus x_3 \cdot 1 \oplus (x_2 \vee 1)}] = \\ & = \overline{x_1} \cdot (\overline{0 \oplus 1 \oplus 0 \oplus 1}) \vee x_1 \cdot (\overline{x_2 \oplus x_3 \oplus x_3 \oplus x_2}) = \overline{x_1} \cdot 1 \vee x_1 \cdot 1 = \overline{x_1} \vee x_1 = 1. \end{aligned}$$

З теоремою розкладу зв'язані тотожності:

$$\left. \begin{aligned} \overline{x_p} \cdot f(x_1, \dots, x_p, \dots, x_n) &= \overline{x_p} \cdot f(x_1, \dots, 0, \dots, x_n) \\ x_p \cdot f(x_1, \dots, x_p, \dots, x_n) &= x_p \cdot f(x_1, \dots, 1, \dots, x_n) \end{aligned} \right\} \quad (1.27)$$

За принципом подвійності цим тотожностям відповідають подвійні тотожності:

$$\left. \begin{aligned} \overline{x_p \vee f(x_1, \dots, x_p, \dots, x_n)} &= \overline{x_p \vee f(x_1, \dots, 1, \dots, x_n)} \\ x_p \vee f(x_1, \dots, x_p, \dots, x_n) &= x_p \vee f(x_1, \dots, 0, \dots, x_n) \end{aligned} \right\} \quad (1.28)$$

Дані тотожності є потужним засобом для спрощення логічних виразів. Нехай потрібно спростити функцію

$$f(v) = \overline{x_2 x_1 \oplus x_3 x_2 \oplus x_1 \vee x_3 x_2 \cdot x_2}$$

Використовуючи першу з тотожностей (1.3.25) відносно x_2 , отримаємо:

$$f(v) = \overline{0 \cdot x_1 \oplus x_3 \cdot 0 \oplus x_1 \vee x_3 \cdot 0 \cdot x_2} = \overline{x_3 \oplus x_1 \vee x_3 \cdot x_2}$$

Для спрощення виразу $\overline{x_3 \oplus x_1 \vee x_3}$ можна використати другу із тотожностей (1.28), тоді

$$f(v) = \overline{0 \oplus x_1 \vee x_3 \cdot x_2} = \overline{x_1 \vee x_3 \cdot x_2} = \overline{x_1 \cdot x_2 \cdot x_3}$$

Первинні терми. Змінні x_p та їх інверсії $\overline{x_p}$ називаються первинними термами, для яких використовується символічне позначення

$$x_p^{e_p} = \overline{e_p x_p \vee e_p x_p} = \overline{e_p \oplus x_p} \quad (1.29)$$

де $e_p = 0$ або 1 .

Дане символічне позначення об'єднує в одному символі $x_p^{e_p}$ обидва первинних терма x_p і $\overline{x_p}$. Дійсно, при підстановці в (1.29) значень $e_p = 0$ і 1 отримаємо

$$x_p^{e_p} = \begin{cases} \overline{x_p}, & \text{при } e_p = 0, \\ x_p, & \text{при } e_p = 1. \end{cases}$$

Тільки завдяки введенню даного символічного позначення можна формалізувати виведення загальних відношень для функцій перемикання. Очевидно, що два первинних терми $x_p^{e_p}$ і $x_p^{e'_p}$ рівні лише в тому випадку, коли $e_p = e'_p$. Для первинних термів справедливі співвідношення:

$$x_p^0 = \overline{x_p^1} = \overline{x_p};$$

$$\overline{x_p^{e_p}} = x_p^{\overline{e_p}} = \overline{\overline{x_p^{e_p}}}; \quad (1.30)$$

$$\begin{aligned} x_p^{e_p} \cdot x_p^{\overline{e_p}} &= 0 \\ x_p^{e_p} \vee x_p^{\overline{e_p}} &= 1 \end{aligned} \quad ; \quad (1.31)$$

$$x_p^{e_p} = \begin{cases} 0, & \text{якщо } x_p = \overline{e_p}, \\ 1, & \text{якщо } x_p = e_p. \end{cases} \quad (1.32)$$

Істинність цих співвідношень елементарно перевіряється на основі визначення первинних термів (1.27).

Мінтерми і макстерми. *Мінтермом (конституентною одиницею)* називається функція n змінних

$$K_i(v) = x_1^{e_1} \dots x_n^{e_n} = \prod_{p=1}^n x_p^{e_p} \quad (1.33)$$

де $v = (x_1, \dots, x_n)$, $e_p = 0$ або 1 , $i = e_1, \dots, e_n$. Із даного визначення слідує, що є 2^n різних мінтермів n змінних, оскільки є 2^n різних n -розрядних двійкових чисел $i = 0, 1, \dots, 2^n - 1$

Мінтерми мають такі властивості:

$$K_i(v) = \begin{cases} 1, & \text{якщо } v = v_i, \\ 0, & \text{якщо } v = v_j \neq v_i; \end{cases} \quad (1.34)$$

$$K_i(v) \cdot K_j(v) \equiv 0, \text{ якщо } i \neq j \quad (1.35)$$

$$\bigvee_{i=0}^{2^n-1} K_i(v) \equiv 1. \quad (1.36)$$

Властивість мінтермів (1.34), яка полягає в тому, що будь-який мінтерм $K_i(v)$ рівний 1 тільки в одній точці v_i області визначення, що складається із 2^n точок, легко доказати, використовуючи властивості первинних термів (1.32). Властивості (1.35) і (1.36) доводяться на основі властивості (1.34).

Запишемо всі мінтерми двох змінних x_1, x_2 :

$$K_0(v) = x_1^0 x_2^0 = \overline{x_1} \cdot \overline{x_2}; \quad K_1(v) = x_1^1 x_2^0 = x_1 \cdot \overline{x_2};$$

$$K_2(v) = x_1^0 x_2^1 = \overline{x_1} \cdot x_2; \quad K_3(v) = x_1^1 x_2^1 = x_1 \cdot x_2;$$

де $v = (x_1, x_2)$. Таким самим способом можна записати будь-який мінтерм $K_i(v)$ більшого числа змінних. Нехай,

наприклад, $n = 4, i = 13$, тоді $K_{13}(v) = x_1^1 x_2^0 x_3^1 x_4^1 = x_1 \cdot \overline{x_2} \cdot x_3 \cdot x_4$

Макстермом (конституєнтою нуля) називається функція n змінних

$$M_i(v) = \overline{K_i(v)} = \prod_{p=1}^n x_p^{e_p} = \bigvee_{p=1}^n \overline{x_p^{e_p}}.$$

Згідно з властивістю первинних термів (1.30) можна записати

$$M_i(v) = \bigvee_{p=1}^n \overline{x_p^{e_p}}, \quad (1.37)$$

де $v = (x_1, \dots, x_n), i = (e_1, \dots, e_n)$.

Макстерми мають такі властивості:

$$M_i(v) = \begin{cases} 0, & \text{при } v = v_i, \\ 1, & \text{при } v = v_j \neq v_i; \end{cases}$$

$$M_i(v) \vee M_j(v) \equiv 0, \quad \text{при } i \neq j$$

$$\prod_{i=1}^{2^n-1} M_i(v) \equiv 0.$$

Для більшої наочності в таблиці 1.5 (таблиця істинності) приведені всі мінтерми і макстерми двох змінних x_1 і x_2 .

Таблиця 1.5 - Макстерми і мінтерми двох змінних

i	x_1	x_2	K_0	K_1	K_2	K_3	M_0	M_1	M_2	M_3
0	0	0	1	0	0	0	0	1	1	1
1	1	0	0	1	0	0	1	0	1	1
2	0	1	0	0	1	0	1	1	0	1
3	1	1	0	0	0	1	1	1	1	0

Мінтерми (макстерми) - це функції, що приймають мінімальне (максимальне) значення із значень своїх первинних термів $x_p^{e_p}$, тобто якщо хоча б один із первинних термів $x_p^{e_p}$ рівний 0(1), то і мінтерм (макстерм) рівний 0(1).

Досконала диз'юнктивна нормальна форма (ДДНФ). Теорему розкладу (1.23) для функцій n змінних можна використовувати n разів, тобто функцію можна розкласти по всім n змінним x_p . Як приклад розглянемо розклад функції $f(x_1, x_2)$ двох змінних x_1 та x_2 .

$$\text{За теоремою розкладу (1.25) отримаємо } f(x_1, x_2) = \overline{x_2} \cdot f(x_1, 0) \vee x_2 \cdot f(x_1, 1).$$

Далі кожен з функцій $f(x_1, 0)$ і $f(x_1, 0) \vee x_2 \cdot f(x_1, 1)$ можна розкласти по змінній x_1

$$\begin{aligned}
f(x_1, x_2) &= \overline{x_2} \cdot [\overline{x_1} \cdot f(0,0) \vee x_1 \cdot f(1,0)] \vee x_2 \cdot [x_1 \cdot f(0,1) \vee x_1 \cdot f(1,1)] = \\
&= \overline{x_1} \cdot \overline{x_2} \cdot f(0,0) \vee x_1 \cdot \overline{x_2} \cdot f(1,0) \vee \overline{x_1} \cdot x_2 \cdot f(0,1) \vee x_1 \cdot x_2 \cdot f(1,1) = \\
&= \bigvee_{i=0}^3 x_1^{e_1} x_2^{e_2} \cdot f(e_1, e_2) = \bigvee_{i=0}^3 f(v_i) K_i(v),
\end{aligned}$$

де $v = (x_1, x_2)$, $v_i = (e_1, e_2)$, $i = e_1 e_2$, $K_i(v) = x_1^{e_1} x_2^{e_2}$ - мінтерми двох змінних

Оскільки $f(v_i) = a_i = 0$ або 1 (значення функції в точці v_i), то $f(v) = \bigvee_{i=0}^3 a_i K_i(v)$.

Така форма подання функції двох змінних називається ДДНФ. Розклад функції n змінних буде представляти собою диз'юнкцію членів виду

$$x_1^{e_1} \dots x_p^{e_p} \dots x_n^{e_n} f(e_1, \dots, e_p, \dots, e_n) = f(v_i) K_i(v) = a_i K_i(v)$$

$$f(v) = \bigvee_{i=0}^{2^n-1} a_i K_i(v) \quad (1.38)$$

Вираз (1.38) представляє собою ДДНФ функції n змінних. Оскільки значення функції $a_i=0$ чи 1, то $a_i K_i(v) = 0$, якщо $a_i=0$, і $a_i K_i(v) = K_i(v)$, якщо $a_i = 1$. Тому ДДНФ функції можна представити у вигляді

$$f(v) = \bigvee_{i_s} K_{i_s}(v) \quad (1.39)$$

де i_s - номери тих точок, в яких функція $f(v)$ дорівнює 1.

Як приклад розглянемо функцію $f(v)$ трьох змінних, задану таблицею істинності (таблиця 1.6).

Таблиця 1.6 - Таблиця істинності функції трьох змінних

i	x_1	x_2	x_3	$f(v)$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Із даної таблиці випливає, що

$$a_0 = a_3 = a_4 = a_6 = 0, \quad a_1 = a_2 = a_5 = a_7 = 1$$

тому згідно з (1.39)

$$f(v) = K_1(v) \vee K_2(v) \vee K_5(v) \vee K_7(v) = \overline{x_1} \cdot \overline{x_2} \cdot x_3 \vee \overline{x_1} \cdot x_2 \cdot \overline{x_3} \vee \overline{x_1} \cdot x_2 \cdot x_3 \vee x_1 \cdot \overline{x_2} \cdot x_3.$$

Це і є досконала кон'юнктивна нормальна форма (ДКНФ) функції. Таку форму функції n змінних можна отримати на підставі подвійної теореми розкладу (1.26). Однак ДКНФ можна отримати і більш простим шляхом, записавши ДДНФ інверсної функції $\overline{f(v)}$. Інверсія функції в кожній точці v_i повинна мати інверсні значення по відношенню до значень a_i самої функції, тобто $\overline{f(v_i)} = \overline{a_i}$, якщо $f(v_i) = a_i$.

На підставі (1.38) запишемо ДДНФ інверсної функції

$$\overline{f(v)} = \bigvee_{i=0}^{2^n-1} \overline{a_i} K_i(v).$$

Із даного співвідношення на підставі закону подвійності випливає

$$f(v) = \overline{\bigvee_{i=0}^{2^n-1} \overline{a_i} K_i(v)} = \prod_{i=0}^{2^n-1} \overline{\overline{a_i} K_i(v)} = \prod_{i=0}^{2^n-1} [a_i \vee \overline{K_i(v)}].$$

Із визначення макстермів випливає, що

$$f(v) = \prod_{i=0}^{2^n-1} [a_i \vee M_i(v)] \quad (1.40)$$

Така форма подання функції двох змінних називається ДКНФ.

Оскільки значення функції $a_i = 0$ чи 1 , то $a_i \vee M_i = M_i(v)$, якщо $a_i = 0$, і $a_i \vee M_i = 1$, якщо $a_i = 1$. Тому ДКНФ можна подати у вигляді

$$f(v) = \prod_{i_s} M_{i_s}(v), \quad (1.41)$$

де i_s - номери тих точок, в яких функція $f(v)$ дорівнює 0 .

Як приклад розглянемо функцію трьох змінних, задану таблицею 6. Оскільки тільки значення функції $a_0 = a_3 = a_4 = a_6 = 0$, то на підставі (1.41):

$$f(v) = M_0 \cdot M_3 \cdot M_4 \cdot M_6 = (x_1 \vee x_2 \vee x_3) \cdot (\overline{x_1} \vee x_2 \vee x_3) \cdot (x_1 \vee \overline{x_2} \vee x_3) \cdot (x_1 \vee x_2 \vee \overline{x_3}).$$

Це і є ДКНФ функції.

Досконалі нормальні форми в базисах І-НІ та АБО-НІ. Сукупність елементарних функцій, за допомогою яких можна записати будь-яку функцію $f(v)$, називається *функціонально повною системою функцій або базисом*. Із виразів (1.39) і (1.41) випливає, що для подання будь-якої функції $f(v)$ в ДДНФ і ДКНФ можна використовувати лише функції (операції) І, АБО і НІ, тобто сукупність цих функцій є базисом.

Перетворимо ДДНФ функції (1.38) за допомогою закону подвійного заперечення і закону подвійності

$$f(v) = \overline{\bigvee_{i=0}^{2^n-1} a_i K_i(v)} = \prod_{i=0}^{2^n-1} \overline{a_i K_i(v)}. \quad (1.42)$$

Дана форма подання функцій називається *досконалою нормальною формою (ДНФ)* в базисі І-НІ, оскільки вона вимагає використання тільки функцій (операцій) І-НІ.

Перетворимо тепер ДКНФ функції (1.40) за допомогою закону подвійного заперечення і закону подвійності

$$f(v) = \prod_{i=0}^{2^n-1} [a_i \vee M_i(v)] = \overline{\bigvee_{i=0}^{2^n-1} \overline{a_i \vee M_i(v)}}. \quad (1.43)$$

Дана форма подання функцій називається ДНФ в базисі АБО-НІ, оскільки вона вимагає використання тільки функцій (операцій) АБО-НІ.

На підставі (1.42) та (1.43) із ДДНФ і ДКНФ функції, що задана таблицею 1.6, можна отримати ДНФ цієї функції в базисах І-НІ та АБО-НІ:

$$f(v) = \overline{x_1 \cdot x_2 \cdot x_3 \cdot x_1 \cdot x_2 \cdot x_3 \cdot x_1 \cdot x_2 \cdot x_3 \cdot x_1 \cdot x_2 \cdot x_3}$$

$$f(v) = \overline{x_1 \vee x_2 \vee x_3 \vee x_1 \vee x_2 \vee x_3 \vee x_1 \vee x_2 \vee x_3 \vee x_1 \vee x_2 \vee x_3}$$

Ці дві форми являються основними в булівському базисі.

Приклади інших функціонально повних систем ЛФ базисів такі:

- Базис Жегалкіна: інверсія, сума по модулю 2, константа 1;
- Базис Запрет: константа 1, операція "запрет", інверсія;
- Базис імплікації: константа 0, операція імплікації, інверсія;
- Базис мажоритарний: інверсія, мажоритарні операції, константи 0 та 1.
- Базис рівнозначності: інверсія, диз'юнкція, константа 0, рівнозначність;
- Базис Пірса: операція Пірса;
- Базис Шеффера: операція Шеффера

Два останні базиса, які складаються кожний з однією функції, мають назву універсальних. Використовуються співвідношення:

$$\overline{x_2 \cdot x_1} = x_2 / x_1; \quad \overline{x_2 \vee x_1} = x_2 \downarrow x_1;$$

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

$$\overline{\overline{X_2 \cdot X_1}} = \overline{X_2} \cdot \overline{X_1}$$

За основу беруть МДФН та МКНФ булівського базиса. Перейти до універсального базиса Шиффера та Пірса можна від кожної мінімальної форми.

Вважається, що оптимальний перехід такий:

МДНФ → Базис Шеффера,

МКНФ → Базис Пірса.

Який перехід взяти, залежить від виду функції. Алгоритм переходу базується на використанні правил де Моргана та подвійної інверсії (Рис.1.8).

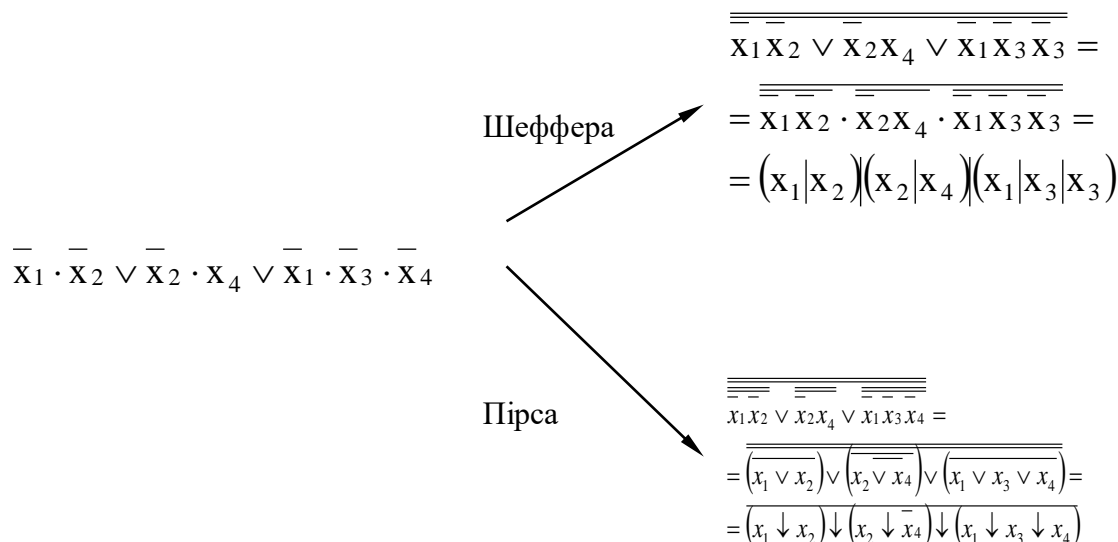


Рис.1.8 Приклад переходу до універсального базису Шиффера та Пірса

1.3 Мінімізація функцій перемикання

1.3.1. Алгебраїчний метод мінімізації

Побудова логічних схем на підставі булевих функцій, представлених у ДДНФ або ДКНФ, складених безпосередньо на основі таблиць істинності, у більшості випадків нецільно. Після складання структурної формули, можливо її перетворення до спрощеного виду, що приводить до меншої витрати логічних елементів у схемі. При проектуванні апаратури прагнуть до реалізації структурних схем, що забезпечують мінімальну витрату встаткування та мінімальну вартість за умови забезпечення заданого рівня надійності.

Під мінімізацією булевої функції найчастіше розуміють знаходження найбільш простого її подання з мінімальним числом вхідних у неї змінних. Після мінімізації, що розуміє в такому змісті, можливе виконання інших перетворень функцій, що мають метою скорочення числа типів елементів, приведення функцій до виду, зручному для реалізації на заданих елементах, наприклад, на елементах І-НІ та т.д.

Існує ряд методів мінімізації булевих функцій, серед яких можна виділити алгебраїчний і метод карт мінтермов.

При алгебраїчному методі мінімізації булевих функцій виконуються наступні операції. Булева функція представляється в ДДНФ, відшукуються сусідні що складають (мінтерми) і виробляється їхнє склеювання. Для зручності склеювання, якщо це необхідно, додають що складають, уже наявні у вихідній структурній формулі. Після виконання можливих склеювань робиться спроба виключати надлишкові складають застосуванням закону поглинання. У результаті виходить не надлишкова, або тупикова форма. Варто помітити, що виключення доданків може вироблятися різним образом, внаслідок чого булева функція може мати кілька тупикових форм. Мінімальні форми також є тупиковими. Тому процес відшукування мінімальних форм зводиться до одержання всіх тупикових форм заданої функції й вибору з них форм із

мінімальною кількістю букв. Потім виконуються перетворення на підставі теореми де Моргана та винесення загальних змінних за дужки.

Приклад. Знайти тупикові форми функції $F = \overline{ABC} + A\overline{BC} + ABC$. У цій функції закон склеювання можна застосувати до мінтермам \overline{ABC} і ABC (склеювання по змінній В) або $\overline{A}BC$ і ABC (склеювання по змінній А). У першому випадку одержимо

$$F = \overline{ABC} + AC(\overline{B} + B) = \overline{ABC} + AC = C(\overline{A}B + A)$$

у другому випадку

$$F = BC(\overline{A} + A) + A\overline{B}C = BC + A\overline{B}C = C(B + A\overline{B}).$$

Можна скористатися властивістю булевих функцій 1.6 (див. табл. 1.3) і до вихідного вираження функції F додати ще раз доданок ABC.

$$F = \overline{ABC} + ABC + A\overline{BC} + ABC$$

Виконавши угруповання доданків, одержимо

$$F = (\overline{ABC} + ABC) + (A\overline{BC} + ABC) = BC(\overline{A} + A) + AC(\overline{B} + B)$$

Після склеювання

$$F = C(B + A)$$

Всі три форми булевої функції є тупиковими. Остання форма булевої функції простіше двох попередніх, отриманих тільки на основі склеювання мінтермов. У цій формі виконуються тільки дві операції - логічного додавання й множення, у той же час у двох формах логічних операцій більше.

Розглянуті алгоритми відшукування тупикових форм є досить трудомісткими, тому що при виконанні операцій склеювання необхідно порівнювати всі можливі пари членів вихідного вираження. Існують методи, що дозволяють спростити пошук тупикових форм булевих функцій. Для цілей логічного проектування мікроелектронних апаратурі зручний метод карт мінтермов - Вейча та Карно.

1.3.2. Мінімізація булевих функцій за допомогою карт мінтермов

Одна із основних задач, що виникає при синтезі комбінаційних схем, є мінімізація функцій перемикання, які ці комбінаційні схеми реалізують. Чим простіший логічний вираз, тим простіша і дешевша комбінаційна схема, що реалізує його.

Спрощення функцій перемикання можна здійснювати за допомогою законів та тотожностей алгебри логіки. Такий шлях мінімізації трудомісткий, тому в більшості випадків використовують графічні методи мінімізації (діаграми Вейча, діаграми Венна, карти Карно). Перевагою таких методів є наочність і простота використання при невеликій кількості змінних ($n \leq 4$).

В 1953 році Моріс Карно опублікував статтю про розроблену ним систему графічного подання і спрощення функцій перемикання. Карта Карно показана на рис. 1.9. Тут і надалі приймемо позначення $f(v) = y$.

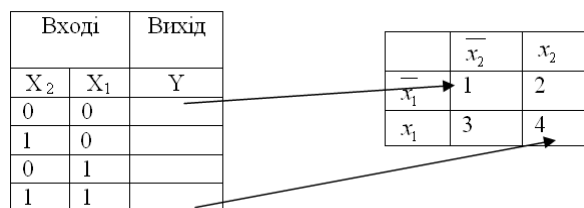


Рис. 1.9 Позначення квадратів на карті Карно

Чотири квадрати (1, 2, 3, 4) відповідають чотирьом можливим комбінаціям x_1, x_2 в таблиці істинності функції з двома змінними. При такому зображенні квадрат 1 відповідає добутку $\overline{x_1} \cdot \overline{x_2}$, квадрат 2 - $\overline{x_1} \cdot x_2$ і т.д.

Припустимо тепер, що потрібно скласти карту Карно для функції перемикання, записаній у ДДНФ: $\overline{x_1} \cdot \overline{x_2} \vee x_1 \cdot \overline{x_2} \vee x_1 \cdot x_2 = y$.

Розташуємо логічні одиниці у всіх квадратах, яким відповідають добутки у вихідній функції перемикання на рис. 1.10. Заповнена таким чином карта Карно тепер готова для побудови. Сусідні одиниці об'єднуються в один контур групами по дві, чотири або вісім одиниць.

	$\overline{x_2}$	x_2
$\overline{x_1}$		1
x_1	1	1

Рис. 1.10 Нанесення одиниць на карту Карно

Побудова контурів продовжується до тих пір, поки всі одиниці не опиняться всередині контурів. Кожний контур - це новий член спрощеної функції перемикавання. Відмітимо, що на рис. 1.10 отримано тільки два контури. Це означає, що нова, спрощена функція перемикавання буде складатися тільки з двох членів, що пов'язані функцією АБО. Тепер спростимо функцію перемикавання, приймаючи до уваги два контури на рис. 1.11.

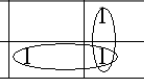
	$\overline{x_2}$	x_2
$\overline{x_1}$		
x_1		

Рис. 1.11. Об'єднання одиниць групами в один контур.

Взявши спочатку нижній контур, замітимо, що x_1 тут зустрічається у комбінації з x_2 і $\overline{x_2}$. У відповідності з правилами алгебри логіки x_2 і $\overline{x_2}$ доповнюють один одного і їх можна опустити.

Тоді в нижньому контурі залишається тільки член x_1 . Аналогічно цьому вертикально розташований контур вміщує x_1 і $\overline{x_1}$, які також можна опустити, залишивши тільки x_2 . Елементи x_1 і x_2 , що залишилися, об'єднуються функцією АБО, що приводить до спрощеної функції перемикавання $x_1 \vee x_2 = y$.

Алгоритм мінімізації функції перемикавання записується таким чином:

1. Переведення функції перемикавання в ДДНФ.
2. Нанесення одиниць на карту Карно.
3. Об'єднання сусідніх одиниць контурами, що охоплюють два, чотири або вісім квадратів.
4. Проведення спрощення, виключаючи члени, які доповнюють один одного всередині контуру.
5. Об'єднання членів, що залишилися (по одному у кожному контурі), функцією АБО.
6. Запис мінімізованої функції перемикавання в ДДНФ.

Карта Карно з трьома змінними.

Розглянемо функцію перемикавання

$$\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \vee \overline{x_1} \cdot x_2 \cdot \overline{x_3} \vee \overline{x_1} \cdot x_2 \cdot x_3 = y.$$

Карта Карно для цієї функції показана на рисунку 1.12.

Нижній контур вміщує x_2 і $\overline{x_2}$, їх можна опустити. Після цього у складі нижнього контуру залишається тільки член $x_1 \cdot \overline{x_3}$. У верхній контур входять x_3 і $\overline{x_3}$, які теж опускаються, після чого залишається тільки член $\overline{x_1} \cdot \overline{x_2}$.

Підсумкова функція перемикавання має вигляд $x_1 \cdot \overline{x_3} \vee \overline{x_1} \cdot \overline{x_2} = y$.

Суттєво, щоб карта Карно була складена саме так, як показано на рисунку 1.12. Замітимо, що в міру того, як зміщатись вниз по лівій частині карти, на кожному кроці змінюється лише одна змінна. Зверху зліва записаний добуток $\overline{x_1} \cdot \overline{x_2}$, а рядком нижче $\overline{x_1} \cdot x_2$ (заміна $\overline{x_2}$ на x_2). Далі, при пересуванні від $\overline{x_1} \cdot x_2$ до $x_1 \cdot x_2$ вниз переходить в x_1 і т.д. Якщо карту Карно скласти неправильно, то вона не принесе очікуваного результату.

		$\overline{x_3}$	x_3
$\overline{x_1}$	$\overline{x_2}$	1	1
$\overline{x_1}$	x_2		
x_1	x_2	1	
x_1	$\overline{x_2}$	1	

		$\overline{x_3}$	x_3
$\overline{x_1}$	$\overline{x_2}$	1	1
$\overline{x_1}$	x_2		
x_1	x_2	1	
x_1	$\overline{x_2}$	1	

Рис. 1.12 Спрощення перемикаючої функції на основі карти Карно

Карти Карно з чотирма змінними. Таблиця істинності для чотирьох змінних вміщує 16 можливих комбінацій. Розглянемо функцію перемикання

$$\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot x_4 \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot x_4 \vee \overline{x_1} \cdot x_2 \cdot \overline{x_3} \cdot \overline{x_4} \vee \overline{x_1} \cdot x_2 \cdot \overline{x_3} \cdot x_4 \vee \overline{x_1} \cdot x_2 \cdot x_3 \cdot \overline{x_4} \vee \overline{x_1} \cdot x_2 \cdot x_3 \cdot x_4 = y.$$

Карта Карно для функції перемикання з чотирма змінними допускає 16 можливих комбінацій x_1, x_2, x_3, x_4 (рис. 1.13).

		$\overline{x_3} \overline{x_4}$	$\overline{x_3} x_4$	$x_3 \overline{x_4}$	$x_3 x_4$
$\overline{x_1}$	$\overline{x_2}$		1	1	
$\overline{x_1}$	x_2		1	1	
x_1	x_2				
x_1	$\overline{x_2}$	1	1		

		$\overline{x_3} \overline{x_4}$	$\overline{x_3} x_4$	$x_3 \overline{x_4}$	$x_3 x_4$
$\overline{x_1}$	$\overline{x_2}$		1	1	
$\overline{x_1}$	x_2		1	1	
x_1	x_2				
x_1	$\overline{x_2}$	1	1		

Рис. 1.13. Мінімізація функції перемикання з чотирма змінними

Ці комбінації подані відповідно 16-ма квадратами карти. Нанесемо на карту 6 одиниць, які відповідають 6-ти членам у заданій функції перемикання. Групи із двох і чотирьох одиниць об'єднані контурами. Нижній контур із двох одиниць дає можливість опустити x_4 і $\overline{x_4}$. Після цього в ньому залишається член $\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3}$. Далі у верхньому контурі із чотирьох одиниць попарно опускаються x_3 і $\overline{x_3}$, x_4 і $\overline{x_4}$, так, що в результаті цього верхній контур дає член $\overline{x_1} \cdot x_4$. Спрощена функція перемикання в ДДНФ має вигляд $\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \vee \overline{x_1} \cdot x_4 = y$.

Інші різновиди карт Карно.

Розглянемо функцію перемикання

$$\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot x_4 \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot x_4 = y.$$

Чотири її члени подають на карті Карно чотирма одиницями (рисунк 1.14).

	$\overline{x_3} \overline{x_4}$	$\overline{x_3} x_4$	$x_3 \overline{x_4}$	$x_3 x_4$
$\overline{x_1} \overline{x_2}$				
$\overline{x_1} x_2$	1			1
$x_1 \overline{x_2}$	1			1
$x_1 x_2$				

Рис. 1.14. Спрощення функції перемикання на основі карти Карно

На цьому ж рисунку поданий правильний спосіб побудови контуру. Зверніть увагу на те, що при цьому попарно опускаються члени x_1 і $\overline{x_1}$, x_3 і $\overline{x_3}$. Спрощена функція перемикання має вигляд $x_2 \cdot \overline{x_4} = y$.

Інші способи побудови контуру покажемо на прикладі функцій перемикання (рисунок 1.15 та рисунок 1.16 відповідно):

$$\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot x_4 \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot x_4 = y,$$

$$\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot x_4 \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot \overline{x_4} \vee \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot x_4 = y$$

Мінімізовані функції відповідно будуть мати вигляд:

$$\overline{x_2} \cdot \overline{x_3} = y,$$

$$\overline{x_2} \cdot \overline{x_4} = y.$$

	$\overline{x_3} \overline{x_4}$	$\overline{x_3} x_4$	$x_3 \overline{x_4}$	$x_3 x_4$
$\overline{x_1} \overline{x_2}$	1	1		
$\overline{x_1} x_2$				
$x_1 \overline{x_2}$				
$x_1 x_2$	1	1		

Рис. 1.15. Перший варіант спрощення

	$\overline{x_3} \overline{x_4}$	$\overline{x_3} x_4$	$x_3 \overline{x_4}$	$x_3 x_4$
$\overline{x_1} \overline{x_2}$	1			1
$\overline{x_1} x_2$				
$x_1 \overline{x_2}$				
$x_1 x_2$	1			1

Рис. 1.16. Другий варіант спрощення

1. 3.3. Інтегральні цифрові мікросхеми логічних елементів

Логічний елемент - це електронний прилад, що реалізує одну з логічних функцій. В склад серій мікросхем, що розглядаються, входить велике число логічних елементів. На принциповій схемі логічний елемент зображають прямокутником, всередині якого ставиться зображення покажчика функції. Лінії з лівої сторони прямокутника показують входи, з правої - вихід елемента. На рис.1.17 зображені основні логічні елементи, що використовуються у цифрових приладах:

Елемент І (кон'юнктор): $y = x_1 \cdot x_2$ (а);

елемент АБО (диз'юнктор): $y = x_1 \vee x_2$ (б);

елемент НІ (інвертор 1): $y = \overline{x}$ (в).

Окрім означених існує множина логічних елементів, що виконують більш складні логічні перетворення. Ці перетворення є комбінаціями найпростіших логічних операцій. До числа таких елементів відносяться:

елемент І-НІ: $y = \overline{x_1 \cdot x_2}$;

елемент АБО-НІ: $y = \overline{x_1 \vee x_2}$;

елемент І-АБО: $y = \overline{x_1 \cdot x_2 \vee x_3 \cdot x_4}$;

елемент І-АБО-НІ: $y = \overline{x_1 \cdot x_2 \vee x_3 \cdot x_4}$;

суматор за модулем 2: $y = \overline{x_1 \cdot x_2 \vee x_1 \cdot x_2}$.

Число входів в логічних елементах різного призначення може бути різним, але входи кожного елемента рівнозначні. Деякі з них можуть при роботі в конкретних приладах не використовуватися. Входи, які не

використовуються в схемах І, І-НІ з'єднують із +U_{дж.}, а в схемах АБО, АБО-НІ, суматора за модулем 2 - із загальним проводом (0 В).

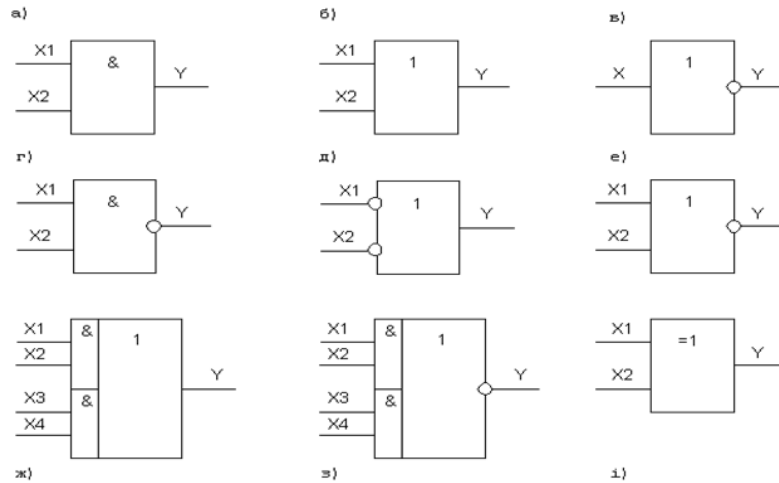


Рис. 1.17. Графічні позначення логічних елементів.

1.3.4. Синтез схеми ЦП за заданою логічною функцією

Основні етапи синтезу цифрових пристроїв.

Перший етап:

- а) Словесна постановка та формулювання задачі синтезу.
- б) Задається базис, в якому потрібно побудувати схему.
- в) Задаються умови функціонування та обмеження.

Другий етап: Отримання таблиці (або таблиць) істинності — таблиці функціонування пристрою.

Третій етап: Запис логічної функції в аналітичному виді — це завжди запис в булівському базисі, мінімізація функції відомими методами за допомогою діаграм Вейча або карт Карно.

Четвертий етап: Перехід до заданого базису та побудова схеми.

При побудові схеми враховуються обмеження на деякі параметри:

- кількість входів логічних елементів,
- глибина схем (впливає на затримку сигналів),
- уникнення гонок в схемі та інші.

В інженерній практиці кожен етап іноді потребує творчого підходу і тому може доповнюватися допоміжними етапами (підетапами).

Наприклад, може статися, що логічні функції залежать від змінних, кількість яких більше 4. Діаграми Вейча при цьому не дуже допомагають отримати мінімальну форму. В цьому випадку може допомогти етап декомпозиції: розділ пристроїв на окремі вузли, які менш складні. Іноді може допомогти деяке ускладнення схеми, щоб при цьому можна було б виділити типові вузли, для яких схеми уже відомі.

На підставі отриманої логічної формули можна побудувати комбінаційну схему, що складається з елементів АБО, І, НІ, яка буде мати задані властивості, тобто реалізує задану булеву функцію. Логічні елементи, призначені для виконання вказаних у формулі операцій, розташовуються на схемі, починаючи від входів, у такому ж порядку, в якому виконуються логічні операції. Спочатку зображуються інвертори для отримання вхідних змінних, потім елементи І і потім елемент АБО. Відповідно з формулою до елементів підводяться або вхідні сигнали, або результати виконання логічних операцій над цими сигналами з виходів інших елементів.

Приклад. Для булевої функції $F = \bar{A}BC + A\bar{B}C + ABC\bar{C} + ABC$ синтезуємо схему, показану на рис. 1.18.

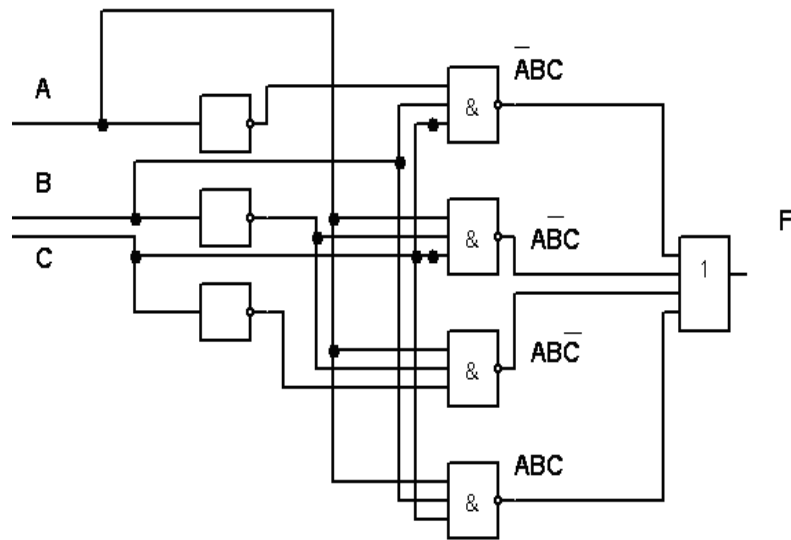


Рис.1.18 Схема ЦП синтезованого по булевої функції

1.3.5 Методи спрощення логічних задач

Виробники ІС полегшили вирішення багатьох задач побудови комбінаційних логічних схем, створивши так звані *селектори даних*, використання яких часто дозволяє вирішити складну логічну задачу за допомогою лише одної ІС. Селектор даних складається із великої кількості логічних елементів, розміщених у одному корпусі ІС.

Селектор даних типу «1 із 8» наведений на рис. 1.19. Зверніть увагу на наявність з лівої сторони селектора восьми *інформаційних входів*, пронумерованих цифрами від 0 до 7, і трьох *селекторних входів* у нижній частині селектора даних, позначених А, В і С.

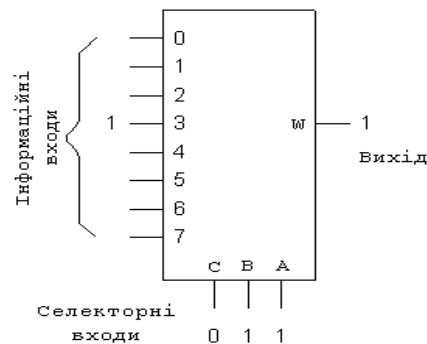


Рис. 1.19. Умовне позначення селектора даних «1 із 8»

Основне призначення селектора даних - пересилка з певного входу на вихід W. Вибір входу, з якого пересилаються дані, визначається двійковим кодом, що поступає на селекторні входи (рис. 1.20). Селектор даних працює за тим самим принципом, що і поворотний перемикач. Для зміни входу для пересилки даних змінюється двійковий код на селекторних входах.

Наведемо приклад конструювання логічних схем. Нехай задано спрощену функцію перемикання:

$$y = x_1 \cdot x_2 \cdot x_3 \cdot x_4 \vee x_1 \cdot x_2 \cdot x_3 \cdot x_4 \vee x_1 \cdot x_2 \cdot x_3 \cdot x_4 \vee x_1 \cdot x_2 \cdot x_3 \cdot x_4 \vee x_1 \cdot x_2 \cdot x_3 \cdot x_4 \vee x_1 \cdot x_2 \cdot x_3 \cdot x_4 \vee x_1 \cdot x_2 \cdot x_3 \cdot x_4 \vee x_1 \cdot x_2 \cdot x_3 \cdot x_4$$

Наведемо на рис.1.21 схему, що відповідає цій функції перемикання. Якби для побудови цієї функції було використано стандартні ІС, то їх потрібно було б 6-9 штук. Таке рішення було б дуже неекономічним. З меншими затратами ця задача вирішується за допомогою селектора даних. З метою пояснення функція перемикання на рис. 2.38 записана у вигляді таблиці істинності. До цієї таблиці доданий селектор даних «1 із 16». Відмітимо, що логічні 0 і 1 подаються на шістнадцять інформаційних входів селектора у відповідності із стовпцем значень виходу у таблиці істинності. Ці з'єднання залишаються постійними для даної таблиці істинності. На селекторні входи (D, C, B і A) подаються двійкові числа, що відповідають різним комбінаціям входів в таблиці істинності. Якщо, наприклад, на селекторні входи D, C, B, A подано двійкове число 0000, то на вихід W селектора даних поступає логічна 1.

Перший рядок таблиці істинності вимагає, щоб на виході W логічна одиниця появлялась в тому випадку, коли всі змінні D, C, B і A приймають значення 0. Якщо на селекторні входи D, C, B і A поступить двійкове число 0001,

то у відповідності до вимог таблиці істинності на виході W появиться логічний 0. Будь-яка комбінація вхідних змінних D, C, B і A генерує необхідний вихід згідно з таблицею істинності.

Використання селектора даних є простим, зручним і ефективним методом вирішення задач побудови комбінаційних логічних схем. Селектори даних називаються також *мультиплексорами*.

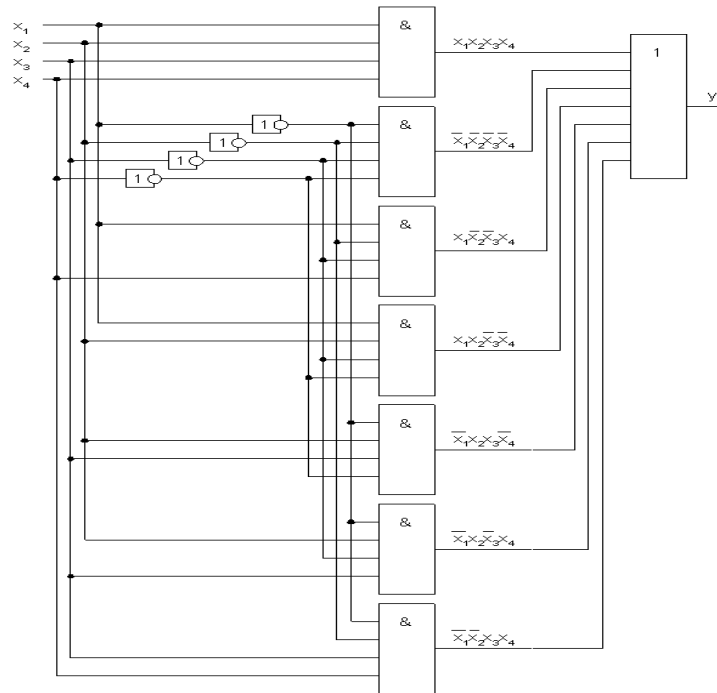


Рис.1.20. Логічна схема для функції перемикання

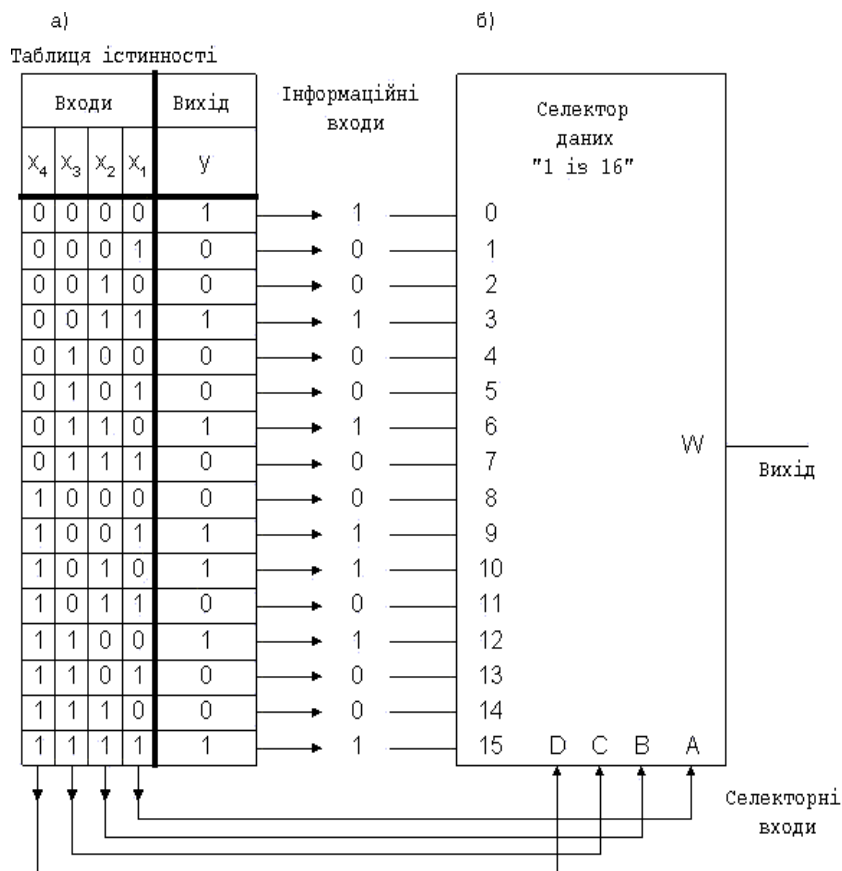


Рис.1.21. Вирішення логічної задачі за допомогою селектора даних

Список питань для самоконтролю:

1. Яка різниця між цифровими і аналоговими сигналами?
2. Що таке квантування?
3. Що таке система числення?
4. Як утворюють числа в різних системах числення?
5. Що таке двійково-десятковий код?
6. Як перевести число із двійкової системи числення в десяткову і навпаки?
7. Як перевести число із шістнадцяткової системи числення в десяткову і навпаки?
8. Як проводять додавання і віднімання чисел у двійковій системі числення?
9. Що таке біт і байт?
10. Що таке диз'юнкція, кон'юнкція і заперечення?
11. Які властивості задовольняє закон заперечення?
12. Назвіть аксіоми алгебри логіки.
13. Назвіть закони алгебри логіки.
14. Що таке операція "сума за модулем два"?
15. Назвіть властивості функцій перемикавання.
16. Що таке закон подвійності?
17. Що таке теорема розкладу?
18. Що таке первинні терми?
19. Що таке макстерми і мінтерми?
20. Як записується диз'юнктивна нормальна форма?
21. Як записується кон'юнктивна нормальна форма?
22. Що таке функціонально повна система функцій?
23. Що таке досконально нормальна форма?
24. Як мінімізуються функцій перемикавання за допомогою карт Карно?

РОЗДІЛ 2. ЛОГІЧНІ ЕЛЕМЕНТИ ЦИФРОВОЇ ТЕХНІКИ

2.1 Поняття комбінаційна схема та цифровий автомат

Будь-який символ інформації в цифрових приладах кодується в двійковому алфавіті, тому сигнали можуть приймати тільки два значення: високий або низький рівень напруги, наявність або відсутність імпульсу напруги. Обов'язковою умовою при цьому є можливість впевненого розпізнавання елементами цифрових схем двох значень сигналів, відповідних символам 0 та 1, в умовах зміни температури довкілля, напруги джерела живлення, вплив інших дестабілізуювальних факторів.

Ми вже говорили, що значення рівня сигналів (U_c) елементами цифрових приладів сприймаються не безперервно, а в дискретні моменти часу, інтервал між якими називають робочим тактом T . Як правило, за один робочий такт у цифрових приладах здійснюється одне елементарне перетворення кодових слів, що поступили на вхід. Дискретизація часу забезпечується спеціальними приладами управління, що виробляють *синхронізуючі імпульси* (СІ). В дискретних приладах використовують два способи подання інформації: потенційний та імпульсний. При потенційному способі значенням логічного 0 та логічної 1 відповідають напруги низького та високого рівня. Якщо логічному нулю відповідає напруга низького рівня, а логічній одиниці – високого, то таку логіку називають позитивною, і навпаки, якщо за логічний нуль приймають напругу високого, а за логічну одиницю – напругу низького рівня, то таку логіку називають негативною. В подальшому будемо розглядати прилади тільки з позитивною логікою.

Далі в основному використовуються терміни «напруга високого і низького рівня» (сигнали високого і низького рівня), відповідні рівню логічної 1 і логічного 0.

Інформація в цифрових приладах може бути подана в послідовному і паралельному кодах. При використанні послідовного коду кожний такт відповідає одному розряду двійкового коду. Номер розряду визначається номером такту, що відраховується від такту, який збігається з початком подання коду.

Графіки, показані на рис. 2.1, ілюструють послідовний код байтового двійкового числа 10011011 при потенційному і імпульсному способах подання інформації. При першому способі (рис. 2.1, а) сигнал зберігає низький або високий рівень протягом одного або декількох тактів. У моменти переходу сигналу від одного рівня до іншого його значення є невизначеним. При імпульсному способі подання цифрової інформації (рис. 2.1, б) одиничному і нульовому значенню двійкової змінної відповідає наявність або відсутність імпульсу кінцевої

тривалості. При послідовному коді числа всі його розряди можуть бути зафіксовані на одному елементі і передані по одному каналу передачі інформації. Для передачі усього числа вимагається вісім тактів (рис. 2.1, в).

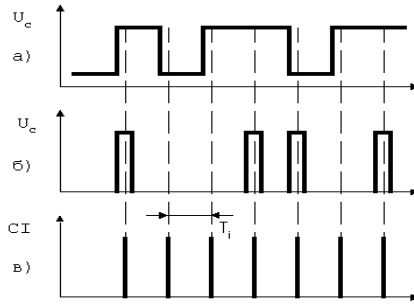


Рис. 2.1. Послідовний код двійкового числа при потенціальному та імпульсному способах подання інформації

Паралельний код дозволяє істотно скоротити час обробки і передачі інформації. Для прикладу рис. 2.2 ілюструє паралельний код семирозрядного числа 1101101. В цьому випадку як при імпульсному (рис. 2.2, а), так і при потенційному (рис. 2.2, б) способах подання інформації всі розряди двійкового коду подані в одному часовому такті, можуть фіксуватися окремими елементами і передаватися по розділних каналах (розрядних шинах). Цифрові прилади, що проводять обробку і перетворення на її входи інформації, що надходить, називають цифровими автоматами.

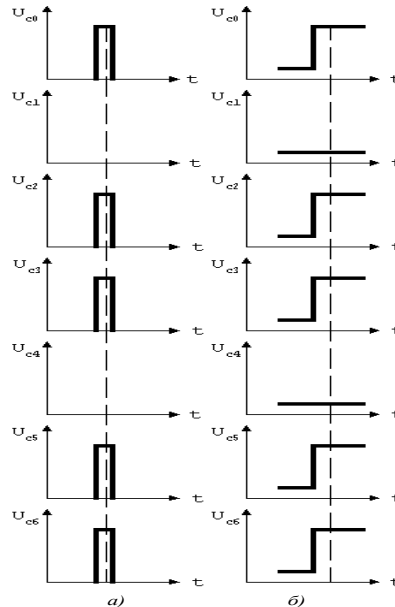


Рис. 2.2. Паралельний код двійкового числа при потенційному і імпульсному способах подання інформації

Умовне графічне зображення найпростішого цифрового автомата показано на рис. 2.3. На входи автомата подають комбінацію двійкових змінних X_1, X_2, \dots, X_n , з виходу знімають комбінацію двійкових змінних Y_1, Y_2, \dots, Y_m . На входах і виході цифрового автомата діють сигнали логічних 0 та 1, що називаються двійковими. Задача побудови цифрового автомата, що виконує певні дії над двійковими сигналами, полягає у виборі елементів і способу їхнього сполучення, що забезпечують задане перетворення. Ці задачі вирішує математична логіка або алгебра логіки.

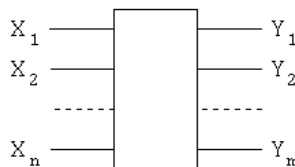


Рис. 2.3. Умовне позначення найпростішого цифрового автомата

Прилади, що формують функції алгебри логіки, називають *логічними* або *цифровими* і класифікують за різноманітними ознаками. Цифрові прилади за характером інформації на входах і виході поділяють на прилади послідовної, паралельної і змішаної дії.

На входи приладу послідовної дії символи кодових слів надходять не водночас, а один за одним (рис. 2.4). У такій же послідовності формується вихідне кодове слово.

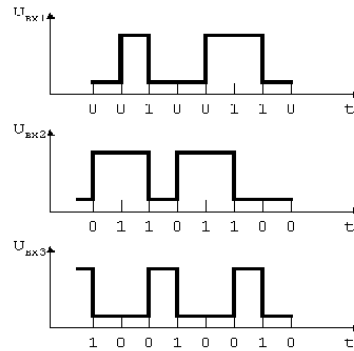


Рис. 2.4. Подання інформації у цифровому приладі послідовної дії

Прилад, що розглядається, формує на виході сигнал $U_{вих}$, код якого рівний сумі кодів вхідних сигналів $U_{вх1}$ і $U_{вх2}$. Для реалізації приладу паралельної дії, що виконує аналогічну функцію, необхідні дві групи входів по вісім розрядів у кожній групі і вісім виходів (у відповідності із розрядністю вихідного слова). Відомі також прилади змішаного типу, в яких, наприклад, вхідне слово подається в паралельній формі, а вихідне - в послідовній (це - перетворювачі коду).

За схемним рішенням і характером зв'язків між вхідними і вихідними змінними з обліком їхньої зміни по тактах роботи розрізняють комбінаційні і послідовні цифрові прилади.

В комбінаційних приладах сукупність сигналів на виході в кожний конкретний момент часу повністю визначається вхідними сигналами, діючими в цей момент на його входах. Якщо вхідні і вихідні функції в n -такті позначити як X_n і Y_n , то зв'язок між ними буде визначатися виразом

$$Y^n = l(X^n),$$

де l - знак виконуваного приладом логічного перетворення.

Цифрові прилади, на відміну від аналогових, дозволяють реалізувати перетворення практично будь-якого виду, але комбінаційні прилади не мають пам'яті. В цифрових приладах послідовнісного типу (або автоматах з пам'яттю) значення вихідних змінних Y_n в n -такті визначаються не тільки значеннями вхідних змінних X_n , діючих в даний момент часу, але й залежать від внутрішніх станів приладу C_n . В свою чергу, внутрішні стани приладу C_n залежать від значень змінних, що діяли на вході в попередні такти. Таким чином, послідовнісні цифрові прилади зберігають відомості про попередні такти роботи приладу і тому мають пам'ять. Функціонування послідовнісного приладу аналітично можна записати у вигляді

$$\begin{aligned} Y_n &= J(X^n, C^n); \\ C_n &= F(X^{n-1}, C^{n-1}), \end{aligned}$$

де X^{n-1} і C^{n-1} — відповідно набір вхідних змінних і внутрішніх станів приладу в попередній такт.

Прикладом послідовнісного приладу може бути лічильник імпульсів, стани виходу якого залежать від загального числа імпульсів, що надійшли на його вхід. Набір змінних на виході лічильника в n -такті залежить від наявності імпульсу на вході лічильника в такті, а також від станів лічильника, що визначаються загальним числом імпульсів, що надійшли на його вхід у тактах $1, 2, \dots, n-1$.

2.2 Класифікація та характеристики елементної бази цифрових пристроїв

Логічні елементи (ЛЕ) майже всі виробляються у вигляді інтегральних мікросхем (ІМС) на базі напівпровідникових пристроїв (транзисторів та діодів). Класифікація ЛЕ ведеться в основному за типом активних компонентів, які використовують для їх побудови, та з врахуванням зв'язку між цими компонентами

У процесі розвитку мікросхемотехніка виділялося кілька типів базових логічних елементів (БЛЕ), що мають досить високі значення параметрів і зручних для інтегрального виконання. Класифікація їх наведена на рис. 2.5.

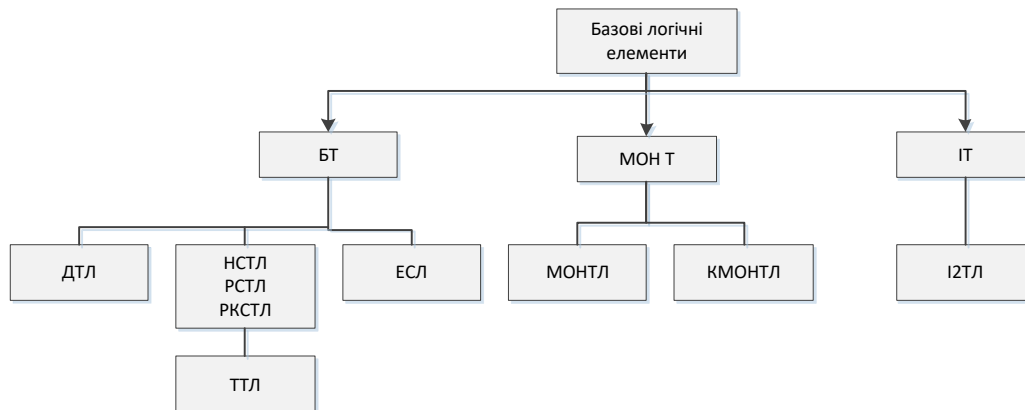


Рис.2.5 Класифікація типових логічних елементів

Активними приладами є біполярні транзистори (БТ), МДП-транзистори (МДПТ) і транзистори з інжекційним живленням (ІТ).

Залежно від типу активних приладів, за допомогою яких виконується вхідна і вихідна операції, базові логічні елементи і цифрові ІС на їх основі прийнято підрозділяти на наступні типи логіки:

- діодні ЛЕ (ДЛ);
- діодно-транзисторні ЛЕ (ДТЛ);
- транзисторно-транзисторні ЛЕ (ТТЛ);
- емітерно-зв'язані ЛЕ (ЕЗЛ);
- ЛЕ на польових транзисторах (МОНЛ);
- ЛЕ з інжекційним живленням (ІЛ).

Для порівняльної оцінки властивостей ЛЕ вводиться ряд характеристик, основними з яких є: статична характеристика передачі, швидкодія, кількість входів, навантажувальна здатність, завадостійкість та споживана потужність.

Розробка електронних пристроїв почалася приблизно 50 років тому. Цифрові пристрої в залежності від використаних елементів та технології їх виготовлення відносять до того чи іншого покоління. Елементною базою цифрової техніки першого покоління були електронні лампи, електромагнітні реле, резистори, конденсатори та інші дискретні елементи. Швидкодія ЕОМ на базі цих пристроїв складала кілька десятків тисяч операцій в секунду. Основними недоліками, що гальмували їх широке використання в різних сферах економіки були складність, мала швидкодія, значна споживана потужність. В наступні роки з появою транзисторів і інтегральних мікросхем стан суттєво змінився. Елементною базою ЕОМ другого покоління були напівпровідникові прилади і мініатюрні дискретні деталі, як запам'ятовувальні пристрої використовувались ферит-транзисторні комірки. Швидкодія напівпровідникових елементів ЕОМ у порівнянні з ламповими виросла більше ніж на порядок.

Основною елементною базою ЕОМ третього покоління стали мікросхеми з малим і середнім рівнем інтеграції. Після появи інтегральних мікросхем фізики і інженери направили свої зусилля на розробку удосконалених технологічних процесів, що дозволяють здійснити більш компактне розміщення елементів на одному кристалі. З 1972 року почалася інтенсивна розробка ЕОМ четвертого покоління. Використання у виробництві оптичних методів технології отримання тонких плівок і способів осадження тонких плівок у вакуумі призвело до створення великих інтегральних схем (ВІС), які вміщують десятки і сотні тисяч елементів і з'єднань. З кожним роком вдвічі збільшувалось число елементів на одному кристалі, в результаті чого в 1980 році рівень інтеграції досяг порядку сотень тисяч елементів на кристалі - це вже надвеликі інтегральні мікросхеми.

В залежності від технології виготовлення інтегральні схеми підрозділяються на серії (сімейства), які відрізняються фізичними параметрами базових елементів, а також числом мікросхем, що входять в їхній склад. В наш час розроблено кілька десятків технологій виготовлення інтегральних схем (ІС). Найбільш широке використання знайшли ІС, виготовлені за технологіями: *транзисторно-транзисторна логіка (ТТЛ)*; *комплементарні метал-окисел-напівпровідник (КМОН)*; *n-МОН*. Кожна технологія безперервно вдосконалюється з метою збільшення швидкодії ІС, зменшення споживаної потужності і збільшення порядку інтеграції - числа елементів, що розміщені на кристалі заданої площі.

ТТЛ ІС. В таблиці 2.1 наведені основні параметри базових елементів різних серій зарубіжних ІС, що виготовлені за ТТЛ-технологією, а в таблиці 2.2 – відповідність зарубіжних ІС вітчизняним.

Крім серій SN74, зарубіжними фірмами випускаються серії SN54 і SN84, які відрізняються тільки температурним діапазоном і допустимим відхиленням напруги джерела живлення від номіналу (таблиця 2.3).

Найбільш важливим параметром, що характеризує якість серій ІС, є робота перемикачів - добуток середнього часу затримки сигналів у вентилі $t_{з,сеп}$ на потужність споживання вентиля P (таблиця 2.1).

Інші параметри мають такі позначення:

- $I_{вх}^0$ - вхідний струм при подачі на вхід рівня логічного нуля;
- $I_{вх}^1$ - вхідний струм при подачі на вхід рівня логічної одиниці;
- F_{max} - максимальна частота перемикавання;
- $I_{вих}^0$ - вихідний струм при рівні логічного нуля на виході;
- $I_{вих}^1$ - вихідний струм при рівні логічної одиниці на виході;
- n - навантажувальна здатність (відношення $I_{вих}^0 / I_{вх}^0$).

Таблиця 2.1 - Основні параметри ІС ТТЛ-технології

Серія ІС	$t_{3, сер.}, нс$	$P, мВт/вент$	$I_{вх}^1, мкА$	$I_{вх}^0, мА$	$F_{max}, МГц$	$I_{вих}^1, мкА$	$I_{вих}^0, мА$	n	$t_{3, сер.}, нс$
SN74	10	10	40	-1.6	35	-	-	10	100
SN74L	33	1	10	-0.18	3	-400	3.6	10	33
SN74H	6	22	50	-2	50	-200	20	10	132
SN74LS	9.5	2	20	-0.36	45	-500	8	20	19
SN74S	3	19	50	-2	125	-400	20	10	57
SN74AL	4	1	10	-0.2	50	1000	8	40	4
SN74AS	1.5	22	-	-	200	-400	20	100	33
SN74F	2	4	20	-0.6	130	-1000	20	33	8

Таблиця 2.2 - Відповідність зарубіжних ІС вітчизняним

Серія ІС		Серія ІС	
Вітчизняна	Зарубіжна	Вітчизняна	Зарубіжна
155	SN74	133	SN54
158	SN74L	136	SN54L
131	SN74H	130	SN54H
555	SN74LS	533	SN54LS
531	SN74S	530	SN54S
KP1533	SN74ALS	1533	SN54ALS
KP1531	SN74F	1531	SN54F

Серії ІС розшифровуються таким чином:

SN74/SN54 - стандартна - 1963р.;

SN74H/SN54H - High speed - швидкодіюча - 1967р.;

SN74L/SN54L - Low power - малопотужна - 1967р.;

SN74S/SN54S - з використанням діодів Шоттки - 1969р.;

SN74LS/SN54LS - Low power Schottky - малопотужна з використанням діодів Шоттки - 1971р.;

SN74AS/SN54AS - Advanced Schottky - удосконалена з використанням діодів Шоттки - 1982р.;

SN74ALS/SN54ALS - Advanced Low power Schottky - удосконалена малопотужна з діодами Шоттки - 1980р.;

SN74F/SN54F - Fairchild's Advanced Schottky TTL - удосконалена з діодами Шоттки фірми Fairchild - 1979р.;

Діоди Шоттки не допускають режиму глибокого насичення транзисторів, що значно збільшує швидкість перемикавання вентилів.

Всі серії сумісні за входами і виходами: вихідна напруга логічного нуля $U_{вих}^0 \leq 0.4$ В; вхідна напруга логічного нуля

$U_{вх}^0 \leq 0.4$ В; вихідна напруга логічної одиниці $U_{вих}^1 \geq 2.4$ В; вхідна напруга логічної одиниці $U_{вх}^1 \geq 2.4$ В.

Таблиця 2.3 - Відмінності між параметрами серій зарубіжних ІС

Серія ІС	Діапазон робочих температур, °C	Напруга живлення, В
SN54	-55...+125	4.5 ... 5.5
SN74	0...+70	4.75 ... 5.25
SN84	-25...+85	4.75 ... 5.25

КМОН ІС характеризуються малим споживанням потужності в статичному режимі і більшою завадостійкістю у порівнянні з ТТЛ ІС. В таблиці 2.4 приведено відповідність вітчизняних і зарубіжних серій ІС.

У КМОН ІС швидкодія суттєво залежить від напруги джерела живлення (із збільшенням напруги живлення збільшується швидкодія).

У порівнянні з ТТЛ ІС необхідно відмітити такі переваги КМОН ІС:

- мала споживана потужність у діапазоні частот до 2 МГц;
- великий діапазон напруги живлення (3...15 В);
- можна використовувати не стабілізоване джерело живлення;
- дуже високий вхідний опір ($10^3...10^6$ МОм);
- велика навантажувальна здатність ($n = 50$; $n = 1000$ на частотах до 10 кГц);
- незначна залежність характеристик від температури.

Недоліки КМОН ІС серій 561 і 1561:

- завищений вхідний опір (0.5 ... 1 кОм);
- вплив ємності навантаження і напруги живлення на час затримки, тривалість фронтів і споживану потужність;
- великі тривалості затримок і фронтів.

Таблиця 2.4 - Відповідність між вітчизняними та зарубіжними КМОН ІС

Серія ІС		Фірма	Напруга живлення, В
Вітчизняна	Зарубіжна		
164, 176	CD4000	RCA	9 і 3...15
564, 561	CD4000A	RCA	3...15
-	MC14000A	Motorola	-
КР1561	CD4000B	RCA	3...18
-	MC14000B	Motorola	-
1561	54НС	NationalSemiconductor Corp.Motorola	2...6

При побудові схем на ІС часто використовуються не всі їх входи. Виходячи з логіки роботи схеми, на ці входи потрібно подати або логічний рівень 0, або 1. Логічний рівень 0 як в ТТЛ, так і в КМОН ІС подається під'єднанням не використовованого входу до корпусу (0 В). Логічний рівень 1 подається на невикористовувані входи під'єднанням їх до джерела напруги живлення, але входи ТТЛ ІС рекомендується приєднувати до джерела через струмообмежувальний резистор для захисту від стрибків напруги, що виникають, наприклад, при ввімкненні живлення. Якщо вхід ТТЛ ІС не приєднаний ні до корпусу, ні до джерела живлення, то ІС буде працювати так, як ніби на неї поданий логічний рівень 1, але надійність роботи буде низькою (імпульсні завади, що викликані перемикаваннями сусідніх входів і виходів, можуть привести до непередбаченого спрацювання ІС). В КМОН ІС не можна залишати не під'єднаними жодного не використовованого входу).

Статична характеристика передачі є залежністю $U_{вих}=f(U_{вх})$, де $U_{вх}$ — напруга на одному з входів. На інші входи при знятті цієї характеристики подається постійний рівень, який відповідає або 0 (для елементів АБО, АБО-НЕ), або ж 1 (для елементів І, І-НЕ). Типова форма характеристики для інвертуючого елемента показана на рис. 2.6. Тут же позначені її основні параметри:

логічні рівні вихідної напруги $U_{вих}^0$ та $U_{вих}^1$, які відповідають 0 та 1; такі ж рівні $U_{вх}^0$ та $U_{вх}^1$ показані й на вході;

$$\text{величина логічного перепаду } U_m = U_{вих}^1 - U_{вих}^0 ;$$

порогові рівні $U_{пор}^1$ та $U_{пор}^2$, які визначаються умовно в тих точках характеристики, де дотична до неї проходить під кутом 45° до горизонту;

ширина активної області $\Delta U = U_{пор}^2 - U_{пор}^1$;

середній коефіцієнт передачі (підсилення) в активній області $K_{ср} = U_m / \Delta U$.

Логічні можливості елемента характеризуються логічною функцією, яку він виконує, а також такими двома параметрами:

$K_{об}$ — коефіцієнтом об'єднання по входу, тобто максимальною кількістю входів, обмеженою в основному кількістю виводів з корпусу МС; як правило буває від 2 до 8;

$K_{розг}$ — коефіцієнтом розгалуження за виходом, тобто максимальною кількістю елементів такого ж типу, які можна підключити паралельно до виходу без порушення нормального функціонування даного елемента. В залежності від типу елемента цей параметр, який виражає навантажувальну здатність, може бути від 3 до 100. Збільшення $K_{об}$ та $K_{розг}$ дозволяє зменшити загальну кількість елементів, необхідну для побудови логічної схеми.

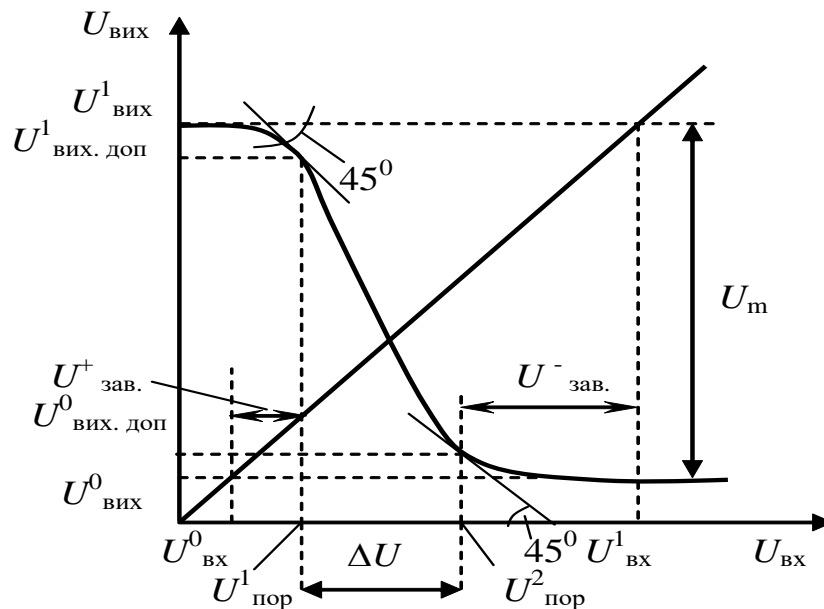


Рис. 2.6. Типова статична характеристика інвертуючого логічного елемента

Завадостійкість визначається допустимою величиною напруги завади $U_{зав}$, при якій не відбувається помилкового спрацювання елемента (переходу з 0 в 1 або навпаки). Для грубої оцінки завадостійкості застосовують величини $U_{зав}^+ = U_{пор}^1 - U_{вих}^0$ та $U_{зав}^- = U_{вих}^1 - U_{пор}^2$ (рис. 2.6).

Допустима динамічна (імпульсна) завада може перевищувати ці значення, і вона може бути тим більша, чим коротший імпульс завади.

Швидкодія ЛЕ — основна його динамічна характеристика. Вона визначається затримкою розповсюдження перепаду напруги через елемент $t_{зр}$, яка вимірюється на рівні половини величини перепаду (рис. 2.7). Як правило $t_{зр}^{01} \neq t_{зр}^{10}$, і швидкодія характеризується середньою затримкою $t_{зр.ср} = (t_{зр}^{01} + t_{зр}^{10}) / 2$.

Для різних типів елементів ця величина коливається від частин до сотень наносекунд.

Споживана потужність у статичному режимі $P_{сп}$ залежить від стану елемента 0 або 1, тому що струм, який йде від джерела живлення, може бути різним. При великій кількості елементів у схемі приблизно половина з них опиниться в стані 0, інша половина — в стані 1. Тому доцільно використовувати величину середньої потужності

$$P_{сп.ср} = (P_{сп}^0 + P_{сп}^1) / 2.$$

У динамічному режимі споживана потужність зростає з підвищенням частоти переключення ЛЕ.

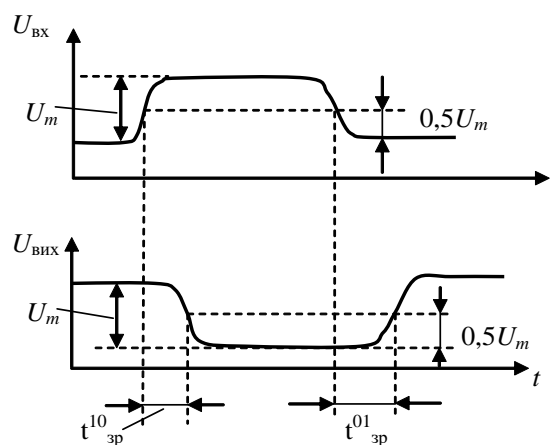


Рис. 2.7. Вимірювання часової затримки логічного елемента

Для узагальненої характеристики елемента часто застосовують величину $A = P_{\text{сп.ср}} \cdot t_{\text{зр.ср}}$, яку називають роботою перемикання. Між $P_{\text{сп.ср}}$ та $t_{\text{зр.ср}}$ існує тісний взаємозв'язок. Тому перевага віддається таким технологіям виробництва ЛЕ, при яких A зменшується, тобто підвищення швидкодії досягається без збільшення споживаної потужності. Сучасні ЛЕ можуть мати величину A меншу ніж 1 пДж (10^{-12} Дж).

2.3 Діодні логічні елементи (ДЛ)

У основу діодних ЛЕ покладено діодний ключ — найпростішу схему, яка складається з діода VD та резистора R (рис. 2.8). Активний компонент схеми — діод працює в ключовому режимі.

Якщо напруга на вході ключа нижче від порогу відкриття діода: $U_1 < U_{\text{дв}}$, то, як видно з характеристики діода $I_{\text{д}} = f(U_{\text{д}})$, діод замкнутий ($I_{\text{д}} = 0$), напруга на резисторі R : $U_R = I_{\text{д}} R = 0$. При $U_1 > U_{\text{дв}}$ діод відкритий, опір діода у відкритому стані $R_{\text{д}}$ малий — (50-100) Ом, і якщо вибрано $R \gg R_{\text{д}}$, то, зневажаючи падінням напруги на діоді $U_{\text{д}}$, напругу U_R можна вважати приблизно рівною U_1 .

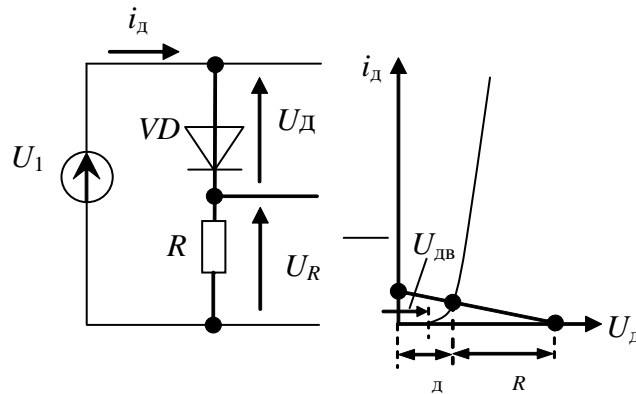


Рис. 2.8. Діодний ключ та його статична характеристика

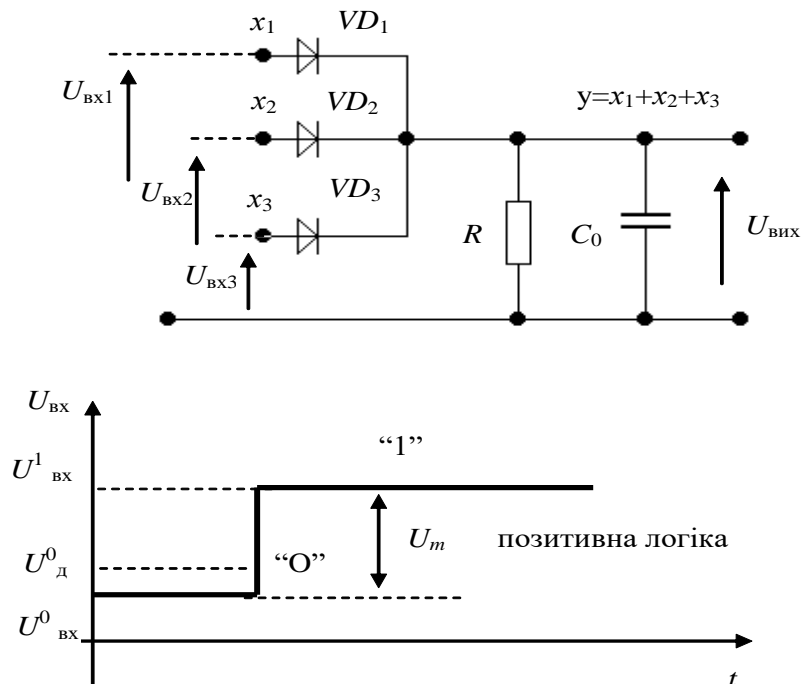


Рис. 2.9. Діодний елемент АБО та рівні його вхідних напруг

Логічний елемент АБО. Схема ЛЕ та рівні вхідної управляючої напруги, які відповідають позитивній логіці, показані на рис. 2.9. При подачі на всі входи 0 ($U_{вх}^0 < U_{дв}$) діоди закриті і вихідна напруга $U_{вих}=0$ ($y=0$). Достатньо лише на один з входів подати 1 ($U_{вх}^1 > U_{дв}$), щоб відповідний діод відкрився, і напруга на виході $U_{вих}=U^1_{вх}-U_{д}$ приблизно ($U_{д} \neq 0$) повторила б напругу на вході ($y=1$). При цьому діоди, на входи яких подається 0, залишаються закритими, тобто вихідна напруга з одного з входів не попадає на інші входи — тим самим ЛЕ забезпечує розв'язку по вхідним колам. Як впливає з розглянутого принципу роботи, елемент виконує операцію АБО: $y=x_1+x_2+x_3$.

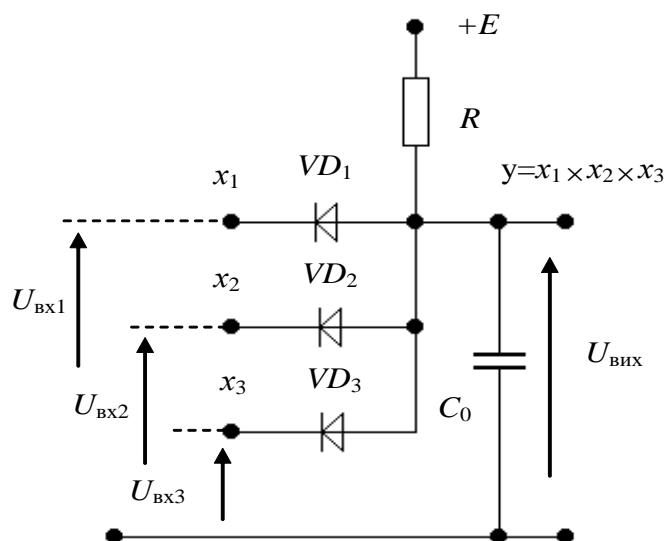


Рис. 2.10. Діодний елемент I

Логічний елемент I. Схема ЛЕ приведена на рис. 2.10. В ній на відміну від схеми АБО вихідна напруга знімається не з резистора R , а з діодів, які включені паралельно один до одного. Тому, якщо хоча б на один з входів буде поданий 0, відповідний діод під дією ЕРС джерела живлення E опиниться відкритим і напруга на виході $U_{вих}=U_{вх}^0+U_{д}$ буде мало відрізнятися від $U_{вх}^0$, тобто на виході — логічний 0 ($y=0$).

Тільки якщо на всі входи подана 1, діоди закриті, на виході високий рівень E ($y=1$). Відмітимо, що якщо $E > U_{вх}^1$, діоди не закриваються, але на вихід через них подається високий потенціал $U_{вх}^1$, що не змінює логіки роботи елемента. Він, таким чином, виконує логічну операцію I: $y=x_1 \cdot x_2 \cdot x_3$.

Розв'язка по вхідним колам тут забезпечується тим, що вхідний сигнал 1, який діє на одному з входів, закриває відповідний діод, і висока напруга з цього входу не попадає на інші входи.

Швидкодія діодних ЛЕ обмежується часом комутації, обумовленим інерційністю діодів, і, головне, процесами установаження, зв'язаними із зарядом та розрядом ємності C_0 , яка включає в себе ємність навантаження та паразитні ємності схеми. Швидкодія зв'язана з кількістю входів елемента та його навантажувальною здатністю ($K_{розг}$). Збільшення кількості входів (кількості діодів) призводить до зростання загальної вихідної ємності C_0 за рахунок паразитних ємностей закритих діодів і до зниження швидкодії. Крім того, при цьому зменшується еквівалентне навантаження паралельно включених закритих діодів, яке шунтує вихід і тим самим зменшує величину вихідного перепаду напруги.

Навантажувальна здатність діодних ЛЕ визначається шунтуючою дією наступних елементів на попередній, тим більше що вхідне навантаження елемента приблизно дорівнює опорі резистору R . При великій кількості навантажувальних елементів може не виконуватись умова $R \gg R_{д}$. Практично значення $K_{розг}$ не перевищує 2–3. З цих причин діодові ЛЕ у чистому вигляді, без підсилювальних транзисторних ключів, застосовуються дуже рідко.

2.4 Діодно-транзисторні логічні елементи (ДТЛ)

У діодно-транзисторних ЛЕ в ролі активних компонентів поряд з діодами використовуються біполярні транзистори в ключовому режимі, при якому транзистор може знаходитися в одному з двох станів — закритий або відкритий та насичений.

Для надійного запирання транзистора, як видно з його характеристик (рис. 2.11), напруга на базі повинна бути менше напруги запирання $E_{б0}$, при цьому струм колектора дорівнює тепловому струму $I_{к0}$. Залишкова напруга на колекторі в режимі насичення $U_{кн}$ близька до нуля.

Такий режим можна забезпечити, подаючи до кола бази напругу зміщення від окремого джерела. Щоб не ускладнювати схему, обмежуються режимом умовного запирання, забезпечуючи напругу на базі менше порогового значення $U_{\beta 0}$. При цьому струм колектора I_K не перевищує величини $(1 + \beta)I_{\kappa 0}$ і ним можна знехтувати.

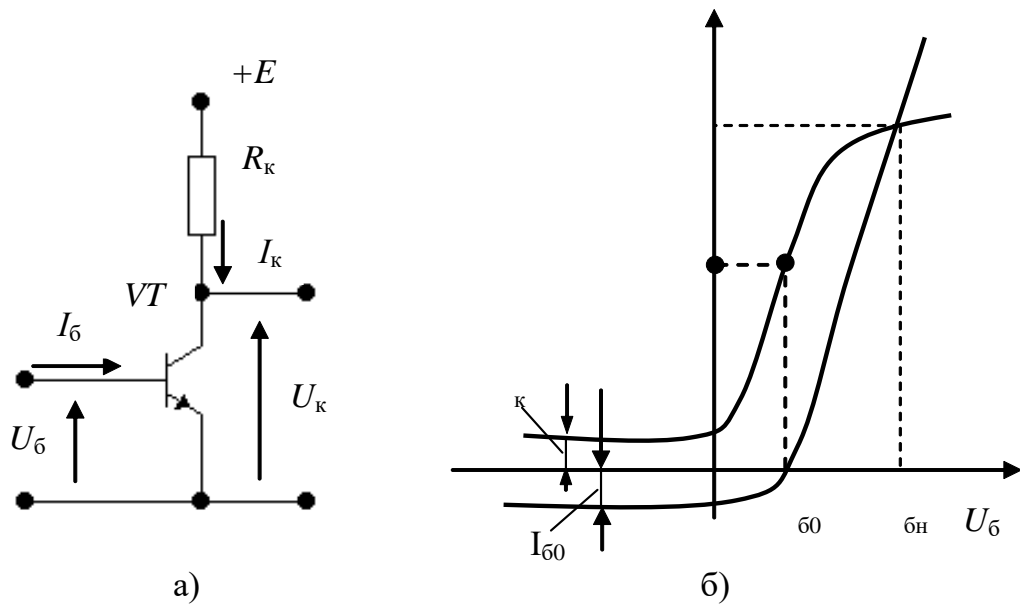


Рис. 2.11. Транзисторний ключ (а) та його статичні характеристики (б)

Для насичення транзистора необхідно створити струм бази, що перевищує струм бази насичення:

$$I_{\beta} > I_{\beta_{н}} = I_{\kappa_{н}} / B,$$

де $I_{\kappa_{н}}$ — струм колектора в режимі насичення;

B — коефіцієнт передачі струму бази в режимі великого сигналу.

Найпростіша схема ДТЛ-елемента на три входи приведена на рис. 2.12,а. Він виконує операцію І-НЕ. Дійсно, діоди $VD_1...VD_3$ разом з резистором R_0 та джерелом E_0 — це елемент І (рис. 2.9), VT — інвертор, який виконує операцію НЕ, а діоди $VD_{зм1}, VD_{зм2}$ разом з резистором R — коло зв'язку.

Розглянемо роботу елемента більш детально.

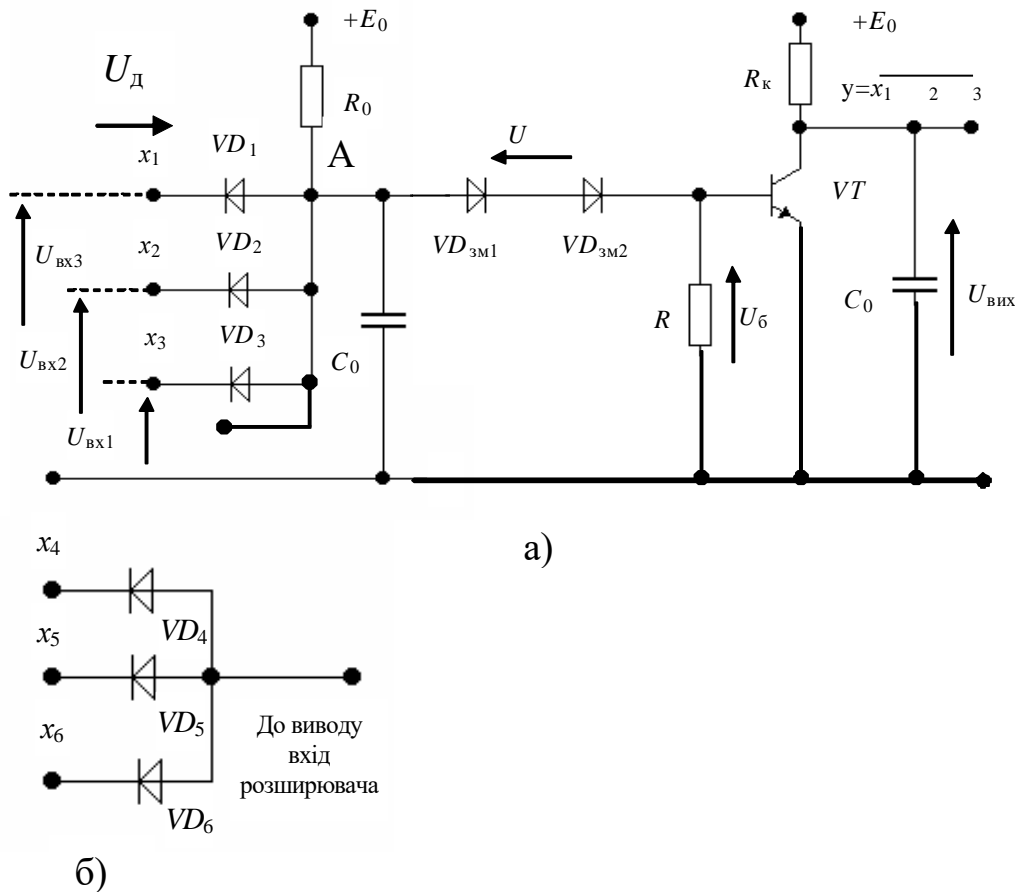


Рис. 2.12. Діодно-транзисторні ЛЕ а) – принципова схема; б) – розширювач по вході

Коли хоча б на один вхід подається рівень логічного нуля $U_{\text{вх}}^0$, напруга в точці А складає: $U_A = U_{\text{вх}}^0 + U_d$.

Якщо врахувати, що $U_{\text{вх}}^0$ – це напруга на колекторі насиченого транзистора $U_{\text{кн}}$ аналогічного елемента і вона мала, а напруга на відкритому діоді складає 0,5–0,7 В, то U_A опиняється біля 1 В. Діючи безпосередньо на базу, вона не забезпечила б запирання транзистора VT . Діоди зміщення $VD_{3\text{м}1}$ та $VD_{3\text{м}2}$ разом з резистором R складають дільник, у якого напруга на плечі R дорівнює $U_R = U_{\delta} = U_A - U_{3\text{м}} < U_{\delta 0}$. Транзистор VT при цьому стає умовно закритим, $I_{\text{к}} \approx 0$, і напруга $U_{\text{вих}} = U_{\text{к}} \approx E_{\text{к}}$. Це означає, що на виході ЛЕ буде 1, тобто рівень $U_{\text{вих}}^1$.

Коли на всі входи буде подана 1 ($x_1 = x_2 = x_3 = 1$), діоди закриються, джерело ЕРС E_0 створить у колі бази струм

$$I_{\delta} = \frac{E_0 - U_{3\text{м}} - U_{\delta\text{н}}}{R_0} - \frac{U_{\delta\text{н}}}{R},$$

і, якщо цей струм перевищить $I_{\delta\text{н}}$, то транзистор VT буде насичений, напруга $U_{\text{вих}} = U_{\text{к}} \approx 0$ – на виході ЛЕ буде 0, тобто $U_{\text{вих}}^0$.

Слід підкреслити, що струм колектора в режимі насичення не тільки визначається струмом $I_{\text{Рк}}$, але й залежить від вхідного струму навантаження: $I_{\text{кн}} = I_{\text{Рк}} + I_{\text{вх}} K_{\text{розг}}$. При постійному струмі бази I_{δ} збільшення навантаження може вивести транзистор з насичення, збільшиться $U_{\text{вих}}$ – порушиться умова нормального функціонування ЛЕ. Для підвищення навантажувальної здатності (збільшення $K_{\text{розг}}$) замість транзисторного ключа встановлюється складний інвертор. Аналогічний складний інвертор застосовується і в транзисторно-транзисторних елементах.

Отже, якщо хоча б на одному вході розглянутого елемента діє 0, на виході буде 1, і тільки коли на всіх входах – 1, на виході встановлюється 0. Інакше кажучи, цей елемент реалізує функцію І-НЕ, тобто $y = x_1 \cdot x_2 \cdot x_3$.

Для збільшення кількості входів елемента в мікросхемі роблять вивід "Вхід розширювача", до якого можна підключити діодну збірку (рис. 2.12,б), тобто здійснити "Розширення по I", тоді $y = x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_5 \cdot x_6$. Швидкодія елемента обмежується інерційністю транзистора (часом накопичення заряду в базі до граничного значення при включенні транзистора і часом виведення заряду з бази при виключенні), а також процесами встановлення, зв'язаними із зарядом та розрядом паразитних ємностей та ємності навантаження. Заряд ємності C_o при запертому VT ($y=1$) відбувається через резистор R_k , а розряд ($y=0$) – через малий опір відкритого транзистора.

2.5 Транзисторно-транзисторні логічні елементи (ТТЛ)

2.5.1 Схема та принцип дії ТТЛ- елемента

Транзисторно-транзисторні логічні елементи виготовляються тільки з так званим складним інвертором. Мета його застосування полягає в тому, щоб зробити вихідний опір елемента $R_{вих}$ малим не тільки в стані 0 (як у простому інверторі), але і в стані 1.

Один з варіантів схеми ТТЛ-елемента приведено на рис. 2.13. На вході схеми встановлений багатоємітєрний транзистор VT_6 , розроблений спеціально для такого застосування. Кількість емітерів цього транзистора визначає кількість входів логічного елемента. Транзистори VT_1 , VT_2 , VT_3 та діод зміщення $VD_{зм}$ утворюють складний інвертор.

Якщо емітерні переходи VT_6 розглядати як діоди, а колекторний перехід — як діод зміщення, то разом з транзистором VT_1 ця частина схеми стає аналогічною схемі ДТЛ-елемента, тобто забезпечує виконання логічної операції І-НЕ. Однак спільна база, що об'єднує емітерні та колекторні переходи VT_6 , а також транзистори VT_2 та VT_3 створюють особливості в роботі схеми.

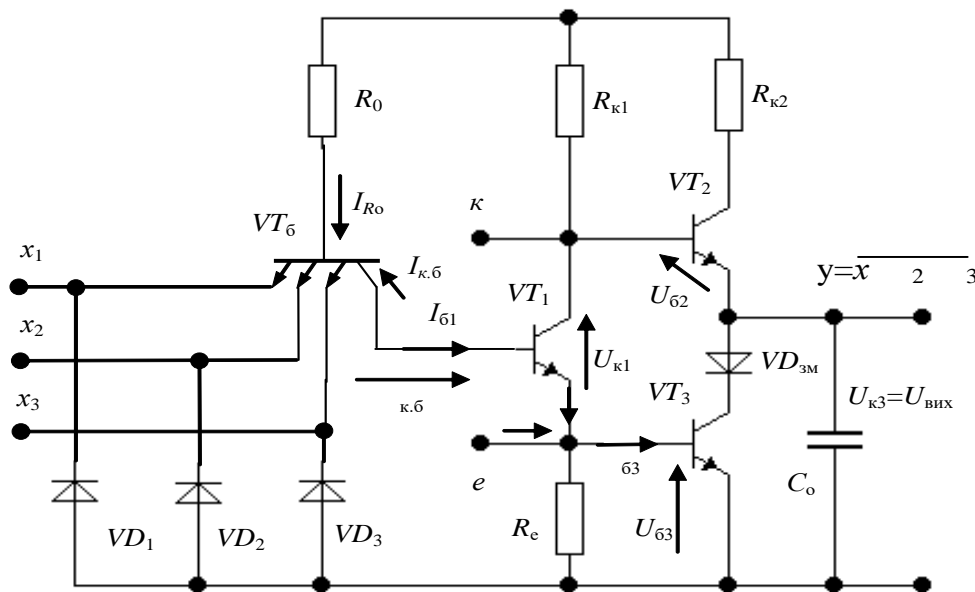


Рис. 2.13. Транзисторно-транзисторний ЛЕ з складним інвертором

Розглянемо роботу елемента з врахуванням цих особливостей, маючи на увазі дію сигналів у позитивній логіці. Коли на всі входи транзистора VT_6 подана 1 (високий рівень напруги $U_{вх}^1$), емітерні переходи закриті, а колекторний перехід відкритий, то багатоємітєрний транзистор працює в активному інверсному режимі: роль емітера виконує колектор, а емітери виступають в якості колекторів. Струм бази транзистора VT_1 визначається колекторним струмом інверсно включеного транзистора VT_6 :

$$I_{61} = -I_{к6} = I_{R0} + \sum_{i=1}^3 I_{вх} \cdot$$

Коефіцієнт передачі струму бази VT_6 в інверсному режимі $\beta_{\text{інв}}$ малий. Вхідні струми ("ті що входять в схему") $I_{\text{вх}} = \beta_{\text{інв}} \cdot I_{\text{Ро}}$ малі, ними можна знехтувати і, якщо $I_{\text{б1}} \approx I_{\text{Ро}} > I_{\text{бн1}}$, транзистор VT_1 буде насичений. Струм бази транзистора VT_3 $I_{\text{б3}} \approx I_{\text{е1}} \gg I_{\text{бн3}}$ і, значить, VT_3 теж насичений. Транзистор VT_2 при цьому буде закритим. Це забезпечується завдяки діоду зміщення $VD_{\text{зм}}$. Дійсно, як видно з схеми, в розглянутому випадку $U_{\text{б2}} = U_{\text{кн1}} + U_{\text{бн3}} - U_{\text{кн3}} - U_{\text{зм}}$. Вважаючи напруги на колекторах насичених транзисторів рівними: $U_{\text{кн1}} = U_{\text{кн3}}$, не важко побачити, що умова запирання VT_2 $U_{\text{б2}} \cong U_{\text{бн3}} - U_{\text{зм}} < U_{\text{б02}}$ може бути забезпечена тільки за рахунок напруги $U_{\text{зм}}$ на діоді $VD_{\text{зм}}$.

Напруга на виході ЛЕ при $x_1=x_2=x_3=1$ $U_{\text{вих}}=U_{\text{кн3}} -$ мала, відповідає $U_{\text{вих}}^0$, тобто $y=0$.

Якщо на один або декілька входів подано 0 — низький рівень напруги $U_{\text{вх}}^0$, то відповідні емітерні переходи VT_6 будуть відкриті. Багатоемітерний транзистор опиниться в режимі прямого включення — струм бази $I_{\text{бм}}$ та струм колектора $I_{\text{км}}$ протікають у бік відкритих переходів (вхідні струми змінюють напрямок — "виходять з схеми"). Струм колектора VT_6 обмежений малим тепловим струмом бази транзистора VT_1 . Тому при заданому струмі $I_{\text{бм}} = I_{\text{Ро}} \approx E_{\text{к}} / R_{\text{о}}$ багатоемітерний транзистор глибоко насичений, напруга на його колекторі відносно корпусу мала: $U_{\text{к}} = U_{\text{кбн}} + U_{\text{вх}}^0 < U_{\text{б01}} + U_{\text{б03}}$, транзистори VT_1 та VT_3 закриті. Високий потенціал колектора VT_1 відносно корпусу забезпечить відпирання VT_2 . У колі емітера VT_2 включений великий опір запертого транзистора VT_3 , тобто транзистор VT_2 працює в режимі емітерного повторювача (опір резистора $R_{\text{к2}}$ малий і лише обмежує кидки струму колектора при відпиранні транзистора). Коефіцієнт передачі емітерного повторювача близький до 1. Тому напруга на виході приблизно повторює високий рівень напруги на колекторі VT_1 : $U_{\text{вих}} = U_{\text{вих}}^1$, тобто $y=1$. Швидкодія елемента обмежується інерційністю транзисторів та процесами заряду та розряду ємності навантаження та паразитних ємностей. Прискоренню процесу виведення зарядів бази при запиранні транзисторів VT_1 та VT_3 сприяє струм колектора багатоемітерного транзистора. Заряд та розряд ємностей навантаження в ТТЛ — елементі здійснюються через малий вихідний опір складного інвертора як у стані 1, так і в стані 0. У першому випадку воно визначається малим вихідним опором емітерного повторювача, в другому — малим опором відкритого транзистора VT_3 .

У різних серіях ІМС швидкодія ТТЛ-елементів характеризується величиною $t_{\text{зр,оп}}$ від 10 до 50 нс. Складний інвертор забезпечує також високу навантажувальну здатність. Звичайно гарантується $K_{\text{розг}}=10$.

Розширення по АБО. Для розширення логічних можливостей елемента в деяких типах мікросхем (МС) з корпусу роблять додаткові виводи колектора (К) та емітера (Е) транзистора VT_1 (рис. 2.14).

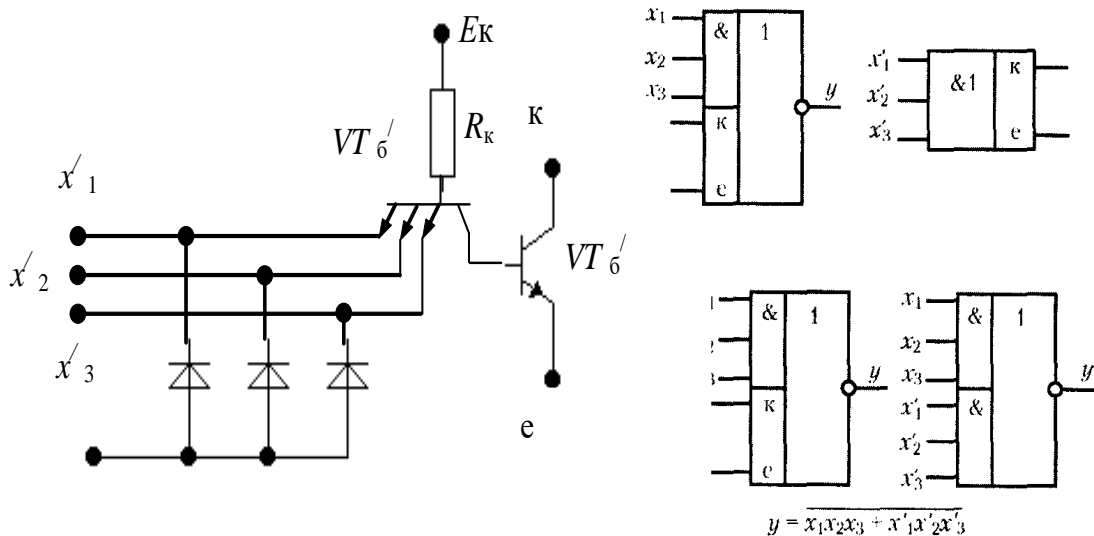


Рис. 2.9. Розширювач по АБО для ТТЛ-елемента

Рис. 2.14. Розширювач по АБО для ТТЛ – елемента:

а)– принципова схема; б) – позначення на функціональних схемах

Наявність цих виводів дозволяє підключити до МС "Розширювач по АБО", схема якого показана на рис. 2.13,а. Стан транзистора VT_1 розширювача визначається комбінацією вхідних змінних x'_1, x'_2, x'_3 . Наявність у складному інверторі двох паралельно включених транзисторів VT_1 та VT'_1 забезпечить низький рівень вихідної напруги $U_{\text{вих}}^0$, якщо буде відкритий хоча б один з транзисторів — VT_1 або VT'_1 , а для цього необхідно, щоб

$x_1=x_2=x_3=1$ або $x'_1=x'_2=x'_3=1$. Тільки у випадку, коли обидва ці транзистори заперті ($x_i=0$ та $x'_i=0$), на виході інвертора з'явиться високий рівень напруги $U_{\text{вих}}^1$. Таким чином, ЛЕ з розглянутим розширювачем буде виконувати логічну операцію І-АБО-НЕ.

2.5.2 ТТЛ-елементи з відкритим колектором

Іноколи виникає необхідність використовувати ЛЕ для управління навантаженням, яке має свої окреме джерело живлення та споживає великий струм. Таким навантаженням можуть бути обмотка реле, світловий індикатор і т.д. Для цієї мети служать елементи, у яких колектор вихідного транзистора залишився вільним (відкритим).

Спрощена схема ЛЕ з відкритим (вільним) колектором показана на рис. 2.15,а.

Для нормальної роботи такої схеми колектор вихідного транзистора VT_2 необхідно підключити, як видно з малюнка, до джерела живлення через зовнішнє навантаження. При цьому підключені зовнішні пристрої можуть працювати від інших джерел живлення $U_{\text{дж}}$ з підвищеною напругою.

ЛЕ з відкритим колектором допускають паралельне підключення декількох входів до спільного навантаження (рис. 2.15,б). При такому з'єднанні, якщо на виході одного з елементів буде низький потенціал $U_{\text{вих}}^0$, то на виході всієї системи буде 0. Щоб забезпечити високий рівень потенціалу $U_{\text{вих}}^1$ на спільному виході, необхідно закрити вихідні транзистори всіх ЛЕ, тобто встановити їх в стан 1.

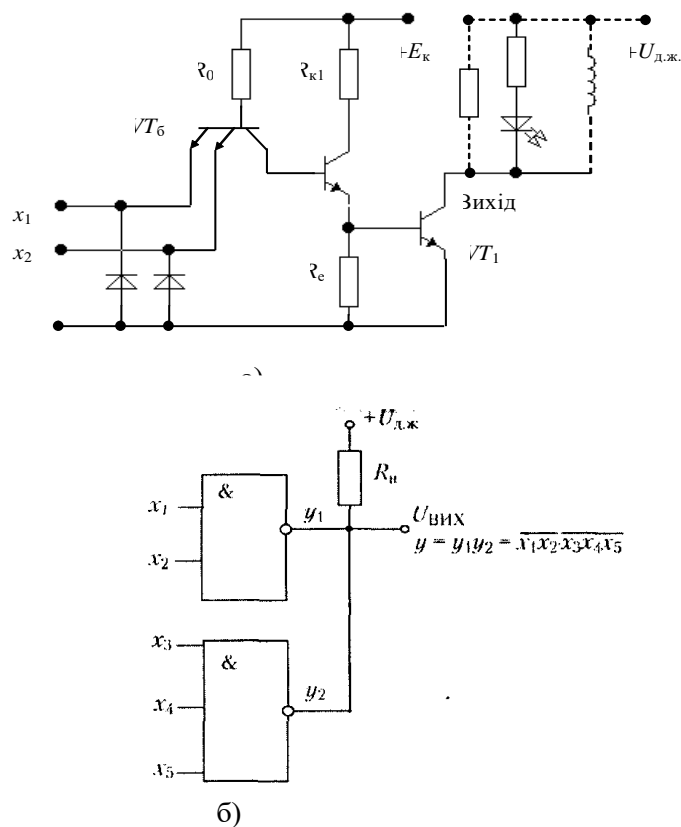


Рис. 2.15. ТТЛ-елемент з відкритим колектором:
а) – принципова схема; б) – з'єднання елементів у монтажне І

Таким чином, паралельним з'єднанням декількох ЛЕ з відкритим колектором до спільного навантаження створюють систему, що виконує операцію І: $F = y_1 \cdot y_2 = x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_5$. Таке з'єднання ЛЕ з відкритим колектором прийнято називати "монтажним І".

2.5.3 ТТЛ-елементи на транзисторах Шотткі

Особливість транзистора Шотткі полягає, як відомо, в тому, що при роботі його в ключовому режимі обмеження колекторного струму зверху відбувається без заходу до області насичення, тобто без відмикання

колекторного переходу. Тому виключається необхідність приймати будь-які додаткові заходи для прискорення виводу надлишкового заряду з бази при виключенні транзистора, тому що такий заряд не створюється. Застосування транзисторів Шоттки в ТТЛ-елементах дозволяє підвищити їх швидкодю без збільшення споживаної потужності або знизити споживану потужність збільшенням опору всіх резисторів без зниження швидкодії. Такі ЛЕ називають ТТЛШ-елементами.

Принципова схема одного з варіантів ТТЛШ-елементів зображена на рис. 2.16. За своєю будовою і загальним принципом роботи вона не відрізняється від схеми на рис. 2.13.

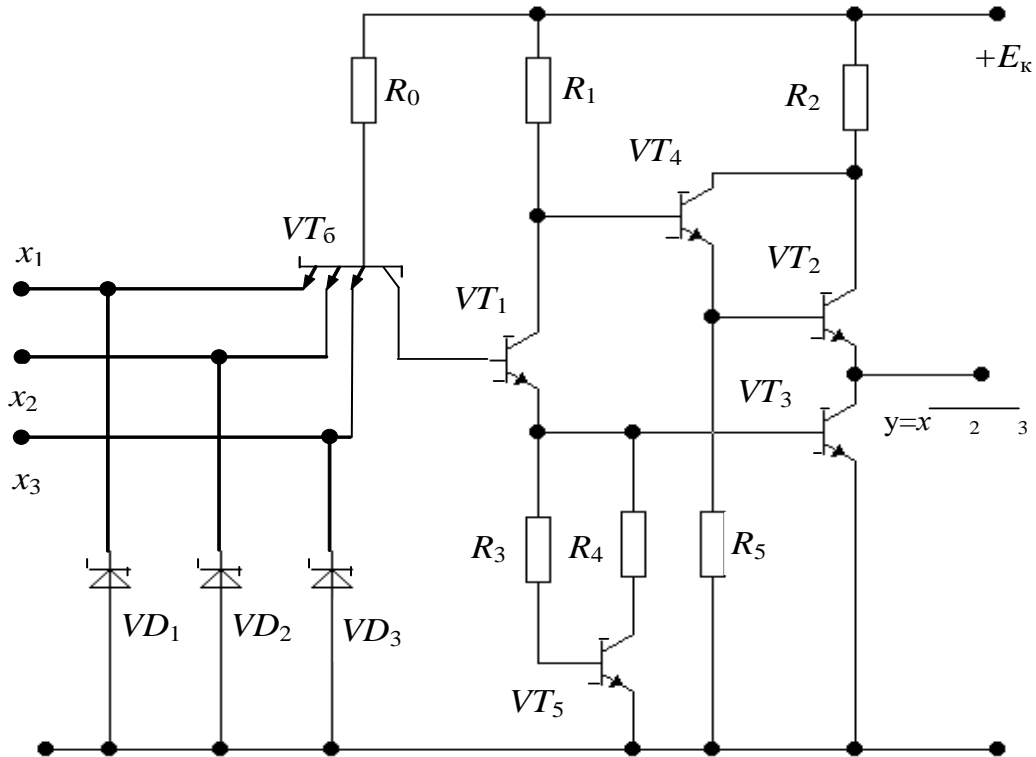


Рис. 2.16. ТТТ-елемент на транзисторах Шоттки

Особливість полягає в тому, що роль зміщуючого діода виконує тут емітерний перехід транзистора VT_4 . Крім того, пара $VT_2 - VT_4$ створює так званий складений транзистор (схема Дарлінгтона). Величина його коефіцієнта передачі струму бази близька добутку коефіцієнтів передачі $B_2 \cdot B_4$ транзисторів VT_2 та VT_4 . Це призводить до зменшення вихідного опору емітерного повторювача при закритому VT_3 і, як наслідок, до прискорення процесу заряду ємності навантаження при переході елемента з стану 0 до стану 1. В ТТЛШ-елементах вдається реалізувати величину затримки $t_{зр,ср}$ біля 5 нс.

2.5.4. ТТЛ елементи з трьома вихідними станами

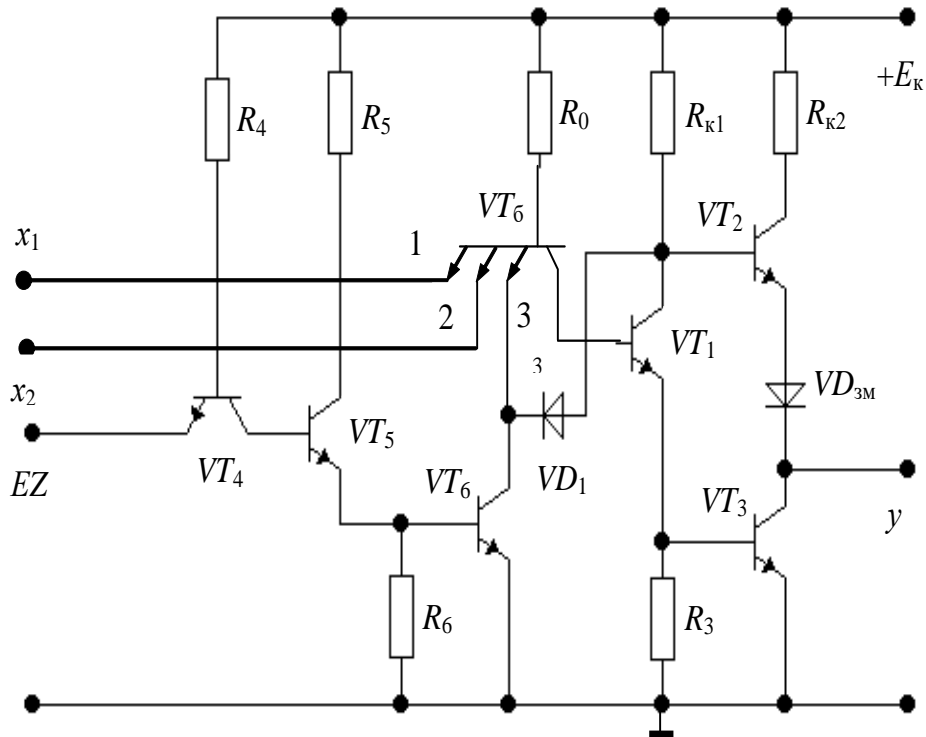
Розглянуті ЛЕ, як відомо, в залежності від комбінації вхідних сигналів можуть знаходитися в одному з двох станів — 0 або 1. В кожному з цих станів на виході елемента встановлюється відповідний рівень напруги — $U_{вих}^0$ або $U_{вих}^1$, і елемент має певним вихідний опір. У стані 0 вихідний опір малий відносно загальної шини (відкритий та насичений транзистор). У стані 1 — відносно шини живлення (вихідний опір емітерного повторювача). Такі властивості елементів виключають можливість об'єднання їх виходів для роботи на спільне навантаження. Існують мікросхеми різного функціонального призначення, вироблені на основі різних технологій, здатні при подачі відповідного керуючого сигналу приймати третій стан, при якому вихідний опір стає дуже значним як відносно загальної шини, так і відносно шини живлення. Такий стан рівнозначний відключенню елемента від навантаження і отримав назву високоім'єданного.

При черговий дії таких пристроїв їх виходи можна з'єднати між собою та підключити до спільного навантаження, зокрема з'єднувати їх з загальними магістральними шинами. Спрощена принципова схема елемента І-НЕ з трьома вихідними станами приведена на рис. 2.17,а.

Основна її відміна від схеми базового елемента (рис. 2.13) полягає в наявності діоду VD_1 , що забезпечує односторонній зв'язок одного з емітерів транзистора VT_6 (третього) з базою транзистора VT_2 .

Якщо $x_3=1$, тобто немає шляху для струму через цей емітер і через діод VD_1 , стан елемента визначається сигналами x_1 та x_2 , і він виконує властиву йому функцію $y = x_1 \cdot x_2$. Якщо ж $x_3=0$, тобто в цій точці низький потенціал, то за принципом роботи елемента І-НЕ транзистор VT_3 закривається, і на виході повинен бути стан 1. Але

при цьому сигнал 0 через діод VD_1 закриває транзистор VT_2 . Таким чином, вихід елемента стає відключеним як від загальної шини, так і від шини живлення. Встановлюється високоімпедансний стан Z. При $x_3=0$ сумарний струм третього емітера та діода VD_1 достатньо великий. Щоб не навантажувати ним джерело керуючого сигналу, цей сигнал EZ подається через особливий інвертор на транзисторах VT_4, VT_5, VT_6 . Якщо $EZ=0$, VT_4 глибоко насичений, VT_5 та VT_6 закриті, це означає $x_3=1$.



а)

x_1	x_2	EZ	y
0	0	0	1
0	1	0	1
1	0	0	1
1	1	0	0
x	x	1	Z

б)

Рис. 2.17. ТТЛ – елемент із трьома станами: а) принципова схема; б) таблиця станів

При $EZ=1$ VT_4 закривається, а VT_5 та VT_6 відкриваються і заходять у насичення. Струм емітера та діода VD_1 протікає через насичений транзистор VT_6 . Елемент переходить до третього стану. Таблиця станів елемента приведена на рис. 2.17,б.

2.5.5. Особливості практичних схем ТТЛ-елементів

ТТЛ-елементи мають достатньо високу швидкодію, особливо це стосується ТТЛШ-елементів. Напруги та струми роблять різкі стрибки, причому струми на входах під час стрибків навіть змінюють напрямок. Внаслідок цього на коливальних контурах, створених паразитними ємностями та індуктивностями, виникають затухаючі коливання, які часто називають дзвоном.

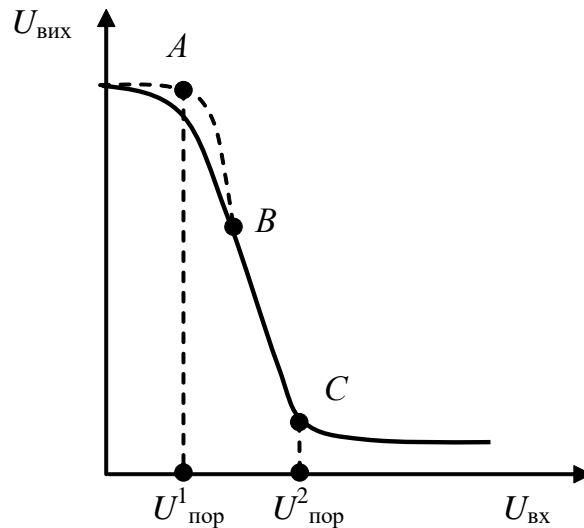


Рис. 2.18. Характеристика передачі елемента ТТЛ

Особливо небезпечні коливання напруг на входах. Наприклад, після позитивного перепаду, який замикає емітерний перехід транзистора VT_6 , негативний напівперіод коливання може призвести до відмикання переходу і помилкового спрацювання елемента. Для усунення цього ефекту входи з'єднуються з корпусом через антидзвоні діоди із зворотним включенням, які шунтують їх для негативної напруги і тим самим демпфують коливання (рис. 2.13-2.16).

У серіях ТТЛ-елементів, які випускаються в наш час, в колі емітера транзистора VT_1 замість резистора R_e (рис. 2.13) ставиться резисторно-транзисторне коло $VT_5-R_3-R_4$ (див. рис. 2.16). Це зроблено для покращання форми статичної характеристики передачі. Справа в тому, що при схемі на рис. 2.13 характеристика передачі відрізняється від зображеної на рис. 2.18.

Ще до досягнення вхідною напругою порогового рівня $U^1_{пор}$ має місце достатньо різке падіння вихідної напруги $U_{вих}$ на ділянці А В (рис. 2.18).

Воно пояснюється тим, що, коли відкривається транзистор VT_1 і через нього починають протікати колекторний та емітерний струми (при $U_{вх}=0,6-0,65$ В), підвищення $U_{вх}$ призводить до падіння напруги на колекторі VT_1 . Це зниження рівня напруги передається через емітерний повторювач на транзисторі VT_2 на вихід (ділянка А В). У той же час транзистор VT_3 ще закритий, тому що напруга на R_e за рахунок протікання через нього емітерного струму VT_1 недостатня для відкриття VT_3 .

При $U_{вх}$ біля 1,4В напруга на базі VT_3 досягає рівня відкриття, коефіцієнт передачі (підсилення) значно зростає, і на ділянці В С характеристика спадає крутіше. У схемі на рис. 2.16 емітерний та колекторний струми транзистора VT_1 не можуть з'явитися раніше, ніж відкриється транзистор VT_5 . А він відкривається приблизно при тій же вхідній напрузі, що і VT_3 . Тому похилої ділянки А В немає (пунктир на рис. 2.17). Така поліпшена характеристика забезпечує більш високу завадостійкість елемента. Дійсно, тепер при $U_{вх} < U^1_{пор}$ напруга позитивної завади на вихід не передається.

При практичному застосуванні ІМС деякі входи елемента можуть стати не потрібними для даної логічної схеми. За принципом роботи елемента І-НЕ на ці входи повинен бути поданий постійний сигнал 1. Це можна зробити різними способами: з'єднати надлишковий вхід з одним з використаних, подати на цей вхід напругу від джерела живлення (як правило +5В) через резистор 2-3 кОм або залишити його вільним, не підключеним ні до якого кола. Віддається перевага з'єднанням з джерелом живлення. При цьому не збільшується вхідний струм і знижується рівень завади, яка обумовлена наведенням від зовнішніх електромагнітних полів. ТТЛ-елементи перекривають широкий діапазон значень показників швидкодії та енергоспоживання. При цьому чим вище припущена робоча частота, тим більше і споживана від джерела потужність. Елементи серій К134 та К158 найекономніші, але вони призначені для застосування при тактовій частоті лише до 3 МГц. Швидкодіючі елементи серії 130 або К131 можуть працювати при частотах до 30 МГц, але споживають значно більшу потужність. ТТЛШ-елементи, які мають підвищену швидкодію, з успіхом витискують ТТЛ-елементи. При тому ж або навіть меншому енергоспоживанні вони можуть застосовуватися в цифрових вузлах з тактовою частотою до 50 МГц (серії 531, 1531). Всі ТТЛ-елементи мають порівняно високу завадостійкість; припустиму напругу статичної завади до 0,5В.

2.6 Емітерно-зв'язані логічні елементи (ЕЗЛ)

2.6.1. Спрощена схема та принцип дії ЕЗЛ-елементів

ЕЗЛ-елементи будуються на основі транзисторного перемикача струму. Тому такі елементи іноді називаються ПСТЛ-елементами (транзисторна логіка на перемикачі струму). Схема перемикача струму (рис. 2.19,а) включає в себе два транзистори VT_1 та VT_2 з резисторами $R_{к1}$ та $R_{к2}$, які виконують роль колекторних навантажень.

Емітери транзисторів з'єднані між собою, і до їх кола включений резистор R_e , що забезпечує глибокий негативний зворотній зв'язок по струму. Звичайно схема виконується симетричною ($R_{к1}=R_{к2}$) і транзистори мають, по можливості однакові параметри.

Якщо до баз транзисторів подати входні напруги $U_{вх1}$ та $U_{вх2}$, а навантаження підключити між колекторами, то це буде так званий диференційний підсилювач. Знак та величина напруги на навантаженні залежить від знаку та величини різниці $U_{вх1} - U_{вх2}$. Подібна схема застосовується, наприклад, на вході операційних підсилювачів.

Але в даному випадку ця схема працює в ключовому режимі, коли приблизно однаковий струм протікає або через транзистор VT_1 (VT_2 закритий), або через транзистор VT_2 (VT_1 закритий). Розглянемо роботу перемикача струму більш докладно.

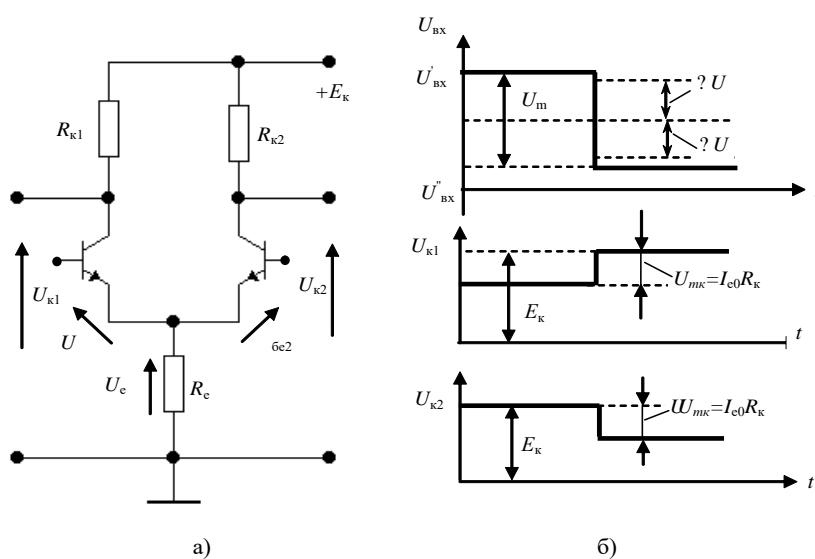


Рис. 2.19. Транзисторний перемикач струму:
а) – спрощена принципова схема; б) – часові діаграми

Транзистор VT_1 називається керуючим, до його бази подається входна керуюча напруга $U_{вх}$. Транзистор VT_2 – опорний, на його базі дії постійна опорна напруга $U_{оп}$. При $U_{вх}=U_{оп}$ емітерні струми транзисторів I_{e1} , I_{e2} будуть однаковими і дорівнювати половині сумарного струму I_{e0} , який протікає через резистор R_e :

$$I_{e1} = I_{e2} = I_{e0} / 2.$$

Будуть однаковими і напруги на колекторах $U_{к1}$ та $U_{к2}$. Але за законом Ома $I_{e1} = (U_{вх} - U_{е1}) / R_e$; $I_{e2} = (U_{вх} - U_{е2}) / R_e$ і, таким чином рівність $I_{e1} = I_{e2}$ можлива тільки, поки $U_{вх} = U_{оп}$. Якщо $U_{вх}$ буде зростати, напруга на емітері, повторюючи $U_{вх}$, збільшиться. При постійному потенціалі бази VT_2 , що дорівнює $U_{оп}$, це призведе до зменшення напруги $U_{е2}$ на емітерному переході і до зменшення струму колектора $I_{к2}$. Струм I_{e0} буде розподілятися між транзисторами вже не порівну – оскільки більша його частина потече тепер через VT_1 . Коли завдяки збільшенню $U_{вх}$ напруга U_e перевищить $U_{оп}$, то на переході $U_{е2}$ вона стане менше порогової $U_{е2}$, тоді VT_2 закриється, весь струм I_{e0} буде протікати через VT_1 , а напруга $U_{к2}$ буде дорівнювати E_k . Навпаки, зменшення $U_{вх}$ відносно $U_{оп}$ призведе до зменшення $U_{е1}$ (напруга U_e повторює найбільший потенціал на базах транзисторів) і, як наслідок, до зниження колекторного струму $I_{к1}$ першого транзистора. Більша частина струму I_{e0} буде протікати тепер через VT_2 . При $U_{е1} < U_{е2}$ транзистор VT_1 закриється, і весь струм I_{e0} буде протікати через транзистор VT_2 , $U_{к1} = E_k$.

Подальше зменшення $U_{вх}$ вже ніяк не впливає на струми транзисторів.

Для переключення струму з одного транзистора на інший потрібне дуже мале відхилення ΔU напруги $U_{вх}$ від значення $U_{оп}$ – десяти долі вольта. Перепад управляючої напруги $U_m \geq 2\Delta U$, який забезпечує переключення

струмів, не перевищує 1В. Зауважимо, що значне перевищення цієї напругою верхнього порогу недопустиме, бо воно може призвести до насичення транзистора VT_1 .

Величина перепаду напруги на колекторах

$$U_{mk} = \alpha I_{e_o} R_k \approx I_{e_o} R_k$$

Вона повинна перевищувати $U_m = 2\Delta U$, що забезпечить управління наступним елементом за допомогою вихідної напруги попереднього при їх послідовному з'єднанні. Ця умова виконується вибором опору резисторів R_{k1} , R_{k2} .

Часові діаграми, що ілюструють рівні вхідної та вихідної напруг перемикача струму, приведені на рис. 2.19,б.

Верхній рівень вхідної напруги приймається за 1, нижній — за 0.

Спрощену схему логічного елемента ЕЗЛ показано на рис. 2.20, а.

Схема містить перемикач струму, причому замість одного транзистора керування VT_1 паралельно підключено три транзистори — $VT_1^{(1)}$, $VT_1^{(2)}$, $VT_1^{(3)}$, на які подаються вхідні сигнали керування x_1 , x_2 , x_3 , а також емітерні повторювачі на транзисторах VT_3 і VT_4 .

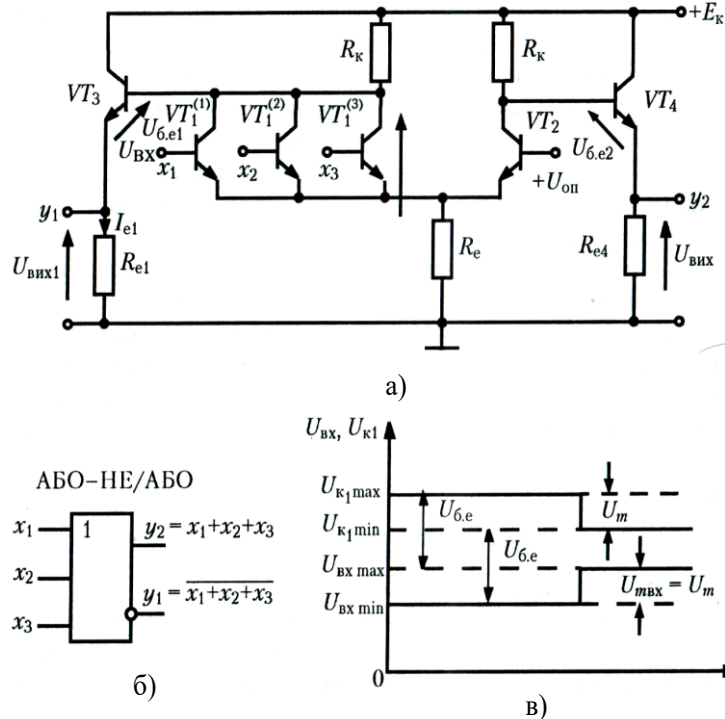


Рис. 2.20. ЕЗЛ – елемент: а) – спрощена принципова схема; б) – умовне позначення; в) – часові діаграми вхідної та колекторних напруг

Емітерний струм I_e протікає через ліве або праве плече перемикача і визначається в основному значенням опору R_e , обраного достатньо великим. Тому значення струму I_e під час проходження його через ліве плече мало залежить від того, чи відкритий один із транзисторів VT_1 , два з них або всі три. Тому достатньо подати сигнал лог. 1 на один із входів, наприклад $x_1 = 1$, як ліве плече відкривається, а транзистор VT_2 закривається. При цьому вихідні сигнали будуть $y_1 = 0$, $y_2 = 1$. І тільки при $x_1 = x_2 = x_3 = 0$ струм із лівого плеча перемикається в праве, і на виходах буде $y_1 = 1$, $y_2 = 0$.

Отже, цей тривходовий ЛЕ по одному виходу виконує операцію АБО-НЕ

$$y_1 = x_1 + x_2 + x_3,$$

а по іншому - операцію АБО

$$y_2 = x_1 + x_2 + x_3.$$

Він одержав назву елемента АБО-НЕ/АБО і на функціональних схемах його позначають, як показано на рис. 2.20, б. Емітерні повторювачі в ЛЕ виконують подвійні функції. По-перше, маючи малий вихідний опір, вони забезпечують швидке зарядження і розрядження вихідних ємностей, збільшуючи тим самим навантажувальну здатність без зниження швидкодії, по-друге, вони знижують рівні колекторних напруг на значення $U_{6.e}$ транзисторів VT_3 і VT_4 , що потрібно для узгодження їх із необхідними рівнями вхідної керувальної напруги наступних елементів. Якби вихідні напруги знімалися безпосередньо з колекторів, то, як показано на рис. 2.120, б, їхні рівні перевищували б потрібні вхідні рівні. При цьому транзистор VT_1 елемента керування не міг бути закритим і навіть заходив би в насичення, що недопустимо.

Невеликий вихідний опір емітерних повторювачів, робота транзисторів без переходу в режим насичення, мале значення необхідного логічного перепаду $U_m = I_e R_k$ дозволяє використовувати резистори R_k невеликих опорів, що забезпечує високу швидкодію ЕЗЛ-елементів. Найкращі елементи ЕЗЛ, наприклад, 1500-ї серії мають затримку розповсюдження перепаду $t_{c.z.p}$ менше, ніж 1 нс. Проте вони мають малий логічний перепад, що знижує завадостійкість елемента, а значний струм I_{e0} , який протікає в елементі незалежно від його стану, обумовлює високе енергоспоживання, що є істотним недоліком.

2.6.2. Особливості реальних схем ЕЗЛ-елементів

Принципова схема одного з варіантів реального базового ЕЗЛ-елемента приведена на рис. 2.21а. Заземлення позитивного полюса джерела колекторного живлення не вносить принципових змін у роботу елемента. Змінюється лише рівень відліку потенціалів (рис. 2.21б). Логічні рівні напруг $U_{к1}$ та $U_{к2}$, що відповідають закритим станам транзисторів VT_1 та VT_2 стають при такому включенні живлення близькими до 0, і нестабільність напруги живлення мало відбивається на величині цих рівнів.

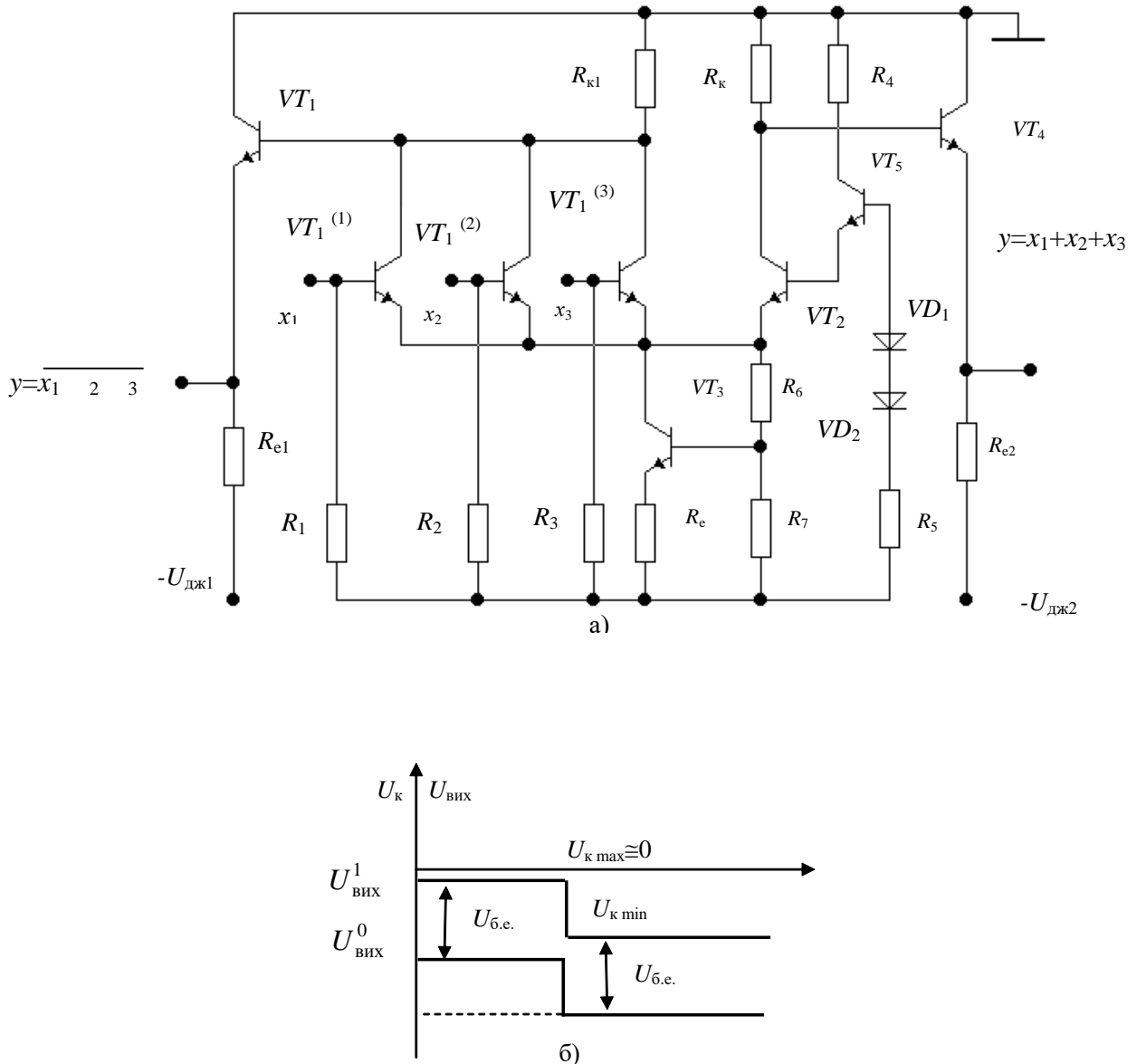


Рис. 2.21. Принципова схема реального базового ЕЗЛ-елемента(а) та рівні його вихідних напруг (б)

Опорна напруга $U_{оп}$ на базі транзистора VT_2 подається з емітерного повторювача (обведений пунктиром), вбудованого в мікросхему (МС), причому він може обслуговувати не один, а декілька елементів, які містяться в цій МС.

Діоди VD_1, VD_2 у базовому дільнику емітерного повторювача забезпечують термостабілізацію опорної напруги. Збільшення температури призводить до зменшення напруги на діодах, при цьому знижується потенціал бази VT_2 , що забезпечує стабілізацію його колекторного струму, який зростає з підвищенням температури.

У схемі, як емітерне навантаження перемикача струму застосований струмостабілізуючий двополюсник у вигляді транзистора VT_6 , охопленого глибоким негативним зворотнім зв'язком по струму за рахунок резистора R_c . До бази VT_6 подається постійна напруга з дільника R_6-R_7 . Невеликий опір двополюсника постійному струму забезпечує достатньо велике значення струму I_{e0} . В той же час двополюсник має дуже великий диференційний опір. Тому необхідна для управління транзисторами VT_1 та VT_2 зміна напруги U_e , обумовлена дуже малими приростами струму I_{e0} .

У вхідних колах управляючих транзисторів $VT_1(i)$ встановлені резистори R_1, R_2, R_3 , які дозволяють залишати вільними невикористовані входи. Опір цих резисторів порядку 50 кОм, а неприєднаний вхід еквівалентний подачі на нього логічного нуля. Резистори в вихідних емітерних повторювачах напруг VT_3, VT_4 всередині МС не встановлюються — схема з вільними емітерами. Можливість підключення резисторів R_{e1}, R_{e2} поза МС знижує потужність розсіювання усередині МС і зменшує її розігрів. До невикористаних виходів МС зовнішні резистори не підключаються, що дозволяє знизити енергоспоживання.

Для отримання максимальної швидкодії до вільних емітерів підключаються резистори R_{e1} та R_{e2} з опором 50 Ом та друге джерело живлення $U_{дж2}$ із зниженою напругою, що зменшує споживану потужність. При цьому вихідні сигнали можуть подаватись безпосередньо до кабелю з хвильовим опором 50 Ом.

При послідовному з'єднанні однотипних елементів на виході можна ставити резистори з великим опором, підключаючи їх до спільного джерела живлення $U_{дж1}$.

Вільні емітери розширюють логічні можливості елементів. Вони дозволяють об'єднувати однойменні (прямі або інверсні) виходи декількох елементів та підключати їх до спільного резистора R_c .

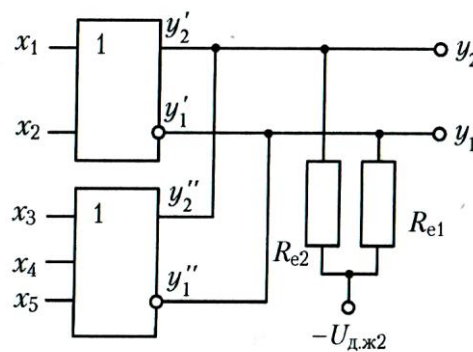


Рис. 2.22. Об'єднання елементів ЕЗЛ у монтажне АБО і монтажне І

На рис. 2.22 показане таке об'єднання для двох елементів. Якщо на виході обох елементів повинні бути однакові сигнали, то вони і будуть на спільному виході. Якщо ж на одному виході 1, на іншому 0, то вихідний транзистор цього другого елемента закривається, тому що різниця рівнів $U_{вих}^1$ та $U_{вих}^0$ перевищує напругу запирання $U_{бe0}$. Тому на об'єднаному виході буде 1. Це еквівалентно операції АБО. При об'єднанні прямих виходів буде отримано:

$$y_2 = y'_2 + y''_2 = x_1 + x_2 + x_3 + x_4 + x_5 \text{ ("монтажне АБО")}$$

При об'єднанні інверсних виходів отримується ("монтажне І")

$$y_1 = y'_1 + y''_1 = \overline{x_1 + x_2 + x_3 + x_4 + x_5} = \overline{(x_1 + x_2)(x_3 + x_4 + x_5)}$$

Слід мати на увазі, що таке з'єднання змінює рівні вихідних сигналів (зменшує перепад) і тому кількість елементів не повинна перевищувати чотирьох. Найзручніше це робити в межах однієї МС підвищеного рівня інтеграції.

2.6.3. Поєднання ЕЗЛ-елементів з ТТЛ-елементами

Один з недоліків ЕЗЛ-елементів полягає в тому, що вони мають дуже малий, менше 1 В, перепад між рівнями вихідної напруги, які відповідають станам 0 та 1. Це особливо незручно при роботі цифрових вузлів або пристроїв на кінцеві виконавчі елементи у вигляді світлових індикаторів, обмоток реле і т.д. Більш для цього підходять, як вже говорилося, ТТЛ-елементи, в тому числі з відкритим колектором. Тому МС, які містять логічні схеми на ЕЗЛ-елементах, мають на виходах перехідні узгоджувальні елементи для перетворення рівнів сигналу з метою підключення до них ТТЛ-елементів.

Як приклад на рис. 2.23 приведена спрощена схема двохвходового елемента АБО-НЕ, у якого вхідні рівні ЕЗЛ, а вихідні – ТТЛ. Він складається з двохвходового ЕЗЛ-елемента на транзисторах $VT^{(1)}_1, VT^{(2)}_1, VT_2$, з негативним джерелом живлення $-U_{дж}$ та ключа на транзисторі VT_3 з позитивним джерелом живлення $+E_k$. Відмінність вхідної частини від звичайної схеми в тому, що транзистор VT_2 має колекторне навантаження у вигляді діода VD.

Коли транзистор VT_2 відкритий ($x_1=x_2=0$), на його колекторі невелика негативна напруга, що падає на відкритому діоді, і вона закриває транзистор VT_3 . На виході – високий потенціал, тобто в системі ТТЛ $y=1$. При закритому транзисторі VT_2 (на одному або обох входах – сигнал 1) струм від джерела E_k може протікати тільки через резистор R_6 і до бази транзистора VT_3 , який стає насиченим (так вибрано співвідношення R_6 та R_k), і тому $y=0$. Таким чином перетворювач виконує логічну операцію АБО-НЕ:

$$y = \overline{x_1 + x_2}.$$

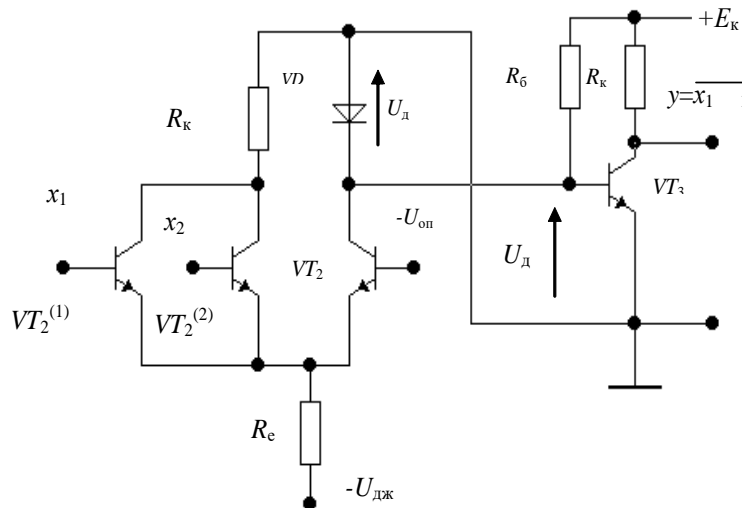


Рис. 2.23. Елемент узгодження ЕЗЛ-ТТЛ

При цьому, як було показано, на входи подаються логічні рівні напруг, властиві ЕЗЛ-елементам, а на виході отримуються рівні напруг, характерні для ТТЛ-елементів.

2.7 Логічні елементи на польових транзисторах

Для побудови ЛЕ застосовуються МДН-транзистори з індукованими каналами n -типу або p -типу, в яких канал наводиться (індукується) під дією вхідної управляючої напруги тільки після того, як ця напруга досягне деякого порогового значення. Ключові каскади на таких транзисторах можна з'єднувати безпосередньо, підключаючи вихід одного каскаду до входу іншого без будь-яких елементів зв'язку.

2.7.1. Логічні елементи на n -канальних МДН-транзисторах

В основу ЛЕ покладений транзисторний ключ з загальним витоком та нелінійним навантаженням (рис. 2.24,а). Транзистор VT_1 є комутуючим. Транзистор VT_2 — навантажувальний, його затвор з'єднаний з стоком, і він таким чином перетворюється на двополюсник з нелінійною вольт-амперною характеристикою. Коли напруга на вході ключа менше порогового значення U_{30} , як видно з вхідної характеристики (рис. 2.24,б), комутуючий транзистор запертий.

Струм стоків через транзистори визначається лише тепловими струмами, при чому транзистор VT_2 знаходиться на межі відкриття. Це пояснюється тим, що транзистор VT_2 не може бути ні відкритим, тому що шлях для струму істока перекриває закритий транзистор VT_1 , ні закритим, тому що тоді напруга на ньому перевищила б значення U_{30} , і він повинен був би відкритися. Тому падіння напруги на ньому $U_n = U_{30} \approx U_{30}$, тобто він на межі відкриття, і на виході ключа $U_{\text{вих}} = U_{\text{дж}} - U_{30}$. У практичних схемах $U_{\text{дж}} = 9$ В, а U_{30} має значення 1,5–2,0 В. Якщо на вхід подати напругу $U_{\text{вх.макс}}$, достатню для переведення VT_1 в круту (тріодну) область вихідних характеристик (рис. 2.24,в), то його опір буде менше опору навантажувального двополюсника і можна отримати $U_{\text{вих.мін}} < U_{30}$.

Відзначимо, що для забезпечення потрібного відношення опорів канал транзистора VT_2 робиться більш вузьким, ніж у транзистора VT_1 . Високий рівень напруги на виході закритого ключа приймається за рівень логічної 1

($U_{\text{вих.мах}} = U_{\text{вих}}^1$), низький рівень напруги на виході відкритого ключа – за рівень логічного 0 ($U_{\text{вих.тін}} = U_{\text{вих}}^0$).
 Умова $U_{\text{вих}}^0 < U_{30}$ забезпечує управління одного ключа другим ключем при їхньому безпосередньому послідовному з'єднанні.

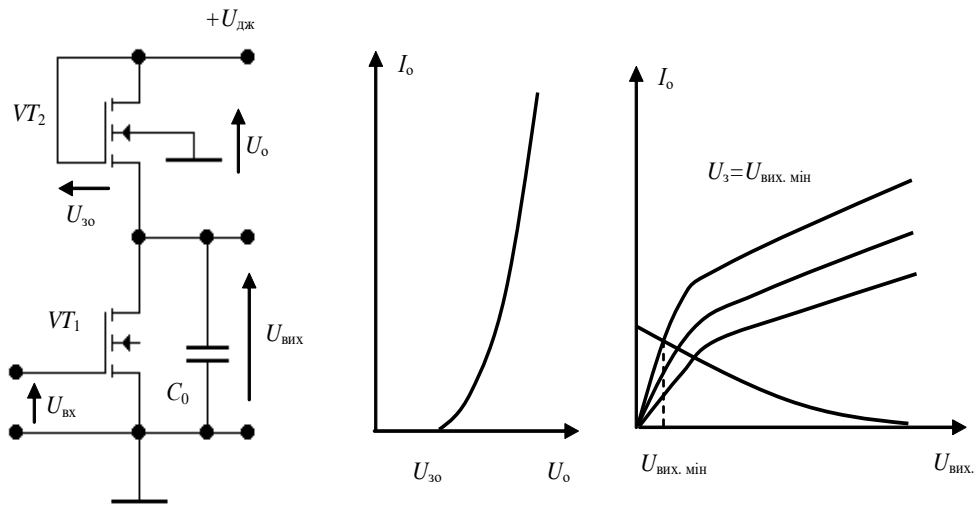


Рис. 2.24. Ключ на МДН-транзисторах з нелінійним навантаженням
 а) – принципова схема; б) – вхідна характеристика;
 в) – вихідні характеристики

Логічний елемент АБО-НЕ створюється паралельним включенням двох і більше комутуючих транзисторів які, працюють на загальне навантаження. На рис. 2.25,а – схема двохходового елемента. Будемо розглядати комутуючі транзистори як ключі: транзистор відкритий – ключ замкнений, транзистор закритий – ключ розімкнений. Якщо на обидва входи поданий низький рівень логічного 0 ($x_1=x_2=0$), транзистори $VT_1^{(1)}$ і $VT_1^{(2)}$ закриті, два ключі розімкнуті (рис. 2.24,а), і на виході високий рівень $U_{\text{вих}}^1$, тобто $y=1$.

Достатньо подати високий рівень 1 хоча б на один з входів, наприклад $x_2=1$, як транзистор $VT_1^{(2)}$ відкриється, тобто ключ замикається і залишкова напруга на виході стане низькою: $U_{\text{вих}}^0 < U_{30}$, тобто $y=0$.

Логічний елемент І-НЕ створюється двома або кількома комутуючими транзисторами, включеними послідовно з загальним навантаженням (рис. 2.25,б). У цій схемі обидва комутуючих ключа будуть замкнені, створюючи шлях для струму і забезпечуючи низький рівень вихідної напруги $U_{\text{вих}}^0$, тільки при $x_1=x_2=1$.

Якщо на один із входів подати 0, наприклад $x_2=0$, комутуючий транзистор $VT_1^{(2)}$ закриється, струм в колі припиниться, і на виході буде високий рівень напруги $U_{\text{вих}}^1$, інакше кажучи, вихідний сигнал y отримується із відношення $y=x_1 \cdot x_2$.

Коефіцієнт об'єднання по входу $K_{об}$ не буває великим, особливо в елементів І-НЕ: при кількості послідовно включених транзисторів більше двох залишкова напруга $U_{\text{вих.мін}}$ може okazaтися дуже великою, більше U_{30} .

Навантажувальна здатність МДН-елементів всіх видів висока, хоча вихідний струм у них невеликий. Це пояснюється дуже високим вхідним опором МДН-транзисторів (10^{12} Ом і більше). $K_{розг}$ досягає 20. Слід пам'ятати, що збільшення кількості навантажуючих елементів, як і збільшення паралельно включених вхідних транзисторів, веде до збільшення паразитних ємностей і як наслідок, до зменшення швидкодії. При оцінці швидкодії можна знехтувати інерційністю МДН-транзисторів, тому що в них не відбувається накопичування і розсмоктування зарядів. Швидкодія визначається процесом заряду і розряду паразитних ємностей через значні за величиною опори транзисторів. Особливо це відноситься до навантажувального транзистора, у якого, як вже відмічалось, канал робиться більш вузьким. Тому заряд вихідної ємності C_0 через VT_2 проходить повільніше, ніж розряд через VT_1 .

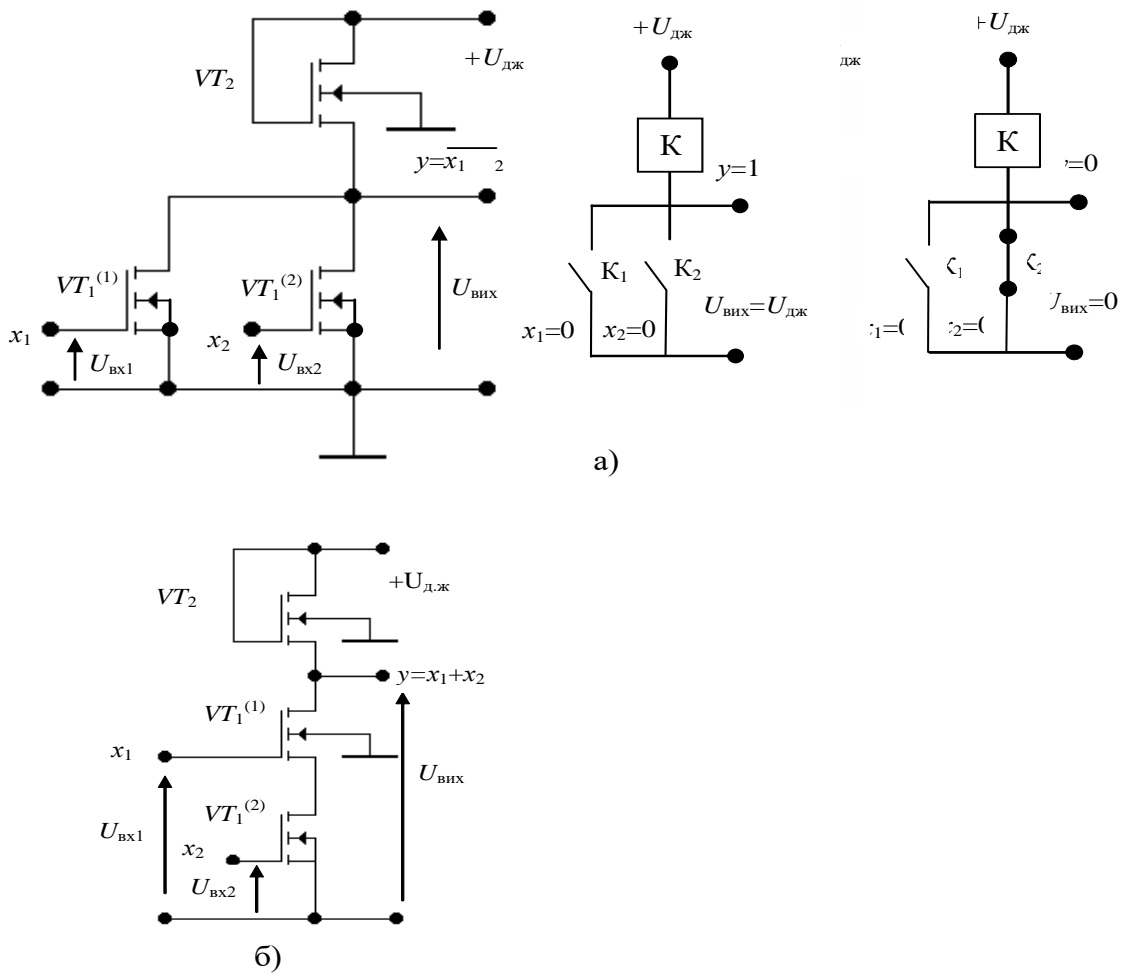


Рис. 2.25. МДН-елемент з нелінійним навантаженням:
а) тип АБО-НЕ; б) тип І-НЕ

2.7.2. Логічні елементи на комплементарних парах МДН-транзисторів

Оснoву КМДН-елементів складає транзисторний ключ, утворений двома МДН-транзисторами з каналами різного типу провідності (рис. 2.26,а). Транзистори з'єднані між собою стоками. Витік транзистора VT_1 з каналом n -типу з'єднаний із загальною шиною, на витік VT_2 з каналом p -типу підключений "+" джерела живлення. При такій схемі напруга затвор — витік $U_{зв1}$ транзистора VT_1 дорівнює входній напрузі $U_{вх}$, а напруга затвор-витік $U_{зв2}$ транзистора VT_2 відрізняється на величину напруги джерела живлення $U_{дж}$, тобто

$$U_{зв1} = U_{вх}, \quad U_{зв2} = U_{вх} - U_{дж}.$$

Якщо входна напруга $U_{вх}$ поступає з виходу аналогічного елемента, то вона змінюється між рівнем $U_{вх.max} \approx U_{дж}$, який приймають за 1, та рівнем $U_{вх.min} \approx 0$, який приймається за 0.

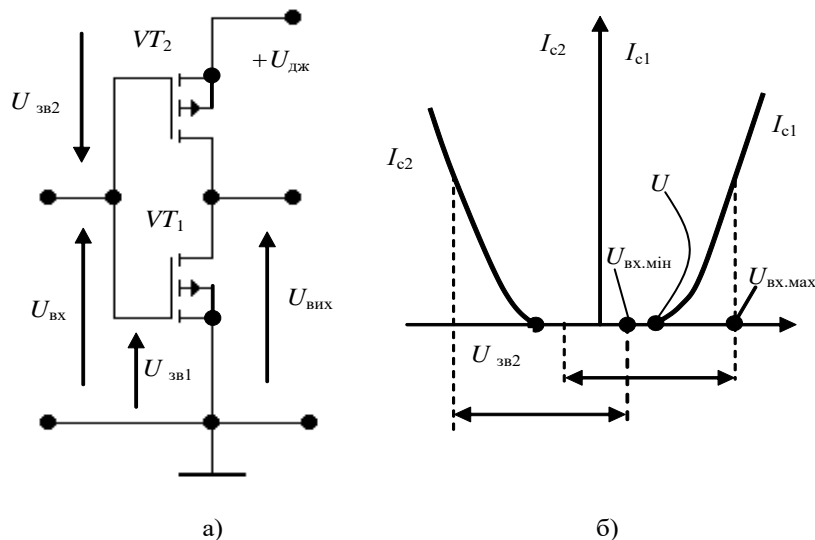


Рис. 2.26. Ключ у вигляді комплементарної пари МДН-транзисторів:
а – принципова схема; б – характеристики транзисторів

Коли на вхід подається $U_{вх.мак}$, VT_1 відкритий, а VT_2 закритий, тому що його напруга $U_{зв2}$ хоч і негативна, але не досягає порогу відкриття $U_{з02}$. При $U_{вх} = U_{вх.мін} < U_{з01}$ VT_1 закривається, а негативна напруга на затворі VT_2 , велика, і він відкривається. Це ілюструється рис. 2.26,б, на якому зображені стокозатворні характеристики обох транзисторів і показані рівні вхідної напруги. Таким чином, у будь-якому з станів один з транзисторів закритий, наскрізний струм у коло живлення відсутній. Напруга на виході ключа $U_{вих}$ змінюється від $U_{вих.мін} \approx 0$, коли транзистор VT_1 відкритий, до $U_{вих.мак} \approx U_{дж}$ при закритому VT_1 . Перепад вихідної напруги близький за величиною до $U_{дж}$. Щоб в процесі комутації уникнути стану, коли обидва транзистори були б закриті, величина напруг живлення повинна задовольняти умові

$$U_{дж} > U_{з01} + |U_{з02}|.$$

Логічний елемент АБО-НЕ реалізується за допомогою двох або більше КМДН-ключів шляхом паралельного включення n -канальних транзисторів та послідовного (ярусного) включення транзисторів з каналами p -типу. На рис. 2.26 зображена схема двовходового елемента АБО-НЕ. Якщо на входи подані низькі рівні, тобто $x_1 = x_2 = 0$, то транзистори $VT_1^{(1)}$ та $VT_1^{(2)}$ закриті (ключі розімкнуті), а транзистори $VT_2^{(1)}$ та $VT_2^{(2)}$ відкриті (ключі замкнені).

Струм у колі живлення визначається лише струмами утічки. Напруга на виході $U_{вих} \approx U_{дж}$, тобто $y=1$. При подачі хоча б на один з входів, наприклад $x_1=1$, транзистор $VT_1^{(1)}$ відкривається, а $VT_2^{(1)}$ переходить в закритий стан (див. рис. 2.27).

Струм у коло живлення як і раніше дуже малий, а вихідна напруга $U_{вих} \approx 0$, тобто $y=0$.

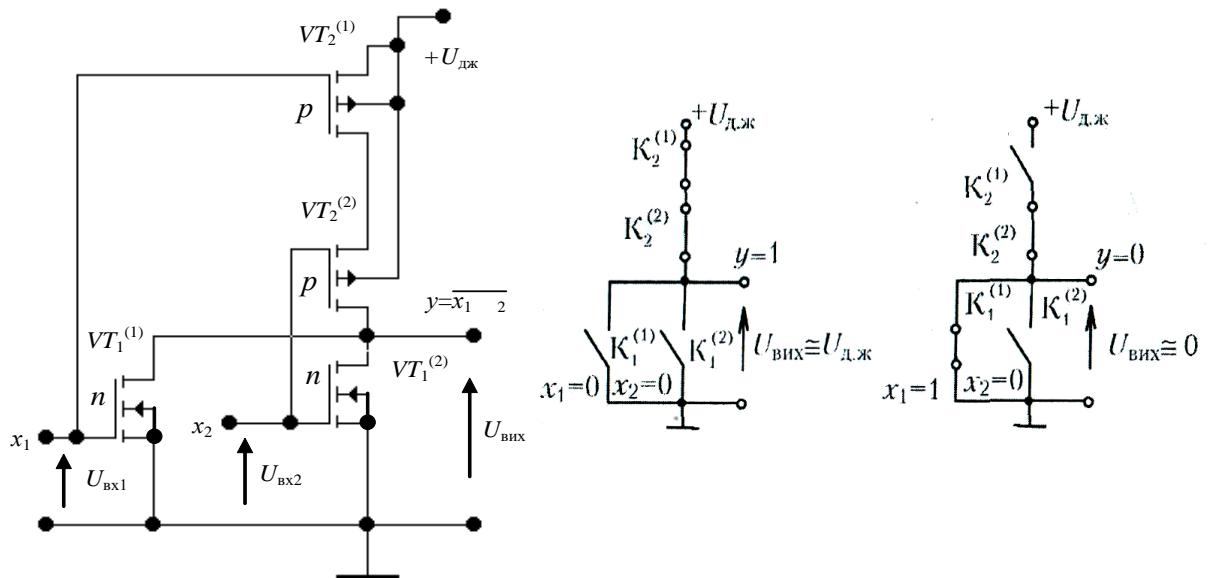


Рис. 2.27. КМДН-елемент типу АБО-НЕ: принципова та ключові еквівалентні схем схеми

Логічний елемент І-НЕ будується аналогічно тільки паралельно включаються p -каналні транзистори, а n -каналні-послідовно (рис. 2.28).

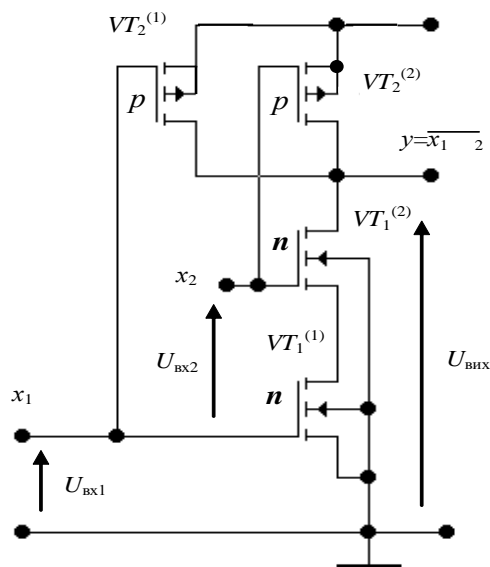


Рис. 2.28. КМДН-елемент типу І-НЕ

Якщо $x_1=x_2=1$, транзистори $VT_1^{(1)}$, $VT_1^{(2)}$ відкриті, а $VT_2^{(1)}$, $VT_2^{(2)}$ закриті. Напряга на виході близька до 0, тобто $y=0$. Якщо хоча б на один вхід подано 0, наприклад $x_1=0$, транзистор $VT_1^{(1)}$ закривається, а $VT_2^{(1)}$ переходить до відкритого стану. При цьому $U_{\text{вих}}=U_{\text{дж}}$, тобто $y=1$.

Основна перевага КМДН-елементів у тому, що в обох статичних станах струм від джерела живлення практично не надходить, тому споживана потужність дуже мала. Однак при роботі елемента струм йде на заряд паразитних ємностей, тому динамічна споживана потужність пропорційна частоті переключення і може на декілька порядків перевищувати статичну. Швидкодія ЛЕ на КМДН-структурах вище, ніж у ЛЕ на МДН-транзисторах з каналами одного типу, тому що тут і заряд, і розряд вихідної ємності відбувається через відкриті транзистори VT_1 та

VT_2 , які мають однаково малий опір. Але як ті, так і інші відносяться до класу елементів низької швидкодії. Припустима частота переключення не перевищує 5 МГц.

Кількість входів ($K_{об}$) та навантажлива здатність ($K_{роз}$) тут також обмежуються в основному впливом цих показників на швидкодію. Транзистори, які включаються паралельно, збільшують паразитну ємність, а ті, які включаються послідовно, збільшують опір, через який відбувається заряд або розряд ємності. Деякі серії КМДН-елементів, наприклад 564-та серія, допускають застосування джерел живлення з напругою від 3 до 15 В. Від цього залежить швидкодія і споживана потужність. При $U_{дж}=5$ В такі елементи за сигналами сумісні з ТТЛ-елементами.

Істотною перевагою КМДН-елементів є їх висока завадостійкість. Для тієї ж 564-ї серії допустима статична завада 2,5 В.

2.7.3 КМДН-логічний елемент з трьома вихідними станами

Принципова схема елемента зображена на рис. 2.29,а. Транзистори VT_1 та VT_2 утворюють інвертуючий КМДН-ключ. Його відміна від зображеного на рис. 2.26 полягає в тому, що за допомогою додаткових транзисторів VT_3 з n -каналом та VT_4 з p -каналом він може відключатись від джерела живлення. На ці транзистори подаються взаємно інверсні управляючі сигнали EZ та \overline{EZ} .

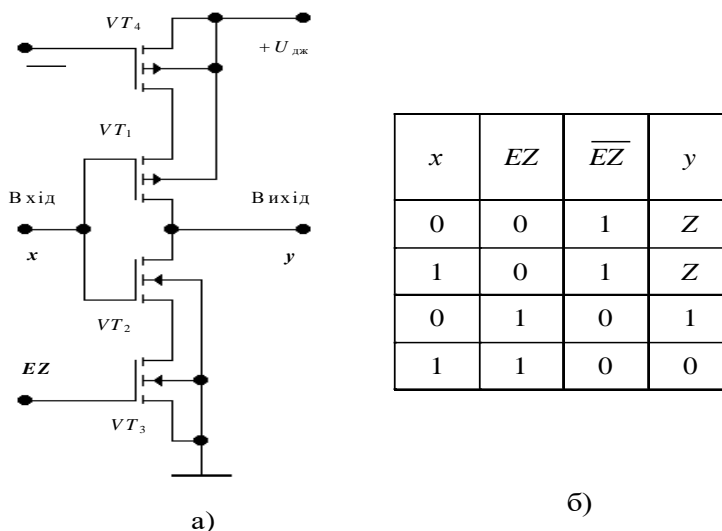


Рис. 2.29. КМДН-елемент з трьома вихідними станами:
а – принципова схема; б – таблиця станів

Якщо на вхід EZ поданий низький рівень 0, а на вхід \overline{EZ} – високий рівень 1, транзистори VT_3 та VT_4 закриті, живлення до ключа не подається, і вихідний вивід має дуже великий опір як відносно шини живлення, так і загальної шини "Земля".

Вхідний сигнал через елемент не проходить. При подачі на вхід EZ рівня логічної 1, а на вхід \overline{EZ} – 0, транзистори VT_3 та VT_4 відкриваються, ключ отримує живлення $U_{дж}$, і напруга на його виході залежить від сигналу на вході. При цьому вихідний опір елемента в будь-якому з станів буде визначатися опором двох послідовно включених відкритих транзисторів. Таблиця станів елемента в залежності від комбінації управляючих сигналів приведена на рис. 2.29,б.

2.7.4. Особливості схем ЛЕ на польових транзисторах та узгодження їх з ТТЛ-елементами

Логічні елементи, які побудовані на МДН-транзисторах, порівняно з ЛЕ інших типів мають деякі особливості, обумовлені властивостями цих транзисторів. Затвор МДН-транзистора та підложка, які розділені шаром діелектрика, утворюють конденсатор. Величезний, порядку 10^{12} Ом, опір утечки цього конденсатора створює сприятливі умови для накопичення на ньому статичних зарядів з потенціалом, здатним викликати необернений пробій діелектричного шару. При цьому затвор транзистора не можна залишати вільним, таким що не має гальванічного зв'язку з загальною шиною або з шиною живлення. Під найбільшу небезпечність пробою діелектрика підпадають транзистори, які встановлені на входах логічної схеми, тому що їх затвори з'єднані з виводами із МС.

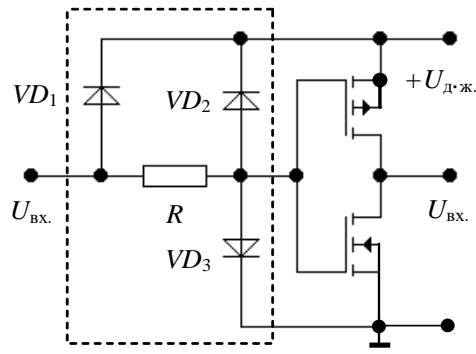


Рис. 2.30. Охоронне коло вхідних МДН-транзисторів

Для захисту таких транзисторів від пошкодження високою напругою в єдиному технологічному процесі виготовлення ЛЕ кожен його вхід забезпечують діодно-резистивним охоронним колом – рис. 2.30 (на принципових схемах ЛЕ кола як правило не зображують). У залежності від значення та полярності напруги перевантаження на вході діоди або проводять у прямому напрямку, або опиняються в режимі лавинного пробію, який настає при зворотній напрузі порядку 30–35 В. Лавинний пробій діодів має обернений характер і на працездатність ЛЕ не впливає. Діоди замикають підвищену вхідну напругу або на джерело живлення $U_{дж}$, або на загальну шину. В нормальних умовах роботи, коли величина вхідної напруги в нормальних межах – $0,7V < U_{вх} < (U_{дж} + 0,7V)$, відкриття діодів не відбувається.

У процесі зберігання виводи МС повинні бути з'єднаними між собою – обгорнутими металеву фольгою.

Підключення КМДН-елементів до виходу ТТЛ (ТТЛШ) – елемента при однаковій напрузі живлення $U_{дж} = 5V$ здійснюється безпосередньо. Важливо лише, щоб вихід ТТЛ-елемента не навантажувався одночасно входами інших ТТЛ-елементів, тому що їх вхідний струм може знизити напругу на виході попереднього елемента до рівня, якого недостатньо для відкриття польових транзисторів. Для узгодження виходу ТТЛ-елемента з входами КМДН-елементів при живленні останніх підвищеною напругою застосовують ТТЛ-елемент з вільним колектором, подаючи на його вихідний транзистор через резистор напругу джерела живлення КМДН-елементів (рис. 2.31).

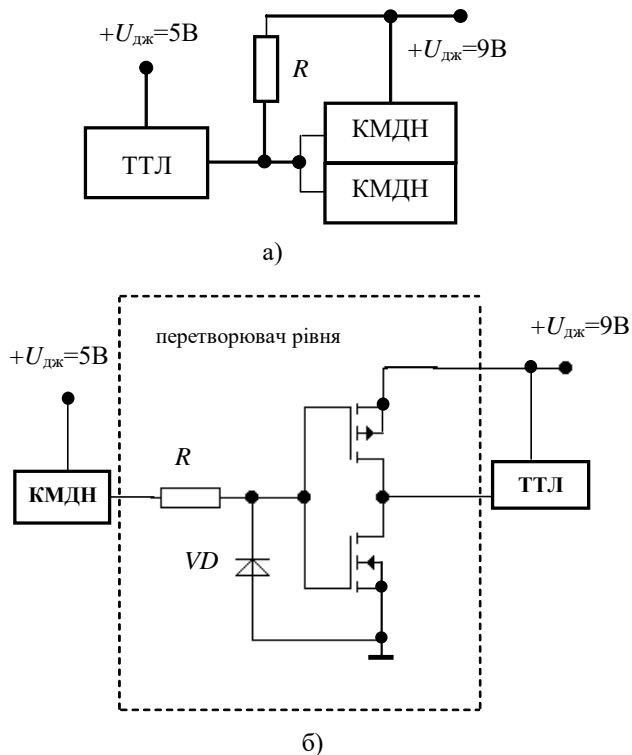


Рис. 2.31. Підключення ТТЛ-елементу до виходу КМДН-елементу при різних напругах живлення

Якщо сигналами з виходу КМДН-елемента необхідно управляти ТТЛ-елементами, то при живленні їх від одного джерела $U_{дж}=5В$ достатньо забезпечити узгодження по струму між елементами. Вхідний струм ТТЛ-елемента при 0 на вході $I_{вх}^0$, як правило, перевищує навантажувальну здатність КМДН-елемента. Як елемент узгодження з підвищеною навантажливою здатністю в стані 0 можна використовувати елемент АБО-НЕ, в якому об'єднані всі входи. Як видно з рис. 2.27, навантажувальна здатність такого елемента визначається кількістю паралельно включених n -каналних транзисторів, тобто в розглянутому випадку подвоюється.

Якщо $U_{дж.КМДН} > U_{дж.ТТЛ}$, то необхідне узгодження як по напрузі, так і по струму.

Узгодження по напрузі забезпечується включенням перетворювача рівня, живлення якого відбувається від спільного з ТТЛ-елементом джерела $U_{дж}=5В$ (рис. 2.31). Відміна перетворювача від інвертора полягає в тому, що в охоронному коло відсутні діоди між затвором та шиною живлення. Це дозволяє, не порушуючи роботи перетворювача, мати на його вході напругу, яка перевищує напругу живлення. Гранична навантажувальна здатність такого перетворювача не більше одного входу ТТЛ-елемента.

2.7.5. Поняття про логічні елементи з інжекційним живленням

ЛЕ з інжекційним живленням (інтегральна інжекційна логіка ІІЛ, ІІЛ) будуються на основі виконаних методами інтегральної технології так званих поздовжньо-поперечних транзисторних структур. Не вдаючись у подробиці електронних процесів в такій структурі, зобразимо її елементарну чарунку у вигляді двох транзисторів VT_1 та VT_2 (рис. 2.32,а). До емітера VT_1 підключене джерело живлення $U_{дж}$ з невеликою напругою, достатньою лише для відкриття емітерного переходу, і через резистор R постійно протікає струм інжекції I_i . Транзистор VT_1 можна розглядати як генератор струму (інжектор), що управляє роботою транзистора VT_2 . Схема заміщення чарунки зображена на рис. 2.32,б. Вхідне коло може мати два стани: розрив або замикання на корпус (зображене пунктиром). Перший стан приймається за 1, другий – за 0.

Якщо на вході коло розірване (1), то струм інжектора I_i , протікаючи до бази транзистора VT , насичує його. Між точками $K-E$ буде дуже малий опір, тобто замикання (0). Якщо на вході коло замкнуто (0), струм інжектора піде через нього, і транзистор VT буде закритий, вихідне коло $K-E$ розімкнене (1). Таким чином, ця чарунка є інвертор. При послідовному з'єднанні чарунок стан входу – це стан попереднього транзистора: закритий (розрив) або насичений (замикання). Об'єднуючи такі інвертори в різні схеми з безпосередніми зв'язками, можна отримати логічні елементи з заданими функціональними можливостями.

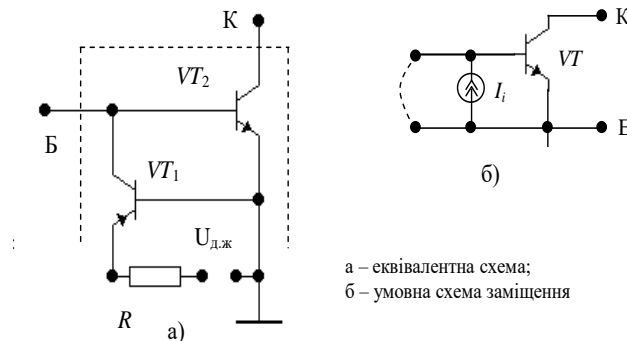


Рис. 2.32. Елементарна чарунка з інжекційним живленням

Якщо інвертори включити паралельно (рис. 2.33,а), то достатньо хоча б на один з входів подати 1 (розірвати вхідне коло) щоб відповідний транзистор перейшов до насичення, тобто вихід став замкненим (0). Таким чином, цей елемент виконує операцію АБО-НЕ: $y = x_1 + x_2$. Підключенням до виходу послідовно ще одного інвертора (рис. 2.33,б) утворюється елемент АБО: $y = x_1 + x_2$.

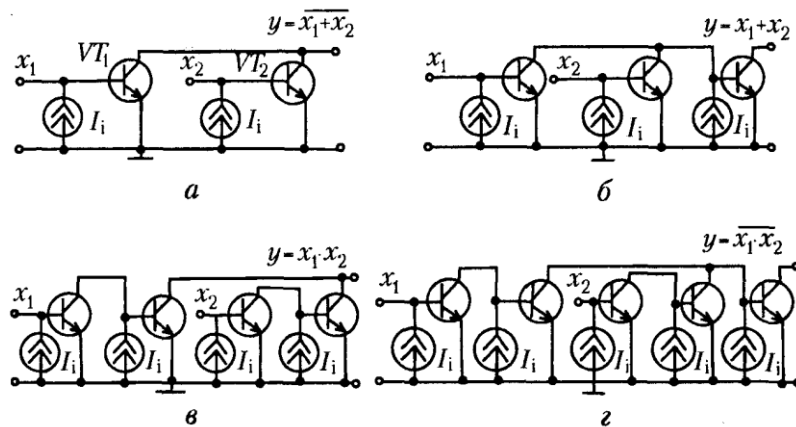


Рис. 2.32. Логічні елементи ПЛ-типу:
а – елемент АБО-НЕ; б – елемент АБО; в – елемент І; г – елемент І-НЕ

Паралельне з'єднання інверторів, до входів яких управляючі сигнали подаються не безпосередньо, а через аналогічні інвертори (рис. 2.33,в), утворюють елемент, що реалізує операцію І. Це випливає з перетворення за теоремою де Моргана:

$$y = \overline{\overline{x_1} + \overline{x_2}} = x_1 \cdot x_2$$

Підключивши до виходу ще один інвертор, отримаємо елемент І-НЕ: $y = \overline{x_1 \cdot x_2}$ (рис. 2.33,г).

Розглянуті елементи працюють без додаткових джерел живлення і без навантажувальних резисторів в колах колекторів. Тому вони відрізняються дуже малою споживаною потужністю і МС на їх базі можуть бути виготовлені з дуже високим рівнем інтеграції. Вони застосовуються в першу чергу в мікропроцесорних наборах (серії 582, 583, 584). На входах і виходах МС типу ПЛ мають спеціальні перетворювачі для узгодження з ТТЛ-елементами, в яких сигнали 1 та 0 відображаються рівнем напруги і які мають достатньо велику вихідну потужність.

Список питань для самоконтролю

1. Що таке комбінаційна схема і цифровий автомат?
2. Які основні параметри мікросхем з транзисторно-транзисторною логікою?
3. Які основні параметри мікросхем з комплементарними метал-окисел-напівпровідник?
4. Що таке логічні елементи і які вони існують?
5. Поясніть ідею способів спрощення логічних задач.
6. Яку логічну операцію виконує ЛЕ, якщо МОН транзистори керування з'єднати паралельно або послідовно?
7. На основі якого елемента будують ЛЕ емітерно-зв'язаної логіки (ЕЗЛ)?
8. З яких схем складається типовий ТТЛ елемент?
9. Які елементи обмежують завадові викиди негативної полярності у схеми ТТЛ?
10. Що забезпечує використання діода Шоттки у схеми ТТЛШ?
11. Які базові елементи серій ІС КМДН ТЛ?
12. Яка структура транзистора з інжекційним живленням?

РОЗДІЛ 3. ЦИФРОВІ ІНТЕГРАЛЬНІ ПРИСТРОЇ КОМБІНАЦІЙНОГО ТИПУ

3.1 Загальні відомості про комбінаційні пристрої

Логічні пристрої можуть бути класифіковані по різних ознаках. По способі уведення/виведення інформації логічні пристрої діляться на послідовні, паралельні й послідовно-паралельні. Послідовним називається пристрій, у якому вхідні змінні подаються на вхід, а вихідні змінні знімаються з виходу не одночасно, а послідовно, розряд за розрядом. Паралельним називається пристрій, у якому всі розряди вхідних змінних подаються на вхід, і всі розряди вихідних змінних знімаються з виходу одночасно. У послідовно-паралельних пристроях вхідні й вихідні змінні представлені в різних формах. Або змінні подаються послідовно символ за символом на вхід, а з виходу вони знімаються одночасно, або навпаки.

За принципом дії всі логічні пристрої діляться на два класи: комбінаційні й послідовні.

Комбінаційними пристроями або автоматами без пам'яті називають логічні пристрої, вихідні сигнали яких однозначно визначаються тільки діючої в цей момент на вхід комбінацією змінних і не залежать от мічений змінних, які діяли на вхід раніше.

Послідовними пристроями, або автоматами з пам'яттю, називають логічні пристрої, вихідні сигнали яких визначаються не тільки діючої в цей момент на вхід комбінацією змінних, але й усе і послідовністю вхідних змінних, які діяли в попередні моменти часу.

Комбінаційну схему можна представити у вигляді $m - k$ полюсного елемента (Рис. 3.1,а). Вхідне слово (вхідний алфавіт) комбінаційної схеми задається набором символів $M = (m_1, m_2, \dots, m_i)$, а вихідне слово (вихідний алфавіт) приймає значення на вихідних символів $K = (k_1, k_2, \dots, k_j)$. У дискретний момент часу сукупність вихідних сигналів однозначно визначається набором вхідних сигналів, які надійшли на вхід у той же момент часу. Комбінаційна схема може характеризуватися: числом вхідних сигналів, числом вихідних сигналів, логічною формулою або таблицею істинності. Для керування роботою комбінаційної схеми вводять керуючі тактові сигнали, щоб новий набір вхідних сигналів надійшов після закінчення перехідних процесів.

Якщо на вихідні сигнали впливає не внутрішній стан комбінаційної схеми, а тільки зміна вхідних сигналів, то така схема називається автоматом без пам'яті або примітивним автоматом.

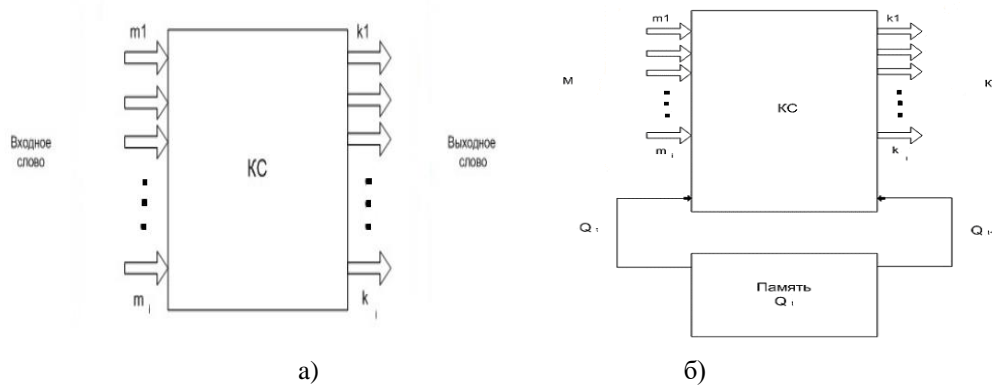


Рис.3.1 Комбінаційну схему у вигляді $m - k$ полюсного елемента

Якщо на набір вихідних сигналів K впливає не тільки набір вхідних сигналів, але й внутрішній стан комбінаційної схеми, то така схема називається автоматом з пам'яттю або повним автоматом (Рис.3.1,б). Автомат з пам'яттю задається трьома наборами змінних: M, K, Q , де Q - набір змінних, які відображають внутрішній стан схеми.

У більшості схем повних автоматів об'єднання комбінаційних елементів дозволяє скласти такий пристрій, сигнали на виході якого будуть безпосередньо залежати не від вхідних сигналів, а лише від об'єднання вхідних сигналів і внутрішнього стану автомата в момент надходження сигналів на вхіді. Тому для опису роботи автомата в таблиці істинності вказують набір вхідних сигналів, набір відповідних внутрішніх станів і набір вихідних сигналів. Крім того, обов'язковим є умова поділу всіх наборів сигналів по тимчасових інтервалах. Спеціально обмовляється внутрішній стан автомата до приходу вхідного сигналу Q_0 , внутрішній стан автомата, що змінюється із приходом вхідного сигналу Q_i та внутрішній стан автомата після нової зміни вхідного сигналу Q_{i+1} .

Розрізняють два типи автоматів. Автомат, у якого вихідна змінна в будь-якому такті t_i , залежить від внутрішнього стану та від вхідний змінної, називається автоматом Милі та задається рівнянням: $K = f(Q_i, M_i)$.

Автомат, у якого вихідна змінна в момент t_i залежить тільки від внутрішнього стану в t_{i-1} м такті, називається автоматом Мура й задається рівнянням $K_j = f(Q_i)$.

3.2. Суматори

3.2.1 Принцип побудови однорозрядного суматора

Суматори призначені для виконання операції додавання багаторозрядних чисел. Звичайно числа представляються в прямих, додаткових або зворотному двійковому кодах, рідше - у двоїчно-десятковому.

Суматори можуть бути нагромаджуючого та комбінаційного типу.

Нагромаджуючі суматори можуть оперувати над декількома (двома й більше) числами, що подаються на їхні входи в різний час, комбінаційні оперують тільки над двома числами, що подаються одночасно. На рисунку 3.2 показане умовне графічне позначення комбінаційного суматора для двох трьохрозрядних чисел $A\{a_2, a_1, a_0\}$ і $B\{b_2, b_1, b_0\}$. На виходах його формується результат додавання S і сигнал переносу P .

Багаторозрядні суматори будуються на основі однорозрядних суматорів, зв'язаних між собою ланцюгами переносу. Залежно від типу ланцюгів переносу розрізняють суматори з послідовним і прискореним переносами.

Напівсуматор. Найпростішим підсумовуючим елементом є напівсуматор (Рис. 3.3). Він має два входи для однорозрядних що складаються A і B і два виходи: C (сума) і Π (перенос).

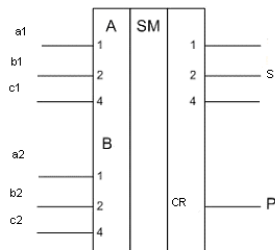


Рис.3.2 Умове графічне позначення трьохрозрядного суматора

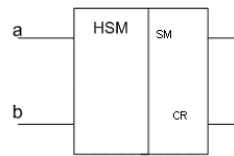


Рис.3.3.Умове графічне позначення напівсуматора

Функціональне призначення його відображає мнемоніка HSM (half sum - напівсума), а роботу - таблиця істинності 3.1.

Таблиця 3.1

a	b	c	П
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

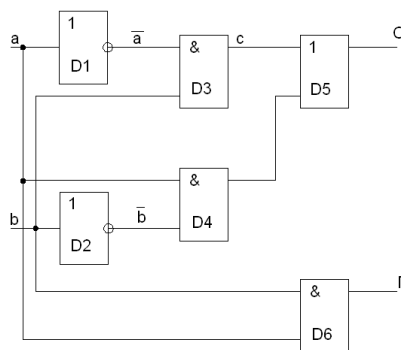


Рис. 3.4. Напівсуматор у базисі простих логічних операцій

Відповідній таблиці аналітичний запис роботи в СДНФ має такий вигляд:

$$C = a\bar{b} \vee \bar{a}b = a \oplus b \quad (3.1)$$

$$\Pi = ab \quad (3.2)$$

Реалізацію цих формул ілюструє рисунок 3.4.

Якщо потрібно побудувати напівсуматор на інших елементах, наприклад І-АБО-НС, необхідно вихідні вирази перетворити відповідним чином:

$$C = \overline{ab} \vee \overline{\bar{a}\bar{b}} = \overline{(a \vee b)} (\bar{a} \vee \bar{b}) = ab \vee \bar{a}\bar{b}, \quad (3.3)$$

$$\Pi = \overline{ab} = \overline{a \vee \bar{b}}. \quad (3.4)$$

Функціональна схема такого напівсуматора наведена на рисунку 3.5.

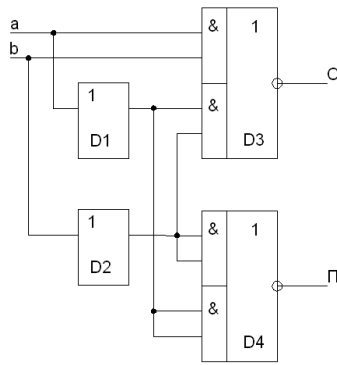


Рис.3.5. Напівсуматор на основі елементів І-АБО- НЕ

Оскільки сума C приймає значення 1, коли $a \neq b$, то операцію виду (3.5) називають ще операцією нерівнозначності або " що виключає АБО". Пристрій, що виконує тільки цю операцію, має самостійне графічне позначення та конструктивне виконання. Тому напівсуматор можна представляти як об'єднання однорозрядного вузла нерівнозначності й формувача сигналу переносу (рис.3.6).

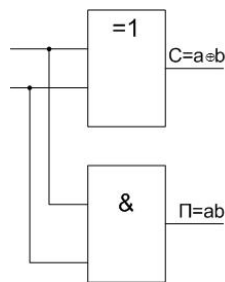


Рис.3.6 Напівсуматор

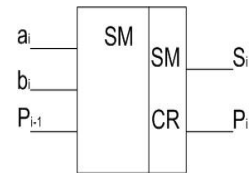


Рис. 3.7. Умовне графічне позначення однорозрядного суматора

Однорозрядний суматор. Напівсуматор придатний для використання тільки в молодшому розряді багаторозрядного суматора. Для другого та наступного розрядів необхідні однорозрядні суматори із трьома входами (рис.3.7): два для однойменних розрядів, що складаються, a_i та b_i чисел A , B и один для сигналу переносу P_{i-1} з попереднього розряду.

Функції виходів S_i и P_i в СДНФ має вид:

$$S_i = \bar{a}_i \bar{b}_i P_{i-1} \vee \bar{a}_i b_i \bar{P}_{i-1} \vee a_i \bar{b}_i \bar{P}_{i-1} \vee a_i b_i P_{i-1}, \quad (3.5)$$

$$P_i = \bar{a}_i b_i P_{i-1} \vee a_i \bar{b}_i P_{i-1} \vee a_i b_i \bar{P}_{i-1} \vee a_i b_i P_{i-1}. \quad (3.6)$$

Для побудови суматора по даних вираженнях необхідно мати сім елементів И на три входи, два елементи АБО на чотири входи й три інвертори.

Трохи економніше його реалізація на двох напівсуматорах. Структурну схему такого вузла можна одержати, виразивши формули (3.5) і (3.6) через (3.1) і (3.2). Для цього треба згрупувати середні та крайні члени формули (3.5) і винести за дужки загальні співмножники:

$$S_i = \bar{P}_{i-1}(\bar{a}_i \bar{b}_i \vee \bar{a}_i b_i) \vee P_{i-1}(a_i \bar{b}_i \vee a_i b_i).$$

Вираження в дужках є сума C_i чисел a_i , b_i і її інверсія.

Дійсно,

$$\bar{C}_i = \overline{a_i \bar{b}_i \vee a_i b_i} = (\bar{a}_i \vee \bar{b}_i)(\bar{a}_i \vee b_i) = \bar{a}_i \bar{b}_i \vee \bar{a}_i b_i. \quad (3.7)$$

Тому

$$S_i = C_i \bar{P}_{i-1} \vee \bar{C}_i P_{i-1} = C_i \oplus P_{i-1}. \quad (3.8)$$

Аналогічно перетвориться вираження (3.6):

$$P_i = (a_i \bar{b}_i \bar{P}_{i-1} \vee a_i b_i P_{i-1}) \vee (\bar{a}_i b_i \bar{P}_{i-1} \vee \bar{a}_i b_i P_{i-1}) = a_i b_i \vee P_{i-1} C_i = \Pi_{11} \vee \Pi_{2i}. \quad (3.9)$$

Схема однорозрядного суматора побудована на рисунку 3.8. У ньому перший напівсуматор складає два числа (a_i, b_i), що належать i -му розряду, та забезпечує одержання їхньої власної суми C_i і переносу Π_1 .

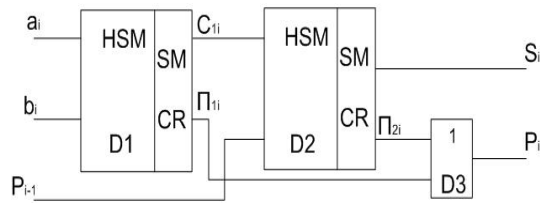


Рис. 3.8. Однорозрядний суматор на двох напівсуматорах

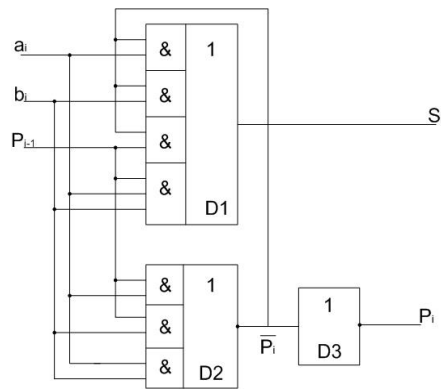


Рис. 3.9. Економічний однорозрядний суматор

Другий напівсуматор складає перенос із попереднього розряду P_{i-1} із сумою C_1 . На його виходах формується повна сума S_i й друга частина загального транзитна складова переносу P_{2i} , що спільно з P_{1i} на виході ЛЕ АБО утворить загальний перенос P_i .

Наступний варіант побудови однорозрядного суматора представлений на рис.3.9. Він виходить у результаті мінімізації виражень (3.5) і (3.6) і приведення їх до виду:

$$\begin{aligned} S_i &= a_i \bar{P}_i \vee b_i \bar{P}_i \vee P_{i-1} \bar{P}_i \vee a_i b_i P_{i-1} \\ P_i &= b_i P_{i-1} \vee a_i P_{i-1} \vee a_i b_i. \end{aligned} \quad (3.10)$$

Цей варіант більше економічний, тому що вимагає найменшого числа вихідних каскадів, що визначають споживання потужності всім пристроєм. Тому йому віддається перевага при створенні повних суматорів у вигляді окремих мікросхем.

3.2.2. Принцип побудови багаторозрядного суматора

Найбільш простим є суматор з послідовним переносом. Він складається з однорозрядних суматорів, число яких відповідає розрядності що складаються А і В з урахуванням знакового розряду (рис.3.10). Суматори вв'язані між собою послідовними ланцюгами переносу з молодших розрядів у старші. Причому молодший розряд зв'язаний зі знаковим SM_{n+1} ланцюгом переносу тільки при використанні зворотних кодів.

Працює суматор у такий спосіб. Всі змінні що складаються А і В подаються на відповідні входи одночасно. У кожному розряді формуються сигнали власної суми та переносу. Останні, через час затримки t_n , надходять на входи сусідніх однорозрядних суматорів та викликають транзитні переноси. Відбувається остаточне коректування результату додавання. Максимальна тривалість підсумовування виходить у випадку, коли перенос передається по всіх розрядах. Протягом всього часу числа А і В необхідно підтримувати на входах незмінними й не робити читання суми. Таким чином час затримки встановлення суматора $T_{зд\ sm} = n * t_n + t_s$.

Якщо числа А і В подані у додатковому й зворотному коді:

$$T_{зд\ sm} = (n+1)t_n + t_s$$

де: n - розрядність додатків, $t_n + t_s$ - час затримки формування сигналу переносу та сигналу суми в одному розряді.

Недоліком суматорів з послідовним переносом є низька швидкодія. Щоб підвищити швидкодію, необхідно прискорити процес формування та поширення переносу.

Найбільшою швидкістю володіють суматори з паралельним переносом. У них сигнал переносу для кожного розряду виробляється з урахуванням аналізу всіх що складаються попередніх молодших розрядів.

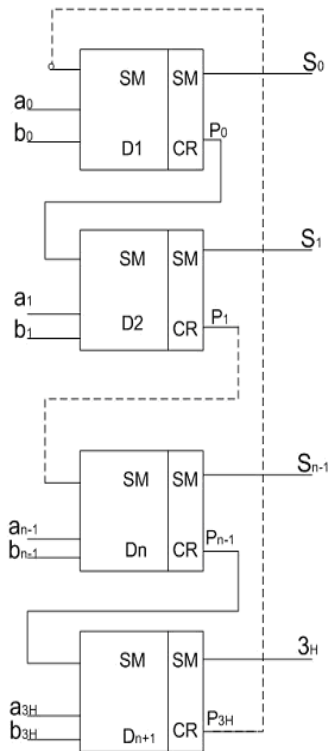


Рис. 3.10. Багаторозрядний суматор з послідовним переносом

Для трьохрозрядного суматора, побудованого на основі напівсуматорів, використовуючи вираження (3.9), запишемо переноси для кожного розряду

$$\begin{aligned}
 P_0 &= \Pi_0, \\
 P_1 &= \Pi_1 \vee P_0 C_1 \\
 P_2 &= \Pi_2 \vee P_1 C_2
 \end{aligned}$$

Виразивши їх через власні переноси й власні суми C_i , одержимо:

$$\begin{aligned}
 P_0 &= \Pi_0, \\
 P_1 &= \Pi_1 \vee \Pi_0 C_1, \\
 P_2 &= \Pi_2 \vee \Pi_1 C_2 \vee \Pi_0 C_1 C_2
 \end{aligned} \tag{3.11}$$

Видно, що для формування сигналів P_i необхідні спеціальні вузли переносу (CRU), які можливо реалізувати за допомогою ЛЕ типу І- АБО (рис.3.11). Час затримки формування переносів визначається тільки часом затримки перемикання цих елементів і не залежить від кількості розрядів.

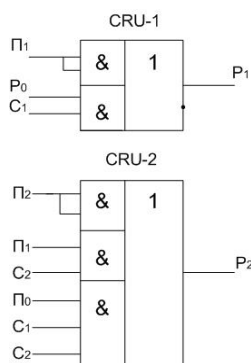


Рис.3.11 Схема вузла переносу (CRU) на ЛЕ типу І- АБО

Схема суматора з паралельним переносом наведена на рисунку 3.12.

Час задержки встановлення його:

$$T_{зд\ sm} = 2t_{зд\ HSM} + t_{зд\ уп},$$

де $t_{зд\ HSM}$ і $t_{зд\ уп}$ - час затримки формування вихідних сигналів напівсуматора й вузла переносу.

Паралельний перенос дозволяє істотно підвищити швидкість суматора, але приводить до збільшення обсягу встаткування за рахунок введення УП, які ускладнюються з ростом розрядності. Тому на практиці для одержання високої швидкості при помірному обсязі встаткування застосовують комбінований перенос. Його сутність полягає в тому, що однорозрядні суматори поєднуються в групи. У середині групи організується одночасний перенос, а між групами - послідовні. Іншим розповсюдженим способом організації прискореного переносу в багаторозрядних суматорах є спосіб, при якому в групах створюється послідовний перенос, а між груп-одночасний. При цьому в порівнянні з попереднім способом виходить вигравш у швидкодії, якщо число груп більше числа розрядів у групах.

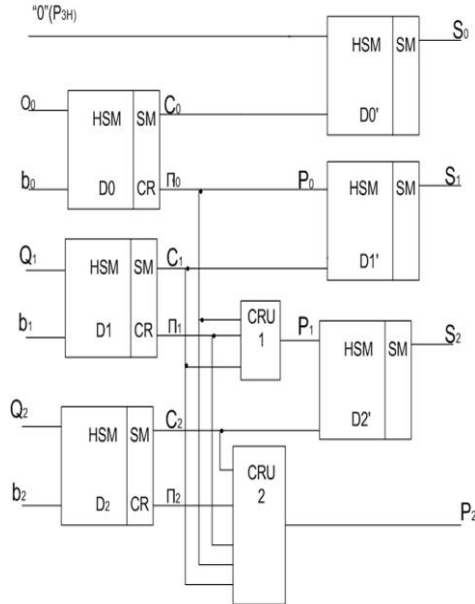


Рис. 3.12. Трьохрозрядний суматор з паралельним переносом

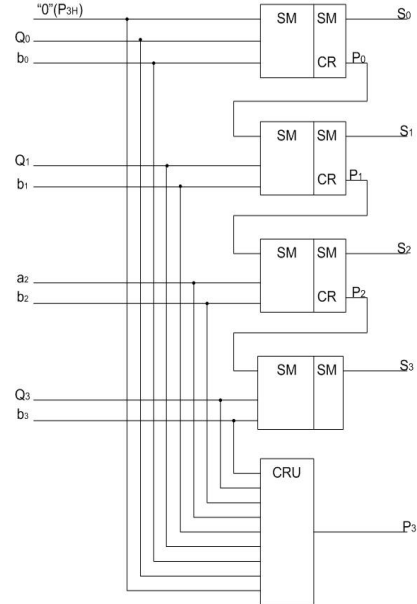


Рис.3.13. Логічна структура суматора 564ИМ1

Прикладом компонування однієї групи розрядів, у якій сигнал переносу останнього розряду виробляється за допомогою спеціального вузла, може служити мікросхема 564 ІМ1, структура якої на рис.3.13.

У таких пристроях однорозрядні суматори звичайно реалізуються за схемою рис.3.9, а вузол переносу поряд з елементами І- АБО містить елементи нерівнозначності для одержання власних сум C_i .

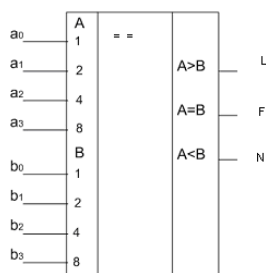
3.3. Цифрові компаратори

Цифрові компаратори ставляться до арифметичних пристроїв комбінаційного типу. Вони призначені для порівняння двох n - розрядних двійкових чисел. Залежно від схемного виконання компаратора результатом порівняння можуть бути ознаки: $A = B$, $A > B$ і $A < B$.

Інтегральні компаратори виконуються функціонально повними, тобто виконують всі три операції порівняння та мають три виходи. На кожному з них виходить результат однієї певної операції у вигляді деякого логічного рівня. Приклад умовного графічного зображення чотирьохрозрядного цифрового компаратора наведений на рисунку 3.14. Вихідні сигнали L , F і N відображаються у вигляді рівня логічної одиниці відповідно при $A=B$, $A > B$ і $A < B$.

Операція рівнозначності F є запереченням операції нерівнозначності. Операції порівняння є порозрядними. Умови роботи однорозрядного функціонально повного компаратора наведені в таблиці 3.2.

Таблиця 3.2



a	b	F	L	N
0	0	1	0	0
0	1	0	0	1
1	0	0	1	0
1	1	1	0	0

$$F = ab \vee \bar{a}\bar{b},$$

$$L = a\bar{b},$$

$$N = \bar{a}b.$$

Рис. 3.14. Умовне графічне позначення чотирьохрозрядного цифрового компаратора

Для реалізації операцій F можна використовувати вузол нерівнозначності на виході, а для операцій L і N – два кон'юнктора D5, D7 (рис.3.15).

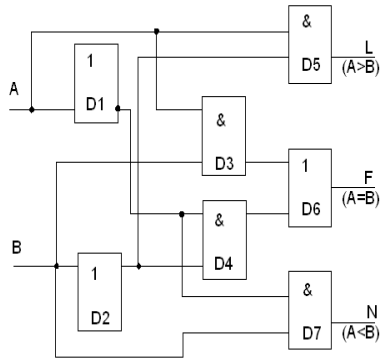


Рис.3.15 Схема повного компаратора

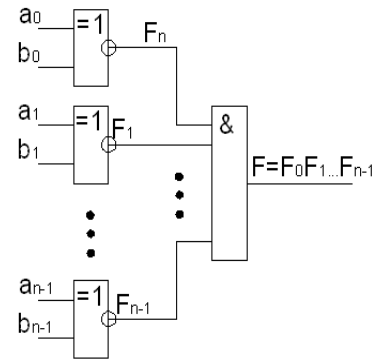


Рис.3.16. Структурна схема багаторозрядного компаратора рівнозначності

У багаторозрядного компаратора факт рівності чисел A та B встановлюється у випадку попарної рівності між собою усіх однойменних розрядів цих чисел (a_i, b_i) . Звідси слідує, що структура багаторозрядного компаратора рівнозначності повинна бути виконана з відносних однотипних однорозрядних вузлів, виходи яких з'єднуються логічним елементом І (рис. 3.16).

У компараторів нерівнозначності розрядні вузли залежать один від одного. При розрядному порівнянні старші розряди користуються пріоритетом над молодшими. Наприклад, в трирозрядному компараторі нерівність $A > B$ забезпечує в наступних випадках: коли $a_2 > b_2$ або $a_2 = b_2$ та $a_1 > b_1$ або $a_2 = b_2$ і $a_1 = b_1$ та $a_0 > b_0$.

В ДДНФ виглядає так:

$$L = \overline{a_2 b_2} \vee (a_2 b_2 \vee \overline{a_2 b_2}) a_1 \overline{b_1} \vee (a_2 b_2 \vee \overline{a_2 b_2}) (a_1 b_1 \vee \overline{a_1 b_1}) a_0 \overline{b_0} = a_2 b_2 \vee F_2 a_1 \overline{b_1} \vee F_2 F_1 a_0 \overline{b_0}$$

Синтезований по даному вираженню компаратор представлений на рис. 3.17. На рисунку, крім цього, показана можливість отримання $A = B$ за допомогою допоміжного вузла рівнозначності молодшого розряду D6 та кон'юнктора D11. Очевидно, для реалізації двох нерівностей $(A > B$ та $A < B)$ компаратор необхідно довершити трьома інверторами для змінних a_i , трьома кон'юнкторами та диз'юнктором аналогічно D7...D10. Типовим прикладом може слугувати чотирирозрядний компаратор K555СП1 (рис.3.18). Він на рівні з чотирма парами входів, які приймають для аналізу два чотирирозрядні числа A та B, має три входи $E_X(A > B)$, $E_X(A = B)$, $E_X(A < B)$, призначені для розширення нарощування розрядності компаратора.

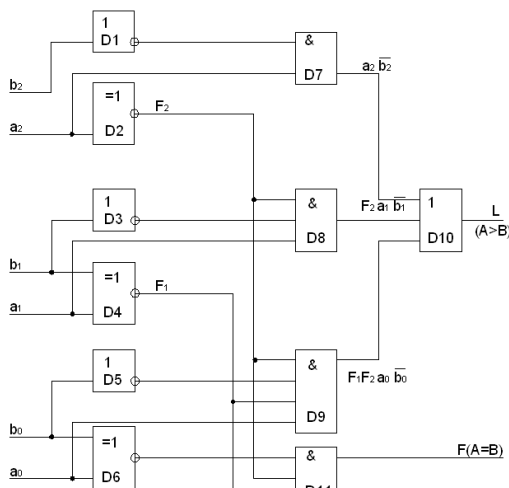


Рис. 3.17. Трирозрядний компаратор операцій $A = B$ і $A > B$

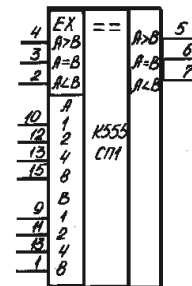


Рис. 3.18. Умовне графічне відображення компаратора K555СП1

Входи нарощування з'ясовують стан виходів L, F та N тільки при $A = B$ відносно таблиці 3.3. Якщо $A \neq B$, то $F = 0$, а $L \neq N$ незалежно від змінних на входах EX.

Таблиця 3.3

Входи нарощування EX			Виходи		
$A > B$	$A = B$	$A < B$	$A > B$	$A = B$	$A < B$
1	0	0	1	0	0
0	0	1	0	0	1
×	1	×	0	1	0
1	0	1	0	0	0
0	0	0	1	0	1

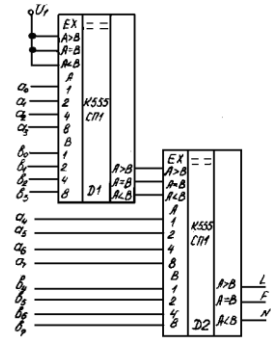


Рис. 3.19. Послідовне з'єднання компараторів K555CP1

Для збільшення розрядності компаратори можна з'єднувати послідовно та пірамідально. При послідовному нарощуванні (рис.3.19) виходи $A > B$, $A = B$, $A < B$ мікросхеми, які аналізують молодші розряди чисел A та B , потрібно під'єднувати до однойменних входів розширення EX мікросхеми, яка аналізує наступні розряди. На розширюючий вхід $EX(A = B)$ першої ІС необхідно подавати логічну одиницю. Цим способом при n мікросхемах можна порівнювати $4n$ —розрядні числа. Основним недоліком послідовного способу нарощування розрядності являється погіршення швидкодії.

На рис. 3.20 приведений приклад структури компаратора с пірамідальним способом нарощування розрядності. Компаратор зібраний на шести ІС K555CP1, П'ять з них створюють першу ступінь і порівнюють числа A та B розрядністю до 24 включно. На їх виходах $A > B$ та $A < B$ формується п'ять пар змінних, які можна представити як два п'ятирозрядних числа x та y , належних порівнянню у другій ступені. У першій ступені ІС молодших розрядів використовується як чотирирозрядний компаратор, а чотири інші – як п'ятирозрядні (входи $EX(A > B)$ та $EX(A < B)$) слугують п'ятою парою розрядних входів. Для забезпечення такого режиму на входи $EX(A = B)$ подано напругу логічного нуля (табл. 3.3). При пірамідальному способу нарощування перехідні процеси встановлюються скоріше, ніж при послідовному, в багатьох випадках потребується великий об'єм обладнання.

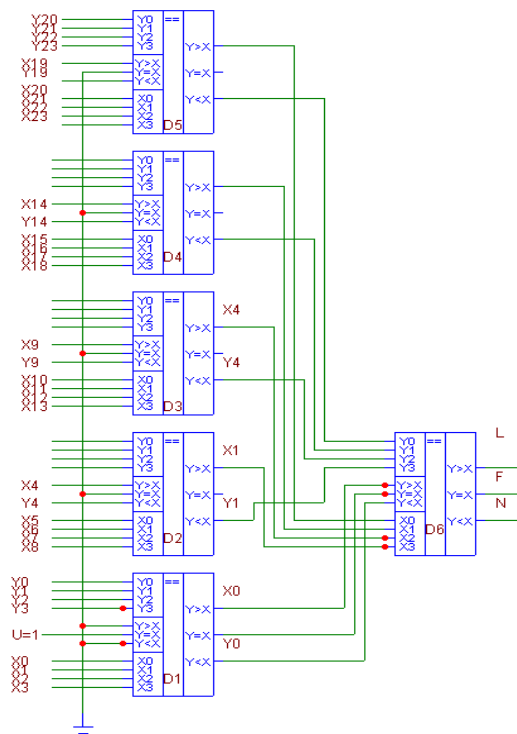


Рис.3.20. Пірамідальне з'єднання компараторів

3.4. Дешифратори

Дешифратори і шифратори використовуються для перетворення інформації та форми її представлення в цифрових системах. Дешифратори призначені для перетворення двійкового коду в інший код. Вони мають n входів та $M \leq 2^n$ виходів. При кількості виходів $M=2^n$ дешифратор називають повним, при меншій кількості – неповним.

У повних дешифраторів кожній комбінації значень вхідних змінних ставиться в залежність напруга логічної одиниці (нуля) на одному з його виходів. У неповних дешифраторів маються перебільшенні комбінації, до яких виходи «нечутливі» або їх стан невизначений.

До неповних відносяться, наприклад, дешифратор двійково-десятичного коду в десятичний. Він має чотири входи для двійкового коду і десять виходів, на яких встановлюється логічна одиниця (нуль) тільки при десяти значеннях вхідного двійкового числа: 0000...1001. Для інших старших комбінацій вхідних змінних стан дешифратора невизначений. Робота дешифратора задана табл. 3.4 і описується наступними вираженнями.

Таблиця 3.4 Таблиця істинності трирозрядного дешифратора

Вхід			Вихід							ДДНФ	
X_1	X_2	X_3	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	
0	0	0	1	0	0	0	0	0	0	0	$Y_0 = \overline{X_1} \overline{X_2} \overline{X_3}$ $Y_1 = \overline{X_1} X_2 X_3$ · · · · $Y_7 = X_1 X_2 X_3$
0	0	1	0	1	0	0	0	0	0	0	
0	1	0	0	0	1	0	0	0	0	0	
0	1	1	0	0	0	1	0	0	0	0	
1	0	0	0	0	0	0	1	0	0	0	
1	0	1	0	0	0	0	0	1	0	0	
1	1	0	0	0	0	0	0	0	1	0	
1	1	1	0	0	0	0	0	0	0	1	

З таблиці та виражень видно, що на виході Y_0 встановлюється логічна одиниця, при вхідному коді 000, на Y_1 при 001, на Y_7 при 111 і т. д. У залежності від способу реалізації функцій виходу Y_i розрізняють дешифратори одноступінчаті (лінійні) і багатоступінчаті. Останні, в свою чергу, діляться на пірамідальні та прямокутні.

При синтезі лінійних дешифраторів кожне логічне вираження y_i реалізується самостійно (рис. 3.21). Лінійні дешифратори прості, економічні і мають високу швидкодію. При великій розрядності вхідного коду перевищуючий максимальне число входів ЛЕ, схема дешифратора суттєво важчає. В такому випадку перевага надається багатоступінчастим дешифраторам. Приклад умовного графічного зображення повного дешифратора на три вхідні змінні представлено на рис. 3.22

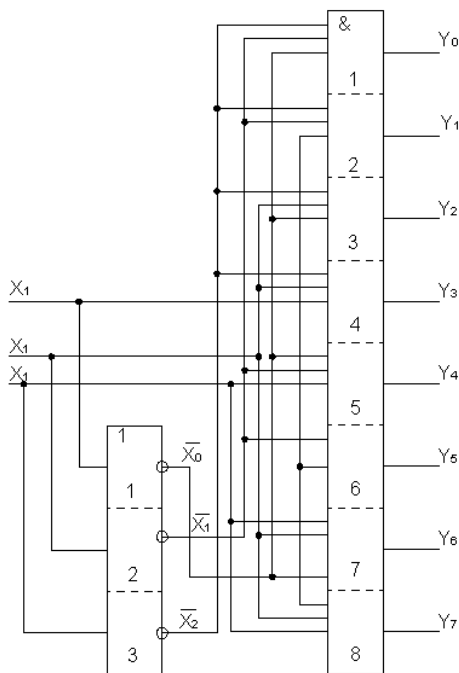


Рис.3.21. Лінійний трирозрядний дешифратор

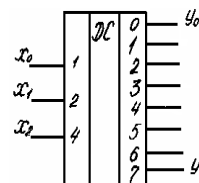


Рис.3.22. Умове графічне зображення

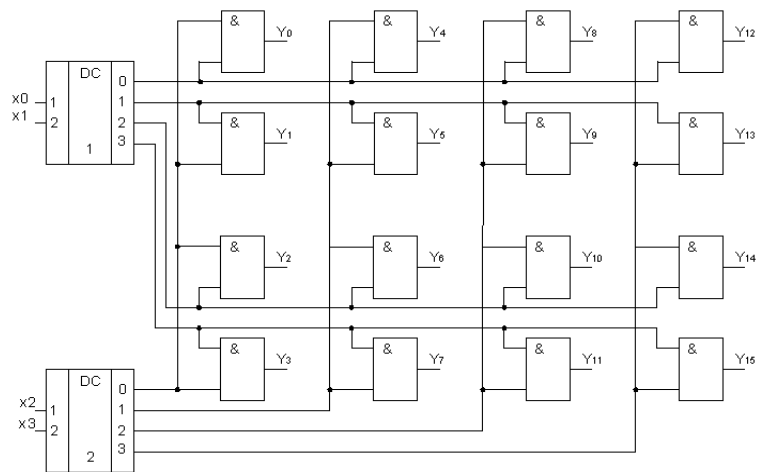


Рис. 3.24. Прямокутний чотирирозрядний двоступінчатий дешифратор

Його збільшена функціональна схема зображена на рис. 3.25. Тут перша ступінь створена двома лінійними дешифраторами на два входи, а до дешифратора другої ступені входить набір (матриця) 4x4 кон'юнктивів. Прямокутні дешифратори по структурній глибині, отже і по швидкодії, займають положення між лінійними і пірамідальними.

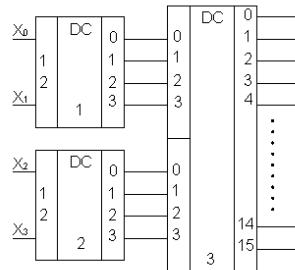


Рис. 3.25. Прямокутний двоступінчатий дешифратор

По об'єму обладнання вони самі економічні при $n > 8$. Тому знаходять використання в БИС та СБИС, де потрібно забезпечити звернення до великого числа (сотні та більше) адресатів, наприклад, до елементів пам'яті в запам'ятовуючих пристроях.

В мікросхемах малої і середньої ступені інтеграції дешифрація виконується лінійним способом. При необхідності розширення числа інформаційних входів мікросхеми-дешифратори об'єднують спеціальним способом. На рис. 3.26 показана схема п'ятирозрядного дешифратора на основі двох чотирирозрядних дешифраторів 133ИД3. У них на збудженому виході встановлюється низький рівень напруги (логічний нуль), а на інших – високий, тобто виходи являються інверсними. Ці ІС з чотирма інформаційними входами мають два інверсних входи розширення, об'єднаних операцією І.

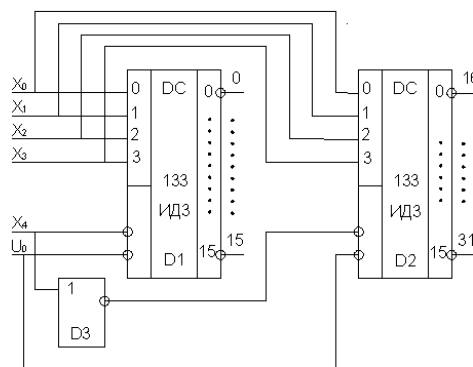


Рис. 3.26. П'ятирозрядний дешифратор на основі двох мікросхем 133ИД1

При наявності логічної одиниці хоча б на одному вході Е на усіх виходах мікросхеми, незалежно від значень змінних на інформаційних входах, встановлюється одиниця, тобто ІС виключається з роботи. Таким чином, завдячуючи інвертору, включеному у ланцюг розширення другої ІС, комбінації вхідних змінних від 0000 до 01111 включно обслуговуються першою ІС, а комбінації 10000...11111 – другою. Найдить застосування і пірамідальний спосіб розширення розрядності вхідного коду.

3.5. Шифратори

Шифратори призначені для виконання операцій, зворотнім операціям дешифраторів. Вони перетворюють різноманітні коди, розрядами котрих являються двійкові змінні, в двійковий код. Табл. 3.5 ілюструє логіку роботи шифратора п'ятиричного коду. Вихідні функції мають вид та реалізуються шифратором на рис. 3.26. Така схема являється пріоритетною. У ній вихідний код визначається старшим значущим розрядом вхідного коду і не залежить від значень змінних більш молодших розрядів. Шифратори в інтегральному виконанні крім інформаційних виводів нерідко мають допоміжні виводи, які дозволяють розширити функціональні можливості мікросхем.

Наприклад, шифратор КМ555ІВ1 (рис. 3.27) має 8 входів десятичного коду, 3 виходи для двійкового коду, а також вхід Е, дозволяючий роботу шифратора, виходи G0 та GS, які сигналізують відносно про нульову та не нульову комбінацію вхідних змінних при наявності дозволяючого сигналу Е.

Поеднуючи визначеним способом додаткові виводу двох та більше шифраторів, можна розширити розрядність перетворюваних кодів.

Таблиця 3.5 Таблиця істинності шифратору п'ятиричного коду

X ₄	X ₃	X ₂	X ₁	X ₀	Y ₂	Y ₁	Y ₀
1	x	x	x	x	1	0	1
0	1	x	x	x	1	0	0
0	0	1	x	x	0	1	1
0	0	0	1	x	0	1	0
0	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0

$$Y_2 = X_4 \vee \overline{X_4} X_3$$

$$Y_1 = \overline{X_4} X_3 X_2 \vee X_4 \overline{X_3} X_2 X_1$$

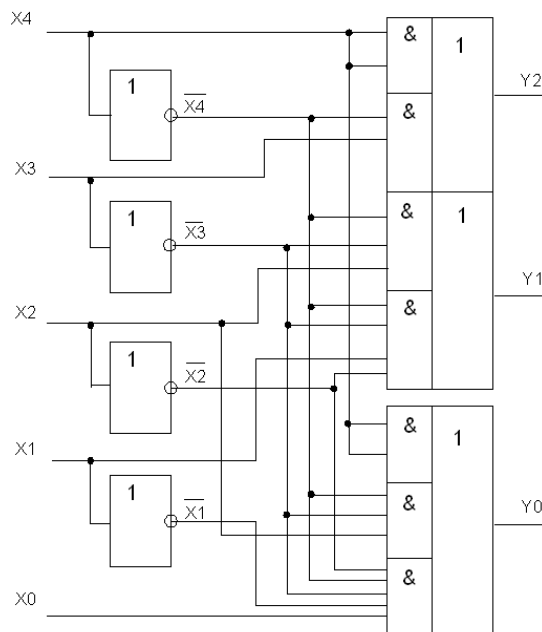
$$Y_0 = X_4 \vee \overline{X_4} X_3 X_2 \vee X_4 \overline{X_3} X_2 \overline{X_1} X_0$$


Рис. 3.26. Пріоритетний шифратор на п'ять входів

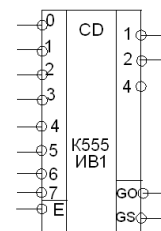


Рис.3.27.Умовнеграфічне позначення шифратора КМ555ІВ1

3.6. Перетворювач двійково-десятькового коду у код семисегментного індикатора

Числа на тарбо і пультах висвічуються, як правило, в десятиковому кодi. Для цього можна використати дешифратор на мікросхемі К555ИД1 разом з газорозрядним індикатором. Однак застосування таких індикаторів в практиці небажано через необхідність використання джерела живлення високої напруги (≈200 В). Зараз широке розповсюдження отримали так звані семисегментні світлодіодні і рідиннокристалічні індикатори, що працюють при тих же напругах, що і мікросхеми. В них індикація здійснюється за допомогою семи елементів (рис. 3.28). Подаючи керуючу напругу на окремі елементи індикатора і викликаючи його світіння (світлодіодні індикатори) або змінюючи його забарвлення (рідиннокристалічні індикатори), можна отримати зображення десятикових цифр 0, 1, ..., 9. Деякі мікросхеми - перетворювачі коду 8421 в семисегментний показані на рис. 3.29.

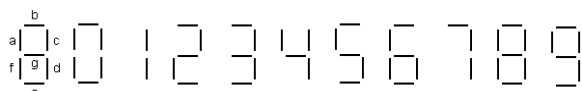


Рис. 3.28. Зображення цифр в семи сегментних семи індикаторах

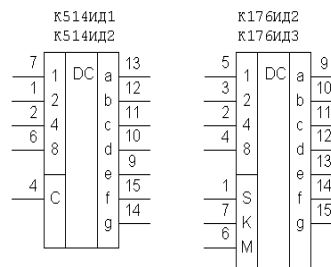


Рис. 3.29. Перетворювачі двійково-десятькового коду в сегментний

На мікросхеми серії К514 подають вхідні сигнали рівня ТТЛ. Сигнал С служить для гасіння індикації (напругою низького рівня). При нормальній роботі рівень сигналу $C=1$. Дешифратор на мікросхемі К514ИД1 працює зі світлодіодними індикаторами, які мають роздільні аноди, а на К514ИД2 - з роздільними катодами. Дешифратор К514ИД2 під'єднують до індикаторів через струмообмежувальні резистори (200 - 500 Ом), а перший має такі резистори у своєму корпусі.

Мікросхеми К176ИД2 і К176ИД3 є перетворювачами коду з вхідним регістром пам'яті. Запис інформації в пам'ять відбувається по фронту тактового сигналу, що подається на вхід S (при цьому сигнал на вході K=0). Якщо сигнал $K=1$, дешифратор блокується. Вихідний код цих дешифраторів - прямий при $M=0$ і зворотний при $M=1$. Дешифратори призначені для роботи з рідинно-кристалічними і люмінесцентними індикаторами. Вони можуть працювати і зі світлодіодними індикаторами при напрузі джерела живлення 9-12 В з зниженою яскравістю світіння (через обмеження струму до 2-3 мА).

3.7 Мультиплектори та демультиплектори

3.7.1. Загальна характеристика мультиплекторів

Мультиплексором називається ЦП, призначений для почергової комутації (перемикання) інформації від одного з n входів на загальний вихід. Номер конкретної вхідної лінії, що підключається до виходу в кожний такт, визначається адресним кодом A_0, A_1, \dots, A_{m-1} . Зв'язок між числом інформаційних n і адресних m входів визначається співвідношенням $n=2^m$. Умовне графічне позначення мультиплекторів показане на рис. 3.30. Таким чином, мультиплексор реалізує керовану передачу даних від кількох вхідних ліній в одну вихідну.

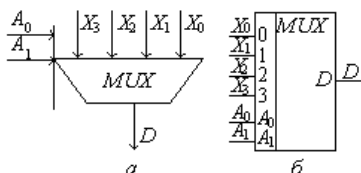


Рис.3.30 Умовне графічне позначення мультиплексора

Функція мультиплекторів записується буквами MUX (multiplexor). Мультиплектори застосовують для таких операцій:

- комутації як окремих ліній, так і груп ліній (шин);
- перетворення паралельного коду в послідовний;
- реалізації логічних функцій;

- побудови схем порівняння, генераторів кодів.

Логіка роботи чотиривходового мультиплексора наведена в табл.3.6, де A_0, A_1 – адресний код; F_0, F_1, F_2, F_3 – виходи внутрішнього дешифратора; X_0, X_1, X_2, X_3 – вхідна інформація; D – загальний інформаційний вихід.

Таблиця 3.6

A_1	A_0	F_0	F_1	F_2	F_3	D
0	0	1	0	0	0	F_0X_0
0	1	0	1	0	0	F_1X_1
1	0	0	0	1	0	F_2X_2
1	1	0	0	0	1	F_3X_3

На основі табл. 3.6, вираз для вихідної функції D можна представити з використанням виходів F_0 – F_3 внутрішнього дешифратора у вигляді:

$$D = F_0X_0 \vee F_1X_1 \vee F_2X_2 \vee F_3X_3,$$

або з мінтермами адресного коду:

$$D = \overline{A_1}\overline{A_0}X_0 \vee \overline{A_1}A_0X_1 \vee A_1\overline{A_0}X_2 \vee A_1A_0X_3$$

Схеми мультиплексорів, відповідні рівнянням (1) і (2), показані на рис. 3.31.

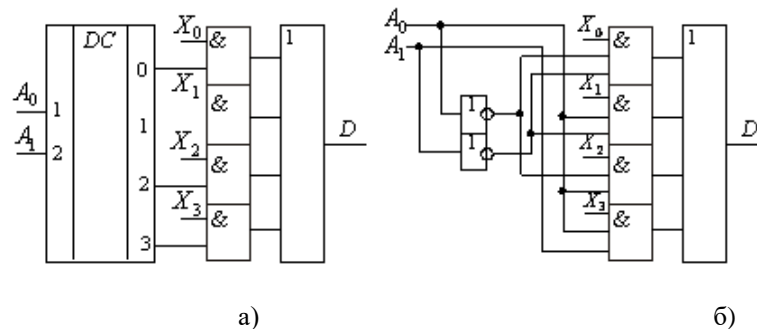


Рис. 3.31. Схеми мультиплексорів

При побудові схеми мультиплексорів з внутрішнім лінійним дешифратором потрібні логічні елементи з меншим числом входів, проте при цьому збільшується час встановлення вихідного сигналу. При виключенні дешифратора швидкодія підвищується, однак потрібні схеми збігу з більшим числом входів.

В інтегральному виконанні мультиплексори випускають на чотири, вісім або шіснадцять входів. Каскадування дозволяє реалізувати комутацію довільного числа вхідних ліній на базі серійних мікросхем мультиплексорів меншої розрядності. Приклад побудови схеми мультиплексора на 16 входів на основі типових чотиривходових мультиплексорів показаний на рис. 3.32. Молодші розряди адреси A_1, A_0 підключаються до адресних входів усіх мультиплексорів першого рівня, на виходах яких виробляються такі функції:

$$D_0' = F_0X_0 \vee F_1X_1 \vee F_2X_2 \vee F_3X_3 ;$$

$$D_1' = F_0X_4 \vee F_1X_5 \vee F_2X_6 \vee F_3X_7 ;$$

$$D_2' = F_0X_8 \vee F_1X_9 \vee F_2X_{10} \vee F_3X_{11} ;$$

$$D_3' = F_0X_{12} \vee F_1X_{13} \vee F_2X_{14} \vee F_3X_{15} ;$$

де $F_0 - F_3$ – виходи внутрішніх дешифраторів:

$$F_0 = \overline{A_1}\overline{A_0} ; F_1 = \overline{A_1}A_0 ; F_2 = A_1\overline{A_0} ; F_3 = A_1A_0$$

$X_{15} - X_0$ – вхідні змінні.

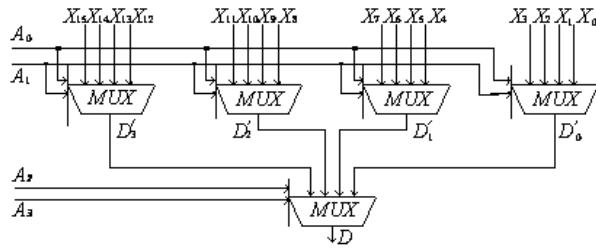


Рис. 3.32. Каскадування мультиплексорів

Старші розряди адреси A_3, A_2 подаються на адресні входи мультиплексора другого рівня, на виході якого формується остаточна такими мінтермами:

$$F_0 = \overline{A_3} \overline{A_2}; \quad F_1 = \overline{A_3} A_2; \quad F_2 = A_3 \overline{A_2}; \quad F_3 = A_3 A_2$$

Нехай, наприклад, функція

$$D = F_0' D_0' \vee F_1' D_1' \vee F_2' D_2' \vee F_3' D_3'$$

де внутрішні виходи дешифратора визначаються значення адреси $A_3 A_2 A_1 A_0 = 1011$.

При цьому на вхід другого рівня комутується змінна D_2^1 (оскільки $F_2^1 = 1$), значення якої визначається з рівнянь першого рівня:

$$D_2^1 = F_3 X_{11} = A_1 A_0 X_{11} = 1 X_{11} = X_{11}.$$

3.7.2. Реалізація логічних функцій мультиплексорами

За допомогою мультиплексорів реалізуються логічні функції з числом змінних m , що дорівнює розрядності адресного коду. Функція, що виконується, має бути представлена в ДНФ. При цьому змінні поступають на адресні входи, а інформаційні входи використовуються як настроювальні – на них подаються константи нуля і одиниці залежно від функції, яка реалізується.

Вихідна функція триадресного мультиплексора на вісім входів описується рівнянням:

$$D(A) = \overline{A_2} \overline{A_1} \overline{A_0} X_0 \vee \overline{A_2} \overline{A_1} A_0 X_1 \vee \overline{A_2} A_1 \overline{A_0} X_2 \vee \overline{A_2} A_1 A_0 X_3 \vee A_2 \overline{A_1} \overline{A_0} X_4 \vee A_2 \overline{A_1} A_0 X_5 \vee A_2 A_1 \overline{A_0} X_6 \vee A_2 A_1 A_0 X_7$$

Якщо потрібно отримати логічну функцію з десятковими еквівалентами мінтермів 1, 3, 5 і 7, то на парні входи X_0, X_2, X_4 і X_6 необхідно подати константу "0", а на непарні X_1, X_3, X_5 і X_7 – константу "1".

У результаті отримуємо (рис. 3.33):

$$D(A) = \overline{A_2} \overline{A_1} \overline{A_0} \vee \overline{A_2} A_1 A_0 \vee A_2 \overline{A_1} \overline{A_0} \vee A_2 A_1 A_0$$

За допомогою додаткових логічних перетворень можна реалізувати логічні функції з числом змінних $m+1$, тобто на одиницю більше розрядності адресного коду мультиплексора.

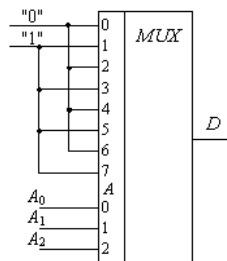


Рис.3.33

Можливі й інші схемотехнічні застосування мультиплексорів. Мультиплексор може виконувати функції перетворювача n -розрядного двійкового паралельного коду в послідовний. Для цього потрібно подати паралельний код на інформаційні входи мультиплексора і потім змінювати код на адреси в необхідній послідовності.

3.7.3. Мультиплексування шин

Мультиплексування шин – це по черговому перемикання шин (груп ліній) від кількох джерел інформації до одного приймача. Такі мікрооперації реалізуються схемами на основі мультиплексорів одиночних ліній. При виборі кількості й типу мультиплексорів враховують: число комутуваних шин дорівнює $2m$, де m – довжина адресного коду;

i -й номер входу всіх мультиплексорів служить для підключення розрядів певної однієї шини. Схема мультиплексора чотирьох $X(n)$, $Y(n)$, $Z(n)$ і $S(n)$ шин показана на рис.3.34. Для її побудови потрібно n двоадресних чотиривходових мультиплексорів, де n – довільна розрядність шин, що комутуються.

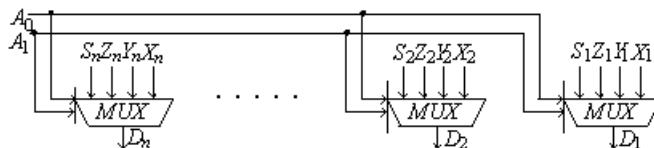


Рис.3.34. Мультиплексор шин

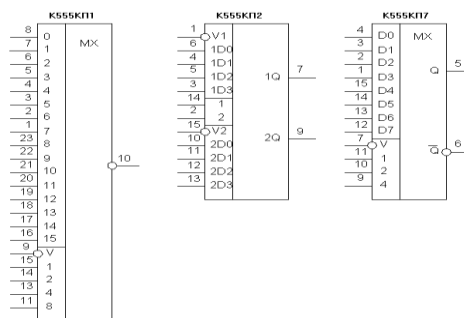


Рис. 3.35. Умовні графічні позначення мультиплексорів

3.7.4. Загальна характеристика демультимплексорів

Демультимплексором називається функціональний вузол комп'ютера, призначений для комутації (перемикання) сигналу з одного інформаційного входу D на один з n інформаційних виходів. Номер виходу, на який в кожний такт машинного часу передається значення вхідного сигналу, визначається адресним кодом $A_0, A_1, A_2, \dots, A_{m-1}$. Адресні входи m та інформаційні виходи n пов'язані співвідношенням $n=2^m$ або $m=\log_2 n$. Демультимплексор виконує функцію, обернену функції мультиплексора. Стосовно мультиплексорів і демультимплексорів користуються також терміном “селектори” даних. В умовних графічних позначеннях (рис.3.36) функція демультимплексора позначається буквами DMX .

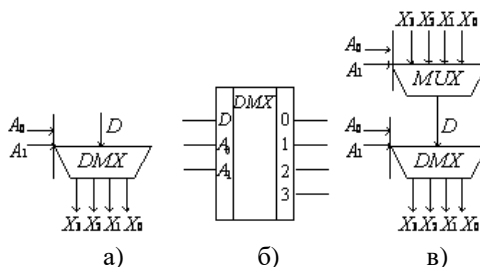


Рис.3.36. Умовні графічні позначення демультимплексорів:

а – на функціональних схемах; б – на принципових схемах; в – типове з'єднання з мультиплексором

Демультимплексори використовують для таких операцій: комутації як окремих ліній, так і багаторозрядних шин; перетворення послідовного коду в паралельний; реалізації логічних функцій та інших.

Логіка роботи двоадресного демультимплексора на мові мікрооперацій наведена в табл. 3.7, де D – інформаційний вхід; F_0, F_1, F_2 і F_3 – виходи внутрішнього дешифратора адреси.

Таблиця 3.7

A ₁	A ₀	F ₀	F ₁	F ₂	F ₃	X ₀	X ₁	X ₂	X ₂
0	0	1	0	0	0	F ₀ D	—	—	—
0	1	0	1	0	0	—	F ₁ D	—	—
1	0	0	0	1	0	—	—	F ₂ D	—
1	1	0	0	0	1	—	—	—	F ₃ D

За даними табл.3.7 запишемо систему рівнянь для інформаційних виходів:

$$X_0 = F_0D = \overline{A_1} \overline{A_0} D; \quad X_1 = F_1D = \overline{A_1} A_0 D; \quad X_2 = F_2D = A_1 \overline{A_0} D; \quad X_3 = F_3D = A_1 A_0 D; \quad (3.12)$$

На основі рівнянь (3.12) побудовані схеми демультиплексорів із внутрішнім дешифратором (рис.3.37, а) із поєднанням адресних і вхідних змінних на тривходових елементах І (рис.3.37, б). Схема демультиплексора з поєднанням адресних і вхідних змінних забезпечує високу швидкодію, проте вимагає застосування логічних елементів з більшим числом входів.

Каскадування дозволяє реалізувати комутацію одного вхідного сигналу на довільне число вихідних ліній на базі серійних мікросхем меншої розрядності. Нехай потрібно реалізувати демультиплексування вхідного сигналу на n вихідних ліній, що визначаються m -розрядним адресним кодом, на базі типових мікросхем меншої розмірності виду " $1 \rightarrow n$ ". Для цього потрібно використати $L = n/n_1$ типових демультиплексорів з числом адресних входів $m_1 = \log_2 n_1$ кожен. Число старших адресних розрядів, що дорівнює різниці $m - m_1$, використовується додатковим "ведучим" демультиплексором, який розташовується у першому рівні схеми каскадування. Ведучий демультиплексор визначає почергове увімкнення одного з L демультиплексорів мікросхем другого рівня. Каскадування демультиплексорів виду " $1 \rightarrow 4$ " для реалізації комутатора " $1 \rightarrow 16$ " показано на рис. 3.38.

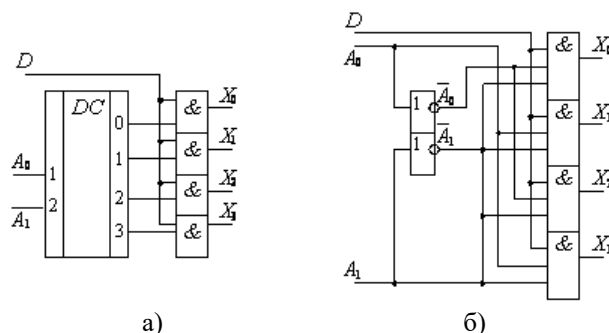


Рис. 3.37. Схема демультиплексорів:

а) – з внутрішнім дешифратором; б) – з поєднанням адресних і вхідних змінних

Нехай для схеми (рис.3.37) адресний код $A_3A_2A_1A_0 = 1010$ і значення вхідного сигналу $D=1$.

Тоді на виході ведучого демультиплексора $D_2' = A_3 \overline{A_2} D = 1$, а на інших виходах встановлюються нульові значення.

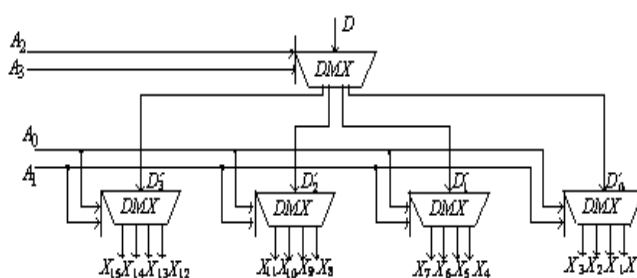


Рис. 3.38. Каскадування демультиплексорів

Одиничне значення сигналу D_2' передається на вихід X_{10} веденого демультиплексора згідно з $X_{10} = A_1 \overline{A_0} D_2 = 1$.

Демультиплексори не випускають як самостійні вироби на інтегральних мікросхемах. Функцію демультиплексора звичайно реалізують на дешифраторах, що мають входи стробування (дешифратори-демультиплексори).

3.7.5. Демультиплексування шин

Під демультиплексуванням шин розуміється почергове перемикування груп ліній від одного джерела інформації до багатьох приймачів. Такі мікрооперації реалізуються звичайно на основі демультиплексорів одиночних ліній. При виборі кількості і типу демультиплексора враховують:

- число шин, які комутуються, дорівнює 2^m , де m – довжина адресного коду;
- кількість демультиплексорів, визначається розрядністю n шин, які демультиплекуються;
- адресні входи всіх мультиплексорів паралельно об'єднуються.

Схема мультиплексора вхідної шини $D(n)$ на чотири вхідні шини $X(n)$, $Y(n)$, $Z(n)$ і $S(n)$ показана на рис.3.39.

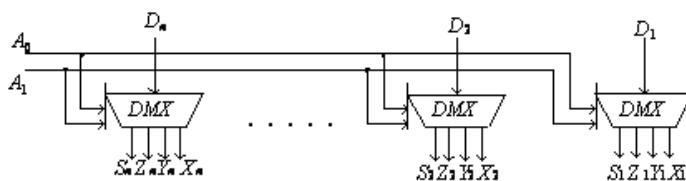


Рис.3.39.Демультиплексор шин

Типове включення мультиплексорів і демультиплексорів для комутації вхідних і вихідних шин n -розрядних регістрів A , B , C і D показано на рис.3.40. У АЛП така комутація забезпечує використання як першого операнда суматора вміст будь-якого регістра і запис результату операції в будь-який регістр, вказаний мікропрограмою команди, що виконується.

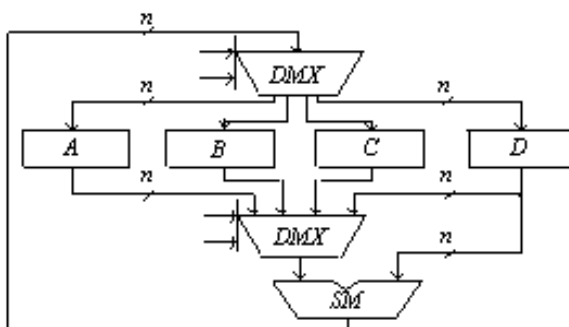


Рис. 3.40. Типова комутація вхідних і вихідних шин регістрів АЛП

Список питань для самоконтролю

1. Що таке шифратор і як він реалізується?
2. Що таке дешифратор і як він реалізується?
3. Наведіть приклади шифраторів і дешифраторів.
4. Як реалізується перетворювач двійково-десятькового коду в код семисегментного індикатора?
5. Що таке мультиплексор і як він реалізується?
6. Що таке суматор і як він реалізується?
7. Які вхідні сигнали повинні бути на входах демультиплексора для того, щоб інформаційний вхід був з'єднаний виходом номер якого відповідає 20?
8. Скільки вихідних шин m повинний мати шифратор, якщо число вхідних шин $n = 7$?
9. Для чого виконується демультиплексування шин?
10. Для яких операцій використовують демультиплексори?
11. Для чого виконується мультиплексування шин?
12. Якому типу дешифраторів надається перевага при більшій розрядності вхідного коду?

РОЗДІЛ 4. ТИПОВІ ЦИФРОВІ ВУЗЛИ з ПАМ'ЯТТЮ

4.1. Загальні відомості про цифрові пристрої накопичувального типу

Функціональні ІС і пристрою накопичувального типу містять елементи пам'яті, здатні зберігати двійкову інформацію тривалий час. Завдяки елементам пам'яті, роль яких звичайно виконують тригери, стан накопичувальних пристроїв визначається не тільки конкретною комбінацією вхідних змінних у цей момент часу, але й станом пристрою до подачі цієї комбінації, тобто визначається послідовністю надходження вхідних сигналів. Тому поряд з терміном «пристрій накопичувального типу» застосовується термін «пристрій послідовного типу».

До пристроїв, що накопичивають, належать тригери, лічильники, регістри, запам'ятовувальні пристрої, суматори що накопичують та ін.

Їхню роботу прийнято розглядати в безрозмірному дискретному часі, для чого реальний час розбивається на інтервали, які нумеруються, починаючи з деякого моменту. Кожний такий проміжок часу називають тактовим інтервалом. Тривалість його для характеристики роботи пристрою не має значення. Тимчасові затримки, обумовлені перехідними процесами, звичайно не враховують, однак коли частота зміни тактів велика й порівнянна зі швидкістю пристрою, з ними доводиться вважатися.

Більшість пристроїв, що накопичивають, є синхронними, тобто мають спеціальні входи, на які подаються синхронізуючі (тактові) сигнали, що є своєрідними виконавчими командами. Вони визначають момент початку роботи пристрою, отже, і момент початку тактового інтервалу. Якщо пристрій асинхронне, то початок тактового інтервалу прив'язують до моменту надходження будь-якого інформаційного сигналу.

Роботу пристроїв що накопичивають, прийнято описувати, зв'язуючи їхнього стану у двох суміжних тактах t^n і t^{n+1} або t^{n-1} і t^n . Відповідно до позначень вхідних і вихідних сигналів, що діють у межах такту, додають номер такту. Так, запис $Q^n=1$ означає, що на виводі Q в інтервалі $t^n < t < t^{n+1}$ діє сигнал логічної одиниці.

Способи опису роботи нагромаджуючих пристроїв подібні до способів опису роботи комбінаційних пристроїв — у вигляді таблиць і виражень, що одержали назву таблиць станів і функцій переходів. Ті й інші виражають зв'язок між вихідними й вхідними сигналами в $(n+1)$ -м такті з урахуванням стану пристрою у попередньому n -м такті.

При великій кількості входів і виходів повні таблиці станів виходять громіздкими. Тому на практиці частіше користуються скороченими таблицями, у яких колонки з вихідними сигналами в n -м такті виключені, а стану в $(n+1)$ -м такті записуються в узагальненому виді, наприклад $Q^{n+1} = \underline{Q}^n$. Це означає, що вихідний сигнал змінив своє значення на протилежне (0 на 1 або 1 на 0).

Як відзначалося вище, що накопичують пристрої можуть бути синхронними й асинхронними.

Асинхронні переходять у новий стан відразу при надходженні на входи певної комбінації інформаційних сигналів.

Синхронні пристрої мають вхід синхронізації й перемикаються тільки при наявності на ньому синхронізуючого сигналу.

Вхід синхронізації може бути статичним і динамічним. Той і іншої, у свою чергу, буває прямим і інверсним.

Статичним входом називають вхід, через який вплив сигналу на пристрій передається протягом усього часу його існування. Причому, якщо під час існування синхронізуючого сигналу будуть мінятися сигнали на інформаційних входах, пристрій буде реагувати на ці зміни. У цьому змісті статичний вхід синхронізації еквівалентний входу дозволу в пристроях комбінаційного типу. Статичними входами звичайно є також входи, на які подаються інформаційні сигнали.

Динамічним входом називають вхід, через який вплив на пристрій передається тільки в момент перепаду вхідного сигналу з 0 в 1 або, навпаки. Властивості такого входу подібно властивості ланцюга, що диференціює виділяти перепади напруги.

При необхідності мітки виводів доповнюють буквами й символами, що показують функціональне призначення виводів. Помітимо, що такими ж буквами й символами позначають виводи й на структурних схемах, якщо виводи прямі, і доповнюють позначення інверсною рисою, наприклад, R , CR , якщо вони інверсні.

Щоб уникнути плутанини в таблицях станів нагромаджуючих пристроїв, умовимося приводити не двійкові змінні, як це роблять деякі автори, а відповідні їм значення сигналів у позитивній логіці, тобто під цифрою 1 будемо мати на увазі напругу високого рівня, а під 0 — низького. Крім того, оскільки входи й вступники на них сигнали звичайно позначають однаковими буквами, для розрізнення будемо вводити в позначення сигналів тактові індекси (наприклад, C^{n+1} , Q^n).

Синтез пристроїв, що накопичивають, ґрунтується на складанні спеціальних таблиць і карт переходів і їхньої обробки за певними правилами.

Основні параметри пристроїв, що накопичивають, характеризують потужні властивості, аналогічні параметрам комбінаційних пристроїв.

4.2 Тригери та їхні характеристики

Найпростішими цифровими автоматами із пам'яттю є тригери. *Тригер* - це прилад послідовного типу з двома стійкими станами рівноваги, призначений для запису і зберігання інформації. Під дією вхідних сигналів тригер може переходити з одного стійкого стану в інший. При цьому напруга на його виході стрибкоподібно змінюється.

Як правило, тригер має два виходи - прямий та інверсний. Число входів залежить від структури і функцій, що виконуються тригером. За способом запису інформації тригери поділяють на асинхронні і синхронізовані. В асинхронних тригерах інформація може записуватися безперервно і визначається інформаційними сигналами, діючими на входах у даний момент часу. Якщо інформація заноситься в тригер тільки в момент дії так званого синхронізуючого сигналу, то такий тригер називають *синхронізованим* або *тактованим*. Окрім інформаційних входів, синхронізовані тригери мають тактовий вхід (синхронізації). В цифровій техніці прийняті такі позначення входів та виходів тригерів:

- Q - прямий вихід тригера;
- \bar{Q} - інверсний вихід тригера;
- S - роздільний вхід установки в одиничний стан (напруга високого рівня на прямому виході Q);
- R - роздільний вхід установки в нульовий стан (напруга низького рівня на прямому виході Q);
- D - інформаційний вхід (на нього подається інформація, призначена для занесення в тригер);
- C - вхід синхронізації;
- T - лічильний вхід.

Найбільше розповсюдження в цифрових приладах отримали RS-тригер з двома установчими входами, тактований D-тригер і лічильний T-тригер. Розглянемо функціональні можливості кожного з них.

Асинхронний RS-тригер. В залежності від логічної структури розрізняють RS-тригери з прямими і інверсними входами. Їхні схеми і умовні позначення наведені на рис. 4.1. Тригери такого типу побудовані на двох логічних елементах: 2АБО-НІ - тригер з прямими входами (рис. 4.1, а), 2І-НІ - тригер з інверсними входами (рис.4.1,б).

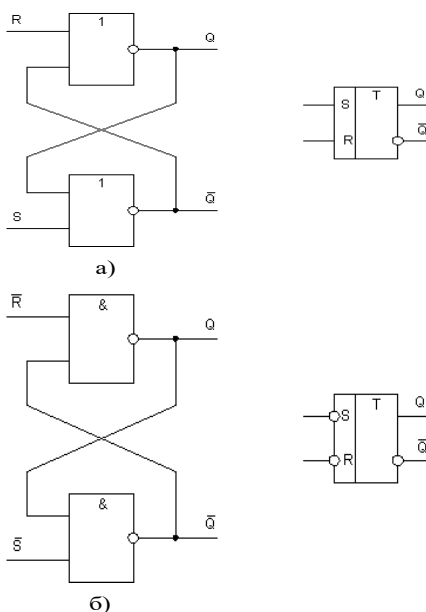


Рис. 4.1. Схеми та умовні позначення RS-тригерів

Вихід кожного з елементів під'єднаний до одного з входів іншого елемента, що забезпечує тригеру два стійких стани. Таблиця істинності наведена у таблиці 4.1 для кожного з цих тригерів.

В таблиці 4.1 сигнали Q^t і $Q^{t'}$ позначають рівні, які були на виході тригера до подачі на його входи активних рівнів. *Активним* називають логічний рівень, що діє на вході логічного елемента і однозначно визначає логічний рівень вихідного сигналу (незалежно від логічних рівнів, що діють на інших входах).

Для елементів АБО-НІ за активний рівень приймають високий рівень, а для елементів І-НІ - низький рівень. Рівні, подача яких на один з входів не призводить до модифікації логічного рівня на виході елемента, називають *пасивними*. Рівні Q^{t+1} і $Q^{t'+1}$ позначають логічні рівні на виході тригера після подачі інформації на його входи.

Для тригера з прямими входами $Q^{t+1} = 1$ при $S = 1$ і $R = 0$; $Q^{t+1} = 0$ при $S = 0$ і $R = 1$; $Q^{t+1} = Q^t$ при $S = 0$ і $R = 0$. При $R = S = 1$ стан тригера буде невизначеним (*), бо під час дії інформаційних сигналів логічні

рівні на виході тригера однакові $Q^{t+1} = Q^{t+1}$, а після закінчення їхньої дії тригер може рівномірно прийняти будь-який із стійких станів. Тому така комбінація є забороненою.

Таблиця 4.1 - Таблиці істинності асинхронних RS-тригерів

S	R	Q	\bar{Q}^t	Q^{t+1}	\bar{Q}^{t+1}	\bar{S}	\bar{R}	Q	\bar{Q}^t	Q^{t+1}	\bar{Q}^{t+1}
0	1	0	1	0	1	0	1	0	1	1	0
1	0	0	1	1	0	1	0	0	1	0	1
0	0	0	1	0	1	0	0	0	1	*	*
1	1	0	1	*	*	1	1	0	1	0	1
0	1	1	0	0	1	0	1	1	0	1	0
1	0	1	0	1	0	1	0	1	0	0	1
0	0	1	0	1	0	0	0	1	0	*	*
1	1	1	0	*	*	1	1	1	0	0	1

Режим $S = 1, R = 0$ називають режимом запису 1 (бо $Q^{t+1} = 1$); режим $S = 0$ і $R = 1$ - режимом запису 0. Режим $S = 0, R = 0$ називається режимом зберігання інформації, бо інформація на виході залишається незмінною. Для тригера з інверсними входами режим запису логічної 1 реалізується при $S = 0, R = 1$, режим запису логічного 0 - при $S = 1, R = 0$. При $R = S = 1$ забезпечується зберігання інформації. Комбінація $R = S = 0$ є забороненою.

Тактовий D-тригер. Він має інформаційний вхід D і вхід синхронізації C і функціонують згідно з таблицею станів 4.2. З подачею синхронізуючого сигналу C тригер встановлюється в стан $Q^{t+1} = D$ і зберігає його до надходження наступного сигналу C. Якщо рівень сигналу на вході C=0, стан тригера стійкий і не залежить від рівня сигналу на інформаційному вході. При цьому на входи RS-тригера з інверсними входами (елементи 3 і 4) надходять пасивні рівні ($R = S = 1$). При подачі на вхід синхронізації рівня C=1 інформація на прямому виході буде повторювати інформацію, що подається на вхід D. Таким чином, при $C = 0$ $Q^{t+1} = Q^t$, а при $C = 1$ $Q^{t+1} = D$. Тут Q^t означає логічний рівень на прямому виході до подачі імпульсу синхронізації, а Q^{t+1} - логічний рівень на цьому виході після подачі імпульсу синхронізації. Таким чином, тригер здійснює затримку двійкової інформації, що надходить на вхід D, на період проходження синхронізуючого сигналу C. Тому його називають ще тригером затримки. Свою назву отримав від слова Delay - затримувати. Одна із можливих структурних схем одноканального D-тригера і його умовне позначення наведені на рис.4.2.

Таблиця 4.2 - Таблиця істинності тактового D-тригера

D	Q	Q^{t+1}
0	0	0
0	1	0
1	0	1
1	1	1

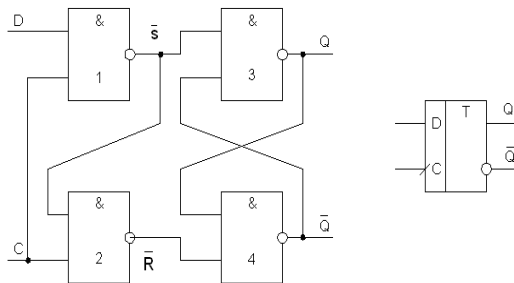


Рис.4.2. Схема та умовне позначення D-тригера з статичним входом синхронізації

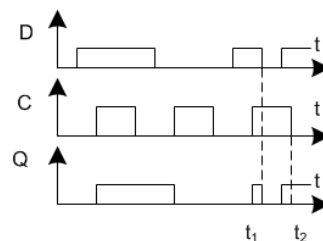


Рис. 4.3 Часові діаграми D-тригера

В такому тригері відбувається затримка сигналу на виході по відношенню до сигналу, поданого на вхід, під час паузи між синхросигналами. Для стійкої роботи тригера необхідно, щоб протягом синхроімпульсу інформація на вході була незмінною.

Тактові D-тригери можуть бути з статичним і динамічним входом синхронізації. У перших з них інформація записується протягом часу, при якому рівень сигналу $C = 1$. На рис.4.2 приведена найпростіша схема D-тригера з прямим статичним входом С. При $C = 0$ логічні елементи ЛЕ1 і ЛЕ2 вимкнені ($\bar{S} = \bar{R} = 1$). Асинхронний RS-триггер (ЛЕ3 і ЛЕ4) є о коміркою пам'яті з інверсними входами, працює в режимі зберігання. Роботу тригера ілюструє рис. 4.3. Сигнал $C=1$ дозволяє передачу інформації зі входу D в комірку пам'яті. Відбувається це так. Якщо $D=0$, то $S\bar{S} = 1$, $\bar{R} = 0$ і $Q^{t+1} = 0$; Якщо $D=1$, то $\bar{S} = 0$, $\bar{R} = 1$ і $Q^{t+1} = 1$.

Зміни інформації на вході D під час дії синхронізуючого сигналу передаються на вихід тригера ($t_1 < t < t_2$). Зазвичай це небажано. Тому, коли необхідно фіксувати інформацію тільки в певний момент часу, застосовують або дуже короткі імпульси С, або D-тригер з динамічним входом С.

В тригерах з динамічним управлінням інформація записується тільки протягом перепаду напруги на вході синхронізації. Динамічні входи зображають на схемах трикутником або лінійкою. Якщо верхівка трикутника звернута в сторону мікросхеми, то тригер «спрацьовує» по фронту вхідного імпульсу, якщо від неї - по зрізу імпульсу. В такому тригері інформація на виході може бути затримана на один такт по відношенню до вхідної інформації.

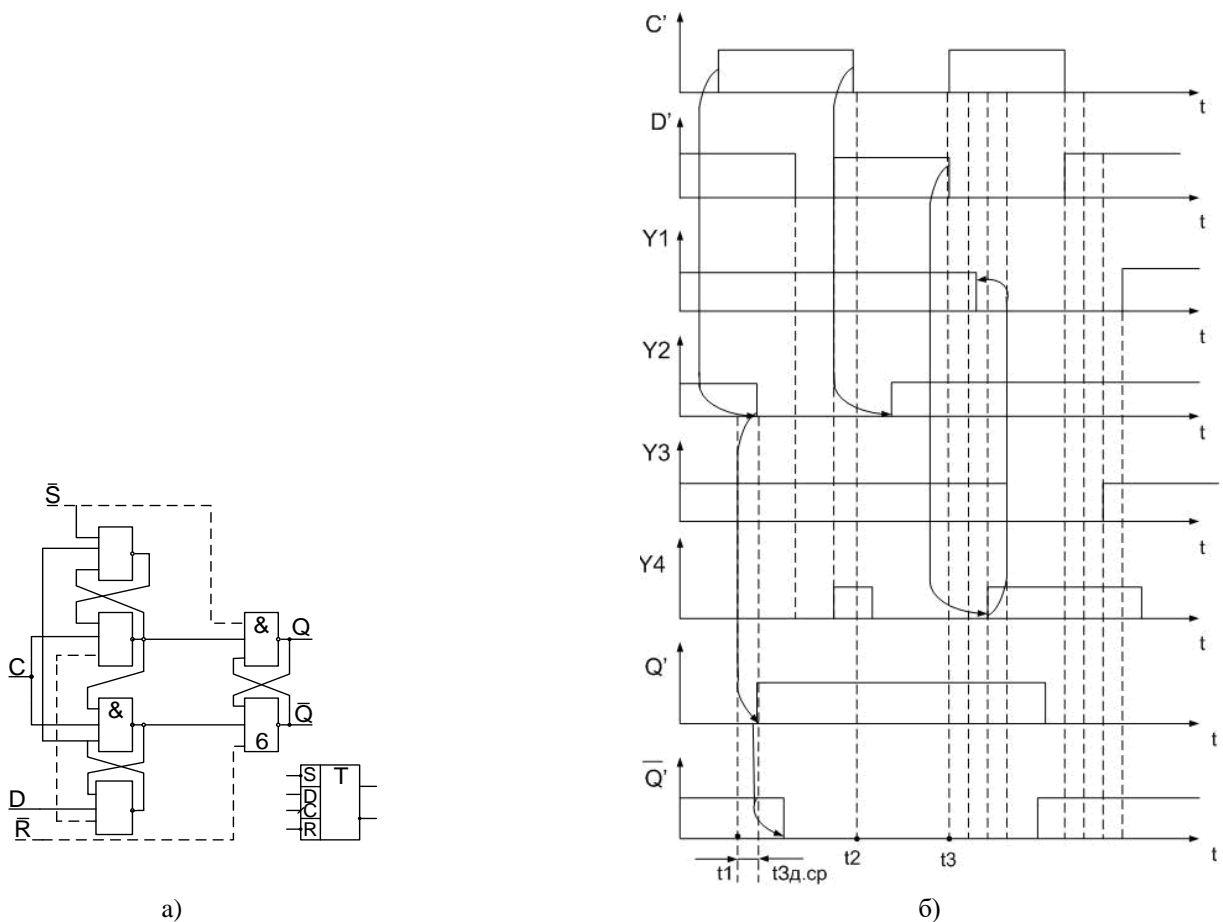


Рис.4.4. D-тригер з динамічним входом синхронізації:
а) схема та УГП, б) часові діаграми роботи

Найбільш поширеною схемою D-тригера з динамічною синхронізацією є схема на трьох асинхронних RS-тригерах (рис.4.4,а). Тригер на ЛЕ5 і ЛЕ6 служить коміркою пам'яті, а два інших - на ЛЕ1, ЛЕ2 і ЛЕ3, ЛЕ4-є допоміжними, утворюють вузол керування. Поряд із входами D і C можуть бути також входи асинхронної установки в одиничний і нульовий стан (S і R). Асинхронність цих входів на умовно-графічному позначенні підкреслюється відділенням їх від області мітки С за допомогою перегородок.

Принцип дії тригера пояснюється тимчасовими діаграмами напруг на виходах логічних елементів (рис.4.4,б). Діаграми побудовані з урахуванням часу затримки переключення ЛЕ $t_{зд,ср}$, що дозволяє підвищити наглядність причинно-наслідкових зв'язків у тригері. Нехай у вихідному стані ($t < t_1$) $D=1$, $C=0$. Сигнал $C=0$ подається на входи ЛЕ2 і ЛЕ3, тому вони вимкнені ($y_2 = y_3 = 1$), комірка пам'яті (ЛЕ5, ЛЕ6) працює в режимі зберігання. Для однозначності прийнято $Q' = 0$. На входах ЛЕ4 дві одиниці, значить, $y_4 = 0$, а пов'язаний з його виходом ЛЕ1 вимкнений ($y_1 = 1$).

З надходженням синхронізуючого сигналу ($t = t_1$) протягом найближчого інтервалу $t_{зд,ср}$ включається ЛЕ2, так як на його входах діють одиничні рівні. Елемент ЛЕ3 свій стан не змінює. Протягом другого інтервалу дискретизації можливо переключення ЛЕ1, ЛЕ3, ЛЕ5, на входи яких надійшов сигнал $y_2 = 0$. Оскільки ЛЕ1 і ЛЕ3 були вимкнені, їх стан не змінюється. Вимкнеться лише ЛЕ5 ($Q' = 1$). Протягом третього інтервалу дискретизації включається лише

ЛЕ6

$$(\overline{Q}^t = 0).$$

На цьому перехідні процеси в тригері закінчуються. Аналогічним чином будуються діаграми при інших комбінаціях зовнішніх сигналів C і D.

З рис.4.4,б видно, що, по-перше, тригер дійсно працює відповідно до функції переходу $Q^{t+1} = D$. По-друге, він переключається тільки в моменти надходження логічного перепаду 0-1 на виході C. Причому, зміна інформації на вході D як під час дії синхронізуючий сигналу ($C = 1$), так і під час паузи ($C = 0$) не призводить до зміни стану тригера.

Отже, вхід синхронізації прямий динамічний. Завдяки цьому тригер фіксує вхідну інформацію тільки по фронту імпульсу C (рис. 4.5).

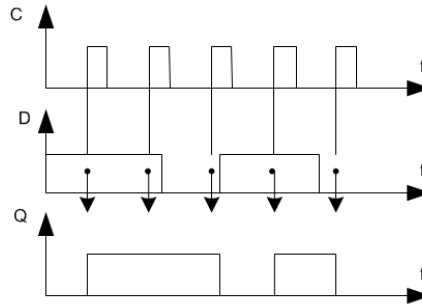


Рис.4.5. Часові діаграми D-тригера з динамічним входом синхронізації

Динамічність синхронізації дозволяє використовувати D-тригер в якості T-тригера. Рахунковий режим виходить шляхом підключення входу D до виходу \overline{Q} (рис. 4.6). До моменту надходження чергового синхронізуючого сигналу ($C = 1$) на інформаційному вході $D = \overline{Q}^t$, отже, в результаті впливу сигналу C стан тригера стане $Q^{t+1} = D = \overline{Q}^t$. Вираз $Q^{t+1} = \overline{Q}^t$ є функцією переходів T-тригера.

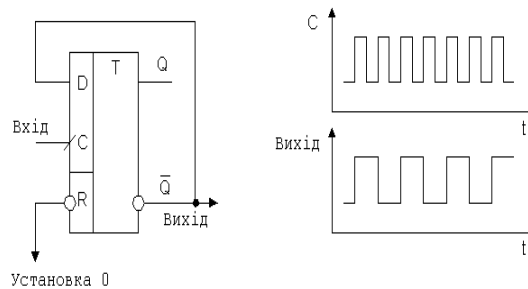


Рис.4.6. Використання D-тригер в якості T-тригера

Швидкодія тригера в рахунковому режимі визначається мінімальним періодом проходження лічильних імпульсів, який, у свою чергу, відповідає логічній глибині, тобто найбільш довгому шляху проходження сигналу. З рис. 4.4,б слід, що логічна глибина (з урахуванням зв'язку D с \overline{Q}) дорівнює п'яти: ЛЕ2-ЛЕ5-ЛЕ:-ЛЕ4-ЛЕ1 і

$$F_{\text{макс}} = 1/5t_{3д.ср}$$

Мінімальна тривалість лічильних імпульсів дорівнює мінімальній тривалості керуючих сигналів на входах елементу пам'яті (асинхронного RS- тригера) і становить $t_{i, \text{мін}} = 2t_{3д.ср}$.

Тригери типу D можуть виконуватися також двоступінчастими. Для цього використовуються D-тригери з статичною синхронізацією (рис.4.7).

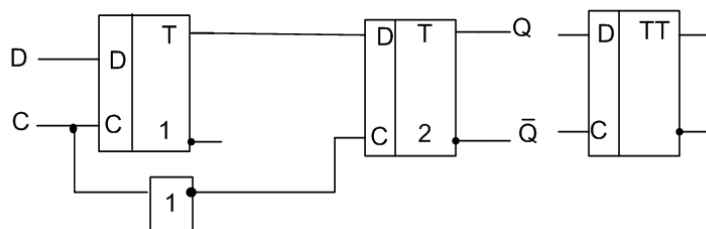
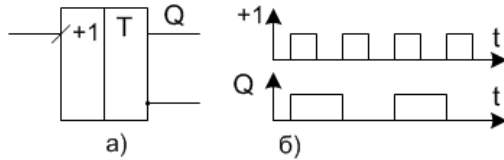


Рис.4.7. Двоступінчастий D-тригер

Порядок роботи двоступінчастого D-тригера такий же, як двоступінчастого RS-тригера. Під час дії сигналу $C=1$ інформація з входу D записується в тригер першого ступеню, а при $C=0$ - другого ступеня. При цьому друга ступінь фіксує лише той стан першого ступеня, який мав місце у момент зрізу імпульсу синхронізації, тобто двоступенева структура надає властивість динамічної синхронізації.

Лічильний T-тригер. T-тригер (рахунковий тригер) має всього єдиний інформаційний вхід, який позначається +1. Його називають також тригером зі лічильним входом (рис.4.8). Зміна станів відбувається при кожному надходженні на нього одиничної інформації (1 або 0), тобто має місце логіка $Q^{t+1}=Q^t$. Якщо вхід тригера прямий, зміна стану відбувається при вступі перепаду 0-1, якщо інверсний то 1-0. Умовне графічне позначення T-тригера з прямим рахунковим входом і ілюструючи його роботу тимчасові діаграми показані



на рис.4.8.

Рис. 4.8. Лічильний тригер

Переключення по кожному вхідному імпульсу можна трактувати як підрахунок їх за модулем 2 (після кожного непарного імпульсу на виході $Q=1$, а після парного $Q = 0$). Така властивість тригера використовується в лічильниках і накопичувального суматора. У інтегральному виконанні у вигляді самостійних виробів T-тригери не виготовляються, так як їх функції можуть виконувати тригери D - і JK-типів. T- тригер може бути створений на базі тактового D-тригера, якщо його інверсний вихід з'єднати з інформаційним входом. Як видно з діаграми, частота сигналу на виході T-тригера в два рази нижче частоти сигналу на вході, тому такий тригер можна використовувати як подільник частоти і двійковий лічильник.

В серіях мікросхем, що випускаються, є також універсальні JK-тригери. При відповідному під'єднанні вхідної логіки JK-тригер може виконувати функції тригера будь-якого іншого типу.

JK-тригер.

JK-тригер має два інформаційні входи (JK) і вхід синхронізації. Працює він згідно функції переходів $Q^{t+1} = \overline{K}Q^t \vee J\overline{Q}^t$, або відповідній їй таблиці станів (рис.4.9).

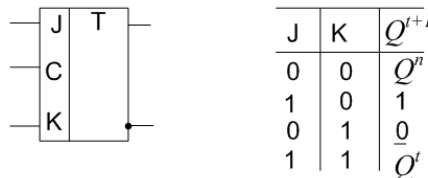


Рис. 4.9. JK-тригер: а) умовно графічне позначення, б) таблиця станів

З надходженням синхронізуючого сигналу C і за наявності на інформаційних входах сигналів $J = K = 1$ тригер перемикається в протилежний стан $Q^{t+1} = \overline{Q}^t$, тобто працює в рахунковому режимі.

В інших випадках він функціонує як синхронний RS-тригер з прямими входами, якщо вхід J вважати входом S, а вхід K - входом R.

Тригер JK-типу може працювати і як D-тригер, якщо не зважати на випадки рівнозначності сигналів J і K.

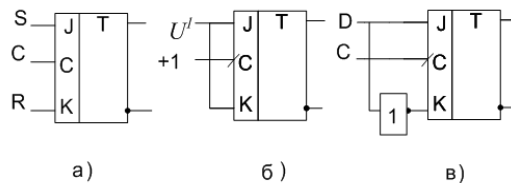


Рис.4.10 JK-тригер в якості тригерів RS (а), T (б) та D (в) типів

На рис.4.10 показані схеми використання його в режимах: а) синхронного RS-тригера при $J = K = 1$, б) T-тригера, в) D-тригера.

На практиці найбільш широке розповсюдження отримали двоступінчасті JK-тригери. Їх структурна схема та умовне графічне позначення представлені на рис. 4.11. За складом і зв'язкам схема аналогічна двоступінчастим

RS-тригерам. Додані лише зворотні зв'язки з виходів другого ступеня на входи першого, що виключають невизначений стан при комбінації $J=K=1$. Для створення зворотних зв'язків тригер першого ступеня містить по два входи S і R, об'єднаних кон'юнктом.

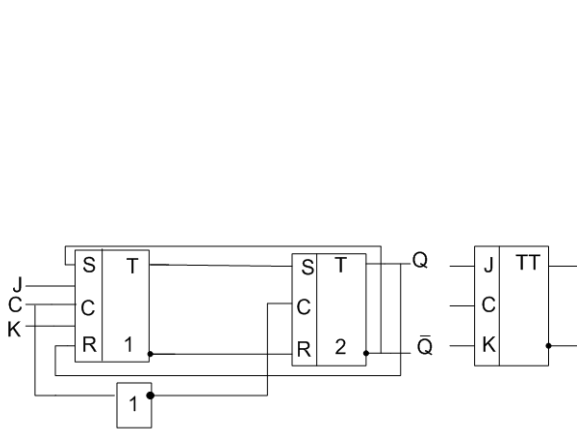


Рис.4.11 Двоступінчастий JK-тригера

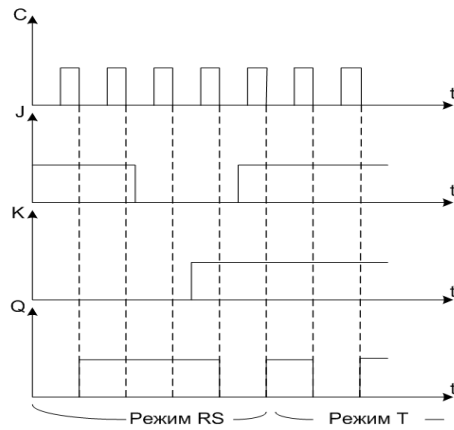


Рис.4.12. Часові діаграми, що ілюструють роботу JK-тригера,

У цілому організація роботи двоступінчастого JK-тригера така ж, як і у всіх раніше розглянутих двоступеневих структурах: почерговий перехід ступенів з режиму прийому інформації в режим зберігання, що визначається синхронізуючим сигналом. Проте перехід тригера в той або інший стан залежить не тільки від інформаційних сигналів J і K але і від стану його в попередньому такті. Покажемо це на прикладі роботи, коли $J = K = 1$. До надходження чергового сигналу $C=1$ на виході $Q^t=0$, то на обох входах S першого тригера буде одиничний рівень, а на одному з входів R - нульовий. Тому в момент надходження сигналу $C=1$ перший тригер встановиться в одиничний стан, а після закінчення його ($C=0$) другий тригер встановиться в одиничний стан ($Q^t=1$). У силу осової симетрії схеми, якщо $Q^t=1$, то $Q^{t+1}=0$. Отже, місце має рахунковий режим: $Q^{t+1} = \overline{Q^t}$. Тимчасові діаграми, що ілюструють роботу JK-тригера, наведені на рис. 4.12.

На рис.4.13 додана функціональна схема універсального тригера IC K133TB1. Він має по три входи J і K, пов'язаних операцією I, а також асинхронні входи S і R для установки тригера в одиничний і нульовий стану незалежно від сигналів на інформаційних і синхронізуючих входах. Тригер першого ступеня (провідний тригер) утворений елементами D1 і D2, а другого ступеня (підпорядкований) - елементами D3 ... D8. Вузол управління RS-тригера (D3... D6) крім звичайної ролі виконує також роль інвертора сигналу C для другого ступеня (D5, D6).

Багато серій цифрових IC поряд з двоступінчастими тригерами містять одноступінчасті з динамічною синхронізацією. З точки зору функціонування між ними відмінностей немає. Але цим самим забезпечується різноманітність деяких споживчих властивостей тригерів. Наприклад, тригер, що входить до складу IC K155TB15, має один інформаційний вхід прямиий (J), а другий-інверсний (K), вхід синхронізації - динамічний. Така особливість дозволяє перетворити його в D-тригер з прямими входами шляхом простого з'єднання виходів J і K.

Швидкодія JK-тригерів (в рахунковому режимі) залежить від конкретної схеми і зазвичай трохи поступається швидкодією D-тригерів.

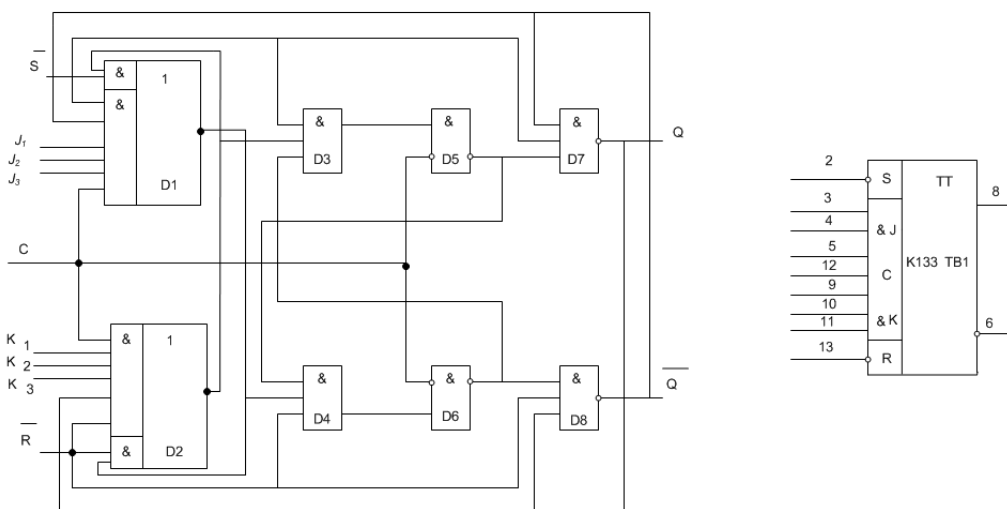


Рис. 4.13. Схема універсального JK -тригера IC K133TB1

- с паралельним прийомом і видачею;
- с послідовним прийомом і видачею;
- с комбінованим прийомом і видачею.

Регістри з паралельним прийомом і видачею називають регістрами пам'яті, а всі інші - регістри зсуву. По напрямку зсуву (передачі) інформації розрізняють односпрямовані й реверсивні регістри.

4.3.1. Принцип роботи регістрів пам'яті

Регістри пам'яті являють собою набір *синхронних тригерів* з незалежними інформаційними входами. Тригери поєднуються лише керуючими сигналами (синхронізації, дозволу запису, високого імпедансу та ін.). Уведення (запис) і вивід (зчитування) інформації виробляється одночасно в усіх (розрядах паралельним кодом. Уведення забезпечується синхронізуючим імпульсом. Із приходом чергового такого імпульсу відбувається відновлення записаної інформації. Якщо вхідна інформація парафазна, тобто кожна змінна надходить одночасно в прямому й інверсному виді, використовуються *RS- або JK-тригери* (рис. 4.4,а), якщо однофазна, — *D-тригери* (рис. 4.16,б). Зчитування може відбуватися в прямому й зворотному кодах, тобто із прямих і інверсних виходів тригерів. У регістрах, призначених для роботи на інформаційну магістраль, у якості вихідних (буферних) каскадів використовуються елементи із трьома станами, керовані сигналом дозволу зчитування *ERD*.

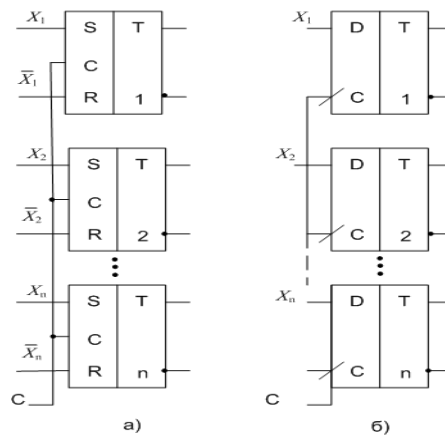


Рис. 4.16. Регістри пам'яті на основі синхронних *RS*-(а) і *D*-тригерів (б)

Типовим прикладом може служити мікросхема 555ИР15 (рис.4.17). Це чотирихрозрядний регістр пам'яті з однофазними інформаційними входами D_1, D_2, D_3, D_4 запис через які виробляється по перепаду 0—1 синхронізуючого сигналу C , якщо на обох входах дозволу PE низька напруга. Подача одиничного рівня хоча б на один вхід PE переводять регістр у режим зберігання. Для зчитування інформації, що зберігається в регістр, на обидва входи ERD (1,2) подається низька напруга, у протилежному випадку виходи відключаються, що відповідає високоімпедансному стану. Нарощування розрядності регістра досягається паралельним з'єднанням керуючих входів декількох мікросхем.

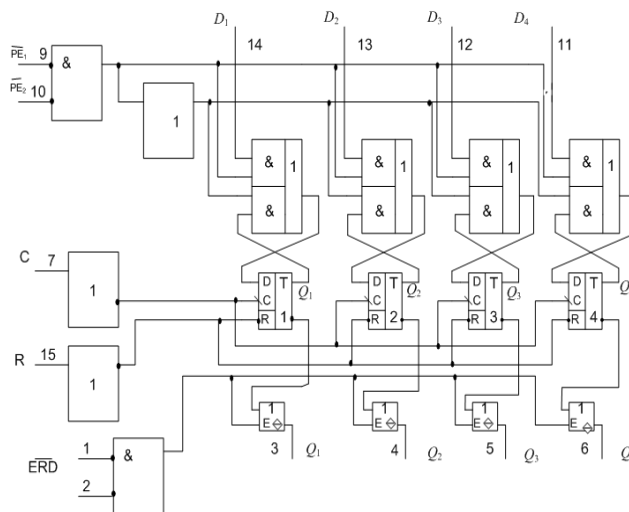


Рис. 4.17. Регістр пам'яті 555ИР15

4.3.2 Регістри зсуву

Регістри зсуву, мають властивість переміщати записану в них інформацію вліво (убік старших розрядів) або вправо (убік молодших розрядів) на задане число розрядів. Тому тригери в них з'єднуються послідовно. Сутність зрушення полягає в тому, що із приходом синхронізуючого імпульсу відбувається перезапис змісту тригера кожного розряду в сусідній розряд. Число зсуву визначається числом синхронізуючих імпульсів, що надійшли. Реальний напрямок зсуву в регістрі визначається напрямком між тригерних зв'язків. Однак з погляду зсуву, щодо числової розрядної сітки один і той же регістр, що має односпрямовані між тригерні зв'язку, може бути регістром зі зсувом вправо й регістром зі зсувом уліво залежно від того, у якому порядку розміщена інформація.

Будуються регістри на *тригерах з динамічним входом синхронізації* або із *внутрішньою затримкою*. Ця умова є обов'язковим. У протилежному випадку при надходженні синхронізуючого сигналу інформація може пересунути не на один розряд, а на стільки, скільки разів встигнуть перемкнутися тригери за час дії цього сигналу.

На рис.4.18 показана схема, умовне графічне позначення та тимчасові діаграми роботи трьох розрядного регістра зсуву на D-тригерах. Він реалізує спосіб послідовного прийому й послідовної видачі. Передбачається, що вхідна інформація надходить молодшими розрядами вперед, тобто вхідним тригером є тригер старшого розряду, а вміст регістра зчитується з виходу Q_1 тригера молодшого розряду.

По кожному фронту імпульсу C двійкова змінна яка зберігається в i -му тригері заноситься у $(i-1)$ -й тригер. Оскільки входи C динамічні, наступні зміни рівнів напруги на входах D у межах даного сигналу $C=1$ не приводять до нових перемикань тригерів. Таким чином, при надходженні одного імпульсу C інформація в регістрі зсувається на один розряд вправо: змінна x із входу 0 записується в тригер D_3 . Зміст D_3 переміщується в тригер D_2 , а попередній зміст D_2 — у D_1 . Для повного завантаження регістра новою інформацією необхідно подати три (по числу розрядів регістра) синхронізуючого імпульсу. Наступні імпульси C забезпечують видачу записаної інформації з виходу тригера D_1 .

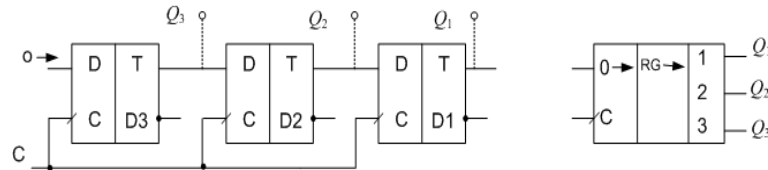


Рис.4.18 Схема трьох розрядного регістра зсуву

Якщо імпульси синхронізації надходять безупинно з періодом T_c , то вихідний сигнал Q повторює вхідний до із затримкою на nT_c , і регістр виступає в ролі пристрою дискретної затримки цифрової інформації. Виходи Q тригерів звичайно мають зовнішні виводи, з яких можна знімати записану інформацію. Такий регістр може служити перетворювачем послідовного коду в паралельний.

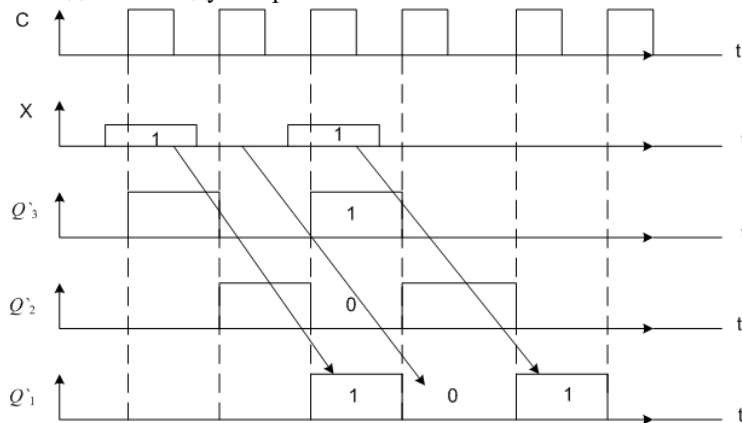


Рис.4.19 Тимчасові діаграми трьохрозрядного регістра зсуву

Щоб змінювати напрямок зсуву, не міняючи послідовності надходження вхідних інформаційних сигналів, використовують реверсивні регістри. Принцип побудови їх заснований на комутації входу i -го тригера до виходу $(i-1)$ -го або $(i+1)$ -го тригера (рис.4.20). Тут при сигналі керування $E=1$ підключається $(i-1)$ -й тригер, а при $E=0$ — $(i+1)$ -й тригер, відповідно встановлюється режим зрушення вліво або вправо.

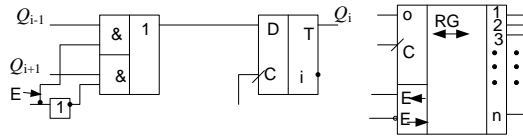


Рис. 4.20. Міжтригерні зв'язки реверсивного регістра

Функціональні можливості регістрів значно розширюються при введенні режиму паралельного запису інформації. Такі регістри додатково можуть виконувати функції регістрів пам'яті, а також перетворювачів паралельного коду в послідовний. На рис. 4.21 показана схема такого регістра. Він зібраний на *JK-тригерах* з асинхронними входами *S* і *R*. Паралельний запис двійкової інформації здійснюється за допомогою суматорів, керованих входними змінними *D*. Залежно від значення D_i команда запису C_2 проходить на вхід *S* або вхід *R* *i*-го тригера. У режимі зрушення виконавчим сигналом служить C_1 .

Однотимчасне надходження синхронізуючих сигналів C_1 і C_2 неприпустимо. Для того, щоб сигнали C_1 і C_2 не впливали на регістр при їхньому випадковому збігу, тобто виключалося накладення різних режимів роботи, у багатьох практичних схемах вводиться керування режимами.

Багато сучасних регістрів, містять вихідні ключі з високоімпедансним станом, що спрощує сполучення з інформаційними магістралями. Споживчі можливості регістрів ще вище, якщо ключі двунправлені. Введення ключів також у вхідні ланцюги регістра дозволяє істотно спростити організацію двунправленої передачі інформації між двома магістралями, причому з можливістю попередньої обробки її. Такою обробкою може бути зрушення двійкового числа на n розрядів уліво або вправо. Якщо при цьому не відбувається втрата знаків числа (що можливо, коли розрядність регістра вище розрядності числа), то зрушення рівноцінне відповідно множенню або розподілу числа на 2^m . Прикладом регістра з можливістю двунправленої передачі слугує ІС 564ІР6.

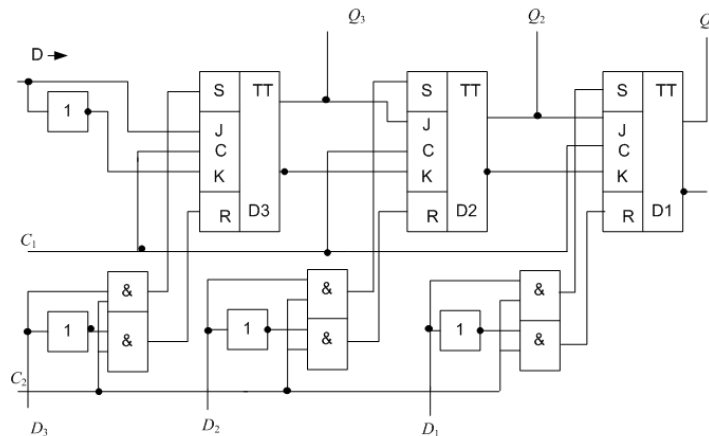


Рис. 4.21. Регістр зсуву з можливістю паралельного запису інформації

Область застосування регістрів зсуву широка. Регістр може бути кільцевим лічильником при підключенні входу до прямого виходу останнього тригера. Регістр зсуву можна перетворити в кільцевий лічильник (рис.4.22), якщо вихід останнього тригера з'єднати з входом *D* першого. Перед початком підрахунку імпульсом початкової установки в нульовий розряд лічильника (Q_0) записується логічна 1, в інші розряди — логічні 0. З початком рахунку кожний з лічильних імпульсів *T*, що приходять, перезаписує 1 в наступний тригер, і число імпульсів, що надійшли, визначається за номером виходу, на якому є 1. Передостанній ($N-1$) імпульс переведе в одиничний стан останній тригер, а *N*-ий імпульс перенесе цей стан на вихід нульового тригера, і підрахунок розпочнеться спочатку. Таким чином, можна побудувати кільцевий лічильник з довільним коефіцієнтом перерахунку (будь-якою основою числення), змінюючи лише число тригерів в ланцюжку.

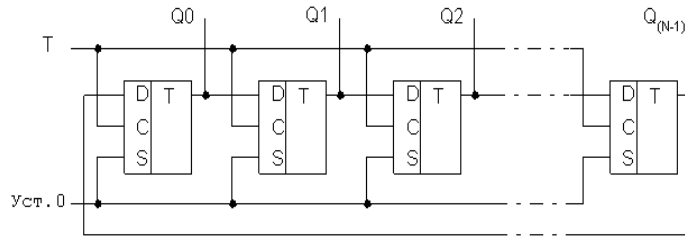


Рис. 4.22. Кільцевий лічильник на регістрі зсуву

Недолік такого лічильника - велике число тригерів, необхідних для його побудови. Більш економічні, а тому і більш розповсюджені лічильники, які побудовані на лічильних Т-тригерах. Після кожного тактового імпульсу Т сигнал на вході D змінюється на протилежний і тому частота вихідних імпульсів вдвічі менша частоти імпульсів, що надходять. Зібравши послідовний ланцюжок з n лічильних тригерів (з'єднуючи вихід попереднього тригера із входом С наступного), ми отримаємо частоту $f_{вих} = f_{вх} / 2^n$.

При цьому кожний вхідний імпульс змінює код числа на виході лічильника на 1 в інтервалі від 0 до $N = 2^n - 1$.

4.4 Лічильники

4.4.1. Призначення, класифікація та характеристики лічильників

Лічильник призначений для підрахунку кількості одиниць інформації (рахункових імпульсів). Надходження одиниці інформації полягає у впливі на вхід лічильника перепаду напруги 0-1 при прямому вході або 1-0 при інверсному вході. Тому що один імпульс містить той і інший перепад, те його й ототожнюють із одиницею інформації не завищимо от тип входу лічильника.

Лічильник має K_n стійких станів, кожне з яких повторюється після підрахунку K_n рахункових імпульсів. Інакше кажучи, рахунок імпульсів здійснюється з коефіцієнтом (модулем) перерахування K_n .

- По способі кодування числової інформації розрізняють лічильники
- с позиційним (одиничним, двійковим, десятковим і т.д.)
 - непозиційним (наприклад, у кодах Грея) кодуванням.

У лічильниках з позиційним кодуванням числове вираження поточного стану визначається формулою

$$A = \sum_{i=0}^{n-1} Q_i M_i$$

де n — кількість розрядів; Q_i — логічне значення розряду ($Q_i=0,1$);

M_i -вага i -го розряду.

У лічильниках з непозиційним кодуванням розряди не мають постійних ваг, і числове вираження стану пропонується кожному набору значень Q_i .

В основу побудови двійкових лічильників покладені n рахункових тригерів. Кожному з них ставиться у відповідність одна певна вага з набору:

$$2^0, 2^1, \dots, 2^{n-1}.$$

Кількість імпульсів, що надійшли, представляється у вигляді суми:

$$A = Q_{n-1}2^{n-1} + Q_{n-2}2^{n-2} + \dots + Q_12^1 + Q_02^0$$

Максимальний коефіцієнт перерахування двійкового лічильника $K_n = 2^n$.

Лічильники підрозділяються ще по других класифікаційних ознаках. По призначенню розрізняють підсумовуючі, що віднімають та реверсивні; по способі запуску - асинхронні та синхронні; по способі організації переносу - з послідовним, наскрізним і паралельним переносами.

Основними параметрами лічильників є розв'язна здатність, час установлення і ємність.

Розв'язна здатність — це мінімальний період проходження рахункових імпульсів $T_{см}$, при яких зберігається працездатність лічильника. Зворотна величина $F_c = 1/T_{см}$ характеризує максимальну частоту рахунку.

Час встановлення $T_{вст}$ — це інтервал між початком подачі на вхід рахункового імпульсу й моментом закінчення самого тривалого перехідного процесу в лічильнику. Параметри $T_{см}$ і $T_{вст}$ характеризують швидкодія лічильника.

Ємність лічильника визначається максимальним числом імпульсів, що геть може зареєструвати. Чисельно ємність дорівнює коефіцієнту перерахування K_n .

4.4.2. Лічильники з послідовним переносом

Підсумовуючі лічильники. Підсумовуючий лічильник повинен функціонувати так, щоб при надходженні на його вхід одного імпульсу записане в ньому число збільшилося на одиницю. Принцип побудови підсумовуючого лічильника заснований на правилу додавання до двійкового числа одиниці. Відповідно до цього правила, наприклад, трьохрозрядний лічильник повинен послідовно приймати стани 000, 001, 010, 011, ..., 111.

Видно, що тригер молодшого розряду перемикається кожним рахунковим імпульсом, тобто входом лічильника служить вхід цього тригера. Стан другого та третього тригерів міняються кожним другим та четвертим імпульсом. Це забезпечується послідовним з'єднанням тригерів. Зміна значення i -го розряду відбувається тоді, коли до додавання чергової одиниці всі попередні розряди були одиницями. Якщо тригери мають прямий рахунковий вхід, то вони підключаються до інверсного виходу попереднього тригера, якщо мають інверсний вхід, то підключаються до прямого виходу. На рис.4.23,а приведена схема трьохрозрядного підсумовуючого лічильника з тимчасовими діаграмами роботи та умовним зображенням. Лічильник може приймати 8 різних станів, які повторюються через кожні 8 вхідних імпульсів ($K_n=8$). З найбільшою частотою перемикається тригер молодшого розряду. Розв'язна здатність лічильника визначається часом затримки перемикавання тригера $T_{cm}=T_{mz}$. Це положення поширюється на всі типи двійкових лічильників.

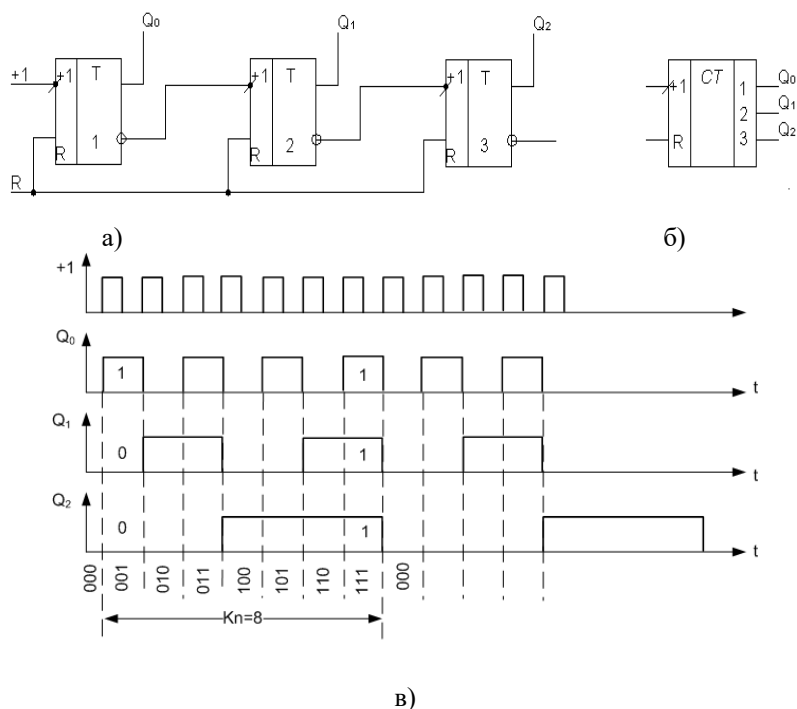


Рис.4.23. Схема (а), УГП (б) та часові діаграми (в) трьох розрядного підсумовуючого лічильника з послідовним переносом

Даний лічильник називають лічильником з послідовним переносом, тому що перемикавання тригера i -го розряду відбувається в результаті послідовного перемикавання всіх попередніх тригерів молодших розрядів, тобто інформація поширюється по ланцюжку тригерів послідовно. Час встановлення лічильника: $T_{вст} = nT_{mz} = nT_{cm}$.

Якщо необхідно знімати інформацію після кожного вхідного імпульсу, період проходження повинен бути $T > nT_{mz}$. Погіршення швидкодії з ростом розрядності це основний недолік лічильників з послідовним переносом.

Лічильники, що віднімають. При подачі на вхід лічильника, що віднімає, одного рахункового імпульсу раніше записане в ньому число зменшується на одиницю. Принципи побудови лічильників, що віднімають, засновані на правилах вирахування двійкових чисел і відрізняються від принципів побудови підсумовуючих лічильників лише тим, що якщо тригери мають прямий вхід +1, то їх підключають до прямого виходу попереднього тригера, якщо вхід інверсний, то підключають до інверсного виходу (рис.4.24). Тут додатковий вхід S дозволяє попередньо встановлювати всі тригери в одиничний стан.

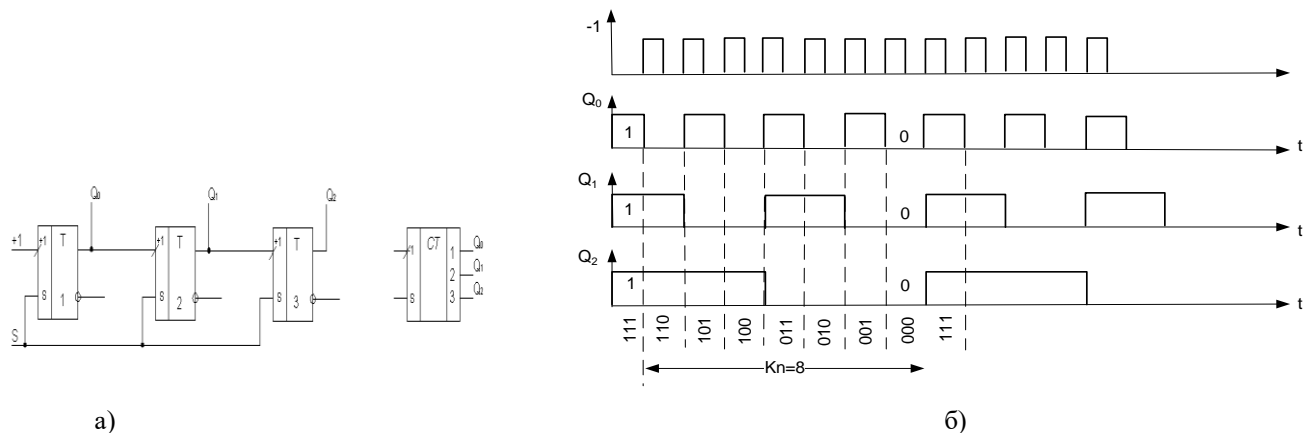


Рис.4.24. Схема та УГП (а) та часові діаграми (б) трьохрозрядного лічильника, що віднімає з послідовним переносом

Помітимо, що даний лічильник можна розглядати як підсумовуючий, а підсумовуючий (рис.4.11) — як, що віднімає при інвертуванні вихідних сигналів Q_i або зніманні інформації з виходів $\overline{Q_i}$.

Реверсивні лічильники. Вони працюють як у режимі підсумовування, так і вирахування. Перемикання режимів здійснюється комутацією облікових входів всіх тригерів (крім тригера молодшого розряду) до інверсних або прямих виходів попередніх тригерів (рис.4.25). Перемикання режиму виконує сигнал дозволу E . При $E=1$ прямих вхід i -го тригера комутується до прямого виходу $(i-1)$ -го тригера, що відповідає режиму вирахування. При $E=0$ вхід підключається до інверсного виходу, і лічильник стає підсумовуючим.

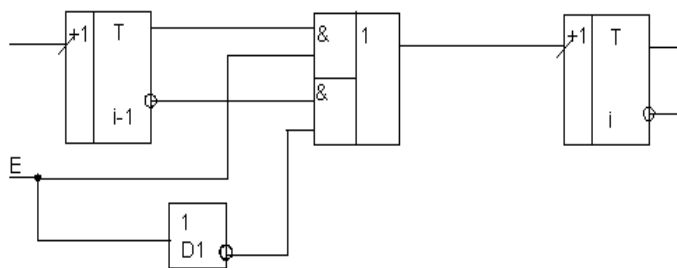


Рис.4.25. Міжтригерні зв'язки реверсивного лічильника

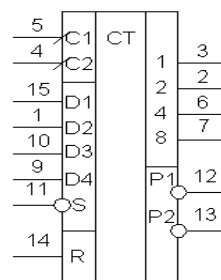


Рис. 4.26. Реверсивний лічильник

Універсальність реверсивного лічильника досягається ціною введення додатково $n-1$ логічних елементів та погіршення швидкодії:

$$T_{вст} = nT_{m2} + (n-1)t_{зм.сп}$$

де $t_{зм.сп}$ — середній час затримки перемикання ЛЕ І-АБО.

Наприклад, мікросхеми К555ІЕ6 і К555ІЕ7 це реверсивні лічильники з попереднім записом. Перший із них - двійково-десятковий, другий - чотирьохрозрядний двійковий (рис.4.14). Установка їх в 0 відбувається при високому рівні на вході R . В лічильник можна записати число, двійковий код якого поданий на входи D_1-D_4 (в К555ІЕ6 від 0 до 9, в К555ІЕ7 від 0 до 15). Для цього на вхід S необхідно подати низький рівень (на входах $C1$ і $C2$ - високий рівень, на вході R - низький). Рахування почнеться із записаного числа по імпульсах низького рівня, що подаються на вхід $C1$ (в режимі додавання) або $C2$ (в режимі віднімання). інформація на виході змінюється по фронту лічильного імпульсу. При цьому на другому лічильному вході і вході S повинен бути високий рівень, на вході R - низький, а стан виходів D байдужий. Водночас з кожним десятим (шістнадцятим) на вході $C1$ імпульсом на виході $P1$ з'являється вихідний імпульс, який його повторює, що може подаватися на вхід наступного лічильника. В режимі віднімання водночас з кожним імпульсом на вході $C2$, що переводить лічильник в стан 9 (15), на виході $P2$ з'являється вихідний імпульс.

Часова діаграма роботи лічильника наведена на рис. 4.27. На діаграмі в режимі паралельного запису ($S=0$) було записане число 6 (високий рівень на входах D_2 і D_3).

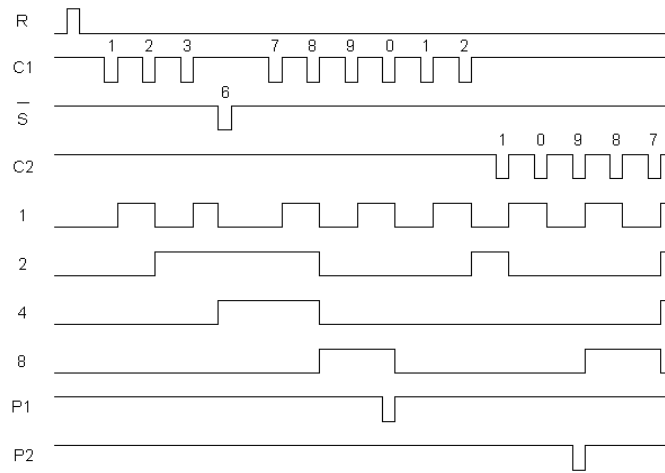


Рис. 4.27. Часова діаграма роботи лічильників К555ІЕ6, 555ІЕ7

4.4.3. Лічильники з наскрізним переносом

Як відомо, тригер i -го розряду перемикається, якщо $(i-1)$ тригер до моменту надходження на його вхід чергового рахункового сигналу T_{i-1} перебував в одиничному стані, тобто має місце $Q_{i-1}T_{i-1}=I$.

Тому з метою прискорення переносу можна вхідний сигнал T_{i-1} пропустити на вхід i -го тригера за допомогою елемента I , минаючи $(i-1)$ тригер (рис.4.28,а). Тригери можуть бути асинхронними та синхронними. Відповідно до цього лічильники виходять асинхронними або синхронними.

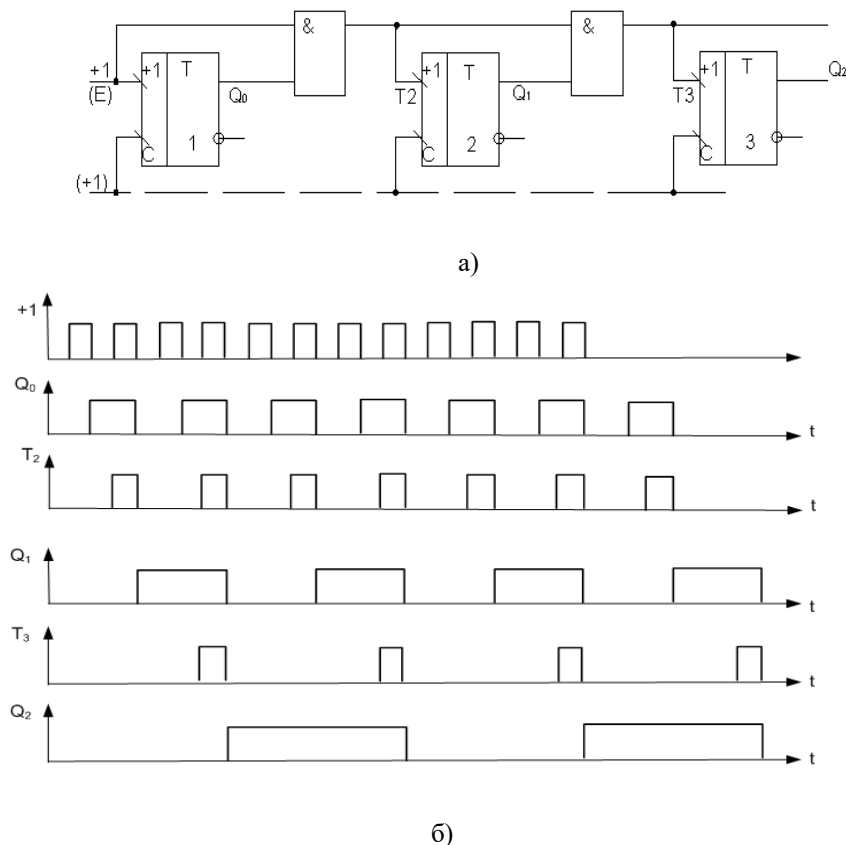


Рис.4.28. Схема (а) та часові діаграми (б) трьох розрядного лічильника з наскрізним переносом

З часових діаграм для асинхронного лічильника, представлених на рис.4.28,б, видно, що завдяки схемі переносу на входи другого, третього і т.д. тригерів транзитом передаються відповідно кожний другий, четвертий і т.д. вхідні імпульси. При цьому проходження імпульсів на вхід останнього тригера затримується на час переносу: $T_{пер} = (n-1)t_{зм.сп}$

Час установлення лічильника: $T_{вст} = T_{мз} + (n-1)t_{зм.сп}$.

Беручи до уваги, що час затримки проходження сигналу через ЛЕ I менше, ніж через тригер, виграш у швидкодії в лічильників з наскрізним переносом у порівнянні з лічильниками з послідовним переносом очевидний, однак тут більше обсяг устаткування.

У синхронного лічильника з наскрізним переносом рахунковим входом є об'єднаний вхід синхронізації C всіх тригерів, завдяки чому вони перемикаються одночасно. Інформаційний вхід першого тригера стає входом дозволу режиму рахунку E . При $E=0$ всі сигнали $T=0$, і лічильник перебуває в режимі зберігання. При $E=1$ устатовлюється режим рахунку.

Час переносу та час устанавлення такі ж, як і у синхронного лічильника. Однак оскільки в асинхронному лічильнику протягом усього часу $T_{уст}$ іде безперервне (із затримкою $t_{зд.сп}$) перемикання тригерів, то для знімання інформації необхідний додатковий час, тобто збільшення періоду проходження вхідних імпульсів.

В синхронному лічильнику тригери перемикаються одночасно (по рахунковому імпульсі) і лише потім відбувається перенос у ціпи логічних елементів, тому, для знімання інформації додаткового часу не потрібно — можна використати час $T_{пер}$. Швидкодія синхронного лічильника вище, ніж асинхронного.

4.4.4. Лічильники з паралельним переносом

Для одержання паралельного переносу використовується та обставина, що перемикання тригера i -го розряду відбувається тоді, коли перед надходженням чергового імпульсу на вхід лічильника всі тригери молодших розрядів перебувають в одиничному стані, тобто: $Q_i \cdot Q_{i-1} = 1$. Тому сигнал на інформаційний вхід кожного тригера можна виробляти шляхом кон'юнкції вихідних сигналів попередніх тригерів. На рис.4.29 показана схема та часові діаграми лічильника з паралельним переносом синхронного типу, що використовує двоступінчасті JK-тригери.

Час встановлення лічильника $T_{вст} = T_{тр} + t_{зд.сп}$ т. е. практично визначається його розв'язною здатністю. Маючи найвищу швидкодію, має один істотний недолік - нерегулярність схем переносу. З ростом ваги розряду число входів ЛЕ збільшується. Тому при великій розрядності прибігають до комбінованого переносу - паралельному усередині груп тригерів і послідовному або наскрізному між групами.

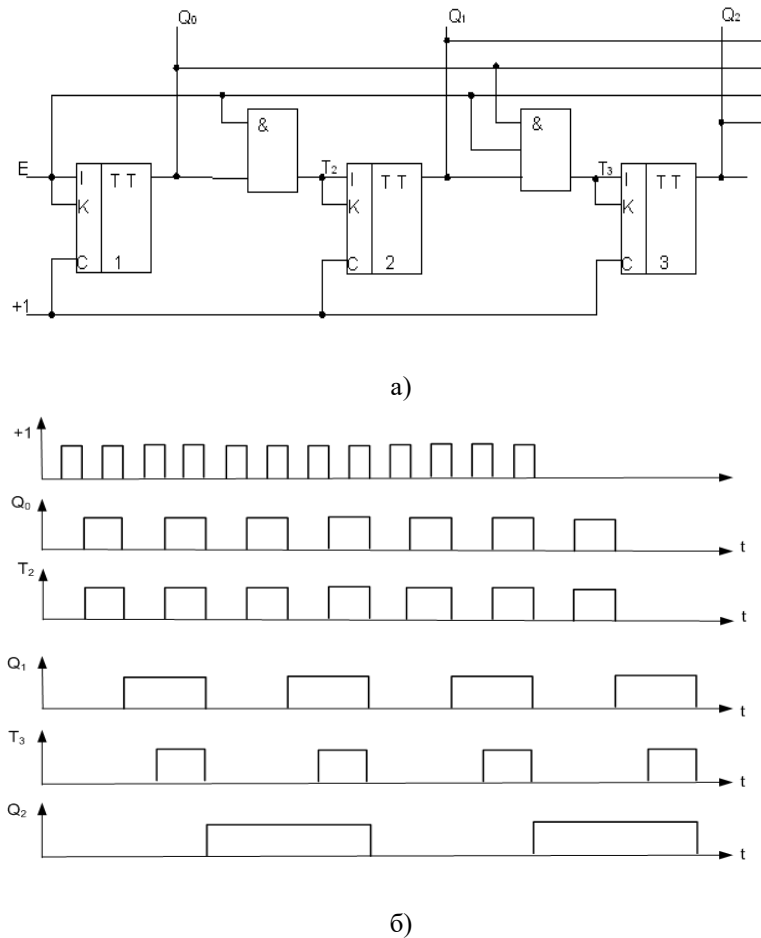


Рис.4.29 Схема (а) та часові діаграми (б) трьох розрядного лічильника з паралельним переносом

У випадку побудови лічильника на JK-тригерах, що мають кілька інформаційних входів, об'єднаних кон'юнкцією, структура його спрощується — виключаються зовнішні елементи I (рис.4.30)

Крім того, відмова от входу дозволу E дозволяє збільшити розрядність лічильника на одиницю без збільшення числа інформаційних входів тригерів. З метою збільшення розрядності лічильники можна з'єднувати послідовно. Для цього використовується сигнал переносу $CR > 15$, що для наступного лічильника є рахунковим. Виробляється він за допомогою стробіруемого входним імпульсом дешифратора старшого (п'ятнадцятого) стану. Позначення « >15 » говорить у тім, що сигнал переносу з'являється в проміжку після 15-закінчення га рахункового імпульсу до надходження чергового - нульового.

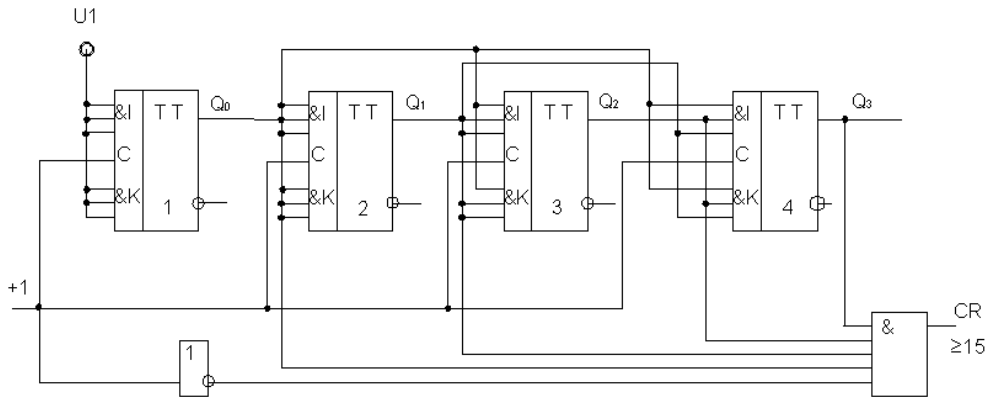


Рис.4.30. Чотирьох розрядний лічильник с пристроєм формування сигналу переносу

Лічильники з паралельним переносом, виконувані у вигляді окремих ІС, звичайно мають додаткові входи для попереднього запису інформації.

4.4.5. Лічильники з коефіцієнтом перерахування, відмінним от 2^n

Велике поширення мають лічильники та подільники із $K_n \neq 2^n$. Так, у цифрових індикаторних пристроях домінують двоїчно-десяткові лічильники із $K_n = 10$. Принцип побудови лічильників із $K_n \neq 2^n$ сводиться до наступного. Беруть таке число n тригерів, щоб виконувалася умова

$$2^{n-1} < K_n < 2^n.$$

Потім схемним шляхом виключають $2^n - K_n$ надлишкових станів. Найчастіше виключають старші стани, рідше - молодші або проміжні. Робиться це за допомогою дешифратора певного стану, що своїм вихідним сигналом примусово встановлює лічильник у вихідний стан, або за допомогою зворотних зв'язків між тригерами.

На рис. 4.31 наведений приклад двоїчно-десятькового лічильника з виключенням старших надлишкових станів. Початковий стан у нього нульове: $A_0\{0000\}$. При надходженні входних імпульсів рахунок іде як у звичайному двійковому лічильнику. Як тільки встановлюється стан $A_{10}\{1010\}$, на виході елемента І дешифратора, виробляється сигнал $V = Q_1 Q_3 = 1$ і лічильник примусово переводиться в початковий стан A_0 .

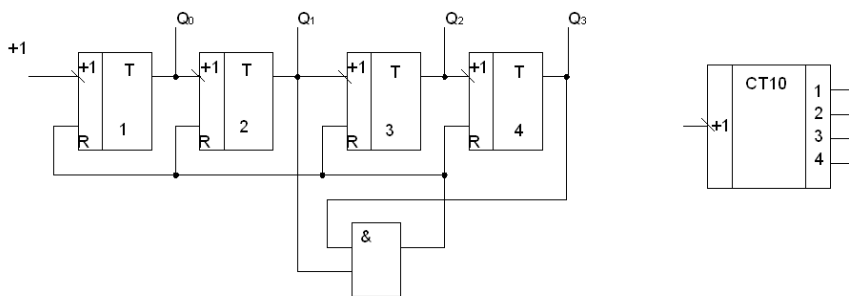


Рис.4.31. Двоїчно - десятковий лічильник з виключенням старших надлишкових станів

Прикладом використання зворотних зв'язків для виключення надлишкових станів може служити ІС 133ІЕ2 (рис.4.32,а). Вона містить T - тригер D_1 і двоїно - п'ятирічний лічильник на тригерах D_2, D_3, D_4 . Завдяки зворотному зв'язку з виходу $\overline{Q_3}$ тригера D_4 на вхід J тригера D_2 , а також зворотного зв'язка в самому тригері D_4 (з виходу $\overline{Q_3}$ на вхід K), забезпечується відповідно блокування дії п'ятого рахункового імпульсу на тригер D_2 і установка тригера D_4 п'ятим імпульсом у нульовий стан (рис.4.32,б). Таким чином, після п'ятого імпульсу виходить $A_0(000)$.

Якщо вихід Q_0 тригера D_1 з'єднати із входом $+1CT$, а рахункові імпульси подавати на вхід $+1T$, то лічильник стає двоїчно-десятьковим з $K_n = 10$.

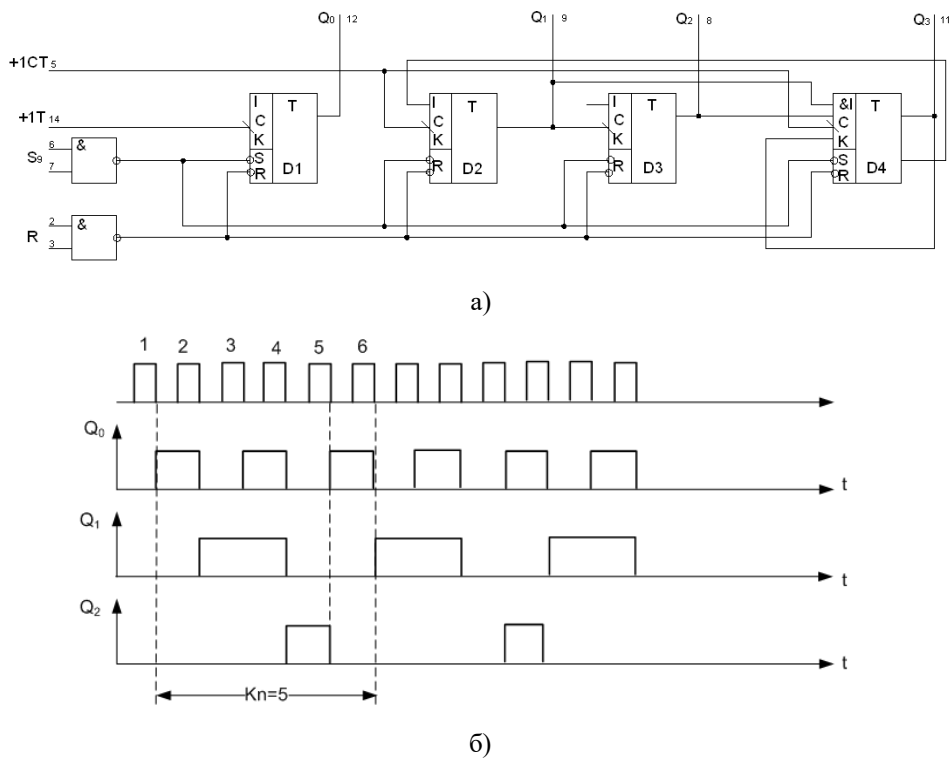


Рис.4.32 Схема (а) та часові діаграми (б) чотирьохрозрядного лічильнику з $K_n=5$

4.5. Подільники

Подільники — це лічильники, які мають один вихід, на якому з'являється імпульсний сигнал після кожних K_n рахункових імпульсів. Нерідко передбачається можливість змінювати коефіцієнт розподілу K_n за допомогою спеціального керуючого коду. У принципі дільники можна будувати й на основі двійкових лічильників, підключивши до виходів дешифратор якого-небудь одного стану. У більшості практичних випадків інтегральні перерахункові пристрої робляться комбінованими - лічильниками-дільниками. Із цією метою лічильник доповнюють дешифратором старшого стану, вихідний сигнал якого може бути використаний як сигнал переносу при нарощуванні розрядності за допомогою декількох мікросхем, або як вихідний сигнал дільника.

Принцип побудови подільників багато в чому аналогічний принципу побудови лічильників. Вони мають, як правило, один вихід, на якому за інтервал перерахування з'являється імпульс у коефіцієнт розподілу раз менше, ніж надходять на вхід ($K_d = N_{вх} / N_{вих}$). Виділяються ці імпульси за допомогою дешифратора станів.

Міняти коефіцієнт K_d у подільниках можна так само, як і в лічильниках із $K_n \neq 2^n$, тобто шляхом виключення різного числа надлишкових станів, але програмно — за допомогою зовнішніх керуючих сигналів. Такий спосіб реалізований, наприклад, в ІС 564ІЕ15. Вона дозволяє одержувати $K_d = 3 \dots 21327$ із кроком одиниця.

Знайшов поширення також спосіб, при якому коефіцієнт перерахування лічильника, що становить основу дільника, не міняється, а виробляється вибір дешифраторів, настроєних на виділення різних станів лічильника. Варто звернути увагу на те, що якщо вхідна послідовність імпульсів періодична, то вихідна послідовність буде періодичною тільки тоді, коли коефіцієнт розподілу - ціле число, у других випадках вихідні імпульси розподілені в часі нерівномірно. З метою збільшення діапазону перебудови коефіцієнта розподілу мікросхеми з'єднують послідовно. Сигнали переносу відіграють роль вхідних стосовно наступних мікросхем.

Наприклад, мікросхема К555ІЕ5 (рис.4.33) містить лічильний тригер (вхід $C1$) і подільник на вісім (вхід $C2$), створений трьома з'єднаними послідовно тригерами. Тригери спрацьовують по зрізу вхідного імпульсу (по переходу з 1 в 0). Якщо з'єднати послідовно всі чотири тригери, то одержимо лічильник за модулем $2^4 = 16$. Максимальне число, що зберігається в лічильнику при повному заповненні його одиницями дорівнює $N = 2^4 - 1 = 15 = (1111)_2$. Такий лічильник працює з коефіцієнтом рахунку K , кратним цілій степені 2, і в ньому відбувається циклічний перебір $K = 2^n$ стійких станів. Лічильник має входи примусової установки в 0.

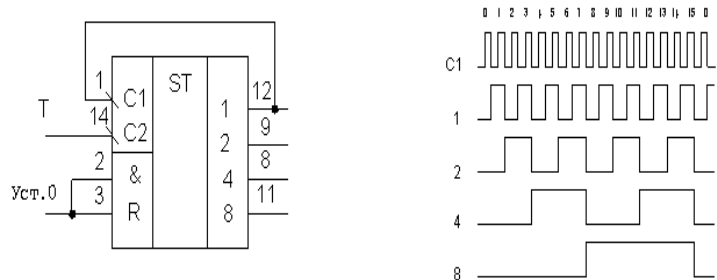


Рис. 4.33. Лічильник з коефіцієнтом перерахування 16 і його часова діаграма

Мікросхема К555ИЕ1 (рис.4.34) - подільник на 10. Установка її тригерів в 0 здійснюється одночасною подачею високого рівня на входи 1 і 2 (елемент І). Лічильні імпульси подають на вхід 8 або 9 (при цьому на іншому вході повинен бути високий рівень), або водночас на обидва входи (елементі).

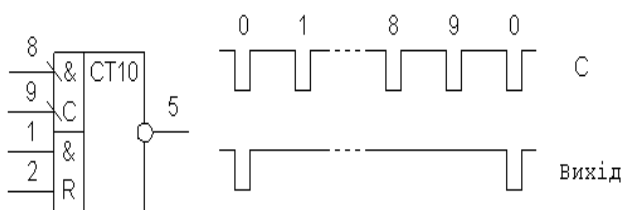


Рис. 4.34. Подільник частоти.

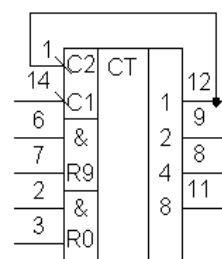


Рис. 4.35. Двійково-десятковий лічильник

У склад мікросхеми К555ИЕ2 (рис.4.35) входять тригер з лічильним входом (вхід C1) і подільник на 5 (вхід C2). При з'єднанні виходу лічильного тригера із входом C2 утвориться двійково-десятковий лічильник. Підрахунок відбувається по зрізу імпульсу. Лічильник має входи установки в 0 (R0 з логікою І) і входи установки в 9 (R9 з логікою І).

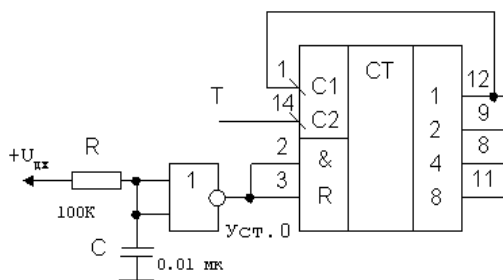


Рис. 4.36. Схема скиду лічильника в 0

Для правильної роботи цих і всіх інших лічильників, виконаних за КМОН- технологією (серії К164, К176, К564, К561), необхідно після ввімкнення живлення (або після зниження напруги джерела живлення до 3 В) встановлювати їх у вихідний нульовий стан подачею імпульсу високого рівня на вхід R.

В протилежному випадку лічильники можуть працювати із випадковими Кп, імпульс скиду після ввімкнення живлення може подаватися автоматично, якщо ввести часозадаюче RC-коло та інвертор (рис.4.36).

Мікросхеми К176ИЕ3 і К176ИЕ4 (рис. 4.37) є лічильниками за модулем 6 і 10 з дешифратором, працюючим на семисегментний індикатор. Їхні корпуси однакові, тільки на місці виходу 2 і 6 (выводи 3 і 2) лічильника К176ИЕ3 у десятичного лічильника К176ИЕ4 вихід 4 і 10. Лічильні імпульси подаються на вхід Т. Напруга на виході може бути як в прямому (при C = 0), так і в зворотному (при C = 1) коді, що дозволяє під'єднати до лічильника індикатори із спільним катодом або спільним анодом. Лічильники можна використовувати спільно з

рідиннокристалічними індикаторами. У цьому випадку на вхід С подають меандр з частотою $f > 50$ Гц. При послідовному з'єднанні лічильників сигнал знімається з виходу 6 (К176ИЕ3) або 10 (К176ИЕ4).

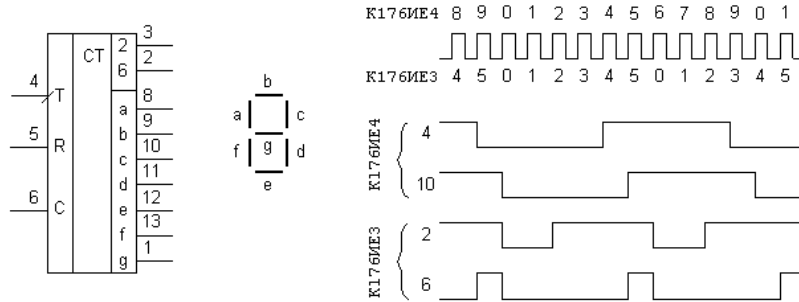


Рис. 4.37. Лічильник за модулем 6 (10) з дешифратором та часові діаграми його роботи

4.6. Генератори і формувачі імпульсів

На базі логічних елементів цифрових приладів можуть бути сконструйовані різноманітні генератори імпульсів. Генератор (рис. 4.38) виробляє імпульси в широкому діапазоні частот - від одиниць герц до декількох кілогерц. Залежність частоти f (кГц) від ємності конденсатора $C1$ (пФ) виражається приблизною формулою

$$f \approx 3 \cdot 10^5 / C1.$$

Скважність імпульсної напруги практично рівна 2. При зниженні напруги джерела живлення на 0.5 В частота генерованих імпульсів зменшується на 20%.

В генераторі, що наведений на рис. 4.39, тривалість імпульсів можна регулювати змінним резистором $R2$ (скважність змінюється від 1.5 до 3), а частоту - резистором $R1$. Наприклад, в генераторі з $C1 = 0.1$ мкФ при вилученні резистора $R2$ тільки резистором $R1$ частоту генерованих імпульсів можна змінювати від 8 до 125 кГц. Для одержання іншого діапазону частот необхідно змінити ємність конденсатора $C1$.

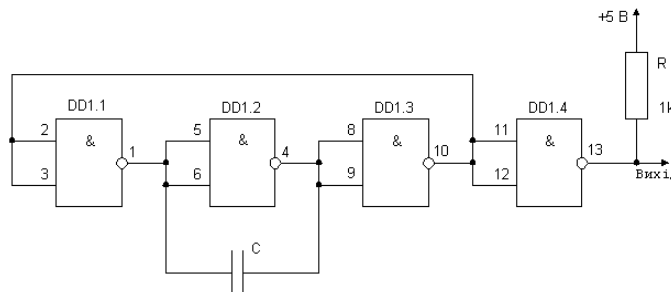


Рис. 4.38. Генератор імпульсів

На рис. 4.40 зображені схема і часова діаграма роботи формувача імпульсів по фронту і зрізу вхідного сигналу. Тривалість кожного імпульсу, що сформувався, рівна $t_{i1} = t_{i2} = n \cdot t^{1,0} + (n + 1) \cdot t^{0,1}$,

де n - парне число елементів, що беруть участь в затримці сигналів; $t^{0,1}, t^{1,0}$ - час затримки роботи елементів при переході від 0 до 1 та від 1 до 0 відповідно.

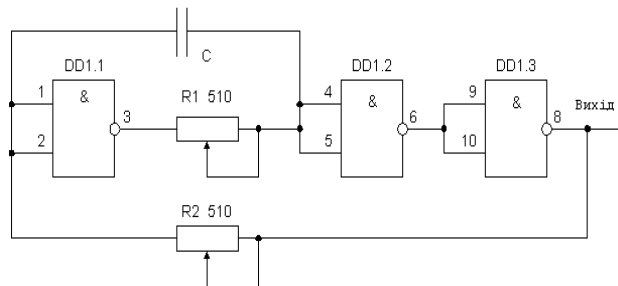


Рис. 4.39. Генератор імпульсів з регульованою тривалістю

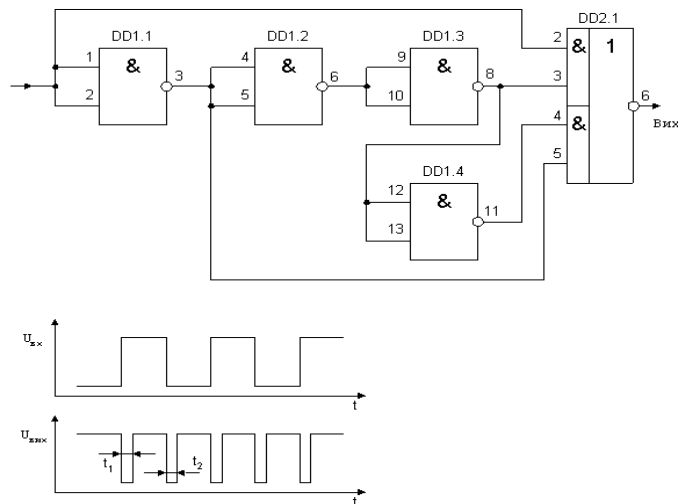


Рис. 4.40. Формувач імпульсів по фронту і зрізу вхідного сигналу

Принцип роботи цього формувача аналогічний принципу роботи описаних раніше формувачів коротких імпульсів.

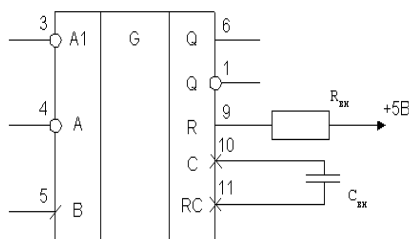


Рис. 4.41. Формувач імпульсів на мікросхемі K555AG1

Формувачі імпульсів є і в складі мікросхем серії K555. Так, мікросхема K555AG1 - це одновібратор з трьома входами, прямим та інверсним виходом і виводами для під'єднання зовнішніх трьох задаючих кіл (рис. 4.41). Одновібратор може запускатися як позитивним, так і негативним перепадами вхідних сигналів при певній напрузі, не залежними від тривалості вхідних імпульсів. Перемикається одновібратор негативним перепадом вхідного сигналу, поданого на один з входів А, в той час як на вхід В подана напруга високого рівня, або позитивним перепадом, поданим на вхід В, якщо на одному з входів А або А1 присутня напруга низького рівня.

При максимальному опорі резистора $R_{\text{вн}} = 40$ кОм тривалість вихідного імпульсу не повинна перевищувати $0.9T$, де T - період слідування вхідних імпульсів.

Тривалість вихідного імпульсу залежить від резистора $R_{\text{вн}} = (0-40)$ кОм і $C_{\text{вн}} = (0...1000)$ мкФ і визначається формулою: $t_n = RC \ln 2$.

Тут $R = 2k + R_{\text{вн}}$, $2k$ - опір внутрішнього резистора.

В склад серії K555 входить також мікросхема K555AG3. В одному корпусі в ній містяться два одновібратори. Варіанти підключення зовнішніх часозадаючих елементів і часова діаграма роботи одновібратора зображені на рис. 4.42. Одновібратор також запускається або негативним перепадом вхідного сигналу на вході А при високому рівні на входах В і R, або позитивним перепадом напруги на вході В при низькому рівні на вході А і високому рівні на вході R. Тривалість імпульсу t_{i1} визначається постійною часу часозадаючого кола, але може бути зменшена за рахунок подачі на вхід R напруги низького рівня при $t_{i2} < t_{i1}$.

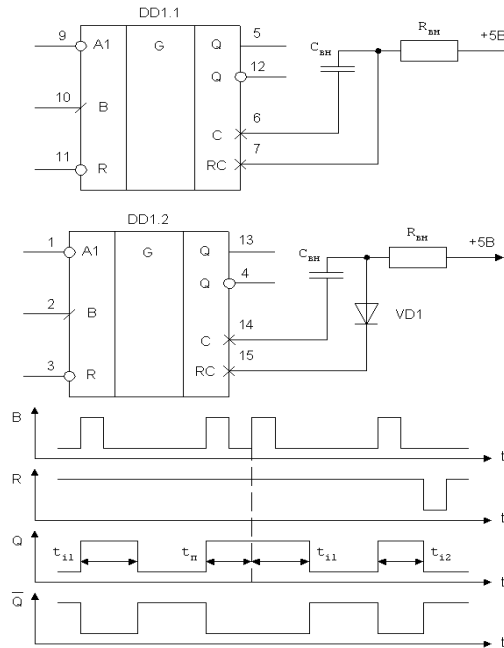


Рис. 4.42. Варіанти формувача імпульсів на мікросхемі К555АГ3

Список питань для самоконтролю

1. Що таке регістр і як реалізуються паралельний і послідовний регістри?
2. Що таке лічильник і як він реалізується?
3. Як реалізуються різні коефіцієнти перерахування в лічильниках?
4. Що таке двійково-десятковий лічильник?
5. Як реалізується реверсивний лічильник?
6. Як реалізується схема скиду лічильника в нуль?
7. Принцип побудови лічильників із $K_n \neq 2^n$
8. Який принцип побудови подільників?
9. От чого залежить тривалість вихідного імпульсу генератора?
10. Яка залежність частоти f імпульсів генератора від ємності конденсатора?
11. На вхід п'яти розрядного лічильника з природним порядком рахунка і коефіцієнтом рахунка $K_{сч} = 25$ надійшло 37 імпульсів. Яке число буде на виході лічильника?
12. Яким способом переноситься код у регістр зсуву із двоичного лічильника?
13. Скільки імпульсів необхідно подати на тактовий вхід регістра зсуву для того, щоб послідовно записати в нього число 101110 якщо число розрядів регістра відповідає розрядності записуваних чисел?
14. Скільки тактових імпульсів необхідно подати на вхід регістру пам'яті для запису в нього числа 101110?
15. У якому з лічильників використовується менше число тригерів — у двоичном або в двоїчно-десятковому?

РОЗДІЛ 5. ЗАПАМ'ЯТОВУВАЛЬНІ ПРИСТРОЇ

5.1. Типи запам'ятовувальних пристроїв

Для зберігання невеликих масивів кодових слів можуть використовуватися регістри. Але вже при необхідності зберігати десятки слів використання регістрів призводить до невиправдано великих апаратних витрат. Для зберігання великих об'ємів слів будують *запам'ятовувальні пристрої* (ЗП) з використанням спеціальних мікросхем, в кожній з яких може зберігатися інформація великого об'єму. Різноманіття напівпровідникових ЗУ обумовило їхню класифікацію по ряду ознак: функціональному призначенню, способу зберігання інформації, способу доступу (адресації) до комірок пам'яті, технології виготовлення й ін.

За функціями, що виконуються, розрізняють такі типи ЗП:

- оперативний запам'ятовувальний пристрій (ОЗП);
- постійний запам'ятовувальний пристрій (ПЗП);
- перепрограмований постійний запам'ятовувальний пристрій (ППЗП).

Оперативні ЗП забезпечують не тільки зберігання й видачу інформації, але й швидке (оперативне) відновлення її. Використовуються вони в ЕОМ або інших обчислювачах для тимчасового зберігання програм, вихідних даних і результатів обчислень. Записана в ОЗУ інформація може зберігатися тільки при підключеному до нього джерелі живлення. При відключенні живлення інформація губиться.

Завдяки цьому, в ОЗП передбачаються три режими роботи:

- режим зберігання при відсутності звертання до ЗП;
- режим читання слів, що зберігаються;
- режим запису нових слів.

ПЗП призначений для зберігання деякої інформації, яка не руйнується при відмиканні джерела живлення.

В ПЗП передбачається два режими роботи:

- режим зберігання;
- режим читання з високою швидкістю.

Постійні ЗУ можуть тільки зчитувати заздалегідь записану інформацію. Запис інформації в них здійснюється шляхом зміни фізичних властивостей ділянок кристала, займаного коміркою пам'яті, або зміною провідності електричних кіл. Тому відключення живлення не позначається на змісті, записаному інформації. Використовуються ПЗУ для зберігання довго тимчасової інформації й підпрограм в ЕОМ, програм для спеціальних обчислювачів, що управляють і індикаторних систем, а також у якості різних функціональних пристроїв: дешифраторів, суматорів, перетворювачів кодів і ін. Наприклад, якщо в групу комірок пам'яті, адреса якої визначається двійковим числом, записати двійково-десятковий еквівалент, то ПЗУ буде функціонувати як перетворювач: при надходженні на адресні входи ПЗУ двійкового числа на виходах його з'явиться це ж число у двійково-десятковому коді.

По способі доступу до комірок пам'яті всі ЗУ діляться на адресні й асоціативні.

В адресних ЗУ звертання до комірок пам'яті виробляється по їхніх фізичних координатах, що задається зовнішнім двійковим кодом - адресою. Адресні ЗУ бувають із довільним доступом, які допускають будь-який порядок проходження адрес, і з послідовним доступом, де вибірка комірок пам'яті можлива тільки в порядку зростання або убуття адрес. Останні функціонально схожі на регістри, що зрушують.

В асоціативних ЗУ пошук інформації виробляється за ознаками, укладеним у збереженій інформації.

По способі зберігання інформації ЗУ бувають статичними й динамічними. Статичні ЗУ ґрунтуються на комірках пам'яті, які при включеному харчуванні можуть перебувати в одному із двох стійких станів як завгодно довго. Прикладом такого осередку є тригер.

У динамічних ЗУ комірками пам'яті служать конденсатори. Після введення в конденсатор заряду, що еквівалентно запису біта даних, відбувається поступове мимовільне розподілення його струмами витоку. Тому така пам'ять має потребу в періодичному відновленні (регенерації) її змісту, що вимагає спеціальних зовнішніх регенераторів і ускладнює організацію доступу. Однак у зв'язку з малими розмірами й високою економічністю комірок пам'яті динамічні ЗУ вигідно відрізняються від статичних ЗУ при великій (десятки тисяч біт і більше) ємності пам'яті. У сучасних ЗУ регенерація сполучається зі зверненням до осередку або групи комірок пам'яті.

Розробляються також ЗУ з динамічними осередками й убудованою системою регенерації - квазістатичні ЗУ. Зовнішні сигнали керування в цьому випадку такі ж, як у повністю статичних ЗУ.

Запам'ятовувальні пристрої характеризуються більшим переліком параметрів. Основними з них є *інформаційна ємність*, час вибірки й *час запису*.

Інформаційна ємність - це максимальне число біт даних, які можуть зберігатися в ЗУ. Визначається числом комірок пам'яті в мікросхемі. Інформаційна ємність часто представляється у вигляді добутку: "число слів" \times "розрядність слів", наприклад 64 \times 4, 1024 \times 1.

Час вибірки - інтервал часу між моментом подачі сигналу вибірки й появою інформації на виході ЗУ за умови, що всі інші необхідні сигнали подані.

Час запису - мінімальний час збігу керуючих сигналів на входах мікросхеми, що забезпечує запис даних.

ППЗП в процесі функціонування цифрового пристрою використовується як ПЗП. Вони відрізняються від ПЗП тим, що дозволяють поновлювати записану інформацію, тобто в ньому передбачається режим запису. Але на відміну від ОЗП, запис інформації потребує відключення ППЗП від цифрового пристрою, і виконується за допомогою

спеціальних призначених для запису пристроїв (програматорів). Перепрограмовані ПЗП дорожчі за ПЗП, і їх використовують в процесі відлагодження розроблюваного пристрою, після чого їх можна замінити дешевшими ПЗП.

5.2. Основні параметри запам'ятовувальних пристроїв

ЗП вміщує деяке число N комірок, в кожній з яких може зберігатися слово з певним числом розрядів n . Комірки послідовно нумеруються двійковими числами. Номер комірки називається адресою. Якщо для подання адрес використовуються комбінації n -розрядного двійкового коду, то число комірок в ЗП може скласти $N = 2^n$.

Кількість інформації, яка може зберігатися в ЗП, визначає його *ємність*. Ємність можна виразити числом комірок N з вказанням розрядності n слів, які зберігаються в них, в формі $N \cdot n$ або її можна визначити добутком $M = N \times n$ біт. Часто розрядність комірок вибирають кратною байту (1 байт дорівнює 8 бітам). Тоді і ємність зручно подавати в байтах. Великі значення ємності виражаються в одиницях $K = 2^{10} = 1024$. Наприклад, $M = 64$ Кбайт означає ємність, що дорівнює $M = 64 \times 1024$ байт, $M = 64 \times 1024 \times 8$ біт.

Швидкодія ЗП характеризується двома параметрами:

- часом вибірки t_B , який є інтервалом часу між моментом подачі сигналу вибірки при запису і моментом вибірки;
- моментом, коли допустиме наступне звертання до пам'яті.

ЗП будуються із набору однотипних мікросхем ЗП з певним їх з'єднанням. Кожна мікросхема ЗП, крім часу звертання і ємності, характеризується споживаною потужністю, набором напруг живлення, типом корпусу (число виводів). Мікросхеми ППЗП додатково характеризуються часом зберігання записаної в них інформації (по закінченні якого інформація, що зберігається в них, може самовільно змінитися), допустимою кількістю циклів перезапису (після чого мікросхема вважається непридатною для використання).

Перелік і основні характеристики різних типів ЗП вітчизняного виробництва наведені в таблиці 5.1 - ОЗП, таблиця 5.2 - ПЗП з однократним електричним програмуванням, таблиця 5.3 - ППЗП.

Таблиця 5.1 - Основні характеристики оперативних запам'ятовувальних пристроїв

Мікросхема	Інформаційна ємність, біт (організація, слово×розряд)	Час вибірки, нс	Напруга джерела живлення, В	Споживана потужність, мВт	Число виводів
K155PY5	256 (256×1)	60	5	700	16
KP188PY2A	256 (256×1)	500	5	$P_{ст}=0.05$, $P_{дин}=10$	16
K500PY410	256 (256×1)	25	-5.2	750	16
K500PY415	1024 (1024×1)	30	-5.2	730	16
KP565PY2A	1024 (1024×1)	450	5	300	16
KP537PY2A	4096 (4096×1)	300	5	50	18
KP541PY1A	4096 (4096×1)	120	5	450	18
KP565PY1A	4096 (4096×1)	200	12; 5; -5	3; 0.25; 0.125	22
KP541PY31	8192 (8192×1)	150	5	550	20
KP541PY3	16384 (16384×1)	150	5	550	20
P581PY4	16384 (16384×1)	200	12; 5; -5	500; 0.05; 2	22

Таблиця 5.2 - Основні характеристики постійних запам'ятовувальних пристроїв

Мікросхема	Інформаційна ємність, біт (організація, слово×розряд)	Час вибірки, нс	Напруга джерела живлення, В	Споживана потужність, мВт	Число виводів
K155PE3	256 (256×1)	50	5	550	16
KP500PE149	1024 (256×4)	35	-5.2	730	16
K541PT1	1024 (256×4)	80	5	400	16
K556PT4	1024 (256×4)	70	5	650	16
KP556PT5	4096 (512×8)	70	5	950	24
KP565PT1	4096 (1024×4)	300	-12; 5; -5	$P_{ст}=3; 10; 0.5$ $P_{дин}=130; 1; 95$	22

Таблиця 5.3 - Основні характеристики перепрограмованих постійних запам'ятовувальних пристроїв

Мікросхема	Інформаційна ємність, біт (організація, слово розряд)	Час виборки, нс	Час зберігання, год.	Число циклів перезапису	Напруга джерела живлення, В	Споживана потужність, мВт	Число виводів
<i>ППЗП із багатократним електричним перепрограмуванням</i>							
KP505PP4A	512 (256×2)	1200	3000	10 ⁴	-9; 5	350; 200	24
K505PP4	1024 (512×2)	1200	3000	10 ⁴	-9; 5	350; 200	24
KP558PP11	1024 (256×4)	5000	3000	10 ⁴	-12; 5	120; 50	24
KP558PP1	1024 (256×4)	5000	3000	10 ⁴	-12; 5	120; 50	24
<i>ППЗП із ультрафіолетовим стиранням та електричним записом</i>							
K573PФ1	8192 (1024×8)	450	15000	10	12; -5; 5	850; 225; 75	24
K573PФ11	4096 (512×8)	450	15000	10	12; -5; 5	850; 225; 75	24
K573PФ13	4096 (1024×4)	450	15000	10	12; -5; 5	850; 225; 75	24
K573PФ2	16384 (2048×8)	900	10000	10	5	225	24
K573PФ21	8192 (1024×8)	900	10000	10	5	225	24
K573PФ23	8192 (2048×4)	900	10000	10	5	225	24

5.3. Оперативні запам'ятовувальні пристрої

На рис. 5.1 наведена типова структура мікросхеми ОЗП. Інформація зберігається у накопичувачі. Накопичувач – це матриця, яка складена із *елементів пам'яті* (ЕП), розташованих уздовж рядків та стовпців. Елемент пам'яті може зберігати 1 біт інформації (логічна 1 або логічний 0). Крім того, він забезпечується керуючими колами для установки елемента в будь-який із трьох режимів:

- *режим зберігання*, в якому він відмикається від входу і виходу мікросхеми;
- *режим читання*, в якому вміщувана в ЕП інформація видається на вхід мікросхеми;
- *режим запису*, в якому в ЕП записується нова інформація, що поступає із входу мікросхеми.

Кожному ЕП присвоєно номер, який називається *адресою* елемента. Для пошуку потрібного ЕП вказуються рядок і стовпець, які відповідають положенню ЕП в накопичувачі. Адреса ЕП у вигляді двійкового числа приймається по шині адреси в регістр адреси. Число розрядів адреси зв'язано з ємністю накопичувача. Число рядків і стовпців накопичувача вибираються рівними цілій степені 2. І якщо число рядків $N_{ряд} = 2^{n1}$, і число стовпців $N_{стовн} = 2^{n2}$, то загальне число ЕП (ємність накопичувача)

$$N = N_{ряд} N_{стовн} = 2^{n1+n2} = 2^n,$$

де $n = n1 + n2$ - число розрядів адреси, яка приймається у регістр адреси.

Наприклад, при ємності $N = 2^{10} = 1024$ число розрядів адреси $n = 10$; при цьому вибирається $n1 = n2 = n/2 = 5$, в цьому випадку число рядків і число стовпців накопичувача рівно $2^{n1} = 2^{n2} = 32$.

Розряди регістра адреси діляться на дві групи: одна група в $n1$ розрядів визначає двійковий номер рядка, в якій в накопичувачі розміщений ЕП, інша група в $n2$ розрядів визначає двійковий номер стовпця, в якому розміщений ЕП, що вибирається. Кожна група розрядів адреси подається на відповідний дешифратор: дешифратор рядків і дешифратор стовпців. При цьому кожний з дешифраторів створює на одному з своїх вихідних кіл рівень логічної 1 (на інших виходах дешифратора встановлюється рівень логічного 0); вибраний ЕП знаходиться під впливом рівня логічної 1 водночас по колах рядків і стовпців. При читанні вміст ЕП подається на підсилювач читання і з нього на вихідний тригер і вихід мікросхеми. Режим запису встановлюється подачею сигналу на вхід дозволу запису (ДЗ). При рівні логічного 0 на вході ДЗ відкривається підсилювач запису і біт інформації зі входу даних надходить у вибраний ЕП і запам'ятовується у ньому.

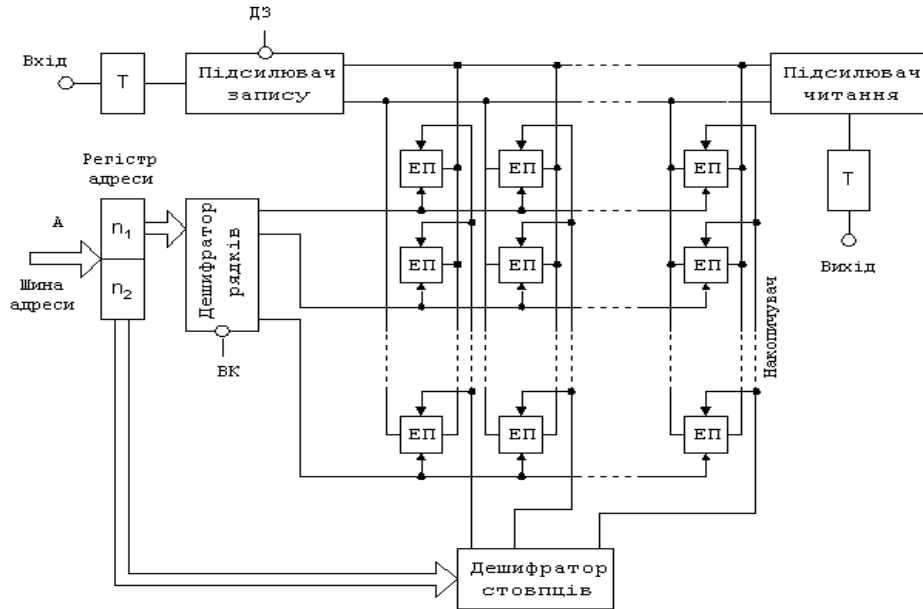


Рис. 5.1. Структура мікросхеми ОЗП.

Дані процеси відбуваються в тому випадку, якщо на вході вибору кристалу (ВК) діє активний рівень логічного 0. При рівні логічної 1 на цьому вході, на всіх виходах дешифратора встановлюється рівень логічного 0 і ЗП переходить в режим зберігання. На рис. 5.2 показано умовне графічне позначення мікросхеми ОЗП.

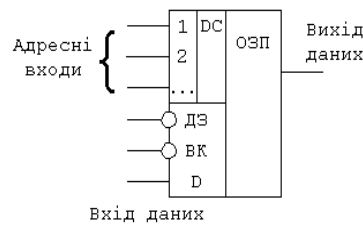


Рис. 5.2. Умове позначення мікросхеми ОЗП

Розглянемо послідовність подачі сигналів в режимах читання і запису. На рис. 5.3а зображена часова діаграма сигналів в режимі читання. З певною затримкою $t_{зам1}$ відносно моменту подачі адреси і сигналу в коло ВК (зв'язаної із процесами дешифрації адреси і ввімкнення вихідних кіл вибраного ЕП) на виході мікросхеми виникає вміст вибраного ЕП. В режимі запису (рис. 5.3б) повинні бути дотримані умови, що виключили би порушення вмісту комірок, в які не проводиться звертання.

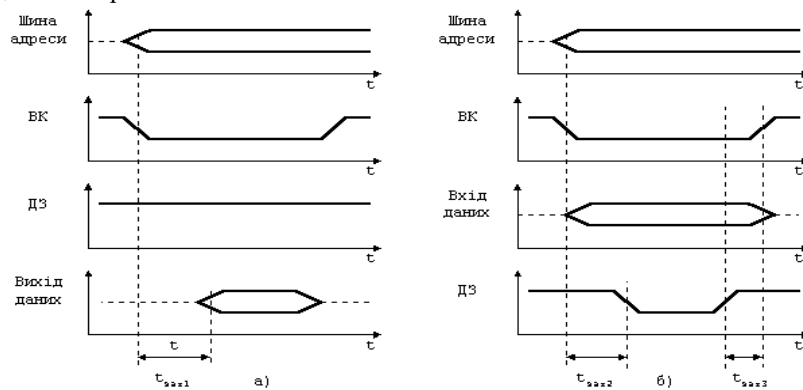


Рис. 5.3. Часові діаграми сигналів:
а) в режимі читання; б) в режимі запису

Це забезпечується тим, що сигнал в коло РЗ подається із затримкою $t_{зам2}$ відносно моменту подачі сигналів у колі адреси, ВК і вхідних даних і знімається сигнал в колі ДЗ раніше, ніж буде знятий сигнал в колі ВК. В

протилежному випадку, при передчасній подачі сигналу ДЗ, може відбутися запис в комірку з адресою, що не збігається з інформацією на адресних входах мікросхеми.

Мікросхеми ОЗП допускають нарощування ємності пам'яті шляхом нарощування розрядності (і, отже, розрядності слів, які вони зберігають) і нарощування числа комірок (і, значить, числа слів, які можна зберігати у пам'яті).

Таким чином, використовуючи відповідне число мікросхем в певному сполученні, можна побудувати пам'ять з необхідною організацією. Розглянемо схему нарощування розрядності комірок (рис. 5.4). На всі мікросхеми подається одна і та сама адреса. При читанні кожною мікросхемою видається певний розряд зчитуваного слова. При запису вхідне слово порозрядно заноситься в ЕП окремих мікросхем. Таким чином, якщо мікросхеми мають організацію $N \times 1$ (N однорозрядних комірок), то для блоку пам'яті з організацією $N \times n$ (N комірок з розрядністю кожної з них, рівною n) потрібно n мікросхем.

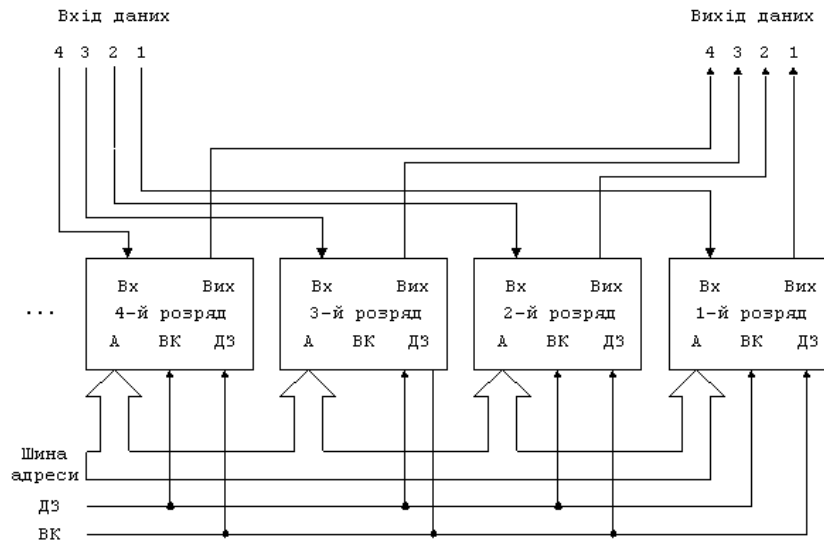


Рис. 5.4. Схема нарощування розрядності комірок ЗП

На рис. 5.5 показана схема нарощування числа і розрядності комірок. Блок пам'яті складається з мікросхем, що утворюють окремі лінії (ряди), кожна з яких будується за схемою нарощування розрядності (рис. 4.4). Розряди адреси блоку пам'яті в цьому випадку діляться на дві групи: A_1 і A_2 . Група розрядів A_2 визначає номер лінії, група розрядів A_1 - номер комірки у вибраній лінії. Вибір лінії здійснюється за допомогою дешифратора, на вхід якого подається A_2 , а кожний з виходів під'єднано до входу ВК певної лінії. Таким чином, в залежності від кодової комбінації, що міститься в A_2 , на відповідному виході дешифратора з'являється рівень логічного 0, що забезпечує вибір певної лінії мікросхем. На входи ВК інших ліній з виходу дешифратора надходить рівень логічної 1, і мікросхеми цих ліній встановлюються в режим зберігання, в якому вони не реагують на адресну групу A_1 .

Розглянемо приклад нарощування ємності блоку пам'яті. Нехай на мікросхемах з організацією 1024×1 необхідно побудувати блок пам'яті, що є організацією 4096×8 , тобто блок пам'яті на 4096 8-розрядних комірках. Нарощування розрядності вимагає в кожній лінії схеми на рис. 4.5 використати 8 мікросхем; для збільшення числа комірок з 1024 до 4096 (в 4 рази) необхідно передбачити 4 лінії мікросхем. Таким чином, загальне число мікросхем $8 \times 4 = 32$. В такому блоку пам'яті адреса для звертання формується таким чином. Для вибору лінії в адресі знадобиться дворозрядна група A_2 , кожній з чотирьох кодових комбінацій цієї групи (00, 01, 10, 11) буде відповідати певна лінія в блоку пам'яті. Вибір комірки в лінії мікросхем потребує наявності в адресі 10-розрядної групи A_1 (число комбінацій 10-розрядної групи $2^{10} = 1024$ рівно числу ЕП в мікросхемі). Таким чином, адреса розглядуваного блоку пам'яті повинна мати 12 розрядів.

В кожному стовпці матриці мікросхем на рис. 5.5 виходи всіх мікросхем об'єднуються в коло відповідного розряду виходу даних блоку, всі входи даних - в коло відповідного розряду входу даних блока пам'яті.

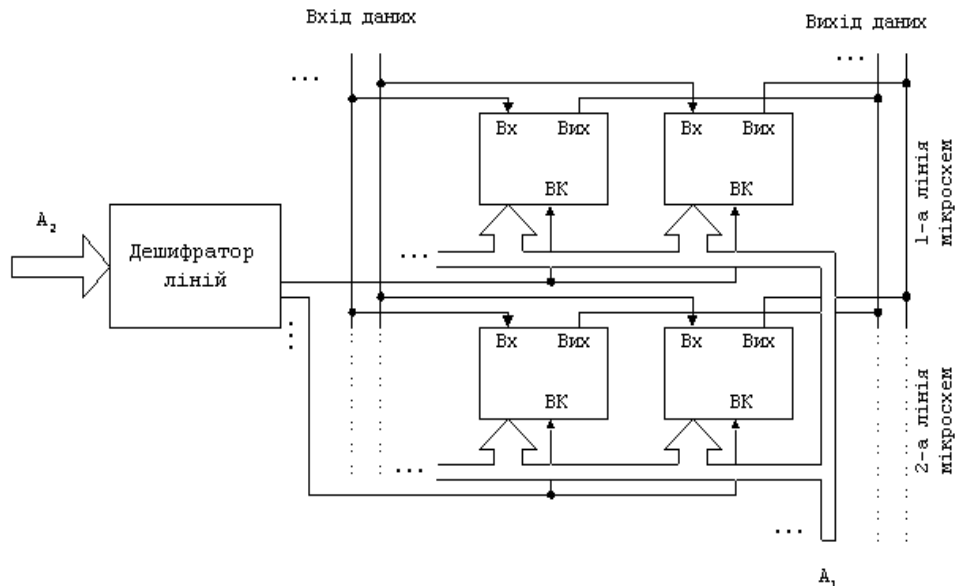


Рис. 5.5. Схема нарощування числа і розрядності комірок ЗП

5.4 Постійні запам'ятовуючі пристрої

Як і ОЗП, ПЗП складаються з комірок, звернувшись до яких, можна вивести їх вміст. Відміна від ОЗП полягає в тому, що інформація в комірки записується одноразово, після чого в процесі експлуатації використовується лише режим читання.

За способом занесення інформації ПЗП поділяють на два види: ПЗП, програмовані маскою на підприємстві-виробнику, і ПЗП, програмовані користувачем.

В перших інформація заноситься в процесі виготовлення мікросхеми за допомогою відповідного фотошаблону. Очевидно, такий спосіб запису придатний у тих випадках, коли проводиться випуск великої партії ПЗП з однією і тією ж записаною в них інформацією. Промисловість випускає такі ПЗП, наприклад, для використання в якості перетворювача двійкового коду в певні двійково-десяткові коди і інших перетворювачів. В них вхідна кодова комбінація слугує адресою комірки, а вміст комірки — вихідною кодовою комбінацією (наприклад, кодовою комбінацією двійково-десятькового коду).

В ПЗП, програмованих користувачем, запис інформації проводиться безпосередньо користувачем за допомогою спеціальних приладів, що називаються програматорами. Програматор подає в мікросхему відповідні напруги для запису інформації, що набирається на клавіатурі. Цими напругами здійснюється перепалювання плавких перемичок в елементах пам'яті. Очевидно, один раз записана в ПЗП інформація в подальшому не може бути змінена. При необхідності змінити вміст ПЗП мікросхеми з раніше записаною інформацією замінюються новими, в які записуються нові дані.

На рис. 5.6 наведена структура ПЗП, програмованого користувачем. Як і в ОЗП, матриця-накопичувач складається з елементів пам'яті (ЕП), які утворюють рядки і стовпці, але на відміну від ОЗП при зчитуванні з накопичувача видається вміст цілого рядка елементів пам'яті. Такий рядок звичайно містить декілька слів. За допомогою селектора з рядка виділяється і передається на вихід потрібне слово.

Нехай, наприклад, ПЗП має ємність $M = 2^{10}$ біт, розбитих на $N = 2^8$ слів по $2^2 = 4$ розрядів у кожному слові. Накопичувач буде містити 2^{10} елементів пам'яті, розташованих вздовж $2^5 = 32$ рядків і $2^5 = 32$ стовпців. При звертанні повинна вказуватися адреса слова; ця адреса в даному прикладі буде містити 8 розрядів, що розбиваються на дві групи розрядів A_2 і A_1 : 5-розрядну групу A_1 і 3-розрядну групу A_2 .

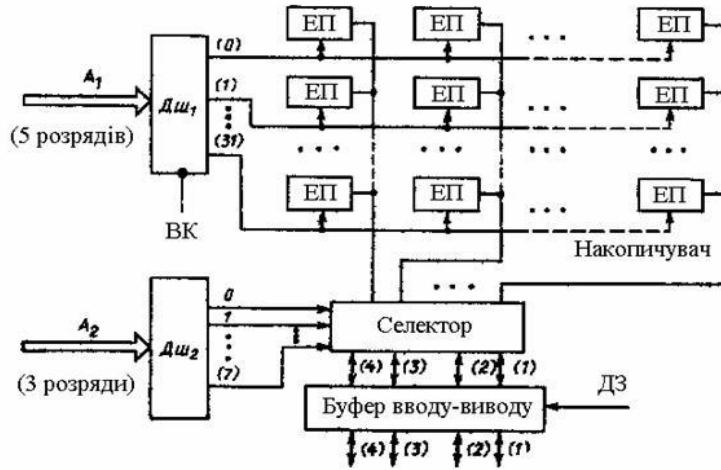


Рис. 5.6. Структура ПЗП, який програмується користувачем.

Група A_1 подається на дешифратор $Dш_1$, який вибирає одну з $2^5 = 32$ рядків накопичувача. Вміст рядка складається з 32 бітів або восьми 4-розрядних слів. Номер слова в рядку задається групою A_2 . Дешифратор $Dш_2$ перетворює цю адресну групу в сигнал на одному з восьми своїх виходів. За цим сигналом в селекторі із вмісту рядка виділяється потрібне слово, яке передається через буфер вводу-виводу на вихід мікросхеми.

На рис. 5.7 показана принципова схема накопичувача і селектора для даного прикладу. Схема побудована на біполярних транзисторах (біполярні транзистори використовуються для побудови ПЗП із високою швидкодією).

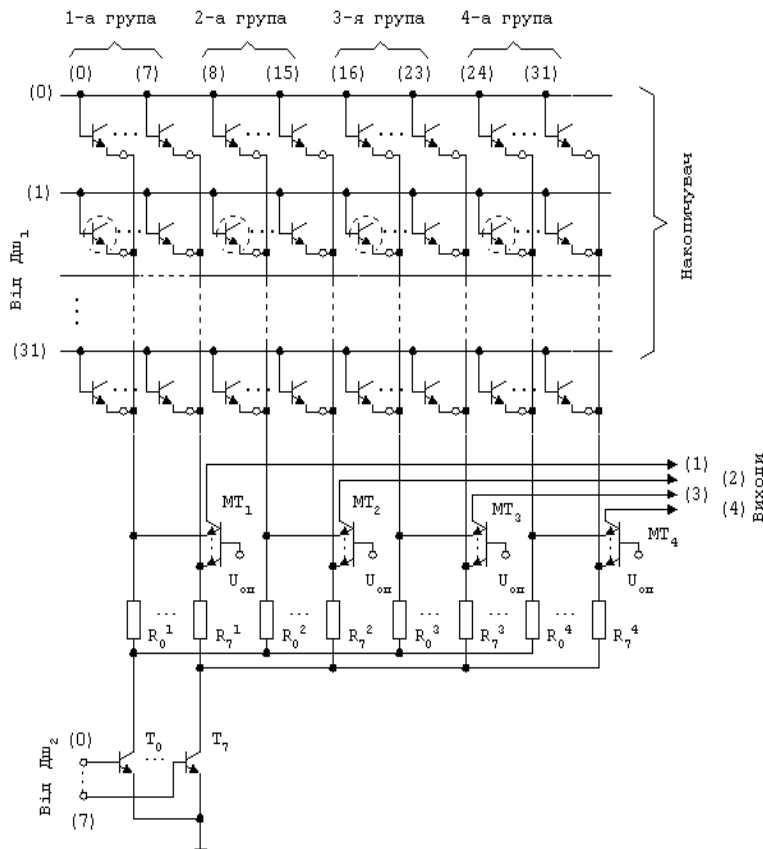


Рис. 5.7. Принципова схема накопичувача і селектора ПЗП

Накопичувач містить 2^{10} транзисторів (елементів пам'яті), що утворюють 32 рядки і 32 стовпці. Колектори транзисторів накопичувача під'єднані до джерела живлення (для спрощення рисунку кола, що з'єднують колектори транзисторів із джерелом живлення, не показані). В коло емітера кожного транзистора ввімкнена плавка перемичка (на рисунку перемички показані кружечками). Перемичка виготовляється з ніхром, полікремнію або титанату вольфраму і має опір в декілька десятків ом.

При програмуванні для перепалювання перемички достатньо через транзистор пропустити імпульс струму 20...30 мА тривалістю порядку 1 мс. При роботі в режимі читання струми в транзисторах накопичувача істотно менші і вони не можуть викликати перепалювання тих перемичок, що в процесі програмування залишені неперепаєними.

Транзистори VT0...VT7 працюють в схемі селектора, багатоємітерні транзистори MT1...MT4 — в схемі буфера вводу-виводу.

Розглянемо процеси при запису інформації. На входи дешифраторів адреси подаються адресні групи A_1 та A_2 і на одному з виходів кожного дешифратора утвориться рівень логічної 1. Нехай від дешифратора Дш₁ рівень логічної 1 надходить в рядок з номером 1, а від дешифратора Дш₂ рівень логічної 1 надходить в коло з номером 0 і в селекторі відкривається транзистор VT₀. При цьому в накопичувачі виявляться відкритими обведені штриховою лінією (рис. 5.7) транзистори (назвемо ці транзистори вибраними). Далі підвищимо напругу U_{on} і подамо на вихід мікросхеми кодову комбінацію, що записується у вибрану четвірку розрядів накопичувача (виводи мікросхеми, які при читанні використовуються як виходи зчитуваного слова, при запису використовуються як входи для подачі слова, що записується).

Нехай, на другий вхід мікросхеми подано рівень логічної 1. При цьому відкривається багатоємітерний транзистор MT₂ в буфері вводу-виводу; емітерний струм цього транзистора, протікаючи через резистор R_0'' , створює напругу, що замикає другий транзистор у вибраній четвірці транзисторів накопичувача. Таким чином, стан транзисторів вибраної четвірки визначається записуванним словом (вибраний транзистор відкритий, якщо у відповідному розряді записуваного слова міститься логічний 0, і, навпаки, цей транзистор закритий, якщо розряд слова, що записується, містить логічну 1). Після цього підвищимо значення напруги колекторного живлення транзисторів накопичувача. Через відкриті вибрані транзистори потече великий струм, що викличе перепалювання перемичок в емітерному колі цих транзисторів. Отже, перемичка в колі емітера вибраного транзистора перепалюється, якщо на відповідний вихід поданий логічний 0. Таким чином може бути записана потрібна інформація у всі елементи накопичувача.

Розглянемо процеси при читанні інформації з ПЗП. При подачі адреси (адресних груп A_1 і A_2) відбувається, як вже розглядалося вище, вибір певної четвірки транзисторів накопичувача. Якщо перемичка в ланцюзі емітера вибраного транзистора не перепалена, струм цього транзистора створює на резисторі напругу, що замикає відповідний багатоємітерний транзистор, якщо ж перемичка перепалена, то багатоємітерний транзистор відкритий. Відкритий або закритий стан багатоємітерних транзисторів MT₁...MT₄ визначає значення розрядів слова.

Розглянемо ПЗП типу K556PT4 та його програмування. Мікросхема ПЗП (рис. 5.8а) має організацію 256 × 4. У відповідності з цим в ній передбачено 8 адресних входів (виводи мікросхеми 5,6,7,4,3,2,1,15) та чотири входи/виходи даних (виводи 12,11,10,9), що виходами в режимі читання і входами в режимі запису. Вивід 16 використовується для під'єднання джерела живлення, вивід 14 (С) - вивід програмування.

Режим читання встановлюється подачею напруги 5 В на вивід 16, на виводи 13 (ВК) та 14 (С) - напруги рівня логічного 0. Входи даних 12,11,10,9 побудовані за схемою з відкритим колектором, тому для зняття даних вони вимагають ввімкнення за схемою, показаною на рис. 5.8б.

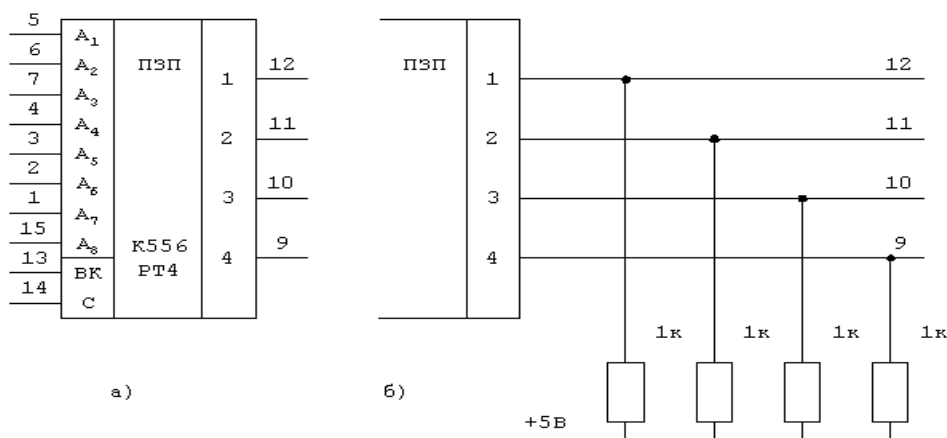


Рис. 5.8. Мікросхема ПЗП K556PT4:

а) умовне позначення; б) схема ввімкнення виходів типу "відкритий колектор"

При програмуванні (в режимі запису) подаються імпульси у відповідності до часової діаграми, що показана на рис. 5.9:

- 1) на адресних входах встановлюється адреса комірки;
 - 2) напруга живлення (на виводі 16) підвищується від 5 В до 10 В (джерело живлення повинно бути розраховане на струм не менше 400 мА);
 - 3) на вивід програмування 14 подається напруга 15 В (струм джерела повинен бути обмежений рівнем 100 мА);
 - 4) на вивід програмування через резистор 300 Ом подається напруга 10 В (при запису логічної 1). В одному циклі можна програмувати тільки один розряд.
- Нарощування ємності ПЗП проводиться за тими ж схемами, що і нарощування ємності ОЗП.

5.5. Перепрограмовані постійні запам'ятовувальні пристрої

Перепрограмовані ПЗП мають всі переваги ПЗП, зберігаючи записану в них інформацію невизначено довго і при відімкненому живленні. В той же час в них допускається стирання записаної інформації і запис нової інформації. Однак, якщо читання здійснюється за частки мікросекунди, то запис вимагає на багато порядків більшого часу.

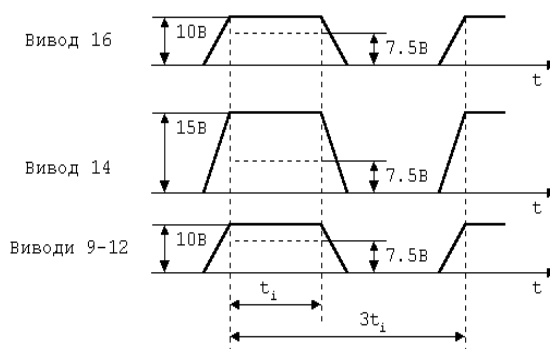


Рис 5.9. Часові діаграми сигналів при запису інформації в ПЗП

Розглянемо принцип роботи наведеного на рис. 5.10а елемента пам'яті з електричним записом інформації і стиранням ультрафіолетовим світлом.

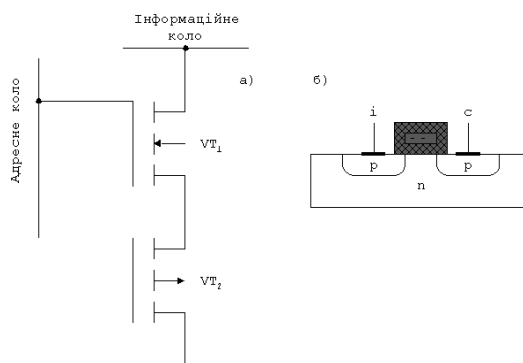


Рис. 5.10. Елемент пам'яті ПЗП: а) схема, б) структура

Транзистор VT_1 служить для вибірки елемента пам'яті. Зберігання інформації здійснюється в транзисторі VT_2 . Особливість транзистора VT_2 , структура якого показана на рис.5.10б, полягає в тому, що він має ізолюваний затвор. При подачі достатньо великої напруги до $p-n$ -переходу виток або сток відбувається інжекція електронів в затвор, після чого цей заряд може утримуватися на затворі тривалий час. Негативний заряд на затворі, притягуючи дірки, створює в n -області провідний p -канал між витком і стоком. Транзистор опиняється в стані логічного 0. Якщо ж до $p-n$ -переходу не прикладалося підвищеної напруги, заряд на затворі буде відсутній, транзистор опиняється в закритому стані (стані логічної 1).

Стирання інформації в одних мікросхемах проводиться шляхом подачі відповідних напруг, в інших — шляхом подачі ультрафіолетового випромінювання через прозору кварцову кришку в корпусі мікросхеми. Під дією напруг або світлового випромінювання, діючого протягом приблизно 10 хв., знімається заряд із затворів

транзисторів і всі транзистори накопичувача встановлюються в стан непровідності. Звичайне кімнатне освітлення практично не впливає на стан транзисторів.

Розглянемо мікросхему ППЗП типу К573РФ1 (рис. 5.11) і її програмування.

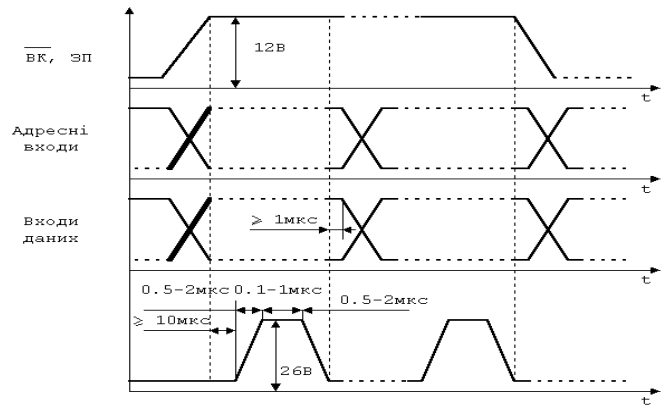
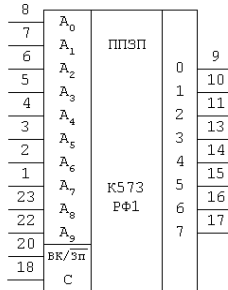


Рис. 5.11 Мікросхема ППЗП К573РФ1 Рис. 5.12. Часові діаграми процесу запису інформації в ППЗП

Дана мікросхема має організацію 1024 на 8, в ній передбачено 10 адресних входів (выводи з номерами 8, 7, 6, 5, 4, 3, 2, 1, 23, 22) і 8 входів-виходів даних (выводи 9, 10, 11, 13, 14, 15, 16, 17), сумісні з ТТЛ-логікою.

Режим читання інформації. Мікросхема вимагає трьох джерел живлення +12 В (вывід 19), +5 В (вывід 24), -5 В (вывід 21) відносно спільного виводу 12. На вході ВК/Зп (вывід 20) і вході програмування (вывід 18) встановлюється напруга рівня логічного 0. Виводи 9...11, 13...17 використовуються як виходи даних, на яких виникає кодова комбінація вмісту комірки пам'яті, адреса якої подана на адресні входи мікросхеми.

Рис. 5.11. Умовне позначення мікросхеми К573РФ1.

Режим стирання інформації. Стирання інформації здійснюється подачею на мікросхему випромінювання високої інтенсивності з довжиною хвилі не більше 400 нм через прозору для променів кришку. Як джерело ультрафіолетового випромінювання можуть бути використані лампи типу ДРТ-220 або ДРТ-375. В процесі стирання, виводи мікросхеми повинні бути закорочені. Після стирання в усіх розрядах всіх комірок пам'яті мікросхеми встановлюється логічна 1.

Режим запису інформації. Цикл запису починається подачею на вхід ВК/Зп (вывід 20) напруги +12 В. На адресних входах послідовно встановлюються адреси комірок пам'яті, інформація, яка записується в комірки у виді паралельних 8-розрядних кодових комбінацій, подається на виводи 9...11, 13...17, які в режимі запису використовуються як входи записуваних даних. Після подачі кодових комбінацій адреси і записуваного числа з деякою часовою затримкою подається імпульс програмування рівня 26 В на вхід програмування 3 (вывід 18). Часова діаграма імпульсів в режимі запису наведена на рис. 5.12. Подача по одному імпульсу програмування на кожен з адреси визначає цикл запису. Протягом кожного циклу запису повинні програмуватися всі комірки пам'яті. Програмування окремих комірок або груп комірок неприпустимо. Необхідне число циклів програмування пам'яті визначається формулою $N = 100мс / t_i$, де t_i тривалість імпульсу програмування, мс. Так, якщо $t_i = 0.5$ мс, то для впевненого запису інформації потрібно $N = 200$ циклів запису.

Перепрограмовані ПЗП дорожчі за ПЗП і їх застосовують в процесі налагодження мікропроцесорних приладів, коли необхідно уточнити інформацію, що повинна зберігатися в пам'яті. Після налагодження ПЗП можна замінити більш дешевим ПЗП.

Список питань для самоконтролю

1. Назвіть типи запам'ятовувальних пристроїв і поясніть режими їх роботи.
2. Які існують основні параметри запам'ятовувальних пристроїв?
3. Яка структура та принцип дії оперативних запам'ятовувальних пристроїв?
4. Яка структура та принцип дії постійних запам'ятовувальних пристроїв?
5. Яка структура та принцип дії перепрограмованих постійних запам'ятовувальних пристроїв?

РОЗДІЛ 6. СИСТЕМИ ВВЕДЕННЯ ТА ВІДОБРАЖЕННЯ ЦИФРОВОЇ ІНФОРМАЦІЇ

6.1. Пристрої введення цифрової інформації

Будь-яка цифрова система вміщує засоби введення-виведення інформації. Як пристрої введення для формування керуючих та кодових сигналів можуть використовуватися кнопкові перемикачі і контактні клавіатури. Як пристрої виведення широко використовують кнопкові, цифрові, знакові індикатори та індикаторні табло на їх основі.

Сигнал за допомогою кнопкових перемикачів формується шляхом замикання-розмикання ними електричного кола (рис. 6.1а). Іншим методом отримання сигналу є використання так званого “контактного щупа” (рис. 6.1б). Тут з виходу контактної площини у вихідному стані знімають потенціальний сигнал $U_{ж}$, а в момент дотику площини щупом рівень сигналу стає рівним нулю. Сигнали, що формуються контактною парою, супроводжуються деренчанням (рис. 6.1в), тривалість якого складає 8-12 мс. Для усунення деренчання в отриманому сигналі (рис. 6.1г) на виході контакту встановлюють спеціальні формувачі. На рис. 6.2 приведені схеми найбільш поширених формувачів. У формувачі на рис. 6.2а використовується принцип безпосередньої установки RS-тригера у відповідності до стану перемикача SB1. Форма сигналу з виходу формувача приведена на рис. 6.1г. Формувач на рисунку 6.2б - це однорозрядний тактований регістр з періодом вхідних тактів $T1 \approx 10-20$ мс. Його роботу ілюструють діаграми на рисунку 6.2в. Перший формувач простий, але для його під'єднання до перемикача SB1 потрібні два резистори ($R1, R2$) і двопровідна лінія. Для другого формувача потрібен один резистор $R3$ і однопровідна лінія, але додатково потрібна подача тактових імпульсів. Перевагою другого способу є часова прив'язка моменту появи вихідного сигналу з внутрішніми процесами пристрою, для якого цей сигнал формується, при цьому як тактові імпульси використовуються вже існуючі в пристрої сигнали.

Формування кодових сигналів виконується в пристроях кодування. Додатковими функціями пристрою кодування є: формування сигналу “Гот.” (готовність) для керування перезаписом сформованого коду; блокування роботи при одночасному натискуванні кількох клавіш і захист від деренчання.

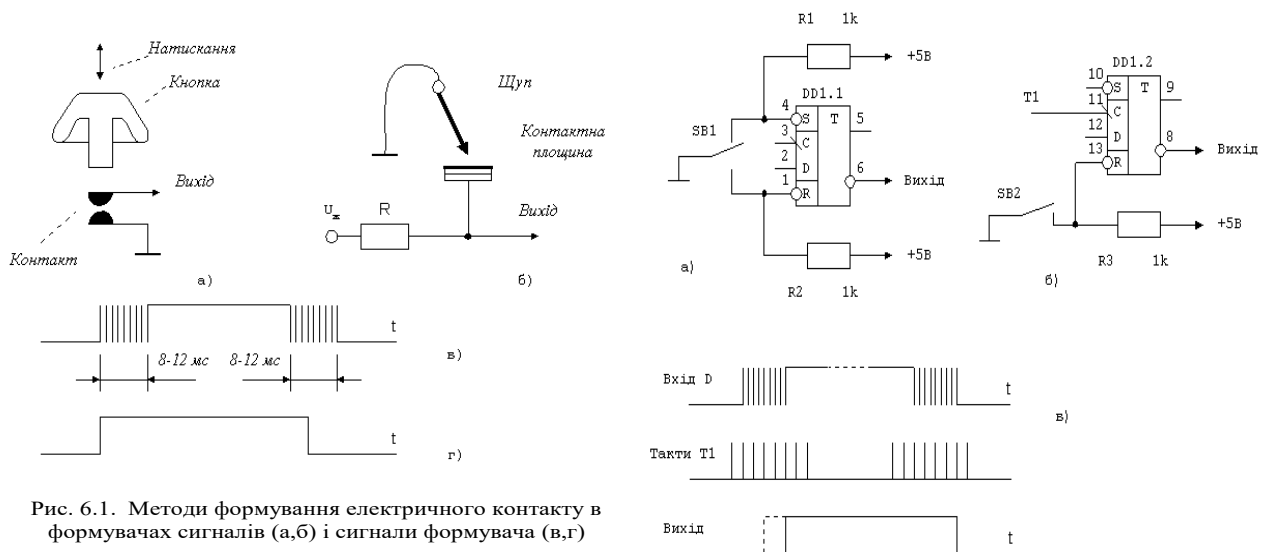


Рис. 6.1. Методи формування електричного контакту в формувачах сигналів (а,б) і сигнали формувача (в,г)

Рис. 6.2. Формувачі сигналів (а,б) і часова діаграма формувача (в)

Пристрій кодування, структурна схема якого наведена на рисунку 5.3, містить: клавіатуру вводу КЛ; кодувальний блок КБ; вузол захисту і формування вихідних сигналів ФС (число n на шині вказує число ліній в цій шині).

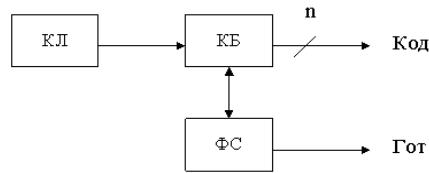


Рис. 6.3. Структурна схема пристрою кодування

Найбільше поширення отримали комбінаційні пристрої кодування і пристрої із сканованою клавіатурою.

Комбінаційний пристрій кодування (рис. 6.4) формує 16 чотири-розрядних кодів. Блок кодування в пристрої утворюють чотири елементи АБО-НІ на DD1-DD4. Вузол ФС пристрою вмикає п'ятивходову збірку АБО-НІ (DD5.1, DD5.2) і формувачі сигналів на тригерах DD6.1 і DD6.2.

При натисканні будь-якої клавіші клавіатури вводу вихідні сигнали з елементів DD1-DD4 утворюють відповідний чотирирозрядний код, на виході тактованого тригера DD 6.2 формується сигнал "Натискання". Тригер DD6.1 встановлюється в одиничний стан по фронту сигналу і скидається після закінчення такту на вході R.

Пристрій кодування на основі клавіатури, що сканується (рис. 6.5) також формує чотирирозрядний двійковий код і вміщує: блок сканування (послідовно ввімкнені чотирирозрядний двійковий лічильник DD2, шістнадцятивходовий мультиплексор DD3), формувач сигналу "Готовність" на тригері DD4.1 і елементі І-НІ DD1.2. Термін "сканування клавіатури" означає послідовне опитування станів всіх її елементів.

В пристрої 16 виходів клавіатури вводу під'єднані до 16 входів мультиплексора DD3. В основному режимі лічильник DD2 працює від зовнішніх тактів, які поступають через елемент DD1.1. Для кожного із станів лічильника мультиплексор проводить опитування стану (натиснутий-відпущений) відповідного контакту клавіатури вводу.

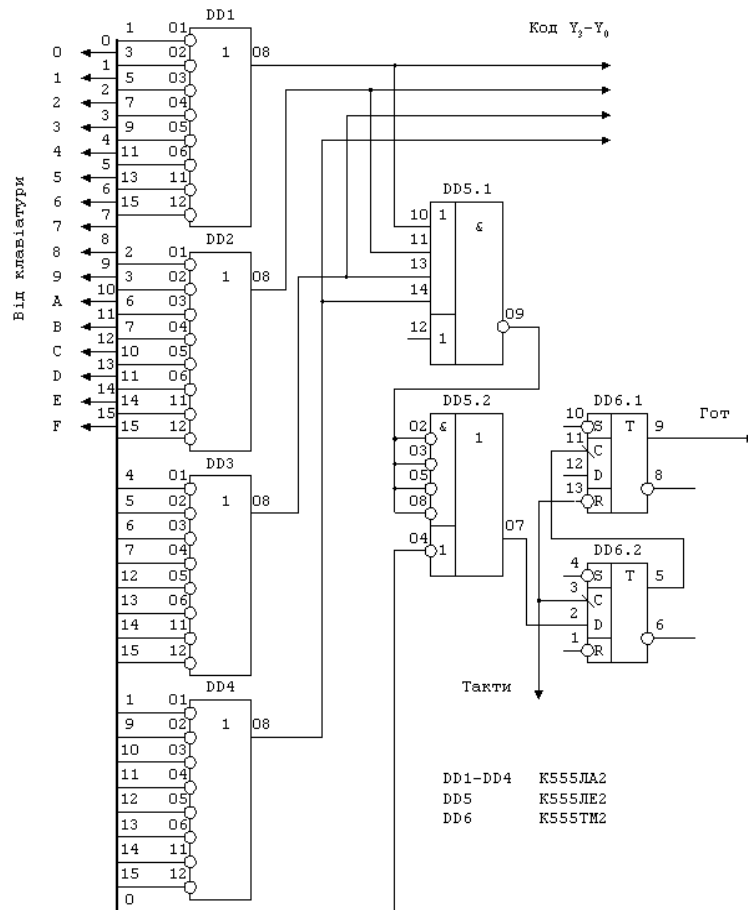


Рис. 6.4. Принципова схема блока кодування комбінаційного типу

В момент опиту натиснутого контакту на виході мультиплексора формується сигнал високого рівня, який готує тригер DD4.1 для установки в 1 по зрізу входного такту. З появою сигналу "Готовність" припиняється подача тактів на лічильник DD2, а стан лічильника відповідає значенню вихідного коду пристрою. При відпущенні клавіші КЛ пристрій повертається у початковий стан.

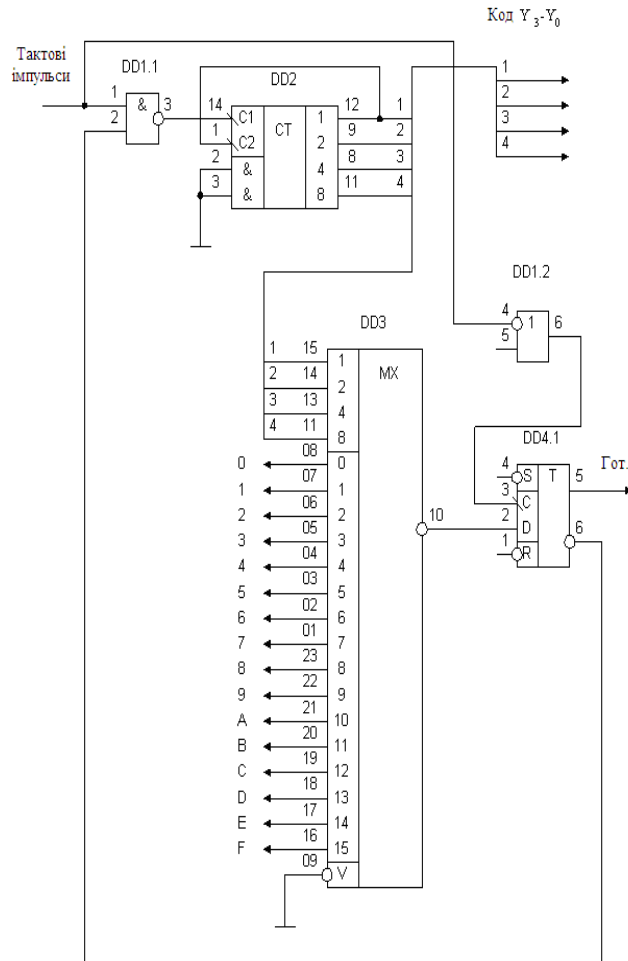


Рис. 6.5. Принципова схема блоку кодування сканувального типу

Перевагою пристрою із сканованою КЛ у порівнянні з комбінаційним є: менші апаратні витрати, наявність захисту від формування хибного коду при одночасному натискуванні кількох клавiш КЛ.

6.2. Елементна база світлових індикаторних пристроїв

Найпростішими приладами для відображення інформації в цифрових пристроях є: світлодіоди, цифрові індикатори і світлові табло.

У напівпровідникових світлодіодах використовується властивість р-п переходу випромінювати світло у видимій частині спектру при протіканні через нього прямого струму. Світлодіоди бувають червоного, жовтого, зеленого і оранжевого кольорів, а також із змінним кольором. Важливим параметром світлодіода є сила світла, яка вимірюється в мілканделах (мКд). Вона пропорційна прямому струму, який протікає через світлодіод. Світлодіоди виготовляють з направленим і розсіяним випромінюванням. Перші з них виконують в металевому корпусі з лінзою, яка забезпечує направлене випромінювання. Розсіяне випромінювання створюють світлодіоди в пластмасовому корпусі із прозорого компаунду.

Постійна пряма напруга для світлодіодів різних типів складає 2-3 В, а значення прямого (робочого) струму – від 5 до 20 мА. Світлодіоди можна використовувати для індикації рівнів на виході мікросхем ТТЛ і транзисторних каскадів. Світлодіод, може засвітитися як при низькому, так і при високому рівнях сигналу на виході мікросхеми. Схеми ввімкнення таких індикаторів зображені на рис. 6.6.

Крім світлодіодів випускають цифрові, знакові, лінійні і матричні світлодіодні індикатори і табло. Цифрові і знакові індикатори бувають однорозрядними і багаторозрядними, їх знаки складаються із семи і більше (9, 11, 16, 18, ...) сегментів. Крім сегментів індикатори мають децимальну крапку чи кому.

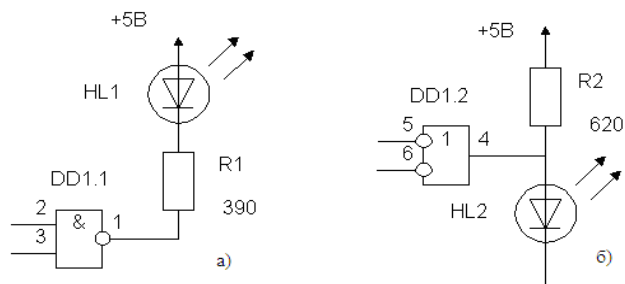


Рис. 6.6. Варіанти ввімкнення світлодіодів для індикації сигналів низького (а) і високого (б) рівнів

Із цифрових найбільше поширення отримали семисегментні індикатори, в яких стилізоване зображення цифр (і деякого набору букв) складають із семи лінійних сегментів, розташованих у вигляді цифри вісім (рис. 6.7). Висвічування сегмента, що вибирається, чи групи сегментів при отриманні зображення знаку забезпечується ввімкненням їх в коло проходження струму.

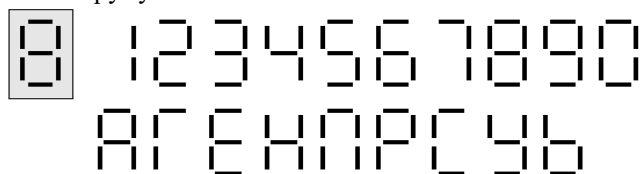


Рис. 6.7. Зображення знаків на основі семисегментного індикатора

На рис. 6.8 подані зовнішній вигляд і принципові схеми індикаторів АЛС324, А, Б. Лінійні шкали на основі світлодіодів - це мікросхеми, що утворені послідовно з'єднаними світлодіодними сегментами, які вмикаються пристроєм керування.

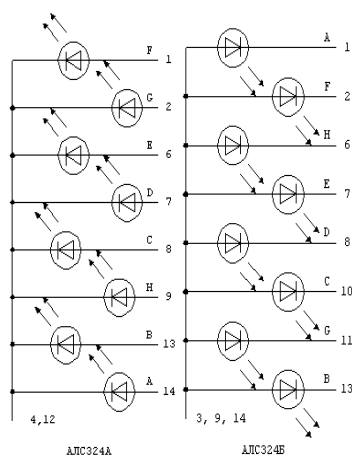


Рис. 6.8. Цифрові світлодіодні індикатори АЛС324

Лінійні шкали використовують для відображення інформації, яка безперервно змінюється, наприклад, для індикації рівня сигналу. Лінійна шкала у ввімкненому стані виглядає як лінія, що світиться, довжина якої визначається числом задіяних сегментів.

Оскільки рідиннокристалічні індикатори не випромінюють світла, то для зчитування з них цифрової інформації потрібне або природне освітлення, або додаткове джерело світла. Принцип дії подібного індикатора ґрунтується на залежності прозорості органічної речовини, що заповнює індикатор, від величини діючого на неї електричного поля. Завдяки цьому змінюється контрастність окремих фрагментів індикатора, з яких формується необхідне зображення.

Розмір знака та індикаційного поля індикатора конструктивно нічим не обмежені. Промисловість випускає багато типів рідиннокристалічних індикаторів, які широко використовуються в електронних годинниках, мікрокалькуляторах та інших пристроях з цифровою індикацією. Всі рідиннокристалічні індикатори працюють на змінному струмі, їх робоча напруга керування складає 4-15 В. Струм індикатора, як правило, не перевищує сотень мікроампер, що і визначило їхнє використання в малогабаритних економічних пристроях.

Окрім світлодіодних і рідиннокристалічних індикаторів в цифрових пристроях можуть використовуватися і газорозрядні індикатори. Ці індикатори мають один чи два аноди і десять катодів, виконаних у вигляді цифр від 0 до 9. Анод газорозрядного індикатора через резистор під'єднують до джерела постійної чи пульсуючої напруги 200-300 В. Середнє значення анодного струму для більшості індикаторів складає 1.5-3 мА. При подачі напруги між анодом і одним із катодів останній починає світитися. В результаті висвічується певна цифра.

6.3. Пристрої індикації

Великий інтерес викликають пристрої відображення цифрової інформації, які побудовані з використанням статичної і динамічної індикації.

Спосіб статичної індикації полягає в постійному підсвічуванні індикатора від одного джерела інформації. Наприклад, в електронних годинниках (рис. 6.9) кожний із цифрових індикаторів блоку індикації через власний перетворювач коду (дешифратор) постійно під'єднаний до “своєї” декади лічильника. В цьому випадку “втрати” на індикацію n знаків є $11n$ з'єднувальних провідників і n дешифраторів. З використанням суміщених мікросхем, наприклад лічильник – дешифратор – індикатор, кількість з'єднувальних провідників значно зменшиться.

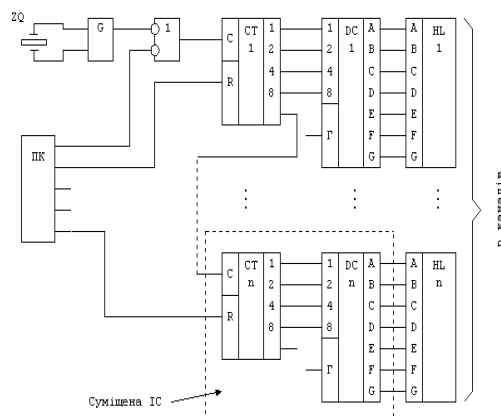


Рис. 6.9. Структурна схема пристрою з блоком статичної індикації

Суть динамічної індикації полягає в почерговому циклічному під'єднанні кожного індикатора до джерела інформації через загальне коло перетворення коду. Перевага цього способу – економія в перетворювачах коду і з'єднувальних проводах. Остання є суттєвою при віддаленні блоку індикації від джерела інформації. Переваги цього способу особливо помітні при числі знаків, що висвічуються, більшому ніж 4-6.

Принципи побудови цифрового табло на n знаків пояснюється на прикладі пристрою, структурна схема якого приведена на рис. 5.10. Пристрій вміщує: n -каналний мультиплексор МХ, виконаний за схемою $n \times 4$ лінії на 1, дешифратор DC1 двійково-десятькового коду в код семисегментного індикатора, блок керування на n станів, який включає двійковий лічильник СТ і дешифратор DC2. Лічильник СТ працює в режимі безперервного підрахунку імпульсів. Коефіцієнт перерахунку лічильника $K=n$. Кожний стан лічильника СТ забезпечує підготовку відповідного індикатора до висвічування інформації від дешифратора DC 1 і передачу через мультиплексор МХ на DC1 інформації від відповідного джерела.

Час світіння одного індикатора протягом циклу роботи лічильника СТ: $t_i = \frac{T_u}{n}$, де T_u - час виконання циклу; $t_i = T_G$ рівне періоду слідування імпульсу G. Для усунення мерехтіння зображення на індикаторі час циклу встановлюють в межах від 10 до 15 мс.

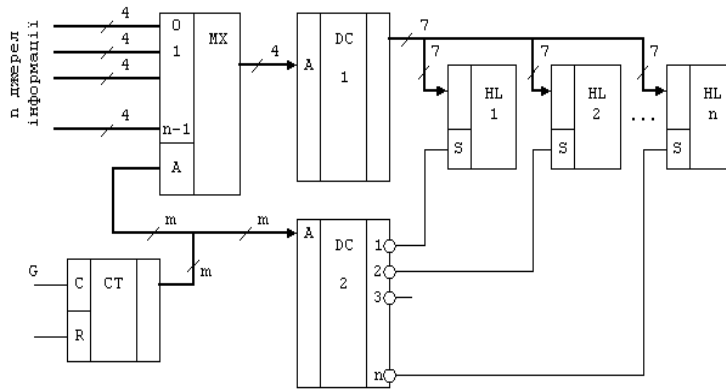


Рис. 6.10. Структурна схема блока динамічної індикації
 Практична схема пристрою, який забезпечує роботу чотирирозрядного цифрового табло, наведена на рис. 6.11.

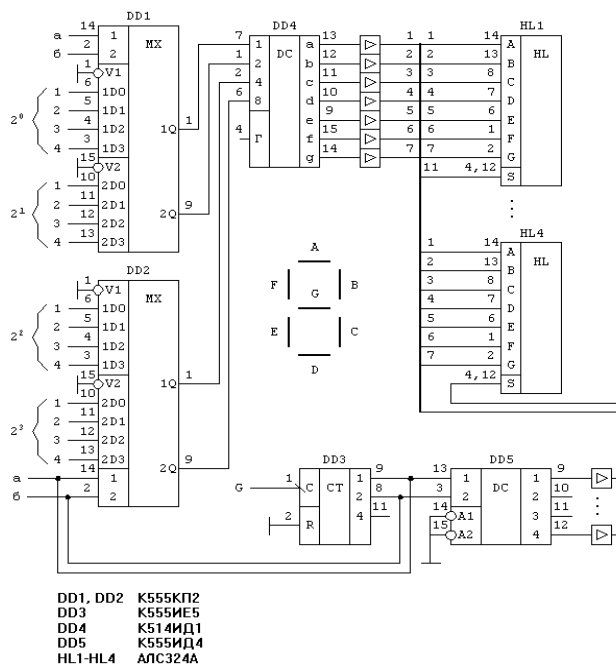


Рис. 6.11. Принципова схема блоку динамічної індикації

Список питань для самоконтролю

1. Назвіть види та принципи дії пристроїв вводу цифрової інформації.
2. Як працюють комбінаційні пристрої кодування?
3. Як працюють пристрої із сканованою клавіатурою?
4. Назвіть основні види та принципи дії світлових індикаторних пристроїв.
5. Наведіть варіанти ввімкнення світлодіодів.
6. Що таке статична індикація і як вона реалізується?
7. Що таке динамічна індикація і як вона реалізується?

РОЗДІЛ 7. ЦИФРО-АНАЛОГОВИ ПЕРЕТВОРЮВАЧІ

7.1. Загальні відомості

Будь-який цифровий пристрій, від найпростішого до самого складного завжди працює за таким принципом: він приймає вхідні сигнали, виконує їх обробку, передачу, зберігання та формує вихідні сигнали. При цьому не обов'язково будь-яка зміна вхідних сигналів призводить до негайної та однозначної зміни вихідних сигналів. Реакція приладу може бути досить складною, відкладеною у часі, неочевидною. В якості вхідних сигналів можуть бути сигнали з виходів інших цифрових пристроїв, перемикачів або клавіш, датчиків фізичних величин. В останньому випадку, зазвичай, необхідно виконати перетворення аналогових сигналів датчиків у потоки цифрових кодів. Вихідні сигнали цифрового пристрою (рис. 7.1) можуть подаватися на інші цифрові пристрої, використовуватися для індикації (екран монітору, цифровий індикатор та ін.) або формування фізичних величин. В останньому випадку необхідно виконати перетворення цифрові коди з цифрового пристрою у неперервні (аналогові) сигнали.



Рис. 7.1. Підключення цифрового приладу

Цифро-аналогові (ЦАП, DAC — "Digital-to-Analog Converter") та аналогово-цифрові перетворювачі (АЦП, ADC — "Analog-to-Digital Converter") використовуються для спряження цифрових пристроїв та систем з зовнішніми аналоговими сигналами. При цьому АЦП перетворює аналогові сигнали у вхідні цифрові сигнали, які потім поступають на цифрові пристрої для обробки або зберігання, а ЦАП — перетворює вихідні цифрові сигнали цифрових пристроїв у аналогові сигнали (див. рис. 7.2).

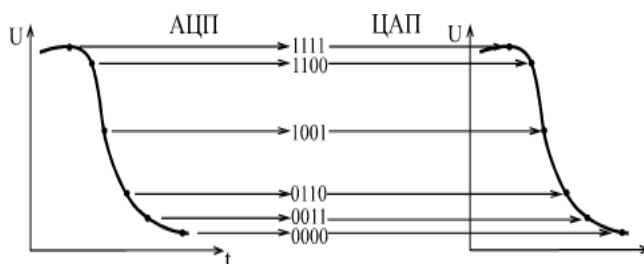


Рис.7.2. Аналогово-цифрове та цифро-аналогове перетворення

ЦАП та АЦП застосовуються у вимірювальній техніці (цифрові осцилографи, вольтметри, генератори сигналів), побутовій апаратурі (телевізори, музичні центри, автомобільна електроніка), комп'ютерній техніці (обробка звукових сигналів, відео монітори, принтери), радіолокаційних пристроях, телефонії і в багатьох інших галузях. Область застосування ЦАП і АЦП постійно розширюється в процесі переходу від аналогових до цифрових пристроїв.

В якості ЦАП і АЦП звичайно застосовують спеціалізовані мікросхеми, що виробляються багатьма вітчизняними та іноземними фірмами. Для вмiлого та професійного використання мікросхем ЦАП та АЦП, необхідне знання аналогової схемотехніки, яка суттєво відрізняється від цифрової. Практичне використання ЦАП і АЦП вимагає розрахунку аналогових кіл, врахування великої кількості похибок перетворення (як статичних, так і динамічних), знання характеристик та особливостей аналогових мікросхем (в першу чергу, операційних підсилювачів).

В даному розділі ми не будемо говорити про особливості вибору та принципах підключення конкретних мікросхем ЦАП і АЦП, а розглянемо лише основи методів з'єднання ЦАП і АЦП з цифровими вузлами.

7.2. Основні параметри та класифікація ЦАП

7.2.1. Статичні та динамічні параметри ЦАП

Цифро-аналоговий перетворювач є декодуєчим пристроєм, на вхід якого надходить цифровий сигнал N , з виходу знімається аналоговий сигнал Y , пов'язаний із вхідним співвідношенням

$$Y=N \cdot q, \quad (7.1)$$

де q - чутливість (ціна молодшого розряду) перетворювача.

В інтегральних ЦАП вхідним сигналом, як правило, є двійковий позиційний код або побудований на його основі десятковий код. Вихідним сигналом звичайно є напруга постійного струму. Такі ЦАП називаються перетворювачами коду в напругу (ПКН). Вихідна напруга ПКН відповідно до (7.1) визначається як

$$U_{вих} = N \cdot U_0 \quad (7.2)$$

де U_0 - ціна молодшого розряду ПКН.

Число N у двійковій системі числення представляється наступним виразом:

$$N = a_{n-1} 2^{n-1} + a_{n-2} 2^{n-2} + \dots + a_1 2^1 + a_0 2^0 = \sum_{i=0}^{n-1} a_i 2^i$$

де a_i - значення розрядів вхідного коду (0 або 1); n - число розрядів вхідного коду. Тому вихідна напруга ПКН відповідно до (7.2) визначається як

$$U_{вих} = \sum_{i=0}^{n-1} a_i 2^i U_0$$

Таким чином, перетворення коду в напругу полягає в підсумовуванні еталонів напруги $2^i U_0$ пропорційних ваги розрядів вхідного коду. У підсумовуванні беруть участь тільки ті еталони, для яких $a_i = 1$. При послідовному зростанні значень вхідного коду N вихідний сигнал $U_{вих}$ утворює східчасту функцію (мал. 7.3,а).

Така залежність називається статичною характеристикою перетворення: $U_{вих} = f(N)$. Вона визначає основні статичні параметри ПКН.

В ідеального ПКН всі сходи характеристики перетворення мають однакові значення U_0 і середні точки сходів розташовані на ідеальній прямій 1 (рис. 7.3,а). Основною характеристикою ідеального ПКН є дозволяюча здатність, обумовлена числом розрядів вхідного коду. Абсолютне значення дозволяючої здатності такого ПКН дорівнює ціні молодшого розряду

$$U_0 = U_{вих} / 2^n - 1 \quad (7.4)$$

де $U_{вих}$ - максимальна вихідна напруга перетворювача.

Зазначена дозволяюча здатність є потенційно можливою й характеризує тільки ідеальні ПКН. Наявність інструментальних погрешностей вузлів реальних ПКН приводить до того, що їхні статичні характеристики перетворення істотно відрізняються від ідеальних розмірами й формою сходів, а також орієнтацією на координатному полі (рис. 7.3,а). Це викликає погрешності перетворювача та погіршення його дозволяючої здатності.

Для кількісного опису сумарної інструментальної погрешності вводиться ряд параметрів.

Абсолютна погрешність перетворення в кінцевій точці шкали (погрешність повної шкали) $\Delta U_{\text{ш}}$ визначає відхилення значення вихідної напруги $U_{вих}$ від номінального (розрахункового) при максимальному вхідному коді $N_{\text{макс}}$ (рис. 7.3,а). Погрешність повної шкали приводиться в довідкових даних в одиницях молодшого розряду: $\delta_{\text{ш}} = \frac{\Delta N_{\text{ш}}}{U_0}$.

Погрешність зсуву нуля на виході $U_{\text{см}}$ — характеризує значення $U_{вих}$ при нульовому значенні вхідного коду (рис. 7.3,а).

Диференціальна нелінійність $\delta_{\text{лн}}$ — характеризує відхилення різниці значень двох вихідних напруг, які відповідають двом сусіднім кодам, від ціни молодшого розряду.

Погрешність нелінійності $\Delta U_{\text{н}}$ — характеризує ідентичність збільшень (сходів) вихідного сигналу у всьому діапазоні перетворення. Вона визначається як максимальне відхилення реальної характеристики перетворення від лінії абсолютної точності 2. Лінія проводиться через точки на характеристиці, що відповідають мінімальному й максимальному значенням вхідного коду (рис. 1.1,а). Абсолютне значення погрешності нелінійності вимірюється в одиницях молодшого розряду: $\delta_{\text{н}} = \frac{\Delta U_{\text{н}}}{U_0}$.

Погрешність нелінійності (рис. 7.3,б) є найбільш важливою характеристикою ПКН і не повинна виходити за межі $\pm \frac{\Delta U_0}{2}$.

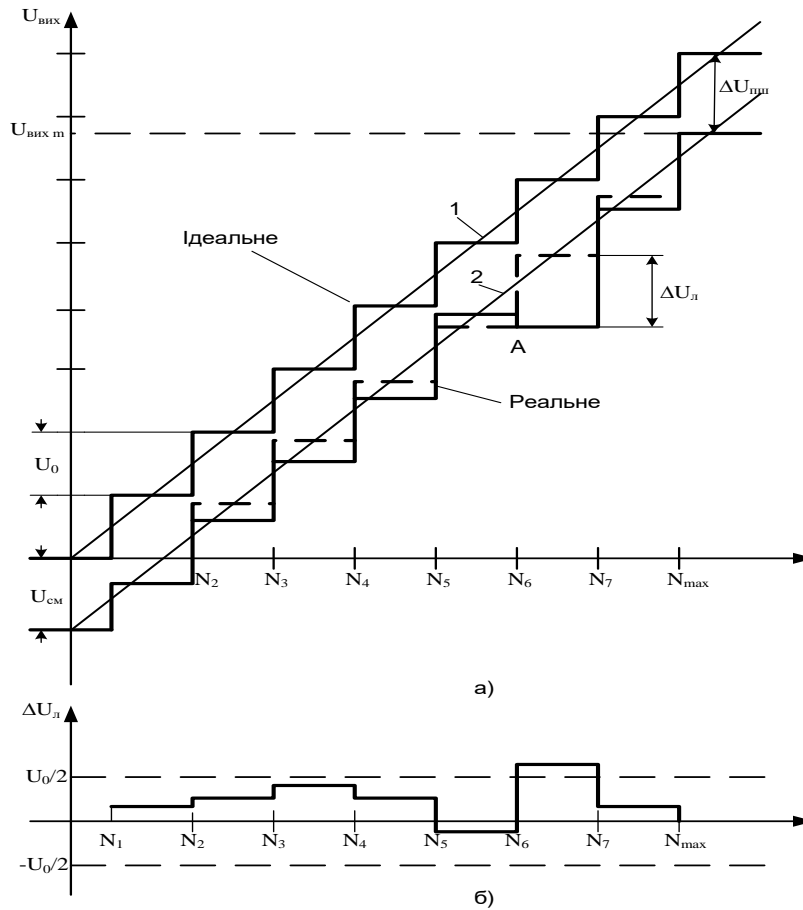


Рис. 7.3. Визначення основних статичних параметрів ПКН:

- а) статичні характеристики перетворення ідеального реального ПКН і основні похибки;
 б) похибка нелінійності реального ПКН

В протилежному випадку характеристика перетворення може бути немонотонною. Це означає, що при подачі вхідного коду, що вимагає збільшення вихідної напруги, воно може зменшитися (ділянку А характеристики реального ПКН на рис.7.3,а).

Сумарна статична погрішність ПКН в основному визначається розглянутими вище складовими. У ряді випадків вона може бути зменшена за рахунок компенсації погрішностей повної шкали й зсуву нуля. Це досягається зміною коефіцієнта підсилення й балансуванням вихідного підсилювача перетворювача.

З динамічних параметрів ПКН найбільш істотними є час установлення вихідної напруги й максимальна частота перетворення.

Час установлення вихідної напруги $t_{\text{уст}}$ — інтервал часу від моменту надходження вхідного коду до моменту встановлення вихідної напруги із заданою точністю.

Максимальна частота перетворення f_{max} — найбільша частота дискретизації (надходження вхідних кодів), при якій параметри ПКН відповідають заданим значенням.

Поряд з розглянутими характеристиками ПКН у довідкових даних приводяться параметри вхідного коду й граничні експлуатаційні параметри.

7.2.2. Класифікація ПКН.

Залежно від набору еталонів розрізняють два методи перетворення коду в напругу:

- метод підсумовування одиничних еталонів
- метод підсумовування з урахуванням маси розрядів.

У першому випадку при формуванні вихідної напруги використовується еталон тільки однієї ваги, рівного U_0 , а на вхід ПКН подається число - імпульсний код.

У другому випадку використовуються еталони, маси яких пропорційні 2^i , де $i = 0, 1, 2, \dots, n-1$, а на вхід ПКН надходить двійковий позиційний код.

По виду вхідного коду ПКН підрозділяються на послідовні та паралельні, на які надходить відповідно послідовний і паралельний двійковий код.

За принципом функціонування розрізняють ПКН *прямого перетворення* і ПКН із *проміжним перетворенням*. У першому випадку вхідний код безпосередньо перетвориться в напругу, у другому - спочатку в проміжний сигнал (число імпульсів, струм, часовий інтервал), що потім перетвориться в напругу.

В інтегральній схемотехніці найбільше поширення одержали паралельні ПКН із прямим перетворенням і підсумовуванням еталонів з урахуванням ваги розрядів. Головною перевагою таких ПКН є висока швидкодія. У залежності від виду використовуваного джерела електричного сигналу вони діляться на дві групи: ПКН зі стабілізацією напруги та ПКН зі стабілізацією струмів.

Ті й інші можуть виконуватися на матриці резисторів з ваговими опорами та на матриці резисторів $R—2R$.

Як ключові елементи в ПКН можуть використовуватися як ключі напруги, так і струму, але в сучасних інтегральних перетворювачах застосовуються більше швидкодіючі ключі й перемикачі струму.

7.3. ПКН на матриці резисторів з ваговими резисторами

Функціональна схема ПКН наведена на рис.7.4. Як джерело електричного сигналу в ньому використовується джерело стабілізованої напруги U_e . Перетворювач складається з n перемикачів струму ($n_0...n_{n-1}$), матриці резисторів, опір яких "зважені" за двійковим законом від 2^0R у старшому $n—1$ у старшому $2^{n-1}R$ у молодшому (нульовому) розряді, і суматора струмів на ОУ, що виконує також перетворення вихідного струму в напругу.

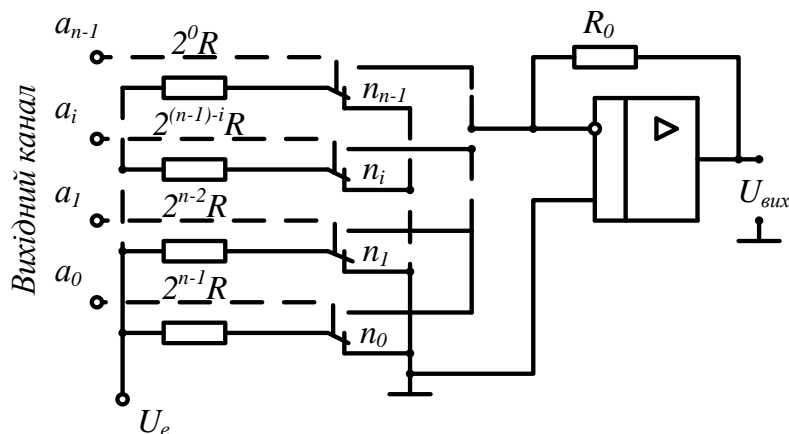


Рис. 7.4. ПКН на матриці резисторів із ваговим опором

На керуючі входи перемикачів ($n_0...n_{n-1}$), подається вхідний код, на комутуючі входи - стабілізована напруга U_e (через відповідні масові резистори матриці).

За принципом дії ПКН аналогічний аналоговому n - входовому суматору на ОУ. Коефіцієнт передачі по кожному вході визначається значенням масового резистора матриці, що підключається до інверсного входу ОУ відповідним перемикачем.

При нульовому значенні вхідного коду ($a_i=0$) перемикачі замикають всі вагові резистори на загальну шину: $U_{вих} = 0$.

Якщо у вхідному коді тільки $a_i = 1$, то i -й перемикач підключить до входу суматора вагомий резистор зі значенням $2^{(n-1)-i}R$, по якому буде протікати струм $I_i = \frac{a_i U_e}{2^{(n-1)-i} R}$, при цьому $U_{вих_i} = I_i R_0 = \frac{a_i U_e R_0}{2^{(n-1)-i} R}$.

У загальному випадку при довільному значенні вхідного коду

$$U_{вих} = \sum_{i=0}^{n-1} U_{вих_i} = \frac{U_e R_0}{2^{(n-1)-i} R} \sum_{i=0}^{n-1} a_i 2^i = \frac{U_e R_0}{2^{(n-1)-i} R} N.$$

З отриманого вираження бачимо, що вихідна напруга ПКН пропорційно значенню вхідного коду N . Коефіцієнт пропорційності визначає ціну молодшого розряду даного перетворювача залежить від співвідношення опорів резистора в ланцюзі зворотного зв'язка ОУ й максимального вагового резистора матриці: $U_0 = \frac{U_e R_0}{2^{n-1} R} (2^n - 1)$.

Якщо на вхід ПКН надходить максимальний вхідний код ($a_i = 1$ у всіх розрядах), то перемикачі підключать всі масові резистори до входу суматора. При цьому максимальне значення вихідної напруги

$$U_{вих_макс} = \frac{U_e R_0}{2^{(n-1)-i} R} (2^n - 1).$$

Основним недоліком ПКН із ваговими резисторами є необхідність використання прецизійних резисторів різних номіналів, максимальний перепад яких дорівнює 2 (від R до $2^{n-1}R$). Відношення потужностей, що розсіюються, на граничних вагових опорах також становить 2^{n-1} . При такому широкому діапазоні потужностей, що розсіюються, важко витримати строго двійкові співвідношення між резисторами. Тому ПКН із ваговими опорами в основному реалізуються на основі гібридних ІС.

У напівпровідникових інтегральних ПКН матриці з масовими опорами застосовуються для перетворення відповідних груп розрядів коду з наступним об'єднанням їхніх вихідних сигналів через масштабні резистори.

На рис.7.5 наведена схема 8- розрядного ПКН, у якому перемикачі й резистори групуються в четвірки з опорами вагових резисторів $R, 2R, 4R, 8R$ і максимальним співвідношенням розрядних струмів 8:1.

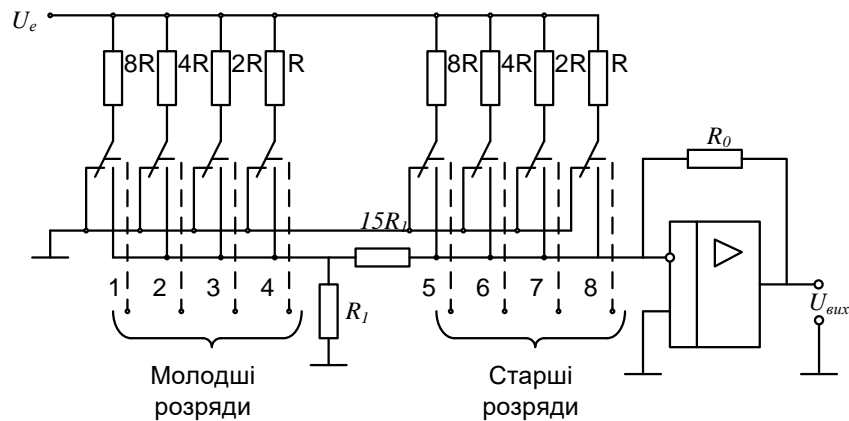


Рис. 7.5. ПКН на комбінованій матриці вагових і масштабних резисторів

Струм перетворювача молодших розрядів до виходу ділиться за допомогою резистивного дільника 16:1. У перетворювачі є лише чотири резистори з різними масовими опорами, що значно зменшує проблеми, пов'язані з узгодженням розрядних струмів. Для забезпечення однакової температурної залежності опорів вагових резисторів і резистора зворотного зв'язка R_0 останній, як правило, виконується в складі матриці резисторів. Розглянутий принцип побудови матриці вагових резисторів широко застосовується в інтегральних ЦАП, наприклад, в 12-розрядному ПКН К594ПА1.

7.4. ПКН на матрице резисторів R-2R

Функціональна схема ПКН зображена на рис. 7.6. У основі 4-розрядного ПКН є резистивна матриця $R-2R$ і ключі (в реальному випадку, ключі – на основі транзисторів). Правому положенню ключа відповідає одиниця у даному розряді вхідного коду N (розряди $D_0...D_3$). Операційний підсилювач може бути як вбудованим (ЦАП з виходом по напрузі), так і зовнішнім (у випадку ЦАП з виходом по струму).

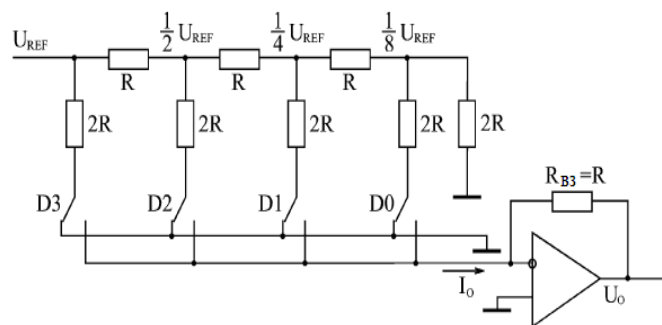


Рис. 7.6. ПКН на матрице резисторів R-2R

Першим (лівим за рис.7.6) ключем комутується струм величиною $U_{REF}/2R$, другим ключем — струм $U_{REF}/4R$, третім — струм $U_{REF}/8R$, четвертим — струм $U_{REF}/16R$. Тобто струми, які комутуються суміжними ключами, відрізняються вдвічі, як і ваги розрядів двійкового коду. Струми, що комутуються усіма ключами, підсумовуються та перетворюються у вихідну напругу за допомогою операційного підсилювача з опором $R_{B3} = R$ у колі від'ємного зворотного зв'язку.

При правому положення кожного ключа (одиниця у відповідному розряді вхідного коду ЦАП), струм, що комутується цим ключем, поступає на сумування. При лівому положенні ключа (нуль у відповідному розряді вхідного коду ЦАП), струм, що комутується ключем, на сумування не поступає.

Тобто внесок першого ключа (старшого розряду коду) у вихідну напругу складає $U_{REF}/2$, другого — $U_{REF}/4$, третього — $U_{REF}/8$, четвертого — $U_{REF}/16$. Отже, при вхідному коді $N = 0000$ вихідна напруга схеми буде рівна нулеві, а при вхідному коді $N = 1111$ вона буде рівна $-15U_{REF}/16$.

В загальному випадку вихідна напруга ЦАП при $R_{B3} = R$, з вхідним кодом N та опорною напругою U_{REF} обчислюється за формулою:

$U_{вих} = -NU_{REF} 2^{-n}$, де n — кількість розрядів вхідного коду. Від'ємне значення отримується внаслідок інверсії сигналу операційним підсилювачем. Цей зв'язок також можна проілюструвати (див. таб. 7.1)

Таблиця 7.1 - Перетворення ЦАП у однополярному режимі

Вхідний код N	Вихідна напруга U_0
000...000	0
000...001	$-2^{-n} U_{REF}$
...	...
100...000	$-2^{-1} U_{REF}$
...	...
111...111	$-(1-2^{-n}) U_{REF}$

Деякі мікросхеми ЦАП передбачають можливість роботи у біполярному режимі, в якому вихідна напруга змінюється не від нуля до U_{REF} , а від $-U_{REF}$ до $+U_{REF}$. При цьому вихідний сигнал ЦАП U_0 множить на 2 та зсувається на величину U_{REF} . Зв'язок між вхідним кодом N та вихідною напругою U_0 , $U_{вих} = U_{REF}(1 - N2^{1-n})$.

Детальніший аналіз формули наведено у таблиці 7.2. Таке біполярне перетворення при можливості зміні знаку опорної напруги називається чотири квадратним множенням (і опорна, і вихідна напруга можуть бути як додатними, так і від'ємними).

Таблиця 7.2 - Перетворення ЦАП у біполярному режимі

Вхідний код N	Вихідна напруга U_0
000...000	U_{REF}
...	...
011...111	$2^{-n} U_{REF}$
100...000	0
...	...
111...111	$-(1 - 2^{1-n}) U_{REF}$

7.5. Перетворювачі коду в часовий інтервал

ЦАП у якому вихідним сигналом є часовий інтервал, називається перетворювачем коду в часовий інтервал (ПКВ) або пристроєм тимчасової затримки, керованої цифровим кодом. Завдання перетворення коду в часовий інтервал виникають у всіляких пристроях і системах керування й виміри. Відомо велика кількість перетворювачів, що виконують таку операцію. Найбільше поширені одержали перетворювачі коду в часовий інтервал на основі двійкового лічильника й на сполученні двійкового лічильника й цифрового компаратора.

7.5.1 ПКВ на основі двійкового лічильника

У ПКВ може використовуватися підсумовуючий або віднімаючий лічильники, що мають входи попереднього запису коду. Функціональна схема ПКВ на основі підсумовуючого двійкового лічильника наведена на м рис.7.7,а. До складу перетворювача входить двійковий лічильник, RS- Тригер, логічний елемент І і формувач вихідного імпульсу. Перед початком перетворення в лічильник записується додатковий код N_0 преутвореного числа N . $N_0=2^n - N$, де n — число розрядів вхідного коду. Перетворення починається з надходженням імпульсу початку інтервалу U_n (рис.7.7,б), що встановлює RS- тригер в одиничний стан.

При цьому через логічний елемент I на вхід лічильника надходять імпульси еталонної частоти з періодом проходження T_e які, як правило, синхронізовані з імпульсами U_n . З надходженням N імпульсів еталонної частоти відбувається переповнення лічильника ($N_0+N=2^n$).

По сигналі переповнення з виходу переносу або старшого розряду лічильника формується імпульс кінця тимчасового інтервалу U_k (рис. 7.7,б). Він затриманий щодо імпульсу U_n на N періодів імпульсів еталонної частоти. Отже, часовий інтервал $T_n=T_e N$ пропорційний вхідному коду. Коефіцієнт пропорційності (ціна молодшого розряду) дорівнює періоду проходження еталонних імпульсів T_e .

Отриманий у результаті перетворення реальний часовий інтервал відрізняється від ідеального через наявність інструментальної погрішності $\Delta t = t_{з.л.е} + t_{уст.чл} + t_{з.Ф} + NT_e$

де $t_{з.л.е}$ — середня затримка логічного елемента І; $t_{уст.чл}$ — час установлення лічильника; $t_{з.Ф}$ — затримка у формувачі; NT_e - нестабільність періоду еталонної частоти.

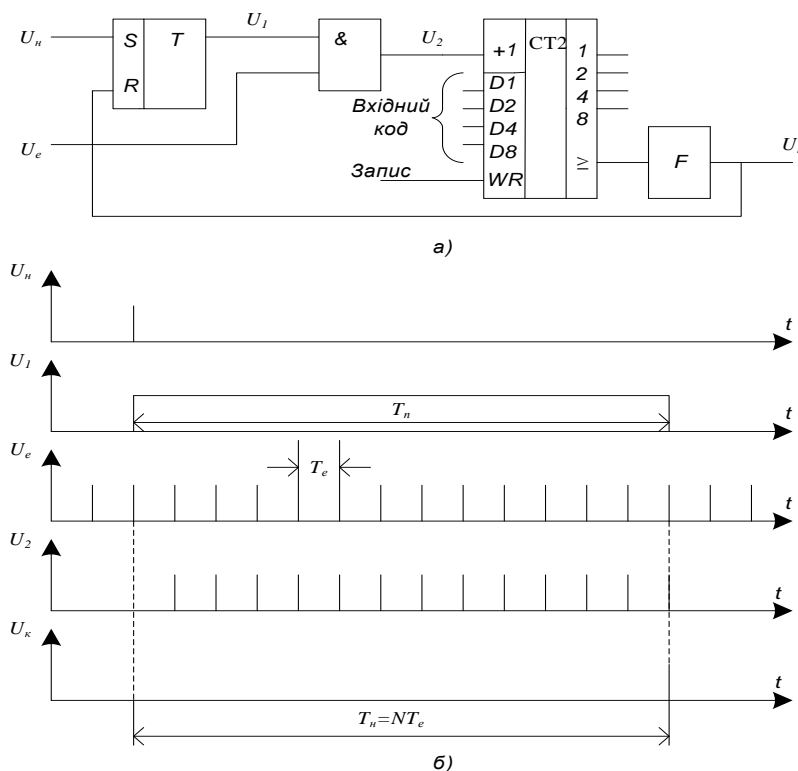


Рис.7.7. ПКВ на основі підсумовуючого лічильника:
а) функціональна схема; б) часові діаграми

Всі складові сумарної погрішності є випадковими. Суму перших трьох складових при відомих середніх значеннях кожної з них, можна компенсувати затримкою імпульсу U_n на час Δt . Для зменшення останньої складової NT_e необхідно підвищувати стабільність частоти (періоду) еталонного генератора. Для цього застосовують генератори із кварцовою стабілізацією частоти.

Робота ПКВ на основі лічильника, що віднімає, відрізняється тим, що в лічильник попередньо записується прямиий код числа N . Момент закінчення тимчасового інтервалу фіксується при обнулінні лічильника.

7.5.2.ПКВ на основі лічильника та цифрового компаратора

Функціональна схема ПКВ наведена на рис.7.8. У його склад входить: RS-тригер, логічний елемент I , двійковий лічильник, цифровий компаратор і регістр.

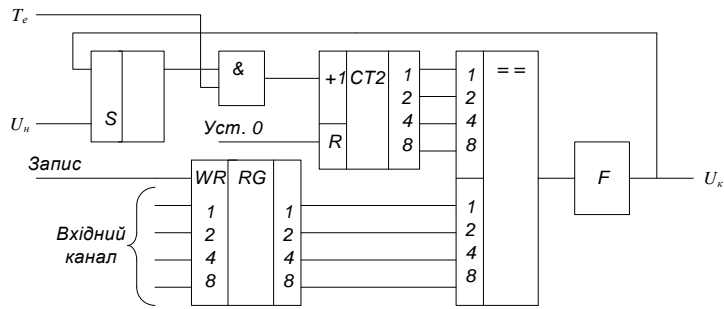


Рис.7.8. ПКВ на основі лічильника і цифрового компаратора

Перед початком перетворення лічильник установлюється в нульовий стан, а в регістр записується вхідний код. По імпульсі початку тимчасового інтервалу U_n RS - тригер установлюється в одиничний стан, при цьому відкривається логічний елемент И , і імпульси еталонної частоти надходять на лічильник. Код з виходу лічильника подається на цифровий компаратор, де рівняється із вхідним кодом. При їхній рівності цифровий компаратор виробляє сигнал рівнозначності, з якого формується імпульс кінця тимчасового інтервалу. Цим же імпульсом RS тригер установлюється в нульовий стан, і рахунок імпульсів еталонної частоти припиняється. Часовий інтервал між імпульсами U_k і $U_k T_n = NT_e$. Методика оцінки погрішності перетворювача й способи її зменшення аналогічні розглянутим раніше для ПКВ на основі підсумовуючого лічильника.

7.6 Особливості застосування ЦАП

В загальному випадку мікросхему ЦАП можна представити у вигляді блоку (рис. 7.9), що має декілька цифрових входів та один аналоговий вхід, а також аналоговий вихід.

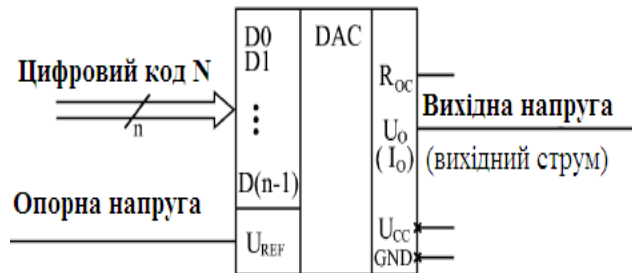


Рис. 7.9. Мікросхема ЦАП

На цифрові входи ЦАП подається n -розрядний код N , на аналоговий вхід – опорна напруга U_{on} (інше поширене позначення — U_{REF}). Вихідним сигналом є напруга $U_{вих}$ (інше позначення — U_o) або струм $I_{вих}$ (інше позначення — I_o). При цьому вихідний струм або вихідна напруга пропорційні вхідному коду та опорній напрузі. Для деяких мікросхем опорна напруга повинна мати строго заданий рівень, для інших – допустимо змінювати її значення у широких межах, а також змінювати полярність. ЦАП з великим діапазоном зміни опорної напруги називають помножуючим ЦАП, оскільки його можна використовувати для множення вхідного коду на опорну напругу.

Крім інформаційних сигналів, мікросхеми ЦАП також вимагають під'єднання одного або двох джерел живлення та загальної шини. Зазвичай цифрові входи ЦАП забезпечують сумісність з стандартними виходами мікросхем ТТЛ.

Здебільше всього, якщо ЦАП має струмів вихід, його вихідний струм перетворюється у вихідну напругу за допомогою зовнішнього операційного підсилювача та вбудованого в ЦАП резистора R_{OC} , один з виводів якого виведено на зовнішній вивід мікросхеми (рис. 7.10). Тому, якщо не сказано інше, надалі, будемо вважати, що вихідний сигнал ЦАП – напруга U_o .

Принцип перетворення вхідного цифрового коду у аналоговий сигнал досить простий. Він полягає у сумуванні декількох струмів (по кількості розрядів вхідного коду), кожний з яких, вдвічі більше за попередній. Для отримання цих струмів використовуються або транзисторні джерела струму або резистивні матриці, які комутуються транзисторними ключами.

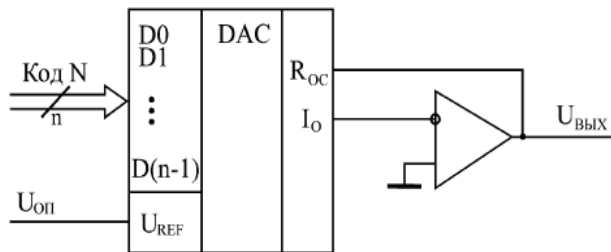


Рис. 7.10. Перетворення вихідного струму ЦАП у вихідну напругу

Промислові мікросхеми ЦАП, відрізняються кількістю розрядів (від 8 до 24), величиною затримки перетворення (від одиниць нс до одиниць мкс), допустимою величиною опорної напруги (зазвичай – декілька вольт), величинами похибок перетворення та деякими іншими параметрами. Розрізняються вони також за технологією виготовлення та особливостями внутрішньої структури, що іноді накладає обмеження на їх використання. Тому, вибір мікросхеми ЦАП для конкретного застосування необхідно проводити з допомогою детальної довідникової інформації, яка надається виробником.

Іноді, необхідно зменшити кількість розрядів ЦАП. Для цього, подають сигнали логічного нуля на потрібну кількість молодших розрядів ЦАП. На рис.7.11 показано, як з 10-розрядного ЦАП можна зробити 8-розрядний, приклавши нулі на два молодших розряди. Збільшення кількості розрядів ЦАП є більш складною задачею, що вимагає побудови складних аналогових схем. Значно простіше підібрати мікросхему з необхідною, або більшою кількістю розрядів.

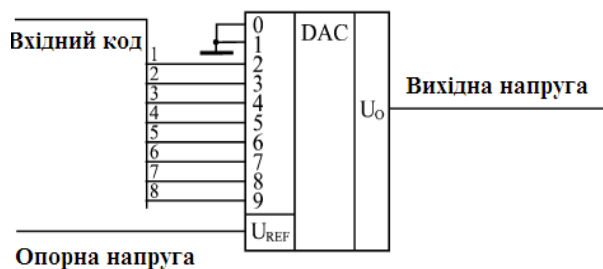


Рис. 7.11. Зменшення розрядності ЦАП

Основною областю застосування мікросхем ЦАП є отримання аналогового сигналу з послідовності цифрових кодів (рис.7.12).

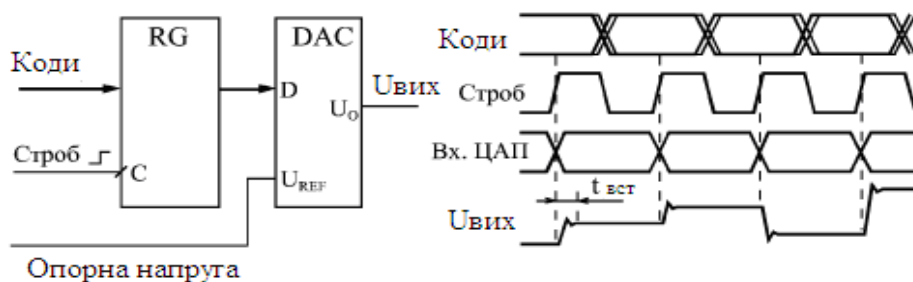


Рис. 7.12. Перетворення послідовності кодів у вихідну напругу

Як правило, коди подаються на входи ЦАП через паралельний регістр, що дозволяє забезпечити одночасність зміни усіх розрядів вхідного коду ЦАП. При неодноразовній зміні розрядів вхідного коду, на виході ЦАП з'являються короткі імпульси напруги великої амплітуди, рівні яких не відповідають жодному з кодів.

Але, навіть при одночасній зміні усіх розрядів вхідного коду ЦАП, рівень напруги, що відповідає поданому коду, встановлюється не одразу, а за час встановлення ЦАП $t_{вст}$, що обумовлене неідеальністю внутрішніх елементів ЦАП. Вихідний струм ЦАП, як правило, встановлюється значно швидше вихідної напруги, оскільки він не залежить від інерційності операційного підсилювача. Зрозуміло, що умова правильної роботи ЦАП полягає у тому,

щоб тривалість зберігання вхідного коду була більшою, ніж час встановлення ЦАП $t_{есм}$, інакше вихідний сигнал не встигне прийняти значення, що відповідає вхідному коду.

Якщо ж подавати коди на вхід ЦАП з великим інтервалом у часі, то приведена на рис. 7.13 схема може використовуватися, наприклад, у керованому джерелі живлення, вихідна напруга якого задається вхідним кодом. Додатково, необхідно застосувати зовнішній підсилювач струму для збільшення вихідного струму.

При зміні вхідного коду ЦАП з високою частотою, можна отримати генератор (синтезатор) аналогових сигналів довільної форми. В такому випадку коди, що поступають на ЦАП, називають кодами вибірок (миттєвих значень) генеруемого аналогового сигналу.

В найпростішому варіанті реалізації в якості джерела вхідних кодів ЦАП можна використати звичайний двійковий лічильник (рис.7.14). Вихідна напруга ЦАП буде наростати при цьому на величину U_{REF} з кожним тактовим імпульсом, формуючи пилкоподібні вихідні сигнали амплітудою U_{REF} . Тривалість кожної сходинки рівна періоду тактового генератора T , а період всього вихідного сигналу - $2^n T$. Кількість сходинок у періоді вихідного сигналу рівна 2^n . Якщо у даній схемі використовувати синхронні лічильники з синхронним переносом, то вхідний регістр Цап не потрібен, оскільки усі розряди лічильника перемикаються одночасно. При використанні асинхронних лічильників або синхронних лічильників з асинхронним переносом, то необхідний вхідний регістр ЦАП.

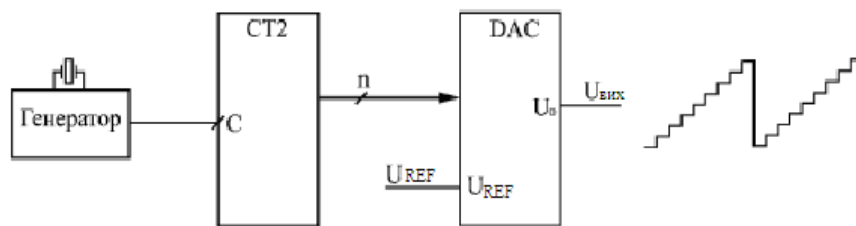


Рис. 7.13. Генератор пилкоподібного аналогового сигналу

При потребі формування аналогових сигналів довільної форми (синусоїдальні, гаусні, шумові, трикутні, імпульсні та ін.), в якості джерела кодів використовують елемент пам'яті, що працює у режимі зчитування (рис. 6.9).

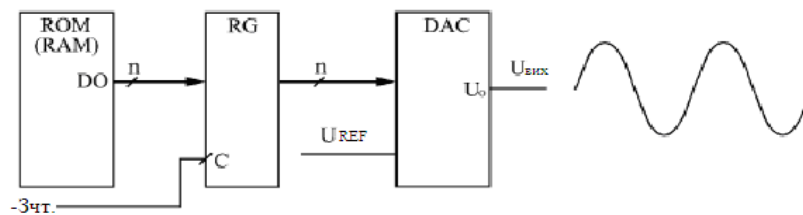


Рис. 7.14. Генерація сигналів довільної форми

Якщо пам'ять постійна, то набір форм генерованих сигналів задається один раз. При використанні оперативної пам'яті створюють однонаправлений інформаційний буфер з періодичним режимом роботи, що дозволяє записувати у пам'ять коди для генерації довільних сигналів. В обох випадках необхідний вхідний регістр ЦАП, інформація у нього записується стробом читання з пам'яті (рис. 7.15). Як і в попередньому випадку, вихідний сигнал ЦАП буде складатися зі сходинок, висота яких кратна U_{REF} . Амплітуда вихідного сигналу не перевищує U_{REF} .

Якщо адреси пам'яті перебирає лічильник, то період вихідного аналогового сигналу рівний $2^m T$, де T — період тактового сигналу зчитування з пам'яті "Зчит.", а m — кількість адресних розрядів пам'яті.

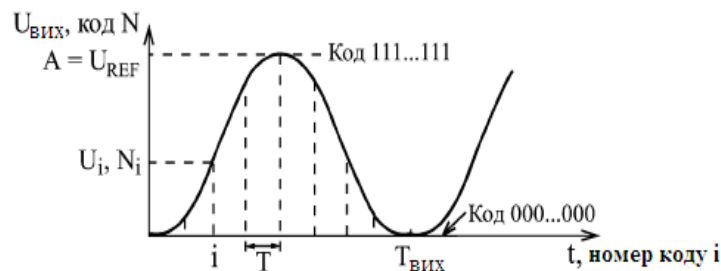


Рис. 7.15. Коди вибірок періодичного сигналу

Перетворення цифрових кодів у аналоговий сигнал – не єдине застосування мікросхем ЦАП. Вони також використовуються для керованої обробки аналогових сигналів, наприклад, для підсилення або ослаблення аналогового сигналу у задану кількість разів. Для цього найбільш придатними є помножуючі ЦАП, які допускають зміну рівня опорної напруги в широких межах.

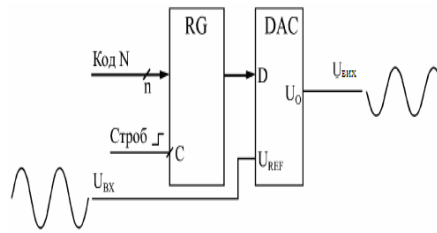


Рис. 7.16. Атенюатор аналогового сигналу на ЦАП

Найбільш проста схема – це цифровий атенюатор (ослаблювач) аналогового сигналу (рис.7.16), що часто застосовується для регулювання амплітуди вихідного сигналу генератора на основі ЦАП.

Опорна напруга являє собою змінний аналоговий сигнал (ЦАП – обов’язково помножуючий). Вихідний сигнал зв’язаний з вхідним за формулою: $U_{\text{в\ddot{e}o}} = -U_{\text{o\ddot{o}}} N 2^{-n}$. Тобто вихідний сигнал пропорційний до вхідного (з інверсією), а коефіцієнт пропорційності визначається вхідним цифровим кодом N . В даному випадку коефіцієнт пропорційності змінюється від нуля і майже до одиниці з кроком 2^{-n} . Вхідний регістр ЦАП при цьому також необхідний, оскільки при неодноточасному перемиканні розрядів вхідного коду на вихідний сигнал ЦАП можуть накладатися короткі імпульси значної амплітуди.

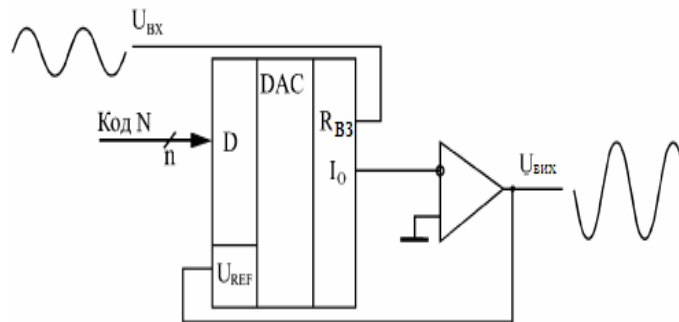


Рис. 7.17. Керований підсилювач вхідного сигналу

Існує також схема включення ЦАП, яку можна використовувати як керований підсилювач аналогового сигналу з коефіцієнтом підсилення, який задається вхідним кодом N (рис. 7.17).

В такому випадку, вихідний струм ЦАП дорівнює U_{ex}/R_{OC} , а оскільки в якості опорної напруги використовується вихідна напруга, то коефіцієнт пропорційності: $U_{\text{вих}} = -U_{\text{ex}} N 2^{-n} / N$.

Код N може змінюватися в даному випадку від 1 до $(2^n - 1)$, що відповідає коефіцієнту підсилення від приблизно одиниці до $2^n - 1$. Наприклад, при використанні 10-розрядного ЦАП, коефіцієнт підсилення схеми може досягати 1024. Як і для попереднього випадку, швидкість перемикання ЦАП не дуже важлива, оскільки коефіцієнт підсилення, здебільшого не потрібно змінювати дуже часто. На схемі для спрощення не показаний вхідний регістр ЦАП, який є необхідним для забезпечення одночасності перемикання усіх розрядів вхідного коду.

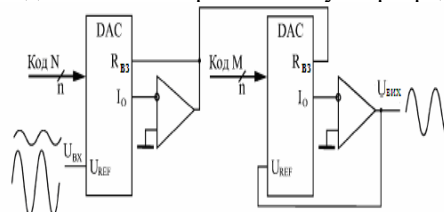


Рис. 7.18. Послідовне ввімкнення атенюатора та підсилювача

Використовуючи послідовне підключення схем на рис. 7.16 і 7.17, можна забезпечити приведення до стандартного рівня вхідної напруги, яка змінюється у досить широких межах (рис.7.18). Така задача часто виникає у аналогово-цифрових системах.

Коефіцієнт передачі усієї системи рівний відношенню вхідних кодів обох ЦАП N/M і може бути встановлений з високою точністю як у діапазоні від 0 до 1 (атенюатор), так і в діапазоні від 1 до $2n$ (підсилювач). На схемі не показані вхідні регістри обох ЦАП, але вони обов'язково потрібні.

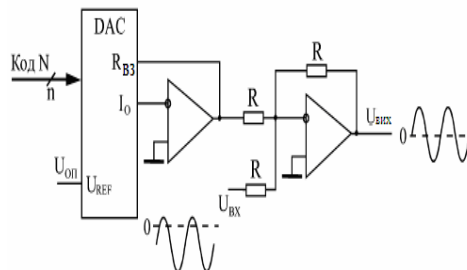


Рис. 7.19 Схема керованого зсуву аналогового сигналу

Остання схема з використанням ЦАП, яку ми розглянемо, - це схема зсуву аналогового сигналу на величину, яка задається вхідним цифровим кодом. Зсув являє собою складання аналогового сигналу з постійною напругою.

Схема зсуву (рис.7.19) містить перетворювач цифрового коду у вихідну напругу та аналоговий суматор на операційному підсилювачі. Величина напруги зсуву вхідного сигналу рівна $N2^{-n}U_{REF}$.

Оскільки використано два інвертуючі операційних підсилювача, інверсії вхідного сигналу на виході не буде. Якщо необхідно забезпечити можливість біполярного зсуву, то необхідно застосувати ЦАП з біполярним вихідним сигналом.

Список питань для самоконтролю

1. Для чого призначені ЦАП?
2. З чого складається загальна інструментальна похибка ЦАП?
3. Які методи перетворення коду в напругу?
4. Як розрізняють ЦАП з а принципом функціонування?
5. Принцип роботи ПКН на матриці резисторів з ваговими резисторами?
6. Принцип роботи ПКВ на основі двійкового лічильника?
7. Принцип роботи ПКВ на основі лічильника та цифрового компаратора?

РОЗДІЛ 8. АНАЛОГОВО-ЦИФРОВИ ПЕРЕТВОРЮВАЧІ

8.1. Основні параметри та класифікація АЦП

Мікросхеми АЦП виконують функцію, протилежну функції ЦАП, — перетворюють вхідний аналоговий сигнал у послідовність цифрових кодів. Мікросхему АЦП можна зобразити у вигляді блоку, що має один аналоговий вхід, один або два входи для подачі опорної напруги, а також цифрові виходи для виводу коду, який відповідає значенню аналогового сигналу (рис. 8.1).

Часто мікросхема АЦП має також вхід для подання тактового сигналу CLK, сигналу дозволу роботи CS та сигналу, що повідомляє про готовність вихідного цифрового коду RDY. На мікросхему подають одну або дві напруги живлення та загальну шину. В цілому, мікросхеми АЦП складніші, ніж мікросхеми ЦАП, їх різновидів помітно більше, а тому, сформувавши для них спільні принципи складніше.

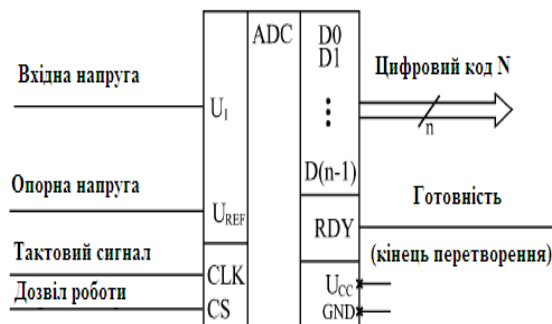


Рис. 8.1. Мікросхема АЦП

Опорна напруга АЦП задає діапазон вхідної напруги, в якому виконується перетворення. Вона може бути постійною або змінюватися в певних межах. Іноді, на АЦП подають дві опорні напруги з різними знаками, тоді АЦП може працювати з біполярними вхідними напругами. Робота АЦП керується тактовим сигналом CLK, який задає частоту перетворення. Гранична тактова частота – другий важливий параметр АЦП. В деяких мікросхемах вбудований генератор тактових імпульсів, частота якого задається зовнішнім кварцовим резонатором. Сигнал CS дозволяє роботу мікросхеми.

АЦП є кодованим перетворювачем, на вхід якого надходить аналоговий сигнал $x_{вх}$ а з виходу знімається цифровий код N , пов'язаний з вхідним сигналом співвідношенням $N = \frac{x_{вх}}{\Delta x}$, де Δx - крок квантування перетворювача.

Серед АЦП найбільшого поширення набули перетворювачі, у яких вхідним аналоговим сигналом є напруга, а вихідним цифровим сигналом - двійковий код. Такі перетворювачі називають перетворювачами напруги в код (ПНК).

Відповідно до узагальненої схеми АЦП в ПНК виконується дискретизація і квантування вхідної напруги та кодування результату перетворення. При цьому вхідній напрузі $U_{вх}$ ставиться у відповідність цифровий двійковий код $N = \frac{U_{вх}}{\Delta U_{кв}}$, де величина $\Delta U_{кв}$ - є кроком квантування (квантом) n -розрядного перетворювача.

Вихідний цифровий код N (n -розрядний) однозначно відповідає рівню вхідної напруги. Код може приймати значень, тобто АЦП може розрізнити рівнів вхідної напруги. Кількість розрядів вихідного коду n є важливою характеристикою АЦП. В момент завершення формування вихідного коду, видається сигнал завершення перетворення RDY, по якому зовнішній пристрій може зчитувати код N .

Статична характеристика квантування ПНК $N = f(U_{вх})$ (рис.8.2.) аналогічна характеристиці перетворення ПКН $U_{вих} = f(N)$. Характеристика квантування (рис1.) також визначає основні статичні параметри ПНК, але по відношенню до вхідної напруги $U_{вх}$.

Основними характеристиками ПНК є: роздільна здатність, точність і швидкодія.

Роздільна здатність ПНК визначається числом розрядів вихідного коду. Абсолютне значення роздільної здатності ПНК дорівнює значенню одного кроку квантування, що жорстко пов'язаний з діапазоном зміни вхідної напруги і

числом розрядів n вихідного коду: $\Delta U_{кв} = \frac{U_{вх_макс} - U_{вх_мін}}{2^n}$

де $U_{вх.макс}$ і $U_{вх.мін}$ - максимальне і мінімальне значення вхідної напруги.

Крок квантування, характеризує потенційну роздільну здатність перетворювача за рівнем, тобто найменше помітне збільшення вхідної величини.

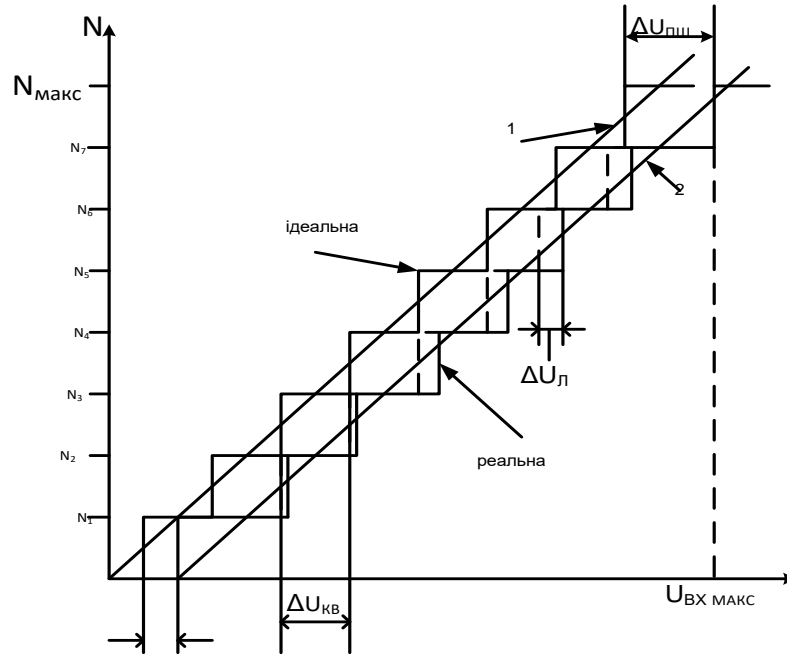


Рис.8.2. Статична характеристика квантування ПНК

Крок квантування також визначає методичну похибку аналого-цифрового перетворювача, що виникає в результаті заміни безперервної аналогової величини квантованими значеннями (найближчими рівнями квантування). Статична точність ПНК крім методичної похибки квантування характеризується також загальною інструментальною похибкою всіх елементів, що утворюють перетворювач.

Основними складовими загальної інструментальної похибки є:

- абсолютна похибка перетворення в кінцевій точці шкали $\Delta U_{ниш}$ (похибка повної шкали);
- похибка зміщення нуля на вході $U_{зм}$;
- диференціальна похибка $\Delta U_{л}$;
- похибка нелінійності ΔU .

Зазначені складові інструментальної похибки визначаються по статичній характеристиці квантування ПНК $N = f(U_{вх})$ (рис.8.2).

Абсолютна похибка перетворення в кінцевій точці шкали $\Delta U_{ниш}$ - відхилення значення вхідної напруги від номінального при максимальному вихідному коді $N_{макс}$ (рис.8.2). Абсолютна похибка повної шкали в довідкових даних наводиться в одиницях кроку квантування

Похибка зміщення нуля на вході $U_{зм}$ - прикладена до входу напруги, відповідає відхиленню початкової ділянки характеристики квантування від номінального значення (рис.8.2).

Диференціальна нелінійність $\Delta U_{л}$ - відхилення різниці значень двох вхідних напруг, що відповідають середнім значенням двох сусідніх рівнів квантування, від величини кроку квантування.

Похибка нелінійності ΔU - максимальне відхилення реальної характеристики квантування від лінії абсолютної точності 2 (рис.8.2). Абсолютне значення похибки нелінійності вимірювання в одиницях кроку квантування:

$$\Delta_l = \frac{\Delta U_{л}}{\Delta U_{кв}}. \text{ Похибка нелінійності інтегральних ПНК не повинна виходити за межі } \pm \frac{\Delta U_{кв}}{2}.$$

Сумарна статична похибка перетворювача не може визначатися у вигляді деякого заданого числа, так як вона залежить від багатьох чинників, що знаходяться під впливом як часу, так і температури.

Швидкодія ПНК характеризується динамічними параметрами; часом перетворення $t_{пр}$ і максимальною частотою перетворення F_m .

Час перетворення $t_{пр}$ - інтервал часу від моменту заданого зміни сигналу на вході до появи на виході відповідного сталого коду.

Вибір часу перетворення пов'язаний зі швидкістю зміни вхідного сигналу. Якщо вимагати, щоб за час $t_{пр}$, вхідна напруга змінювалася не більш, ніж на крок квантування $\Delta U_{кв}$ то: $t_{пр} \leq \frac{\Delta U_{кв}}{U_{вх}}$, де $U_{вх}$ - максимальна швидкість

зміни вхідної напруги.

Якщо ця вимога не виконується, то на вході ПНК необхідно включити пристрій вибірки та зберігання, який запам'ятовує вибірки вхідної напруги в моменти дискретизація на час перетворення.

Максимальна частота перетворення F_m - найбільша їх кількість, при якій параметри ПНК відповідають заданим значенням. Максимальна частота перетворення вибирається в залежності від ширини спектру вхідного напруги і необхідних якісних показників системи, до складу якої входить ПНК.

Поряд з розглянутими характеристиками і параметрами існують параметри, що визначають зовнішні умови роботи перетворювача, такі, як:

- параметри вихідного коду;
- динамічний діапазон вхідної напруги;
- параметри джерел еталонного і живлячої напруги;
- граничні експлуатаційні параметри та інші.

Класифікація ПНК. Основною ознакою класифікації ПНК є алгоритм його функціонування в процесі перетворення. Алгоритм відображає комплекс операцій, за допомогою яких встановлюється чисельна відповідність між вхідною аналоговою величиною та вихідним цифровим кодом.

По виду алгоритму перетворення ПНК діляться на три основні типи:

- послідовного рахунку;
- порозрядного кодування (послідовного наближення);
- паралельні (зчитування).

За принципом перетворення ПНК підрозділяються на перетворювачі прямої дії і з проміжним перетворенням. У ПНК прямої дії вхідна напруга перетворюється безпосередньо в цифровий код. У ПНК з проміжним перетворенням напруга перетворюється спочатку в іншу аналогову величину (часовий інтервал, частоту та ін), яка потім перетворюється в двійковий код.

8.2. Перетворювачі напруги в код послідовного рахунку

У ПНК послідовного рахунку вхідна напруга врівноважується сумою однакових і мінімальних еталонів, рівних кроку квантування. Число еталонів, що використовуються для врівноваження, підраховується лічильником, а момент рівності вхідної напруги і суми еталонів фіксується компаратором. Результатом перетворення є цифровий код лічильника.

Найбільшого поширення серед ПНК послідовного рахунку отримали перетворювачі, в яких еталонні напруги отримують за допомогою ЦАП, включеного в ланцюг зворотного зв'язку. До них відносяться: ПНК із ступінчастою розгортаючою напругою і стежачи ПНК.

8.2.1. ПНК з ступінчастою розгортаючою напругою

Функціональна схема ПНК і тимчасові діаграми, що пояснюють принцип роботи, наведені на рис.8.3,а. Імпульс запуску $T_{ц}$, що визначає початок циклу перетворення, встановлює RS-тригер в одиничний стан. Рівень логічної одиниці з виходу тригера дозволяє проходження через логічний елемент І імпульсів еталонної частоти на вхід лічильника. Послідовно збільшується код з виходу лічильника з допомогою ПКН перетворюється в ступінчасту напругу, яка змінюється за лінійним законом. Величина ступені дорівнює кроку квантування $\Delta U_{кв}$. При перевищенні вихідною напругою ПКН вхідної напруги $U_{вх}$ компаратор спрацьовує, встановлює RS-тригер в нульовий стан і блокує проходження еталонних імпульсів на вхід лічильника. Число імпульсів N , записаних в лічильник, буде дорівнює кількості квантів $U_{кв}$, необхідних для врівноваження вхідної напруги:
$$N = \frac{U_{вх}}{\Delta U_{кв}}.$$

Перед початком чергового циклу перетворення здійснюється видача коду споживачеві і встановлення лічильника в нульовий стан.

Сумарна похибка перетворювача складається з методичної та інструментальної складових. Методична складова характеризується похибкою квантування і є функцією числа розрядів вихідного коду. Максимальна похибка квантування дорівнює кроку квантування $\Delta U_{кв}$.

Основними джерелами інструментальної похибки є компаратор і ПКН. У найгіршому випадку сумарна інструментальна похибка $\Delta U_{i_макс} = \Delta U_{комп} + \Delta U_{кв_пкн}$, а сумарна похибка перетворювача

$$\Delta U_{пнк} = \Delta U_{кв} + \Delta U_{комп} + \Delta U_{пкн}.$$

Інструментальні похибки $\Delta U_{комп}$ і $\Delta U_{пкн}$ містять систематичні і випадкові складові. Систематичні складові можна компенсувати підстроюванням елементів перетворювача. Випадкові складові (дрейф, шуми тощо) необхідно враховувати при оцінці параметрів ПНК.

Інструментальні похибки ПНК характеризуються такими основними параметрами, як похибка нелінійності $\Delta U_{л}$, похибка повної шкали $\Delta U_{пш}$ і похибка зміщення нуля $\Delta U_{зм}$.

Час перетворення $t_{пр}$ ПНК послідовного рахунку непостійний і залежить від $U_{вх}$.

Період $T_{ц}$ вибирається з умови $T_{ц} > t_{пр,макс} + t_{пнк}$, де $t_{пнк}$ - час, необхідний для видачі коду та встановлення елементів перетворювача в початковий стан.

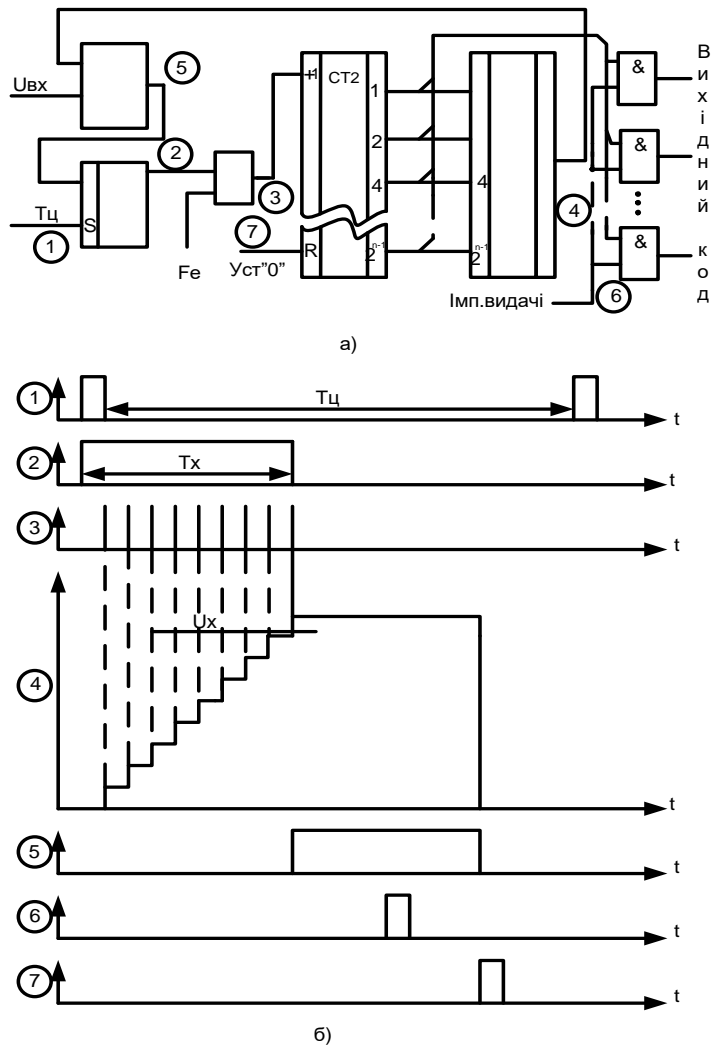


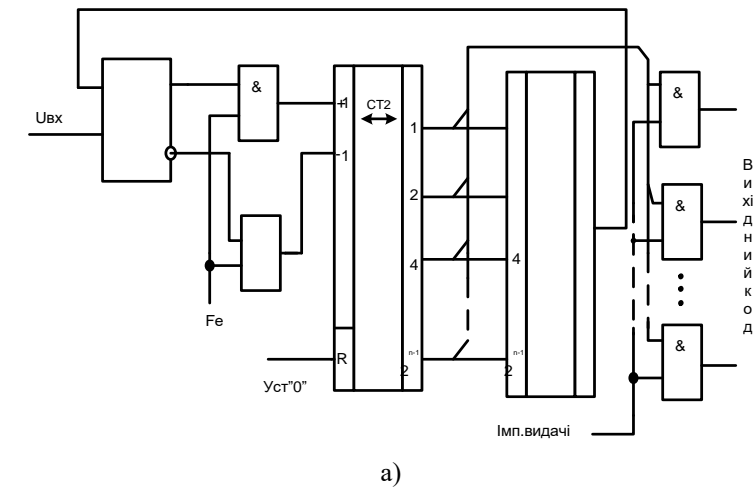
Рис.8.3. ПНК із ступінчастою розгортаючою напругою

Перевагою ПНК послідовного рахунку є порівняна простота і висока точність, основним недоліком низька швидкодія.

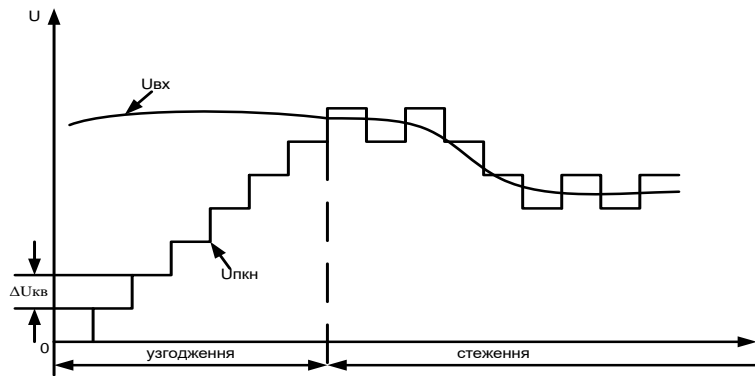
Основні сфери їх застосування - цифрові вольтметри і радіотехнічні системи, призначені для роботи з постійними і напругами, що повільно змінюються. ПНК послідовного рахунку, як правило, не виконуються у вигляді закінчених ІС. Вони легко реалізуються на основі інтегральних типових вузлів: ЦАП, компараторів, лічильників і т. д.

8.2.2. ПНК слідкуючого типу.

ПНК послідовного рахунку зі ступінчастою розгортаючою напругою легко перетворюється в ПНК слідкуючого типу. Для цього необхідно замінити сумуючий лічильник на реверсивний і використовувати прямий і інверсний виходи компаратора для управління режимом рахунку. Функціональна схема слідкуючого ПНК наведена на рис.8.4. На входи компаратора подається вхідна напруга $U_{вх}$ і напруга з виходу ПНК $U_{пкн}$. У залежності від знаку різниці між $U_{вх}$ і $U_{пкн}$ імпульси еталонної частоти надходять на сумуючий або віднімаючий вхід лічильника. Якщо $U_{вх} > U_{пкн}$ вміст лічильника буде збільшуватися, а при $U_{вх} < U_{пкн}$ - зменшуватися. При досягненні рівності вхідної напруги і напруги зворотного зв'язку з виходу ПНК в стежачій системі встановлюється динамічна рівновага і вихідний код ПНК буде коливатися навколо, відповідного вхідній напрузі, середнього значення з точністю до одиниці молодшого розряду, (рис.8.4,б.) Якщо в цьому стані вхідна напруга почне змінюватися, то вихідний код перетворювача буде відслідковувати його з похибкою, що не перевищує одиниці молодшого розряду,



а)



б)

Рис.8.4. ПНК слідкуючого типу

За умови виконання зазначеного співвідношення час перетворення слідкуючого ПНК дорівнює одному тактовому інтервалу T_L .

Недоліком найпростішого слідкуючого ПНК (рис.8.4) є великий час початкового узгодження значень $U_{пкн}$ і $U_{вх}$, тому що в цей час $U_{пкн}$ змінюється від нуля до $U_{вх}$, а перетворювач працює як ПНК послідовного рахунку зі ступінчастою розгортаючою напругою (рис.8.3).

8.3. Перетворювачі напруги в код порозрядного кодування

ПНК порозрядного кодування є одним з найбільш часто використовуваних перетворювачів. Його робота заснована на виконанні послідовних порівнянь вхідної напруги з набором з n еталонів, значення яких пропорційні ступені числа 2.

Цей процес аналогічний процесові зважування, в якому невідома вхідна величина врівноважується опорною сформованою з набору еталонів.

Урівноваження починається з еталона з максимальним значенням $U_e / 2^1$, де $U_{em} = U_{вх.макс}$ - максимально можлива вхідна напруга ПНК.

Якщо $U_{вх} < U_e / 2^1$, то в старшому розряді вихідного коду перетворювача фіксується нуль і далі проводиться урівноваження вхідної величини $U_{вх}$ наступним за значенням еталонів $U_e / 2^2$. Якщо ж $U_{вх} > U_e / 2^1$, то в старшому розряді вихідного коду фіксується одиниця, і далі вхідна напруга врівноважується сумою еталонів $U_e / 2^1 + U_e / 2^2$. Аналогічні операції виконуються при визначенні значень всіх n розрядів вихідного коду.

Схеми ПНК, що реалізують алгоритм порозрядного кодування, діляться на дві групи: виконані у вигляді замкнутих систем з ПКН в колі зворотного зв'язку; добудовані у вигляді розімкнутих систем. У інтегральному виконанні, як правило реалізуються ПНК першої групи. Схема 4-розрядного ПНК з ПКН в колі зворотного зв'язку наведена на рис.8.5.

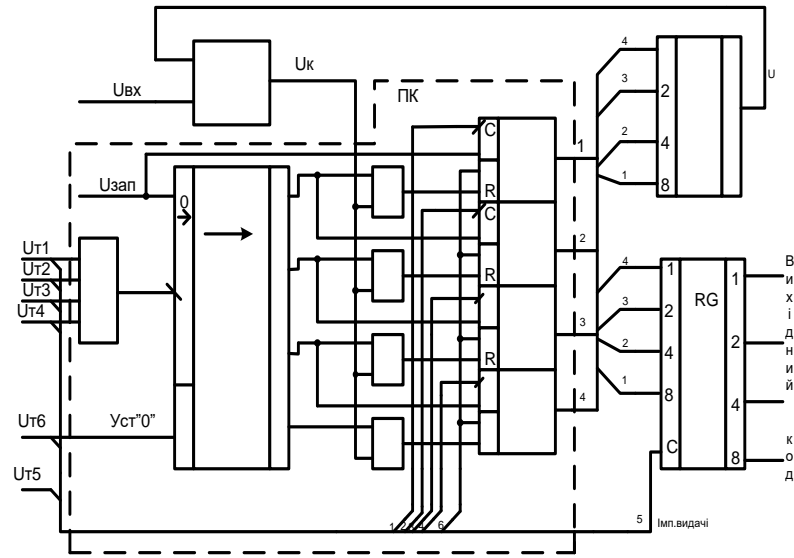


Рис.8.5. Схема 4-розрядного ПНК з ПКН в колі зворотного зв'язку

До складу ПНК входять: компаратор, ПКН, вихідний регістр і пристрій керування (ПК). До пристрою керування відносяться: зрушуючий регістр, регістр пам'яті на D-тригерах і логічні елементи І і АБО. У інтегральній схемотехніці пристрій керування, що містить зазначені елементи, часто виконується у вигляді окремої ІС, яка називається регістром послідовних наблизень. Для забезпечення роботи перетворювача на ПК надходять такі імпульси: імпульс запуску перетворювача $U_{зап}$, тактові імпульси U_{T1} , ..., U_{T4} , імпульс видачі коду U_{T5} і імпульс установки регістрів в нульове стан U_{T6} (Уст. «0»). Тимчасові діаграми, що пояснюють функціонування ПНК, та тимчасове розташування тактових імпульсів, наведені на рис.8.6.

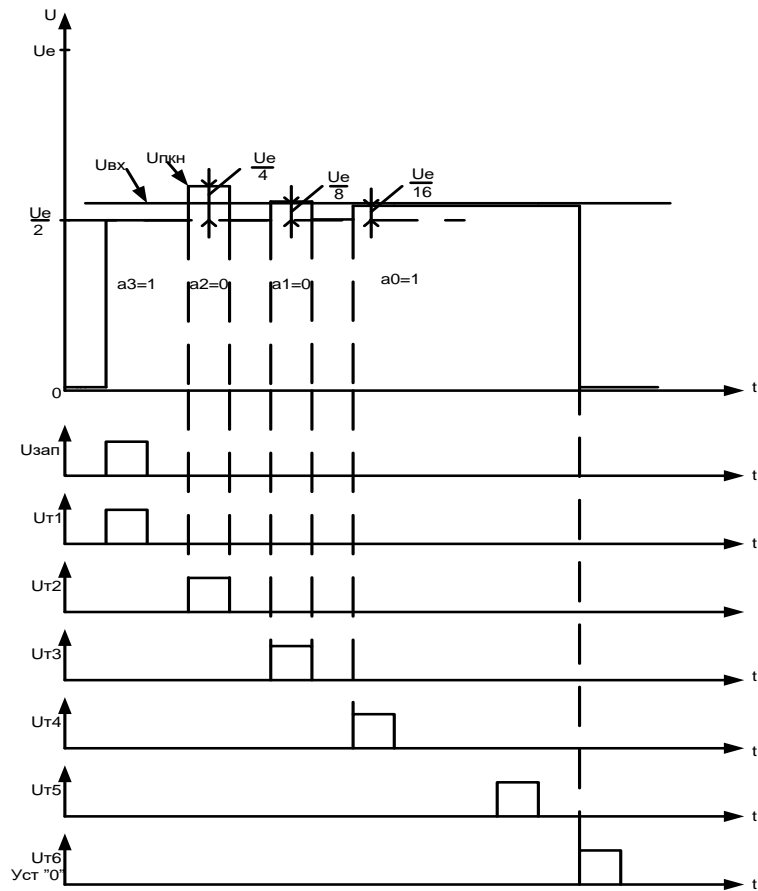


Рис.8.6. Часові діаграми ПНК з ПКН в колі зворотного зв'язку

Цикл перетворення починається з надходження імпульсу запуску $U_{зап}$ і збіжного з ним першого тактового імпульсу U_{T1} . По фронту імпульсу U_{T1} в тригер старшого розряду регістра пам'яті ПК записується одиниця, яка надходить на вхід старшого розряду ПКН.

На виході ПКН виробляється напруга $U_{ПКН} - U_e / 2$. У компараторі вона порівнюється з вхідною напругою $U_{вх}$ і якщо $U_{вх} > U_e / 2$ то вихідна напруга компаратора $U_K = U^0$.

По зрізу тактового імпульсу U_{T1} в перший розряд зрушуючого регістра записується логічна одиниця. Вона надходить на вхід першого логічного елемента І, на другий вхід якого прикладена напруга логічного нуля з виходу компаратора ($U_{до} = U^0$). При цьому вихідна напруга логічного елемента І рівна U^0 , і в тригері старшого значущого розряду (СЗР) регістра пам'яті фіксується логічна одиниця (значення старшого розряду коду $a_3 = 1$). Якщо $U_{вх} < U_e / 2$ то $U_K = U^1$ і після закінчення тактового імпульсу U_{T1} на виході першого логічного елемента І з'явиться логічна одиниця, яка встановить тригер старшого розряду регістра пам'яті в нульовий стан і фіксує нульове значення старшого значущого розряду вихідного коду ($a_3 = 0$).

По фронту наступного тактового імпульсу U_{T2} логічна одиниця записується в другій тригер регістра пам'яті.

При цьому вихідна напруга ПКН
$$U_{пнк} = a_3 \frac{U_e}{2} + \frac{U_e}{4},$$
 де $a_3 = 1$ при $U_{вх} > U_e / 2$, $a_3 = 0$ при $U_{вх} < U_e / 2$

Компаратор порівнює цю напругу з $U_{вх}$. Якщо $U_{вх} > U_{пнк}$, то в другому тригері регістра пам'яті фіксується логічна одиниця, тобто $a_2 = 1$. При $U_{вх} < U_{пнк}$ тригер встановлюється в нульовий стан ($a_2 = 0$). У наступних двох тактах перетворення (по U_{T3} і U_{T4}) аналогічно визначаються значення вихідного коду a_1 і a_0 . При цьому вхідна напруга послідовно зрівноважується напругою з виходу ПКН з похибкою, що не перевищує кроку квантування $U_e / 16$. Цикл перетворення закінчується, і вихідний код $a_3 a_2 a_1 a_0$, що відповідний вхідній напрузі $U_{вх}$ по імпульсу видачі U_{T5} видається споживачеві. Перед початком нового циклу перетворення зрушуючий регістр і регістр пам'яті імпульсом U_{T6} (Уст. «0») встановлюються в нульовий стан. Перетворювачі, побудовані за методом порозрядного кодування з ПКН в колі зворотного зв'язку, є найбільш поширеним типом ПНК.

Серійно випускаються спеціалізовані ІС - перетворювачі напруги в код порозрядного кодування К572ПВ1, К108ПВ1, КШЗПВ1 та інші. Наприклад, ІС К572ПВ1 представляє собою ЦАП із схемою управління і логічним пристроєм, в склад якого входить регістр послідовних наближень. При підключенні компаратора ІС К572ПВ1 може виконувати функції 12 - розрядного ПНК порозрядного кодування.

8.4. АЦП паралельного типу

Усі розряди вихідного коду обчислюються одночасно (паралельно), тому, вони більш швидкі, ніж послідовні АЦП. Але при такому алгоритмі роботи, вимагається більша кількість компараторів ($2^n - 1$), що викликає чисто технологічні труднощі при великій кількості розрядів (наприклад, при 12-розрядному АЦП, необхідно 4095 компараторів).

Схема такого АЦП (рис.8.7) містить резистивний дільник з однакових резисторів, який ділить опорну напругу на $(2^n - 1)$ рівнів.

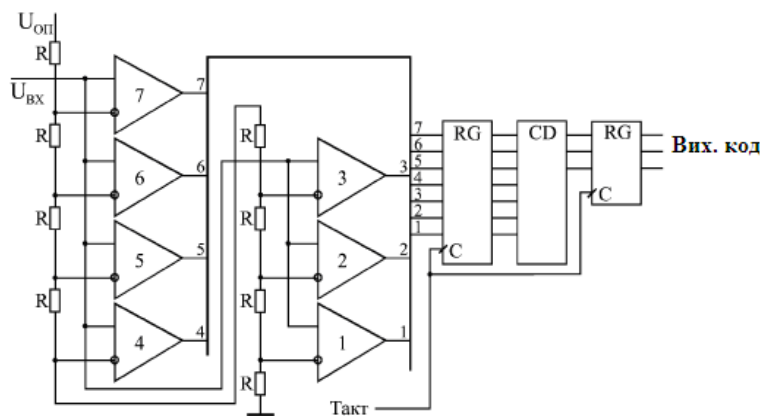


Рис. 8.7. АЦП паралельного типу

Вхідна напруга порівнюється за допомогою компараторів з рівнями, що формуються дільником напруги. Вихідні сигнали компаратора за допомогою шифратора перетворюються у n -розрядний двійковий код. Шифратор видає на вихід номер останнього з спрацювавших (які видали сигнал логічної одиниці) компараторів. Наприклад, у випадку 3-розрядного АЦП (рис.8.7) при величині вхідної напруги від 0 до $1/8$ значення опорної напруги, вихідний код буде 000, при вхідній напрузі від $1/8$ до $2/8$ значення опорної напруги, спрацює перший компаратор і схема сформує вихідний код 001, при вхідній напрузі від $2/8$ до $3/8$ опорної напруги, спрацюють компаратори 1 і 2, і вихідний код буде 010. Процес перетворення у паралельному АЦП відбувається дуже швидко, тому частота перетворення може досягати сотень мегагерц. Для збільшення швидкодії у паралельному АЦП іноді застосовують

конвеєрний принцип: вихідний код компараторів записується у $(2^n - 1)$ -розрядний паралельний регістр. Вихідний код шифратора також записується до n-розрядного паралельного регістру. При цьому обидва регістри тактуються одним тактовим сигналом. Це пом'якшує вимоги до швидкодії компараторів та шифратора. Але, вихідний код АЦП затримується при цьому на два періоди тактової частоти. Громіздкість структури паралельного АЦП призводить до того, що в деяких АЦП застосовується комбінований паралельно-послідовний принцип, який дещо зменшує швидкодію АЦП у порівнянні зі звичайним паралельним АЦП, але в той же час дозволяє більшу кількість розрядів, не збільшуючи кількість компараторів до $(2^n - 1)$.

Для того, щоб АЦП будь-якого типу працював з використанням усіх своїх можливостей, необхідно забезпечити узгодження діапазону зміни вхідного аналогового сигналу з допустимим діапазоном (динамічним діапазоном) вхідної напруги АЦП.

На рис.8.8 зображено чотири можливих випадки співвідношення динамічного діапазону АЦП (від 0 до або від U_{REF1} до U_{REF2}) і вхідного сигналу. У випадках а) і б) вхідний сигнал менше діапазону, тому, АЦП буде працювати правильно, але не буде використовувати усіх своїх можливостей. Випадок в) реалізується, коли амплітуда вхідного сигналу занадто велика, тому частина його значень не буде перетворена. Тільки у випадку г) АЦП буде працювати як n-розрядний і буде перетворювати усі значення вхідного сигналу. Для узгодження вхідного сигналу з динамічним діапазоном АЦП можна використовувати підсилювачі, атенуатор, схеми зсуву. В деяких випадках узгодження можна досягти простим підбором значення опорної напруги.

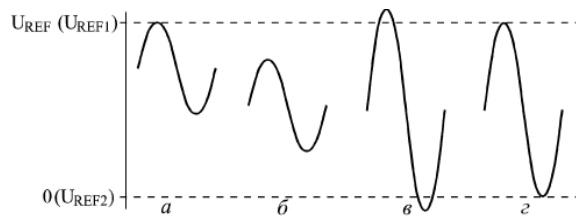


Рис. 8.8. Співвідношення вхідного сигналу та динамічного діапазону АЦП

При необхідності, можна зменшити кількість вихідних розрядів АЦП шляхом від'єднання потрібної кількості молодших розрядів виходу. На рис. 8.9 зображено використання 10-розрядного АЦП у якості 8-розрядного.

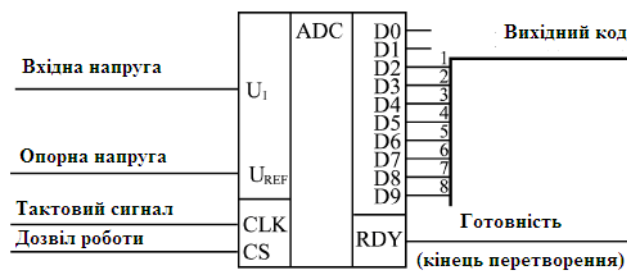


Рис. 8.9. Зменшення кількості розрядів вихідного коду АЦП

Обернена задача – збільшення розрядності АЦП, - виникає частіше. Існує багато схемотехнічних рішень по об'єднанню декількох мікросхем АЦП для збільшення кількості розрядів вихідного коду, але більшість з них вимагає складних розрахунків кінцевих похибок перетворення та застосування аналогових вузлів. Тому, ми розглядати їх не будемо. Зауважимо, що при виникненні необхідності збільшення розрядності, потрібно, перш за все, спробувати знайти мікросхему з відповідною розрядністю, і тільки після цього розглядати можливості об'єднання декількох АЦП.

Розглянемо декілька типових схем підключення АЦП, які використовуються у аналогово-цифрових схемах.

Перша схема (рис.8.10) розрахована на фіксацію моменту перевищення вхідним аналоговим сигналом заданого рівня порогової напруги. Схема формує вихідний сигнал (позитивний фронт) тоді, коли вхідний аналоговий сигнал перевищує встановлений рівень, який задається цифровим кодом порогу. Код порогу порівнюється з вихідними кодами АЦП за допомогою компаратора кодів. Вихідний сигнал компаратора кодів

записується у тригер за сигналом RDY з АЦП, що дозволяє виключити вплив коротких імпульсів, які виникають на виході компаратора під час зміни вхідних кодів. Застосування цього тригера затримує вихідний сигнал на один такт.

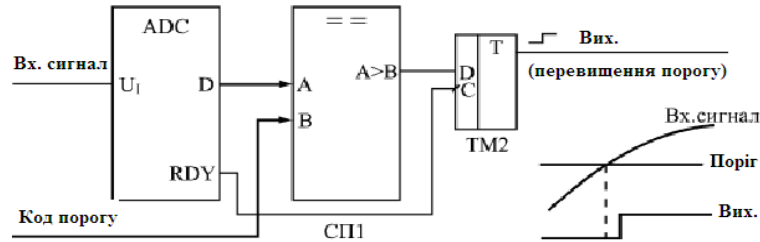


Рис. 8.10. Фіксатор перевищення вхідним сигналом встановленого рівня

Може здатися, що застосування АЦП у даному випадку не доцільне, але потрібно враховувати, що у аналогово-цифрових схемах АЦП, який перетворює вхідний сигнал у послідовність кодів, як правило вже є, тому, додаткового АЦП не потрібно, необхідно лише додати до схеми компаратор кодів та тригер.

АЦП також використовуються при розрахунку амплітуди вхідного аналогового сигналу. Для такого обчислення можна використати схему індикації екстремального значення вхідного коду. В якості джерела послідовності вхідних кодів, використано АЦП (рис. 8.11).

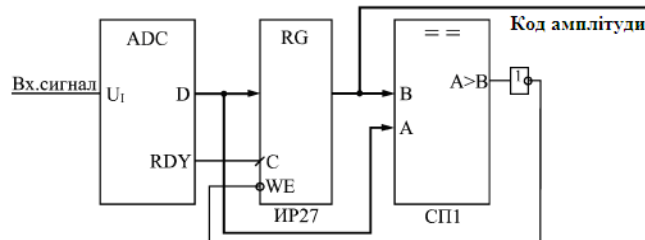


Рис. 8.11. Обчислювач амплітуди аналогового сигналу

У регістрі з входом дозволу запису, записується код з виходу АЦП за сигналом RDY у тому випадку, коли діюче значення коду більше значення, що було раніше збережено у регістрі. Зрозуміло, що вже після першого періоду вхідного сигналу, у регістрі буде код амплітуди вхідного сигналу. За період перетворення АЦП повинні встигнути спрацювати компаратор кодів і регістр.

Якщо такий обчислювач амплітуди вхідного сигналу використовується у складній аналогово-цифровій системі, де вже є АЦП, який непереривно перетворює вхідний сигнал у коди, то додатково знадобляться лише компаратор кодів та регістр.

Найбільш розповсюджене застосування АЦП – це перетворення вхідного сигналу у потік кодів, які потім записуються до буферної пам'яті. В такому випадку найбільш придатним є однонаправлений буфер з періодичним режимом роботи. Тобто, спочатку до буферної пам'яті вноситься масив кодів вибірок вхідного сигналу, а потім, цей масив зчитується для подальшої обробки (рис. 8.12). Саме так, наприклад, будується цифровий осцилограф, розрахований для візуального спостереження аналогових сигналів. У схемі на рис. 8.12, в якості стробу запису використовується сигнал RDY з АЦП. Реальні аналогово-цифрові пристрої набагато складніші, в них реалізовано синхронізацію процесу запису з вхідним сигналом, схеми попередньої обробки аналогового сигналу, але принцип залишається незмінним.

Наостанок, розглянемо схему на рис. 8.13, що дозволяє вдвічі збільшити швидкодію АЦП, точніше, вдвічі підвищити частоту запису кодів вибірок вхідного сигналу до буферної пам'яті.

Принцип роботи схеми досить простий: використовуються два АЦП і два буфера, які працюють по черзі, наприклад, парні вибірки вхідного сигналу обробляє один АЦП зі своїм буфером, а непарні – інший АЦП з власним буфером. Як наслідок, швидкість запам'ятовування кодів вхідного сигналу здійснюється з частотою, яка вдвічі більше частоти перетворення кожного з АЦП.

Тактові сигнали АЦП та сигнали RDY на виходах АЦП повинні бути зсунуті один відносно одного на половину періоду тактового сигналу. Читання зареєстрованих кодів з обох буферів також повинно виконуватися по черзі: перший код зчитується з першого буфера, другий – з другого, третій – знову з першого, і т.д. Об'єм обох буферів у даному випадку складається. Наприклад, при організації кожного буфера 64Кх8, результуючий буфер буде мати організацію 128Кх8.

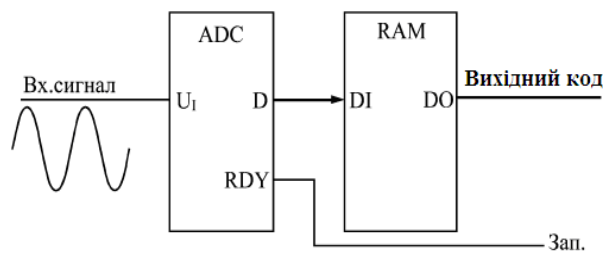


Рис. 8.12. Використання буферної пам'яті для запам'ятовування кодів з виходів АЦП

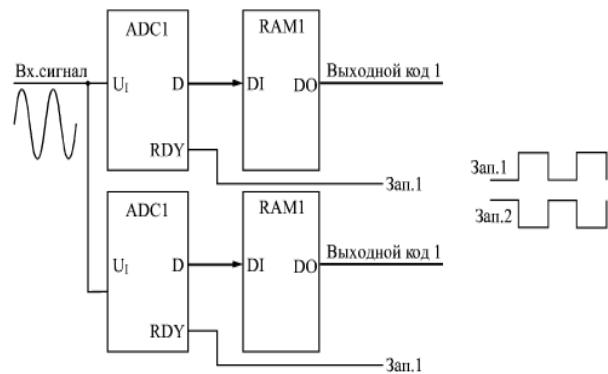


Рис. 8.13. Збільшення вдвічі частоти перетворення вхідного сигналу за допомогою двох АЦП з буферами

Використовуючи цей принцип, можна збільшити частоту обробки вхідного сигналу в три, чотири і більше разів. Необхідно лише узгодити у часі роботу відповідно трьох, чотирьох і т.д. АЦП, у кожного з яких повинна бути своя буферна пам'ять.

Крім розглянутих АЦП послідовного та паралельного типів, існує також АЦП з проміжним перетворенням. В ньому вхідний аналоговий сигнал за допомогою інтегратора перетворюється у часовий інтервал між цифровими імпульсами або в частоту слідування цифрових імпульсів. Вихідний цифровий код, що відповідає вхідному аналоговому сигналу формується в результаті вимірювання тривалості часового інтервалу або частоти слідування цифрових імпульсів (рис. 8.14).



Рис. 8.14. АЦП з проміжним перетворенням

Якщо використовується вихідна частота, то такий АЦП називають "перетворювачем напруга-частота" (ПНЧ). Такий підхід дозволяє за допомогою простих апаратних засобів отримати високу точність перетворення, яка не залежить від багатьох параметрів компонентів та від умов навколишнього середовища.

Вимірювання часових інтервалів і частоти слідування імпульсів виконується простими цифровими схемами. Такі виміри можуть проводитися з великою точністю, внаслідок того, що можна застосувати дуже стабільний часовий еталон – кварцовий генератор. Перевагою ПНЧ є також можливість простої передачі його вихідного сигналу на великі відстані.

На завершення розділу, слід зауважити, що вищенаведені схеми значно спрощені. Для їх практичної реалізації необхідне знання не лише цифрової техніки, але і аналогової та аналогово-цифрової схемотехніки, а також знання особливостей мікросхем ЦАП і АЦП.

Список питань для самоконтролю

1. Для чого призначені АЦП?
2. З чого складається загальна інструментальна похибка АЦП?
3. Що характеризує крок квантування?
4. Як класифікуються АЦП по виду алгоритму перетворення?
5. Принцип роботи ПНЧ з ступінчастою розгортаючою напругою?
6. Принцип роботи ПНЧ слідувального типу?
7. Принцип роботи АЦП паралельного типу?

РОЗДІЛ 9. ОСНОВИ МІКРОПРОЦЕСОРНОЇ ТЕХНІКИ

9.1 Класифікація процесорних пристроїв

Прийнята наступна класифікація процесорних пристроїв, реалізованих на одному кристалі:

- мікропроцесори загального призначення для числової обробки;
- мікроконтролери для простих систем керування/контролю;
- сигнальні процесори для цифрової обробки сигналів;
- програмовані логічні інтегральні схеми.

Дана класифікація в міру вдосконалення архітектури процесорного ядра й впровадження нових технологій перетерпіла істотні зміни. З урахуванням взаємного впливу архітектури мікропроцесорів різних типів надалі можливо будуть потрібні нові принципи класифікації мікропроцесорів.

Універсальні мікропроцесори призначені для використання в обчислювальних системах: персональних ЕОМ, робочих станціях, а в останній час і в масово-паралельних супер-ЕОМ. Основною їх характеристикою є наявність розвинутих приладів для ефективної реалізації операцій із плаваючою точкою над 64-розрядними операндами.

Цифрові сигнальні процесори розраховані на обробку в реальному часі цифрових потоків, утворених шляхом оцифрування аналогових сигналів. Це зумовлює їх порівняно малу розрядність і переважно цілочисельну обробку. Але сучасні сигнальні процесори здатні проводити обчислення із плаваючою точкою над 32-40 розрядними операндами.

Цифрові сигнальні процесори (DSP) — відносно нова категорія процесорів. Призначення DSP полягає в тому, щоб одержувати поточні дані від аналогової системи і формувати відповідний відгук. DSP і їх ALU (Arithmetic Logic Unit - арифметико-логічний пристрій, що є апаратним засобом для виконання обчислень) працюють з дуже високою швидкістю, що дозволяє здійснювати обробку даних у реальному масштабі часу. DSP часто використовуються в активних шумо-придушуючих мікрофонах, що встановлюються в літаках (другий мікрофон забезпечує сигнал навколишнього шуму, що віднімається із сигналу першого мікрофона, дозволяючи в такий спосіб придушити шум і залишити тільки голос) чи для придушення роздвоєння зображення в телевізійних сигналах.

Розробка DSP алгоритмів — це спеціальний розділ теорії управління. Викладання цієї теорії вимагає використання багатьох знань з математики і виходить за рамки даної книги (хоча пізніше ви познайомитеся з «нечіткою логікою», що являє собою нетрадиційний метод управління системами за допомогою комп'ютера).

DSP не призначені для автономного застосування. Звичайно вони входять до складу систем і використовуються як пристрої управління зовнішнім обладнанням, а також для обробки вхідних сигналів і формування відповідного відгуку.

Програмовані логічні інтегральні схеми – це матричні інтегральні схеми, що дозволяють програмно скомпонувати в одному корпусі електронну схему, еквівалентну схемі, що включає в себе від кількох десятків до кількох сотень інтегральних схем стандартної логіки. У порівнянні з іншими мікроелектронними технологіями, технологія програмованих логічних інтегральних схем забезпечує рекордно короткий проектно-технологічний цикл (від кількох годин до кількох днів), мінімальні затрати на проектування, максимальну гнучкість при необхідності модифікації апаратури.

Найбільшою спеціалізацією і різноманітністю функцій володіють мікроконтролери, які використовуються у вбудованих системах вимірювання та керування, у тому числі і в побутових приладах. Загальна кількість кристалів з різними системами команд перевищує 500, і всі вони, в силу існування виробів з їх використанням, мають свою стійку долю ринку.

Основне призначення вбудованих мікроконтролерів - забезпечити за допомогою недорогих засобів гнучке (програмувальне) управління об'єктами й зв'язок із зовнішніми пристроями. Ці мікроконтролери не призначені для реалізації комплексу складних функцій, але вони здатні забезпечити ефективне управління в багатьох областях застосування. Недорогими будемо вважати мікроконтролери, вартість яких становить від 1,0 до 20,0 доларів за штуку (ціна залежить від технічних характеристик, кількості виводів корпуса, обсягу закупок).

Вбудовані мікроконтролери містять значну кількість допоміжних пристроїв, завдяки чому забезпечується їхнє включення в реалізовану систему з використанням мінімальної кількості додаткових компонентів. До складу цих мікроконтролерів звичайно входять:

- Схема початкового запуску процесора (Reset)
- Генератор тактових імпульсів
- Центральний процесор
- Пам'ять програм (E(E)P)ROM і програмний інтерфейс
- Пам'ять даних RAM
- Засоби введення-виведення даних
- Таймери, що фіксують число командних циклів

Загальна структура мікроконтролера показана на рис. 9.1. Ця структура дає уявлення про те, як мікроконтролер зв'язується із зовнішнім світом.

Кристал мікроконтролеру

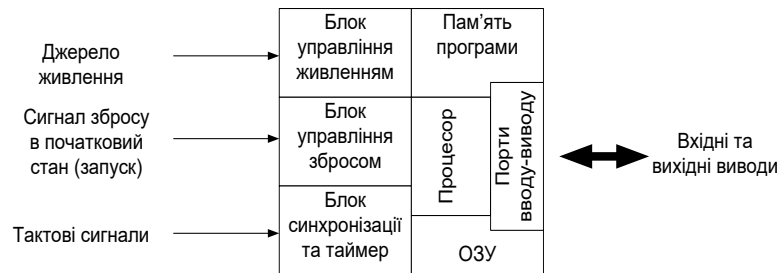


Рис. 9.1. Структура мікроконтролера

Більш складні мікроконтролери, що вбудовуються, можуть додатково реалізувати наступні можливості:

- Вбудований монітор/налагоджувач програм
- Внутрішні засоби програмування пам'яті програм (ROM)
- Обробка переривань від різних джерел
- Аналоговий введення-виведення
- Послідовний введення-виведення (синхронний і асинхронний)
- Паралельний введення-виведення (включаючи інтерфейс з комп'ютером)
- Підключення зовнішньої пам'яті (мікропроцесорний режим)

Усі ці можливості значно збільшують легкість застосування мікроконтролерів і роблять більш простим процес розробки систем на їхній основі. Слід відзначити, що для реалізації цих можливостей у більшості випадків потрібно розширення функцій зовнішніх виводів.

Дотепер десятки фірм, серед яких *Analog Devices*, *Atmel*, *Dallas Semiconductor*, *Oki*, *Philips*, *Infineon Technologies*, *Silicon Storage Technologies*, *Temic* і інші, продовжують випуск аналогів мікроконтролера 8051 (фірма Intel) — родоначальника всіх мікроконтролерів. У мікроконтролері 8051 реалізована CISC (Complex Instruction Set Computer) архітектура процесорного ядра, що оперує з повним набором інструкцій. У класичному мікроконтролері 8051 для виконання більшості інструкцій потрібно 1-2 машинних тактів. Фірма *Dallas Semiconductor* випускає аналог мікроконтролера 8051, у якому основні інструкції виконуються за чотири такти. В аналогах мікроконтролера 8051, що випускаються фірмами *Infineon* і *Philips Semiconductors*, для виконання основних інструкцій потрібно шість машинних тактів. Усього в мікроконтролері 8051 реалізовано 255 інструкцій. Фірмою *Intel* випускаються також мікроконтролери MCS151/251, цілком сумісні на рівні кодів інструкцій з мікроконтролером 8051. У 1994 р. фірма *Philips Semiconductors*, що випускає більш 60 модифікацій мікроконтролера 8051, створила на базі популярної 8-розрядної архітектури оригінальний 16-розрядний мікроконтролер 8051XA, сумісний на рівні кодів інструкцій з мікроконтролером 8051. Мікроконтролер 8051XA може працювати в двох режимах: розширеному й у режимі сумісності. У розширеному режимі використовуються нові можливості мікроконтролера, у тому числі й ефективні інструкції для мультизадачної обробки. У мікроконтролері 8051XA більшість інструкцій типу реєстр-реєстр виконується за три машинних такти (100 нс) при тактовій частоті 30 МГц. Усього в мікроконтролері 8051XA реалізовано 479 інструкцій. У таблиці 9.1 наведено основні параметри найпоширеніших 8-розрядних мікроконтролерів.

Одна з проблем, що існують при використанні повного набору інструкцій (CISC-архітектура), складається в складності реалізації компілятора. Застосування повного набору інструкцій, не формованих компілятором, не має змісту. Переважним є використання скороченого набору інструкцій. У результаті пошуку найбільш оптимальної архітектури для числової обробки, з'явилася архітектура процесорного ядра зі скороченим набором інструкцій — RISC (Reduced Instruction Set Computer) архітектура. Ідея RISC архітектури народилася в розроблювачів великих ЕОМ. Вони встановили, що різні види даних мають різну частоту появи. Найбільшу частоту появи (більш 50 %) мають локальні дані, а на виклик процедур приходиться не менш 30 % усього часу обробки. З іншого боку, однією з типових структур даних при числовій обробці є векторні дані. Поняття вектор, часто застосовуване в обчислювальній техніці, відрізняється від поняття вектора в математиці й фізиці.

Вектор — це впорядкований список даних. Число елементів списку — довжина вектора. Ідеальний випадок, якщо весь список можна зберігати в реєстровому файлі процесорного ядра (тобто кількість реєстрів дорівнює довжині вектора). У такому випадку при звертанні до даних, що містяться в реєстровому файлі, можна досягти максимальної швидкодії. Реєстровий файл чи векторні реєстри можна розглядати як буферну пам'ять, розташовану між операційним блоком процесорного ядра й основною пам'яттю. Крім того, вкрай вигідно мати можливість за допомогою однієї інструкції виконувати операції над безліччю даних, що можуть бути розташовані в пам'яті чи реєстровому файлі. Подібні інструкції дозволяють реалізувати ортогональну (симетричну) структуру, що дає

можливість виконувати кожну операцію з будь-яким регістром, використовуючи при цьому будь-який набір команд. Скорочення набору інструкцій дозволяє зменшити площу, займану процесорним ядром на кристалі, а отже, припускає розширення площі для внутрішніх регістрів (реєстрового файлу). У мікропроцесорах з скороченим набором інструкцій підвищення ефективності досягається також за рахунок застосування компілятора для безпосередньої трансляції програм, написаних на мовах високого рівня. В даний час багато фірм випускають 8- і 16-розрядні мікроконтролери з RISC-подібною архітектурою. Це фірми *Atmel* (серія AVR), *Texas Instruments* (серія MSP430), *Infineon* (серія 166) і ін.

Таблиця 9.1 - Основні параметри мікроконтролерів

Фірма, тип	Тактова частота, МГц	Вбудована пам'ять		Напруга живлення, В	Таймери	АЦП, ЦАП	Послідовний інтерфейс
		OTP/ROM/Flash/EEPROM	RAM				
Analog Devices AduC824	12...16	Flash (8 KB) EEPROM (640 B)	256 B	3/5	3 станд. 8051 (16-розр.)	24- і 16-розр., ЦАП	UART, I2C, SPI
AduC812						12-розр., ЦАП	
Atmel AT89	0...33	Flash (1..32 KB) EEPROM (128..512 B)	128...512 B	2.7...6	1...3 (16-розр.)		UART, SPI
AT90	0...12	Flash (1..8 KB) EEPROM (0..512 B)	0...512 B	2.7...6	1..4 (8- і 16-розр.)	10-розр.	UART, SPI
ATtiny	0...8	Flash (1..2 KB) EEPROM (0..128 B)	0...128 B	1.8...5.5	1...2 (8- і 16-розр.)	10-розр.	UART, SPI
ATmega	0...6	Flash (16..128 KB) EEPROM (512...4 KB)	1...4 KB	2.7...5.5	3..4 (8- і 16-розр.)	10-розр.	UART, SPI
Cybernetic Micro Systems P-51	1...60	-	8 KB програм і 4 KB даних	3.3/5	3 (16-розр.)	-	UART
<u>Infineon</u> SABC500	0...40	ROM (8...64 KB) OTP (8...64 KB)	256...3328 B	4.5...5.5	3...5 (16-розр.)	10-розр.	1...2 (UART, USART)
<u>Philips</u> 87C51Fx	33	OTP (8...32 KB)	256 B	2.7...5.5	3 (16-розр.)	-	UART
Rx2	20	Flash (16..64 KB)	512...1024 B	5	3 (16-розр.)	-	UART
<u>Hitachi</u> H8/3664	10	Flash (32 KB)	2 KB	2.5...5.5	3 (8- і 16-розр.)	10-розр.	UART, I2C, SPI
H8/3802	10	OTP (16 KB)	1 KB	2.5...5.5	5 (8- і 16-розр.)	8-розр.	UART, SPI

9.2. Мікропроцесори у вимірвальних приладах

Загальні відомості. Як уже відзначалося, для сучасного етапу розвитку техніки характерно усе більш інтенсивне і глибоке проникнення в її різні галузі мікропроцесорів, радикально перетворюючи властивості багатьох пристроїв і відкриваючи нові можливості їхнього застосування. За широтою й ефективністю застосування мікропроцесорів одне з перших місць займає контрольно-вимірвальна техніка.

Природно виникає питання: «Що дає застосування мікропроцесорів у вимірвальних приладах, як ефективні схемні рішення, в основі яких лежить мікропроцесорна система.»

У загальному плані відповіддю можуть служити заголовки ряду журнальних статей і інформаційних матеріалів: «Мікропроцесор робить революцію в електронному приладобудуванні». І це дійсно так. Але навіть якщо стримати патетичні інтонації, і перейти до звичайної технічної мови, то варто сказати, що застосування мікропроцесорів у вимірвальній техніці дозволяє різко підвищити точність приладів, значно розширити їхні можливості, підвищити надійність, швидкодію, вирішити задачі, що раніше взагалі не вирішувалися.

Конкретний розгляд функцій мікропроцесорних систем у вимірювальних приладах показує, що за допомогою цих систем досягаються багатофункціональність приладів, спрощення керування процесом вимірювання, автоматизація регулювання, самокалібрування й автоматична перевірка, поліпшення метрологічних характеристик приладу, виконання обчислювальних процедур, статистична обробка результатів спостережень, визначення і переклад у лінійну форму функції вимірюваної фізичної величини, створення програмувальних, цілком автоматизованих приладів. З'явився новий клас «інтелектуальних» приладів, які також називають «думаючими» чи «розумними».

Радикально змінилася ідеологія побудови приладів. МП став основною частиною приладу, що привело до зміни конструкції і схемних рішень, компоновання, керування, включенню обробки даних у вимірювальну процедуру (виконувану без участі експериментатора). Впровадження МП відкрило можливість побудови багатофункціональних приладів із гнучкими програмами роботи, зробило прилади більш економічними, полегшило вирішення задачі виходу на стандартну інтерфейсну шину (канал загального користування) і керування інтерфейсом. Усе це спростило експлуатацію приладів, різко підвищило продуктивність праці їхніх користувачів.

Розглянемо більш докладно основні можливості, особливості приладів, що містять мікропроцесорні системи, і з'ясуємо, в результаті чого досягаються ці можливості.

Багатофункціональність. Ідея побудови багатофункціональних вимірювальних приладів, призначених для виміру декількох параметрів сигналів чи характеристик об'єкта дослідження, не нова. Вона здійснюється вже більш трьох десятиліть. Але до застосування МП багатофункціональні прилади являли собою сукупність декількох функціональних вузлів, об'єднаних в одне конструктивне ціле. При експлуатації таких приладів перехід від однієї функції до іншої виробляється за допомогою комутуючих пристроїв. У результаті комутації сполучних ланцюгів користувач складає, «збирає» з окремих вузлів визначений прилад для виміру конкретного параметра сигналу чи характеристики випробуваного об'єкта. Алгоритм роботи засобу вимірювання, закладений при його розробці, у процесі експлуатації зберігається незмінним. Інакше кажучи, *традиційні* багатофункціональні прилади виконані за схемою з *твердою логікою*. Для неї характерне протиріччя між багатофункціональністю, числом можливих функцій приладу, з одного боку, і економічністю, а також технічною ефективністю — з іншої. Проблема комутації і керування ніколи не втрачала гостроти при конструюванні приладів, призначених для виконання ряду функцій, і далеко не завжди вирішувалась успішно.

Мікропроцесорна система, введена до складу багатофункціонального засобу вимірювання, радикально змінила його, перетворила пристрій із твердою логікою роботи в *програмно-керований пристрій*. Функціональні можливості такого пристрою визначаються виконуваною програмою, і можуть бути легко видозмінені шляхом переходу до іншої програми, збереженої в постійній пам'яті. Тому програмувальну логіку роботи подібних приладів іноді називають «збереженою». Вона створює гнучкість перебудови, дозволяє нарощувати функції при модернізації приладу без істотних змін у його схемі. Застосування програмувальної логіки, як правило, зменшує вартість приладу.

Підвищення точності приладів. Нагадаємо, що під точністю засобу вимірювань розуміють якість засобу вимірювань, що відбиває близькість до нуля його похибок. При цьому близькість до нуля систематичних похибок визначає правильність засобу вимірювань, а близькість до нуля випадкових похибок — збіжність показань засобу вимірювань.

Похибки засобу вимірювань відносяться до його метрологічних характеристик. Можна перерахувати наступні шляхи підвищення точності вимірювального приладу, які досягаються в результаті введення мікропроцесорної системи до складу приладу:

- автоматична компенсація (виключення) систематичної похибки, зокрема, автоматична установка нуля перед початком вимірів;
- автоматичне виконання градуувальної операції (самокалібрування);
- виконання самоконтролю;
- зменшення впливу випадкових похибок шляхом проведення багаторазових спостережень (одиничних вимірів) з наступним усередненням їхніх результатів;
- виявлення і виключення грубих похибок;
- виведення на дисплей інформації про числові значення похибок по ходу вимірів.

Розширення вимірювальних можливостей приладів. Застосування МП дозволяє істотно розширити можливості вимірів широкого переліку параметрів сигналів і характеристик пристроїв. Це зв'язано насамперед з використанням, здавалося б застарілих, видів вимірів: непрямих і сукупних.

Через необхідність застосування декількох приладів, зняття ряду відліків і наступних обчислень непрямі виміри сприймаються як примітивні, несучасні. Навіть при використанні мікрокалькуляторів обчислення в деяких випадках можуть забирати значний час, і, головне, вони, вимагаючи постійної уваги і роботи, не дозволяють досягти високої продуктивності. Крім того, не дуже проста процедура оцінки похибок непрямих вимірів, а без цього ніяке вимірювання не може бути визнано достовірним.

Докорінно змінюється положення при включенні до складу приладу мікропроцесорної системи. По команді, одержуваної з клавіатури, вона автоматично відповідно до заданої програми вибирає режими вимірювань,

запам'ятовує результати прямих вимірювань, проводить необхідні обчислення і видає знайдене значення вимірюваної фізичної величини на дисплей. Хоча вимірювання за своєю природою залишаються непрямими, експериментатор сприймає їхній як прямі, оскільки, підключивши прилад об'єкту дослідження, безпосередньо одержує результат виміру.

Прикладом може служити вимір цифровим вольтметром потужності P , що розсіюється на навантажувальному резисторі. Вимірювання здійснюються згідно формулі $P = U^2 / R$, де U — спадання напруги на резисторі; R — опір резистора. Цифровому вольтметру задається програма, відповідно до якої спочатку вимірюється опір резистора і запам'ятовується отриманий результат, потім вимірюється напруга на резисторі, після чого обчислюється потужність.

Як інші приклади, можна привести вимірювання опорів резисторів на основі формули закону Ома $R = U / I$ та вимірювання коефіцієнта підсилення підсилювача відповідно до визначення $K = U_{\text{вд}} / U_{\text{вх}}$.

Наведені приклади відносяться до порівняно простих математичних співвідношень. Але на практиці нерідко виникає необхідність знаходження непрямим шляхом значень і таких фізичних величин, що залежать від великого числа безпосередньо вимірюваних інших фізичних величин. Застосування мікропроцесорних систем робить ці виміри простими і зручними для користувача, що одержує прямі показання приладу і не відчуває того, що фактично виконуються складні непрямі виміри.

Ще більш ефективні мікропроцесорні системи при сукупних вимірах, тобто одночасних вимірюваннях декількох однойменних фізичних величин, при яких шукані значення величин знаходять вирішенням системи рівнянь, одержуваних при прямих вимірах різних сполучень цих величин.

Спрощення і полегшення керування приладом. На перший погляд розширення функцій, які виконуються програмними приладами, повинно було б привести до збільшення числа органів керування. Але в дійсності це не так. Одним із критеріїв високого рівня програмного забезпечення вимірювального приладу є ступінь складності його передньої панелі.

Прийнято вважати, що «розумний» прилад повинен мати простий набір органів керування. Для сучасних приладів, що містять МП, характерна кнопкова система керування, конструктивно оформлювана у виді клавіатур (виносної чи на передній панелі приладу), що зовні нагадує клавіатуру калькулятора.

Так, наприклад, в одного з цифрових мультиметрів, що володіє багатьма функціональними можливостями, керування зміною функцій, діапазонів вимірів і режимів роботи (всього 44 сполучення) здійснюється за допомогою клавіатури, що складається з 17 клавіш. Цього вдалося досягти внаслідок того, що кожна клавіша керує аналоговими схемами непрямим чином — через мікропроцесор, а останній селекує різні сполучення сигналів, що вводяться при натисканні клавіш.

Іншим прикладом спрощення керування, скорочення числа ручок і кнопок може служити малогабаритний 7-розрядний цифровий частотомір, що працює в діапазоні 10 Гц ... 1 ГГц. На передній панелі цього приладу є тільки два вхідних контакти (для сигналів частотою 10 ... 75 МГц, другий — для сигналів частотою 70 МГц ... 1 ГГц), кнопка включення приладу, ручка регулятора чутливості і двопозиційний кнопковий перемикач тривалості тимчасових воріт («часу виміру»): 1 с і 1 мс.

Радикально зменшує число органів керування автоматизація вибору меж вимірів, інтервалу дискретизації напруги досліджуваного сигналу й інших режимів роботи приладу. У деяких приладах передбачена сигналізація про некоректні кроки експериментатора й видача на дисплей інструкцій, що вказують, що повинний робити експериментатор, яка правильна послідовність дій.

Можливість одержання математичних функцій вимірюваних значень. У залежності від розв'язуваної задачі експериментатора можуть цікавити не безпосередньо одержуване при вимірі значення фізичної величини, а його різні математичні функції. Багато приладів, що містять мікропроцесорні системи, дозволяють автоматично виконувати запрограмовані функціональні перетворення. Прикладами таких перетворень можуть служити:

1. Множення знайденого значення A на константу c . При цьому показання приладу $A_n = cA$. Константа вводиться по команді при натисканні клавіші.

2. Одержання відхилень результату виміру A від номінального значення A_H : абсолютного $A - A_H$ і відносного, вираженого у відсотках стосовно номінального значення, тобто $100(A - A_H) / A_H$.

3. Зсув, що припускає вирахування константи з результату виміру.

4. Обчислення відношень: ділення на константу (наприклад, при визначенні значення постійного струму через резистор по вимірюваному вольтметром значенню спадання напруги на цьому резисторі), знаходження частки від ділення одного результату виміру на інший результат виміру (наприклад, при визначенні коефіцієнта підсилення за результатами вимірів напруг на виході і вході підсилювача).

5. Представлення результату виміру в логарифмічних одиницях. Наприклад, загасання чотириполосника, вираженого в децибелах: $a = 20 \lg(U_{\text{вх}} / U_{\text{вих}})$.

6. Лінеаризація залежностей. Така необхідність особливо часто зустрічається при електричних вимірах неелектричних величин (наприклад, температури), коли напруга електричного сигналу на виході датчика являє собою нелінійну функцію вимірюваної фізичної величини на його вході. У таких ситуаціях значення вихідної напруги датчика перетворюються за допомогою АЦП у числа, що обробляються мікропроцесорною системою за заданою програмою, і в підсумку, одержується лінійний зв'язок між показами приладу і значеннями фізичної величини на вході датчика.

У деяких приладах передбачена можливість обчислення за бажанням користувача довільних (зрозуміло, у визначених межах) математичних співвідношень.

Одержання статистичних характеристик. Ряд вольтметрів, у складі яких є мікропроцесорна система, дозволяють формувати оцінки таких імовірнісних характеристик аналізованої випадкової змінної, як середнє значення, середня потужність, середньоквадратичне значення, дисперсія, середньоквадратичне відхилення, а також коефіцієнт кореляції двох випадкових змінних. Мікропроцесорні прилади, спеціально призначені для виміру статистичних характеристик сигналів, мають більш широкі можливості.

Мініатюризація й економічність апаратури. Різке зменшення числа компонентів у схемі приладу внаслідок виконання багатьох функцій мікропроцесорною системою, їх відносно невисока вартість, значне зниження споживаної потужності дозволяють будувати малогабаритні та економічні прилади.

Підвищення надійності приладів. Воно зумовлено зменшенням числа елементів схем, здійсненням автодіагностики, застосуванням вузлів з некаліброваними характеристиками (наприклад, підсилювача в каналі вертикального відхилення осцилографа), можливістю виконання корекції похибок, що поліпшує метрологічну надійність.

Скорочення тривалості розробки. Часто для одержання нових властивостей приладу, виконуваного на основі мікропроцесорної системи, не потрібно значних змін у схемі і тим більше в конструкції приладу. Основний зміст розробки полягає в створенні необхідного програмного забезпечення. З огляду на те, що для широко застосовуваних МП уже накопичена бібліотека досить розроблених типових прикладних програм та вимірювальних процедур, у багатьох випадках розробка програмного забезпечення приладу в значній мірі зводиться до раціонального вибору наявних програм.

Організація вимірювальних систем. Прилад, що містить МП, звичайно оснащений інтерфейсами, що дозволяють підключати його до стандартної інтерфейсної шини. Це дає можливість поєднувати визначену сукупність приладів у єдину вимірювальну систему (вимірювально-обчислювальний комплекс).

9.3. Поліпшення метрологічних характеристик приладів

Розглянемо основні можливості і способи зменшення похибок у приладах, що містять мікропроцесорні системи.

Виключення систематичної похибки. Найбільш часто систематичні похибки обумовлені зсувом нуля, невідповідністю реального значення коефіцієнта передачі тракту сигналу номінальному значенню, нерівномірністю амплітудно-частотної характеристики тракту передачі сигналу, впливом характеристики аналого-цифрового перетворювача (АЦП).

Наявність у приладі мікропроцесорної системи дозволяє скорегувати, виключити систематичні похибки. Коротко освітимо шляхи розв'язку задачі. Для виключення зсуву нуля, наприклад у цифровому вольтметрі, його вхідні контакти замикаються накоротко і приєднуються до точки з нульовим потенціалом (заземлюються). При цьому число, одержуване на виході АЦП, характеризує зсув нуля. Воно запам'ятовується і віднімається з показань приладу. Надалі, коли вимірюється напруга, що підводиться до вхідних контактів приладу, автоматично вноситься виправлення, що усуває систематичну похибку, викликану зсувом нуля.

Принцип корекції систематичної похибки, пов'язаної з тим, що значення коефіцієнта передачі тракту сигналу (характеризуючи внесені ним посилення чи ослаблення) відрізняється від номінального, полягає в наступному:

У пам'яті мікропроцесорної системи зберігається число B , що не руйнується при відключенні живлення системи і відповідає строго визначеному значенню A_0 вхідної напруги, тобто число, що повинно бути отримане на виході АЦП, якщо на вхід вольтметра надходить напруга A_0 і коефіцієнт передачі тракту сигналу, а також коефіцієнт перетворення АЦП відповідають своїм номінальним значенням. У середині приладу є цифро-аналоговий перетворювач (ЦАП), що містить зразкове джерело живлення. При підведенні числа B до входів ЦАП на його виході утвориться напруга, значення якої дорівнює A_0 . Ця напруга подається на вхід приладу. У результаті аналого-цифрового перетворення виходить B' , що відрізняється від числа B через наявність систематичної похибки. Її характеризує відношення чисел $\alpha = B/B'$. Значення коефіцієнта α обчислює мікропроцесор, і воно фіксується в пам'яті. Таким чином, у пам'яті міститься поправочний множник.

Коли на вхід приладу надходить вимірювана напруга постійного струму, то на виході АЦП виходить число C' , що відповідає значенню цієї напруги. Введення поправочного множника, тобто виконуване мікропроцесором множення числа C' на коефіцієнт α , дає правильний результат виміру — число C .

Задача виключення систематичної похибки, яка обумовлена нерівномірністю АЧХ тракту передачі сигналу, особливо складна при використанні широкодіапазонних вольтметрів. Наявність мікропроцесорної системи в приладі істотно спрощує рішення цієї задачі.

На виносному вимірювальному пробнику (вимірювальній головці) закріплена табличка, що вказує масштабні коефіцієнти (поправочні множники) для сукупності частот, що входять у діапазон приладу 0...2 ГГц. Кожен масштабний коефіцієнт, що відповідає визначеній частоті f_i , може бути введений у прилад для виключення систематичної похибки, обумовленої відмінністю значення коефіцієнта передачі K_i від номінального значення K_0 . У пам'яті каліброваних коефіцієнтів зберігається значення $K_0 = 1000$. Це значення можна висвічувати на дисплеї приладу і змінювати за командою з клавіатури (чи інтерфейсної шини). Якщо експериментатор вводить значення K_i , відмінне від K_0 , то на дисплеї відображається фактичне (введене) значення коефіцієнта з метою залучення уваги експериментатора до факту зміни калібрування.

При вимірюванні напруги мікропроцесор виконує операцію множення на поправочний множник, тобто враховує введений масштабний коефіцієнт, враховуючи тим самим систематичну похибку.

Зменшення впливу випадкової похибки. Ця складова похибки вимірювання, як відомо, не може бути виключена. Її вплив можна зменшити раціональною обробкою результатів спостережень.

Для обліку випадкових похибок користуються ймовірнісними характеристиками. З теорії ймовірностей відомо, що найбільш повно випадкові величини характеризуються законами розподілу ймовірностей. Але при рішенні багатьох вимірювальних задач цілком достатніми характеристиками випадкових похибок служать їхні найпростіші числові характеристики: середнє значення (математичне очікування) і середньоквадратичне відхилення. Оскільки число N спостережень завжди обмежено, то реально користуються статистичними числовими характеристиками, які називаються оцінками характеристик.

Оцінку середнього значення результатів спостережень обчислюють за формулою

$$A_{cep} = \frac{1}{N} \sum_{i=1}^N A_i, \quad (9.1)$$

де A_i — результат i -го спостереження, не спотворений систематичною похибкою; N — число спостережень.

Оцінкою середньоквадратичного відхилення випадкової похибки результату спостережень служить вираз

$$\hat{\sigma}_v = \sqrt{\frac{1}{N-1} \sum_{i=1}^N v_i^2}, \quad (9.2)$$

де $v_i = A_i - A_{cep}$ — відхилення i -го результату спостереження від його середнього значення.

У теорії похибок доводиться, що середньоквадратичне відхилення результату виміру, що визначається як $\hat{\sigma}_A$, обчислене для N груп серії незалежних спостережень (в кожній групі по N спостережень), при великому числі N , набагато менше, ніж середньоквадратичне відхилення $\hat{\sigma}_v$. Розраховують $\hat{\sigma}_A$ за формулою

$$\sigma_A = \hat{\sigma}_v / \sqrt{N} = \sqrt{\frac{1}{N(N-1)} \sum_{i=1}^N v_i^2}. \quad (9.3)$$

Формула (9.2) визначає абсолютну похибку. Для знаходження відносної середньоквадратичної випадкової похибки δ_v значення $\hat{\sigma}_v$, обчислене за (9.2), відносять до A_{cep} .

З написаних виразів видно, що проведення багаторазових вимірів з наступним усередненням — ефективний спосіб зменшення впливу випадкової похибки на результат виміру.

Компенсація внутрішніх шумів. Ця операція дозволяє підвищити чутливість вимірювального приладу, розширити діапазон вимірюваних значень напруги в сторону малих значень. Принцип компенсації, використаний у вимірювачі рівня високочастотних сигналів, зводиться до наступного.

До складу приладу входить вимірювальний перетворювач, що здійснює перетворення високочастотної напруги змінного струму в напругу постійного струму, значення якої відповідає середньоквадратичному значенню напруги змінного струму. Ще до подачі досліджуваного сигналу $s(t)$ протягом інтервалу часу, який витрачається на автоматичне регулювання нуля, вимірюється середній квадрат шумового сигналу $n\{t\}$ на вході перетворювача.

Результат виміру $\overline{n^2(t)}$ запам'ятовується. Після підведення до входу приладу корисного сигналу на вході перетворювача виходить сума сигналу і шуму. В перетворювачі сумарний сигнал $s(t) + n(t)$ зводиться в квадрат, в результаті чого утворюється сигнал $s^2(t) + 2s(t)n(t) + n^2(t)$. Усереднення цього сигналу дає

$$\overline{s^2(t) + 2s(t)n(t) + n^2(t)} = \overline{s^2(t)} + \overline{n^2(t)}$$

(так як сигнали $s(t), n(t)$ незалежні, середнє значення їхнього добутку дорівнює нулю). З результату усереднення віднімається вимірний раніше середній квадрат $\overline{n^2(t)}$ шумового сигналу і отримана різниця дорівнює $\overline{s^2(t)}$. Витяг квадратного кореня дає середньоквадратичне значення “чистого” корисного сигналу $s(t)$, оскільки шумова складова скомпенсована.

9.4. Процесорні похибки вимірювань

Розглядання процедури процесорних вимірювань як послідовності виконання аналогових, аналого-цифрових і цифрових вимірювальних перетворень, дозволяє уявити похибку у вигляді суми відповідних компонент. Відмінність результатів цифрових вимірювальних перетворень від потрібних визначають процесорні похибки вимірювань. Для вивчення цього виду похибок виділяють три причини їх появи:

- алгоритмічна похибка, зумовлена відмінністю прийнятого алгоритму вимірювання від адекватного (гіпотетичного);
- похибка округлення, викликана наявністю округлень проміжних цифрових перетворень в результаті вимірювання;
- динамічна процесорна похибка, джерелом появи якої є обмеженість швидкодії мікропроцесора (процесора).

Розвиваючи ідеї класифікації, засновані на виділенні факторів, що зумовлюють появу похибок, введемо в розгляд поняття гіпотетичного алгоритму вимірювань, що дозволяє одержати істинне значення вимірюваної величини. В тому разі, коли його можна сформулювати, він відповідає значенню вимірюваної величини. Наприклад, застосовуючи закон Ома, можна інтерпретувати відповідні визначення сили струму I , напруги U і опору R , як гіпотетичні алгоритми вимірювань, виконання яких дозволяє визначити істинне значення I , U або R . Тоді виникнення алгоритмічної похибки буде зумовлено відмінністю прийнятого алгоритму вимірювання від гіпотетичного.

Застосування мікропроцесорів у вимірювальних приладах пов'язане з необхідністю оцінювати інструментальні похибки, які можуть виникнути при виконанні обчислювальних процедур, через обмеженість розрядної мережі процесора. Вони називаються похибками округлення.

При виборі алгоритмів обчислень, як правило, застосовується умова, щоб абсолютна похибка обчислень не перевищувала $\Delta_0 = 2^{-n}$, де n - розрядність МП.

Операції округлення в МП здійснюються, як правило, простим відсіканням невраховуваних розрядів або симетричним округленням з урахуванням значення старшого розряду, що відкидається. Якщо l - число неврахованих при округленні розрядів при рівномірному законі розподілу ймовірностей округлення (що справедливо в більшості випадків при $n \geq 8$), то дисперсії похибки простого відсікання і симетричного округлення (при порівняно невеликих l)

$$D_y \cong 2^{-2n} / 12 = \Delta_0^2 / 12. \quad (9.4)$$

Математичне сподівання інструментальної похибки округлення

$$M_y = 2^{-(n+l+1)}. \quad (9.5)$$

Похибки відсікання мають негативний знак для будь-яких арифметичних операцій над числами, поданими у прямому і додатковому кодах. Тому при значній кількості послідовних арифметичних операцій похибки можуть нагромаджуватись і перевищити припустимий рівень. Особливо важливо проводити оцінку цієї похибки при розрядній мережі мікропроцесорної системи, близької до розрядності аналого-цифрового перетворення.

У засобах вимірювання, що працюють під управлінням МП, суттєву роль відіграють часові затримки запуску аналого-цифрових перетворювачів порівняно з потрібною швидкістю вимірювань. Ці затримки значною мірою визначаються затратами процесорного часу на обробку переривань, управління, запам'ятовування, виконання інтерфейсних функцій. Тому при метрологічних випробуваннях мікропроцесорних засобів вимірювання, де похибка

датування відліку залежить від стану вимірювальних модулів і програмних драйверів, необхідно передбачати експериментальне визначення характеристик цієї похибки.

Вплив похибки датування відліку на загальну похибку мікропроцесорного засобу вимірювання Δ залежить від швидкості зміни вхідного сигналу dx/dt :

$$\Delta = \Delta_{МПЗВ} + \frac{dx}{dt} \delta_t, \quad (9.6)$$

де $\Delta_{МПЗВ}$ - абсолютна похибка мікропроцесорного засобу вимірювання; δ_t - похибка датування відліків.

Похибка δ_t може визначатись як різниця астрономічного і реального часу виконання вимірювань або як міра відхилення інтервалу між послідовними опитуваннями АЦП і заданого інтервалу, котрий має підтримуватись постійним.

При організації опитування аналого-цифрових перетворювачів їх швидкість вибирають так, щоб у межах припустимих похибок можна було між двома послідовними вимірюваннями використати лінійну інтерполяцію $dx/dt = const$. Однак при дослідженні швидких процесів, як правило, не можна задовольнити цю умову, що призводить до появи динамічної похибки.

9.5. Архітектура процесорів

Мікропроцесорна система (МПС) – це зібрана в єдине ціле сукупність взаємодіючих інтегральних схем цифрової логіки та аналогових ланцюгів, організована у обчислювальну або в керуючу систему з мікропроцесором (мікроконтролером), як вузлом обробки інформації.

Узагальнена структура мікропроцесорної системи наведена на рис. 9.2.

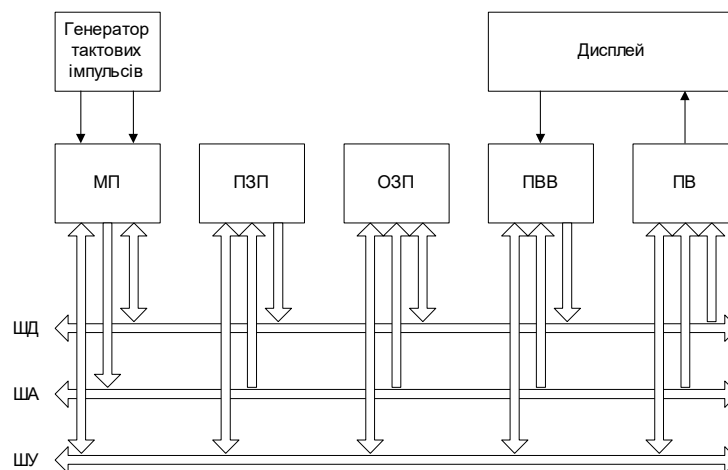


Рис. 9.2. Узагальнена структурна схема мікропроцесорної системи

Коротко охарактеризуємо основні елементи, що входять до її складу.

Генератор тактових імпульсів - джерело послідовності прямокутних імпульсів, за допомогою яких здійснюється управління роботою МП у часі. Для сучасних МП не потрібний зовнішній генератор тактових імпульсів: він міститься безпосередньо в його схемі.

Основна пам'ять системи (зовнішня щодо МП) складається з постійного (ПЗП) і оперативного (ОЗП) запам'ятовуючих пристроїв.

ПЗП – це пристрій, в якому зберігається програма та сукупність констант. Вміст ПЗП не стирається при відключенні живлення. ПЗП використовується як пам'ять програми.

ОЗП - це пам'ять програми і даних, що належать обробці і результатам обчислень.

Пристрій введення-виведення (ПВВ) здійснює введення в систему даних, що належать обробці. Пристрій виведення (ПВ) перетворює вихідні дані (результат обробки інформації) у форму, зручну для сприйняття користувачем або зберігання. ПВВ служать гнучкі магнітні диски, клавіатура, дисплей, аналого-цифрові і цифроаналогові перетворювачі, графобудівники, друкуючі пристрої тощо.

Далі розглянемо системи шин. Шиною називається група ліній передачі, що використовуються для виконання певної функції (по одній лінії на кожен передаючий біт). Особливістю структури МПС є магістральна організація

зв'язків між модулями, що входять у її систему. Вона здійснюється за допомогою трьох шин. Ці шини з'єднують МП із запам'ятовуючим пристроєм (ПЗП, ОЗП) і інтерфейсами введення-виведення, внаслідок чого створюється можливість обміну даними між розглянутими модулями системи.

Шина даних (ШД) - це двонаправлена шина: по ній дані можуть направлятися в МП або з нього. При цьому необхідно підкреслити, що одночасна передача даних в обох напрямках неможлива. Ці процедури рознесені в часі у результаті застосування часового мультиплексування.

По шині адреси (ША) інформація передається тільки в одному напрямі - від МП до модуля пам'яті або ПВВ.

Шина управління (ШУ) використовується для передачі сигналів, що обслуговують взаємодію, синхронізацію роботи всіх модулів системи і внутрішніх вузлів МП.

Перевагою шинної структури є можливість вмикання до МПС нових модулів, наприклад кількох блоків ОЗП і ПЗП, для одержання потрібного обсягу пам'яті.

Порівняння архітектур CISC та RISC. В даний час існує багато RISC процесорів, тому що склалася думка, що RISC швидше чим CISC процесори. Така думка не зовсім вірна. Існує багато процесорів, що називаються RISC, але насправді відносяться до CISC. Більш того, у деяких задачах CISC-процесори виконують програмний код швидше, ніж це роблять RISC-процесори, чи вирішують такі задачі, що RISC-процесори не можуть виконати.

Яка дійсна відмінність між RISC і CISC. CISC-процесори виконують великий набір команд з розвинутими можливостями адресації (безпосередня, індексна і т.д.) і дають розробнику можливість вибрати найбільш придатну команду для виконання необхідної операції. У RISC-процесорах набір виконуваних команд скорочений до мінімуму. При цьому розробник повинний комбінувати команди, щоб реалізувати більш складні операції. Нижче буде показано, як операції завантаження в стек («push») і вивантаження зі стеку («pop») реалізуються в RISC-процесорах за допомогою двох простих команд.

Можливість рівноправного використання всіх регістрів процесора називається «ортогональністю» чи «симетричністю» процесора. Це забезпечує додаткову гнучкість при виконанні деяких операцій. Розглянемо, наприклад, виконання умовних переходів у програмі. У CISC-процесорах умовний перехід звичайно реалізується відповідно до визначеного значення біта (прапорця) у регістрі стану. У RISC-процесорах умовний перехід може відбуватися при визначеному значенні біта, що знаходиться в будь-якому місці пам'яті. Це значно спрощує операції з прапорцями і виконання програм, що використовують ці прапорці.

Успіх при використанні RISC-процесорів забезпечується завдяки тому, що їх більш прості команди вимагають для виконання значно менше число машинних циклів. Таким чином, досягається істотне підвищення продуктивності, що дозволяє RISC-процесорам ефективно вирішувати надзвичайно складні задачі.

Порівняння Гарвардської та Принстонської архітектур. Багато років тому уряд Сполучених Штатів дав завдання Гарвардському і Принстонському університетам розробити архітектуру комп'ютера для військово-морської артилерії. Принстонський університет розробив комп'ютер, що мав загальну пам'ять для збереження програм і даних. Така архітектура комп'ютерів більше відома як архітектура Фон-Неймана за ім'ям наукового керівника цієї розробки (рис. 9.3).

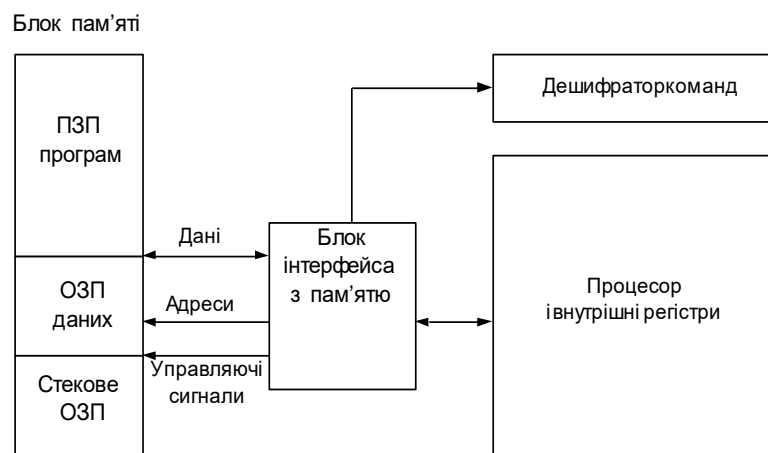


Рис. 9.3. Структура комп'ютера із Принстонською архітектурою

У цій архітектурі блок інтерфейсу з пам'яттю виконує арбітраж запитів до пам'яті, забезпечуючи вибірку команд, читання і запис даних, розташованих у пам'яті чи внутрішніх регістрах. Може показатися, що блок інтерфейсу є найбільше вузьким місцем між процесором і пам'яттю, тому що одночасно з даними потрібно вибирати з пам'яті чергову команду. Однак у багатьох процесорах із Принстонської архітектурою ця проблема вирішується шляхом вибірки наступної команди під час виконання попередньої. Така операція називається попередньою вибіркою («передвибіркою»), і вона реалізується в більшості процесорів з такою архітектурою.

Гарвардський університет представив розробку комп'ютера, у якому для збереження програм, даних і стеку використовувалися окремі банки пам'яті (рис. 9.4).

Прінстонська архітектура виграла змагання, тому що вона більше відповідала рівню технології того часу. Використання загальної пам'яті виявилось кращим через ненадійність лампової електроніки (це було до широкого поширення транзисторів) — при цьому виникало менше відмов.

Гарвардська архітектура майже не використовувалася до кінця 70-х років, доки виробники мікроконтролерів не зрозуміли, що ця архітектура дає переваги пристроям, що вони розробляли.

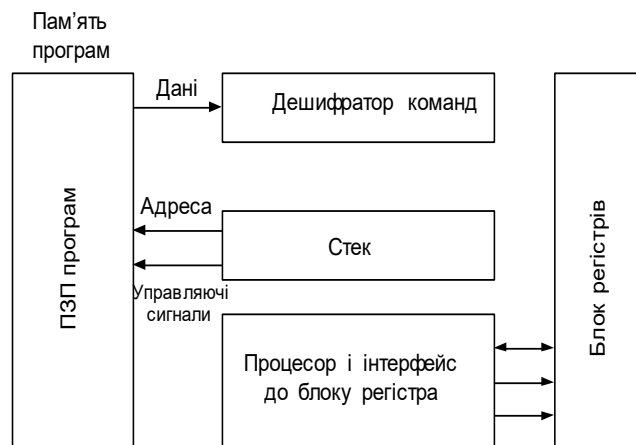


Рис. 9.4. Структура комп'ютера із Гарвардською архітектурою

Основною перевагою архітектури Фон-Неймана є те, що вона спрощує пристрій мікропроцесора, тому що реалізує звертання тільки до однієї загальної пам'яті.

Для мікропроцесорів найважливішим є те, що вміст ОЗП (оперативно запам'ятовуючий пристрій) (RAM - Random Access Memory) може бути використано, як для збереження даних, так і для збереження програм. У деяких додатках програми необхідно мати доступ до вмісту стека. Все це надає велику гнучкість для розробника програмного забезпечення, насамперед в області операційних систем реального часу, про які мова піде пізніше. Гарвардська архітектура виконує команди за меншу кількість тактів, чим архітектура Фон-Неймана. Це обумовлено тим, що в ній більше можливостей для реалізації паралельних операцій. Вибірка наступної команди може відбуватися одночасно з виконанням попередньої команди, і немає необхідності зупиняти процесор на час вибірки команди.

Наприклад, якщо процесорі з Прінстонською архітектурою необхідно зчитати байт і помістити його в акумулятор, то він робить послідовність дій показану на рис. 9.5. У першому циклі з пам'яті вибирається команда, у наступному циклі дані, що повинні бути поміщені в акумулятор, зчитуються з пам'яті.

Цикл 1: - читання (вибір) команди

Цикл 2: - читання даних та їх розміщення в акумуляторі

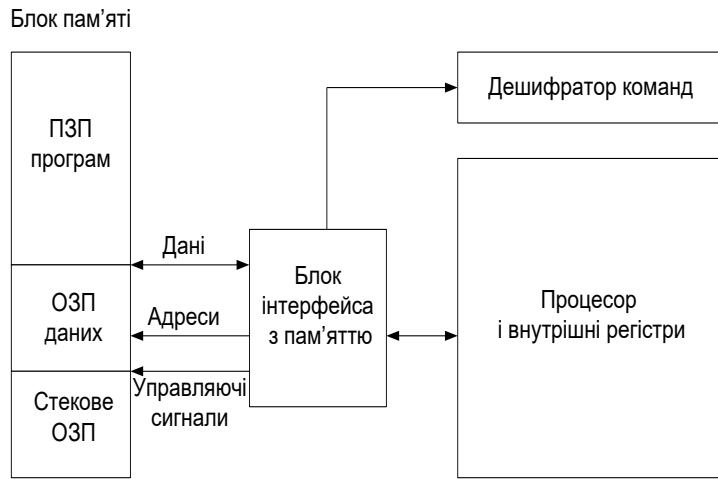


Рис.9.5. Виконання команди *mov Acc, Reg* у Принстонській архітектурі

У Гарвардській архітектурі, що забезпечує більш високий ступінь паралелізму операцій, виконання поточної операції може сполучатися з вибіркою наступної команди (рис 9.6). Команда також виконується за два цикли, але вибірка чергової команди відбувається одночасно з виконанням попередньої. Таким чином, команда виконується усього за один цикл (під час читання наступної команди).

Цей метод реалізації операцій («паралелізм») дозволяє командам виконуватися за однакове число тактів, що дає можливість більш просто визначити час виконання циклів і критичних ділянок програми. Ця обставина є особливо важливою при виборі мікроконтролера для задач, де потрібно забезпечення суворо заданого часу виконання.

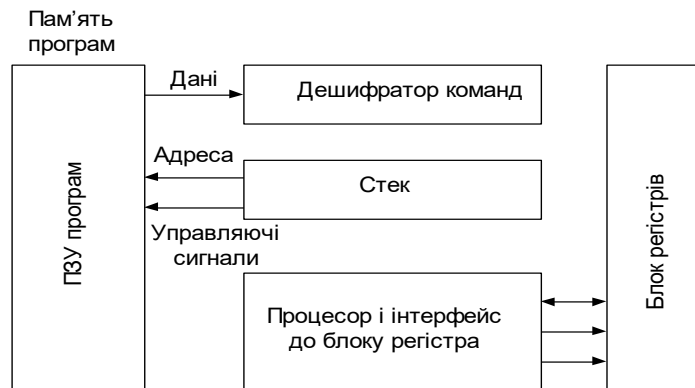


Рис. 9.6. Виконання команди *mov Acc, R* в Гарвардській архітектурі.

Цикл 1: - завершення виконання попередньої команди – читання (вибірка) команди *mov Acc, R*

Цикл 2: - виконання команди *mov Acc, R* – читання (вибірка) наступної команди

Наприклад, мікроконтролер PIC фірми Місгошір виконує будь-яку команду, крім тих, котрі модифікують вміст програмного лічильника, за чотири такти (один цикл). Це спрощує реалізацію критичних до часу процедур у порівнянні з мікроконтролером Intel 8051, де для виконання команд може знадобитися від 16 до 64 тактів. Через це часто не вдається підрахувати точний час виконання програми вручну і приходиться застосовувати симулятори чи апаратні емулятори.

Слід зазначити, що такі загальні способи порівняння продуктивності не слід використовувати для всіх процесорів і мікроконтролерів, у яких реалізуються ці дві архітектури. Порівняння краще проводити стосовно до конкретного додатка. Різні архітектури і пристрої мають свої специфічні особливості, що дозволяють щонайкраще реалізувати ті чи інші додатки. У деяких випадках конкретний додаток може бути виконано тільки з використанням визначеної архітектури і специфічних особливостей мікроконтролера.

9.6. Типи пам'яті мікроконтролерів

Можна виділити три основних види пам'яті, що використовується в мікроконтролерах. Пам'ять програм являє собою постійну пам'ять, призначену для збереження програмного коду і констант. Ця пам'ять не змінює свого

вмісту в процесі виконання програми. Пам'ять даних призначена для збереження перемінних у ході виконання програми. Регістри мікроконтролера - цей вид пам'яті включає внутрішні регістри процесора і регістри, що служать для управління периферійними пристроями.

Пам'ять програм. Для збереження програм звичайно служить один з видів постійної пам'яті: PROM (однократно-програмувальне ПЗУ), EPROM (електрично-програмувальне ПЗУ з ультрафіолетовим стиранням), EEPROM (ПЗУ з електричним записом і стиранням, до цього виду відносяться також сучасні мікросхеми Flash-пам'яті) чи ROM (масово-програмувальне ПЗУ). Усі ці види пам'яті є енергонезалежними — це означає, що вміст пам'яті зберігається після вимикання живлення мікроконтролера. Така пам'ять необхідна, тому що мікроконтролер не містить яких-небудь пристроїв масової пам'яті (магнітних дисків), з яких завантажується програма в комп'ютерах. Програма постійно зберігається в мікроконтролері.

У процесі виконання програма зчитується з цієї пам'яті, а блок управління (дешифратор команд) забезпечує її декодування і виконання необхідних операцій. Вміст пам'яті програм не може мінятися (перепрограмуватися) під час виконання програми. Тому функціональне призначення мікроконтролера не може змінитися, поки вміст його пам'яті програм не буде стерте (якщо це можливо) і перепрограмоване (заповнене новими командами).

Варто звернути увагу, що розрядність мікроконтролера (8, 16 чи 32 біт) указується відповідно до розрядності його шини даних. У Гарвардській архітектурі команди можуть мати більшу розрядність, ніж дані, щоб дати можливість зчитувати за один такт цілу команду. Наприклад, мікроконтролери PIC у залежності від моделі використовують команди з розрядністю 12, 14 чи 16 біт. У мікроконтролерах AVR команда завжди має розрядність 16 біт. Однак усі ці мікроконтролери мають шину даних розрядністю 8 біт.

У пристроях із Принстонської архітектурою розрядність даних звичайно визначає розрядність (число ліній) використовуваної шини. У мікроконтролерах Motorola 68HC05 24-розрядна команда розміщується в трьох 8-розрядних комірках пам'яті програм. Для повної вибірки такої команди необхідно зробити три цикли зчитування цієї пам'яті.

Коли говориться, що пристрій є 8-розрядним, це означає розрядність даних, що здатний обробляти мікроконтролер.

Пам'ять ROM (ПЗУ) використовується тоді, коли програмний код заноситься в мікроконтролер на етапі його виробництва. Попередньо програма налагоджується і тестується, після чого передається фірмі-виробнику, де програма перетворюється на малюнок маски на скляному фотошаблоні. Отриманий фотошаблон з маскою використовується в процесі створення з'єднань між елементами, з яких складається пам'ять програм. Тому таку пам'ять часто називають масово-програмувальне ROM.

ROM є самим дешевим типом постійної пам'яті для масового виробництва. Однак вона має ряд істотних недоліків, що привели до того, що в останні роки цей тип пам'яті майже не використовується. Основними недоліками є значні витрати засобів та часу на створення нового комплекту фотошаблонів і їхнє впровадження у виробництво. Звичайно такий процес займає біля десяти тижнів і є економічно вигідним при випуску десятків тисяч приладів. Тільки при таких обсягах виробництва забезпечується перевага ROM у порівнянні з E(E)PROM. Існує також обмеження, пов'язане з можливістю використання таких мікроконтролерів тільки у визначеній сфері застосування, тому що його програма забезпечує виконання жорстко фіксованої послідовності операцій, і не може бути використана для вирішення яких-небудь інших задач.

Електрично-програмувальна пам'ять EPROM складається з комірок, що програмуються електричними сигналами і витираються за допомогою ультрафіолетового світла. Пам'ять PROM може бути запрограмована тільки один раз. Ця пам'ять звичайно містить плавкі перемички, що перепалюються під час програмування. В даний час така пам'ять використовується дуже рідко.

Комірка пам'яті EPROM являє собою MOS-транзистор із плаваючим затвором, що оточений діоксидом кремнію (SiO₂). Стек транзистора з'єднаний з «землею», а джерело підключене до напруги живлення за допомогою резистора. У стертому стані (до запису) плаваючий затвор не містить заряду, і MOS-транзистор закритий. У цьому випадку на джерелі підтримується високий потенціал, і при звертанні до комірки зчитується логічна одиниця. Програмування пам'яті зводиться до запису у відповідні комірки логічних нулів.

Програмування здійснюється шляхом подачі на керуючий затвор високої напруги. Цієї напруги повинно бути досить, щоб забезпечити пробій між керуючим і плаваючим затворами, після чого заряд з керуючого затвора переноситься на плаваючий. MOS-транзистор переключується у відкритий стан, закорочуючи джерело з землею. У цьому випадку при звертанні до комірки зчитується логічний нуль.

Щоб стерти вміст комірки, він висвітлюється ультрафіолетовим світлом, що дає заряду на плаваючому затворі достатню енергію, щоб він міг залишити затвор. Цей процес може займати від декількох секунд до декількох хвилин.

Звичайно, мікросхеми EPROM виробляються в керамічному корпусі з кварцевим віконцем для доступу ультрафіолетового світла. Такий корпус досить дорогий, що значно збільшує вартість мікросхеми. Для зменшення ціни мікросхеми EPROM укладають у корпус без віконця (версія EPROM з однократним програмуванням). Скорочення вартості при використанні таких корпусів може бути настільки значним, що ці версії EPROM у даний час частіше використовуються замість масово-програмувального ROM.

Раніше мікроконтролери програмувалися тільки за допомогою паралельних протоколів, досить складних для реалізації. В даний час протоколи програмування сучасної EPROM і EEPROM пам'яті істотно змінилися, що дозволило виконувати програмування мікроконтролера безпосередньо в складі системи, де він працює. Такий спосіб програмування одержав назву «in-system programming» чи «ISP». ISP-мікроконтролери можуть бути запрограмовані після того, як їх припаяли на плату. При цьому скорочуються витрати на програмування, тому що немає необхідності у використанні спеціального устаткування — програматорів.

Пам'ять EEPROM (Electrically Erasable Programmable Memory - програмувальна пам'ять, що стирається електрично) можна вважати новим поколінням EPROM пам'яті. У такій пам'яті комірка стирається не ультрафіолетовим світлом, а шляхом електричного з'єднання плаваючого затвора з «землею». Використання EEPROM дозволяє стирати і програмувати мікроконтролер, не знімаючи його з плати. Таким способом можна періодично оновлювати його програмне забезпечення.

Пам'ять EEPROM більш дорога, ніж EPROM (у два рази дорожче за EPROM з однократним програмуванням), а також EEPROM працює небагато повільніше, ніж EPROM. Основна перевага використання пам'яті EEPROM полягає в можливості її багаторазового перепрограмування без видалення з плати. Це дає величезний вигравш на початкових етапах розробки систем на базі мікроконтролерів чи у процесі їхнього вивчення, коли багато часу іде на багаторазовий пошук причин непрацездатності системи і виконання наступних циклів стирання-програмування пам'яті програм.

Функціонально Flash-пам'ять мало відрізняється від EEPROM. Основні відмінності полягають в способі стирання записаної інформації. У пам'яті EEPROM стирання відбувається окремо для кожної комірки, а в Flash-пам'яті стирання здійснюється цілими блоками. Якщо ви хочете змінити вміст однієї комірки Flash-пам'яті, то вам буде потрібно перепрограмувати цілий блок (чи всю мікросхему). У мікроконтролерах з пам'яттю EEPROM можна змінювати окремі ділянки програми без необхідності перепрограмувати весь пристрій.

Часто вказується, що мікроконтролер має Flash-пам'ять, хоча насправді він містить EEPROM. В даний час між цими типами пам'яті є небагато відмінностей, тому деякі виробники використовують ці терміни як еквівалентні. Пам'ять даних. При першому знайомстві з описом мікроконтролера багатьох здивує малий об'єм їхньої оперативної пам'яті даних RAM, що звичайно складає десятки чи сотні байт. Якщо мікроконтролер використовує для збереження даних пам'ять EEPROM, то її об'єм також не перевищує декількох десятків байт.

Якщо ви пишете програми для персонального комп'ютера (PC), то у вас, імовірно, виникне питання, що можна зробити з таким маленьким об'ємом пам'яті. Імовірно, Ваші програми для PC містять змінні, об'єм яких вимірюється в кілобайтах, не враховуючи використовувани масиви даних. При використанні масивів необхідний обсяг пам'яті може складати сотні кілобайт. Так що ж можна зробити, маючи обсяг ОЗП порядку 25 байт.

Справа в тому, що програмування для мікроконтролера виконується за іншими правилами, ніж програмування PC. Застосовуючи деякі нескладні правила можна вирішувати багато задач з використанням невеликого обсягу пам'яті RAM. При програмуванні мікроконтролерів, константи, якщо це можливо, не зберігаються як змінні. Максимально використовуються апаратні можливості мікроконтролерів (такі як таймери, індексні регістри), щоб по можливості обмежити розташування даних у RAM. Це означає, що при розробці прикладних програм необхідно попередньо подбати про розподіл ресурсів пам'яті. Прикладні програми повинні орієнтуватися на роботу без використання великих масивів даних.

У мікроконтролерах RAM використовується для організації виклику підпрограм і обробки переривань. При цих операціях вміст програмного лічильника й основних регістрів (акумулятор, регістр стану, індексні регістри і т.д.) зберігається і потім відновлюється при поверненні до основної програми. Стек — це електронна структура даних, що функціонує аналогічно до своєї фізичної копії — стопки паперів. Коли що-небудь поміщається в стек, то воно залишається там доти, поки не буде вийняте назад. Уявіть різнобарвні аркуші паперу, що укладаються в стопку один на іншій. Коли аркуші віддаляються, то відбувається їхнє переміщення в зворотному порядку. З цієї причини, стек часто називають чергою типу LIFO (Last In, First Out) — «останній прийшов, перший пішов».

У Принстонській архітектурі RAM використовується для реалізації безлічі апаратних функцій, включаючи функції стеку. При цьому знижується продуктивність пристрою, тому що для доступу до різних видів пам'яті необхідні багаторазові звертання, що не можуть виконуватися одночасно. По цій же причині Принстонська архітектура зазвичай вимагає більшої кількості тактів на виконання команди, чим Гарвардська.

Процесори Гарвардської архітектури можуть мати три області пам'яті, що адресуються паралельно (в один і той же час): пам'ять програм, пам'ять даних, що включає простір введення-виводу, і стек. У Гарвардській архітектурі стекові операції можуть реалізовуватись в пам'яті, спеціально виділеній для цієї мети. Це означає, що при виконанні команди виклику підпрограми «call» процесор з Гарвардською архітектурою виконує кілька дій одночасно. У Принстонській архітектурі при виконанні команди «call» наступна команда вибирається після того, як у стек буде поміщений вміст програмного лічильника.

Необхідно пам'ятати, що мікроконтролери обох архітектур мають обмежений об'єм пам'яті для збереження даних. Перевищення цієї межі може викликати проблеми при виконанні програм.

Якщо в процесорі виділений окремий стек, і обсяг записаних у нього даних перевищує його ємність, то відбувається циклічна зміна вмісту покажчика стека, і покажчик стека починає посилатися на раніше заповнену

комірку стека. Це означає, що після занадто великої кількості команд «call» у стеці з'явиться неправильна адреса повернення, що була записана замість правильної адреси. Якщо мікропроцесор використовує загальну область пам'яті для розташування даних і стека, то існує небезпека, що при переповненні стека відбудеться запис в область даних, або буде зроблена спроба запису даних, що завантажуються в стек, в область ROM.

Тепер розглянемо можливості збереження в стеці вмісту регістрів. У деяких архітектурах немає команд, що виконують завантаження вмісту регістрів у стек «push» і вивантаження зі стека «pop». З чотирьох сімейств мікроконтролерів, описаних у цій книзі, тільки два мають такі команди. Однак команди «push» і «pop» можуть бути легко реалізовані за допомогою індексного регістра, що вказує на область стека. При цьому замість кожної з команд «push» і «pop» використовуються дві команди, зазначені нижче:

Push ; Завантаження даних у стек.
move [index], acc; Зберегти вміст акумулятора в стеці.
decrement index ; Перейти до наступної комірки стека.
Pop ; Вивантажити дані зі стека.
increment index ; Перейти до попередньої комірки.
move acc, [index]; Помістити вміст стека в акумулятор.

Звичайно, таке рішення є менш ефективним, чим використання спеціальних команд «push» і «pop», а використовуваний індексний регістр може знадобитися для інших цілей. Однак це рішення забезпечує імітацію стека при використанні процесорів, у яких такі команди відсутні.

Існує ще одна проблема з наведеними вище прикладами. Що станеться, якщо відбудеться переривання між першою і другою командою, що імітують операції «push» і «pop». Якщо програма обробки переривання використовує стек, то записані в ньому дані будуть загублені. Для запобігання цього можна заборонити переривання перед виконанням цих команд чи переставити їх у наступному порядку:

Push ; Завантажити дані в стек.
decrement index ; Перейти до наступної комірки стека.
move [index], aci ; Зберегти вміст акумулятора в стеці.
Pop ; Вивантажити дані зі стеку.
move aci,[index] ; Помістити значення стека в акумулятор.
increment index ; Перейти до попередньої комірки стеку.

Якщо після першої команди програма буде перервана, то після виконання обробки переривання вміст стеку не буде втрачено.

9.7. Регістри мікроконтролера. Простір введення-виведення

Подібно до всіх комп'ютерних систем, мікроконтролери мають безліч регістрів, що використовуються для управління різними пристроями, підключеними до процесора. Це можуть бути регістри процесора (акумулятор, регістри стану, індексні регістри), регістри управління (регістри управління перериваннями, регістри управління таймером) чи регістри, що забезпечують введення-виведення даних (регістри даних і регістри управління паралельним, послідовним чи аналоговим введенням-виведенням). Звертання до цих регістрів може відбуватися різними способами.

Реалізовані мікроконтролером способи звертання до регістрів впливають на їхню продуктивність. Тому дуже важливо зрозуміти, як відбувається звертання до регістрів, щоб писати ефективні прикладні програми для мікроконтролерів. У процесорах з RISC-архітектурою всі регістри (часто й акумулятор) розташовуються по адресах, що явно задаються. Це забезпечує більш високу гнучкість при роботі процесора.

Розглянемо приклад процедури розгалуження програми за умови, що визначений біт у регістрі порту введення-виведення встановлений у 1. Для CISC-процесора відповідний псевдокод буде виглядати в такий спосіб:

<i>Accumulator = IOPort</i> ;	Завантажити вміст регістра IOPort в акумулятор
<i>Accumulator = Accumulator&(1<Bit);</i>	Маскувати всі біти акумулятора, крім Bit ;
<i>if ZeroFlag != 0 goto Address</i> ;	Якщо Zero, то біт Bit=1

Ця процедура буде компілюватися в послідовність операцій:

if IOPort.Bit == 1 goto Address

Асемблерний запис для мікроконтролерів Microchip PIC:

```
btfsc IOPort, Bit ; Пропустити наступну команду, якщо біт Bit=0  
goto Address
```

Більш ефективно ця процедура реалізується в мікроконтролері Intel 8051:

```
jb IOPort.Bit, Address ; Перейти, якщо біт Bit=1
```

Використовуючи процесор, що може безпосередньо звертатися до будь-якого регістра, можна одержати переваги при розробці простих прикладних програм. Наприклад, у мікроконтролері PIC вміст акумулятора і регістра стану не змінюється при передачі управління в залежності від значення біта в регістрі порту IOPort.

Одним з важливих питань є розміщення регістрів в адресному просторі. У деяких процесорах усі регістри і RAM розташовуються в одному адресному просторі. Це означає, що пам'ять сполучена з регістрами. Такий підхід називається «відображенням пристроїв введення-виведення на пам'ять».

В інших процесорах адресний простір для пристроїв введення-виведення відділено від загального простору пам'яті. Основна перевага розміщення регістрів введення-виведення в окремому просторі адрес полягає в тому, що при цьому спрощується схема підключення пам'яті програм і даних до загальної шини. Пристрій введення-виведення звичайно займає маленький блок адрес, що робить незручним декодування їхньої адреси разом з великими блоками основної пам'яті. Окремий простір введення-виведення дає деякі переваги процесорам з Гарвардською архітектурою, забезпечуючи можливість зчитувати команду під час звертання до регістра введення-виведення.

9.8. Зовнішня пам'ять

Незважаючи на величезні переваги використання внутрішньої вбудованої пам'яті, у деяких випадках необхідне підключення до мікроконтролера додаткової зовнішньої пам'яті (як пам'яті програм, так і даних).

Існує два основних способи підключення зовнішньої пам'яті. Перший спосіб — підключення зовнішньої пам'яті до мікроконтролера, як до мікропроцесора. Багато мікроконтролерів містять спеціальні апаратні засоби для такого підключення. Другий спосіб полягає в тому, щоб підключити пам'ять до пристроїв введення-виведення і реалізувати звертання до пам'яті через ці пристрої програмними засобами. Такий спосіб дозволяє використовувати прості пристрої введення-виведення без реалізації складних шинних інтерфейсів. Вибір найкращого з цих способів залежить від конкретного використання мікроконтролера.

Список питань для самоконтролю

1. Для чого призначені мікроконтролери?
2. Назвіть функції, які виконують мікроконтролери у вимірювальних приладах.
3. Що являє собою мікропроцесорна система?
4. Чим обумовлені систематичні похибки мікропроцесорних систем?
5. Які основні види пам'яті, що використовуються в мікроконтролерах, ви знаєте?

Рекомендована література

1. Павловський О.М. Основи цифрової схемотехніки. Комп'ютерний практикум [Електронний ресурс]: навчальний посібник для здобувачів ступеня бакалавра за освітньою програмою «Комп'ютерно-інтегровані системи та технології в приладобудуванні» спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» / О.М. Павловський, І.О. Васильковська; КПІ ім. Ігоря Сікорського.– Київ: КПІ ім. Ігоря Сікорського, 2022. – 62 с.
2. Roger Tokheim, Patrick Hoppe. Digital Electronics: Principles and Applications. 9th edition in New York, USA: McGraw Hill LCC, 2022, 576 p.
3. Комп'ютерна схемотехніка: підручник / [Азаров О. Д., Гарнага В.А., Клятченко Я.М., Тарасенко В.П.]. – Вінниця: ВНТУ, 2018. – 230 с.
4. Paul Scherz, Simon Monk. Practical Electronics for Inventors. 4th edition in New York, USA: McGraw Hill LCC, 2016.
5. Толюпа С.В., Дружинін В.А., Ярцев В.П. Елементи і вузли цифрових пристроїв інфокомунікаційних систем. Навчальний посібник. – К.: ДУІКТ, 2014 – 165 с.
6. Борисенко О.А. Дискретна математика: підручник / О.А. Борисенко. – Суми: Університетська книга, 2007. – 255 с.
7. Лебедев О.М., Ладик О.І. Цифрова техніка. - К.: Політехніка, 2004. - 316 с.
8. Схемотехніка електронних систем: у 3 кн. Кн. 2. Цифрова схемотехніка: підручник / В.І. Бойко, А.М. Гуржій, В.Я. Жуйков та ін. – 2-ге вид. допов. і переробл. – Київ: Вища шк., 2004. – 423 с.
9. Бабич Н.П. Комп'ютерна схемотехніка: навч. посіб. / Н. П. Бабич, І.А. Жуков. – Київ: МК-Прес, 2004. – 412 с.