МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДНІПРОВСЬКИЙ ДЕРЖАВНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

КОНСПЕКТ ЛЕКЦІЙ

з дисципліни

«Комп'ютерна схемотехніка (елементи і схеми комп'ютерних систем)»

для здобувачів вищої освіти першого (бакалаврського) рівня
за освітньо-професійною програмою
«Інженерія програмного забезпечення»
із спеціальності 121 – «Інженерія програмного забезпечення»
галузі знань 12 – «Інформаційні технології»

Затверджено редакційно-видавничою
секцією науково-методичної ради ДДТУ
протокол №___ від _____ 2019р.

Кам'янське
2019

Розповсюдження і тиражування без офіційного дозволу
Дніпровського державного технічного університету ***заборонено***

Конспект лекцій з дисципліни «Комп'ютерна схемотехніка (елементи і схеми комп'ютерних систем)» для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Інженерія програмного забезпечення» із спеціальності 121 – «Інженерія програмного забезпечення» / Укл. М.В.Бабенко.– Кам'янське: ДДТУ, 2019.– 67с.

Укладач:
кандидат технічних наук, доцент **Бабенко М.В.**

Рецензент:
завідувач кафедри прикладної математики
доктор технічних наук, професор **Самохвалов С.Є.**

Відповідальний за випуск: зав. кафедри програмного забезпечення систем Шумейко О.О.

Затверджено на засіданні кафедри програмного забезпечення систем
(протокол №*1* від *30 серпня 2019*р.)

Коротка анотація видання. Видання містить конспект лекцій, питання для самоперевірки засвоєнних знань, список основної та додаткової літератури для першого та другого модулів з дисципліни "Комп'ютерна схемотехніка (елементи і схеми комп'ютерних систем)" для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Інженерія програмного забезпечення» із спеціальності 121 – «Інженерія програмного забезпечення»

КОМП’ЮТЕРНА СХЕМОТЕХНІКА

ЗМІСТ

ВСТУП 4
1. ЛОГІЧНІ ОСНОВИ КОМП'ЮТЕРНОЇ СХЕМОТЕХНІКИ 5
1.1. Алгебра логіки і теоретичні основи синтезу цифрових пристроїв 5
1.2. Основні закони алгебри логіки 5
1.3. Форми логічних функцій і їх використання для синтезу логічних схем 7
1.4. Логічні елементи і схеми. Принцип подвійності 11
1.5. Методи мінімізації булевих функцій 12
2. ОСНОВНІ ХАРАКТЕРИСТИКИ ЦИФРОВИХ МІКРОСХЕМ 16
2.1. Поняття елементів, вузлів і пристроїв комп'ютерної схемотехніки 16
2.2. Характеристики логічних елементів 19
3. ЕЛЕМЕНТИ КОМП'ЮТЕРНОЇ СХЕМОТЕХНІКИ 23
3.1. Транзисторно-транзисторні логчні елементи 23
3.2. Логічні елементи емітерно-зв′язної логіки 25
3.3. Логічні елементи на МДН-транзисторах. 27
3.4. Логічні елементи інтегрально-інжекційної логіки. 28
3.5. Тригери 29
3.5.1. Класифікація тригерів 30
3.5.2. Динамічні параметри тригерів 31
3.5.3. RS-тригер 31
3.5.4. Тригери типу JK. 34
3.5.5. Т-тригер. 36
3.5.6. D-тригер. 36
4. ПОСЛІДОВНІСНІ (НАКОПИЧУВАЛЬНІ) ВУЗЛИ 39
КОМП'ЮТЕРНОЇ СХЕМОТЕХНІКИ 39
4.1. Регістри 39
4.2. Лічильники 43
5. КОМБІНАЦІЙНІ ФУНКЦІОНАЛЬНІ ВУЗЛИ КОМП’ЮТЕРНОЇ СХЕМОТЕХНІКИ 46
5.1. Дешифратори 46
5.2. Шифратори 47
5.3. Мультиплексори 48
5.4. Демультиплексори 50
5.5. Компаратори. 52
5.6. Суматори. Арифметико-логічні пристрої 53
6. АНАЛОГОВО-ЦИФРОВІ, ЦИФРО-АНАЛОГОВІ ПЕРЕТВОРЮВАЧІ 59
КОДОПЕРЕТВОРЮВАЧІ 59
6.1. Цифро-аналогові перетворювачі 59
6.2. Аналого-цифрові перетворювачі 60
6.3. Перетворювачі кодів 62
ЛІТЕРАТУРА 66

ВСТУП

Комп’ютерна схемотехніка – це науково-технічна дисципліна, яка вивчає теоретичні методи аналізу і синтезу схем комп’ютерів (електронних обчислювальних машин) і засоби їхньої технічної реалізації. Розвиток комп’ютерної схемотехніки є основою удосконалення архітектури комп’ютерів, якісного підвищення їхньої продуктивності та надійності, істотного зменшення масових та габаритних показників.

1. ЛОГІЧНІ ОСНОВИ КОМП'ЮТЕРНОЇ СХЕМОТЕХНІКИ

1.1. Алгебра логіки і теоретичні основи синтезу цифрових пристроїв

Математична логіка є частиною формальної логіки і служить теоретичною основою побудови електронних обчислювальних машин і цифрових пристроїв. Найбільше застосування з області математичної логіки отримала алгебра логіки.

Базою алгебри логіки є поняття про висловлювання, істинність і помилковість висловлювання, і зв'язках між висловлюванням. Висловлення, або логічний аргумент, в залежності від значення буває істинним або помилковим. Значення висловлювання може змінюватися із зміною обставин, і таким чином висловлювання міняє оцінку своєї істинності. З точки зору логіки висловлювання можна розділити:

- висловлювання постійно істинне (математично їх приймають рівними 1);
- висловлювання постійно помилкове (математично їх приймають рівними 0);
- висловлювання, яке може бути істинними або помилковими в залежності від певних умов, тобто приймати значення 1 або 0 навперемінно.

За змістом висловлювання бувають прості та складні. Просте висловлювання - логічний аргумент (змінна) входить до складу складного висловлювання логічної функції, що залежить від істинності або помилковості аргументу. Звичайно просте висловлювання позначається малими буквами латинського або російського алфавіту: x, e, z, m, p, a, b. Складні висловлювання або логічні функції, позначають великими буквами латинського або російського алфавіту: A, F, P, X, Y, S, Q. Зв'язки між висловлюванням - аргументами за своєю логікою різні, і від цього значення складного висловлення непостійне.

Різні комбінації значень вхідних змінних в логічних функціях називаються наборами. Функція є цілком заданою, якщо вказані її значення для всіх наборів значень вхідних змінних. Зіставляючи кожному набору значення функції, рівне 0 або 1, можна отримати табличне завдання даної функції, яке називається таблицею істинності або таблицею відповідності.

Розглянемо логічні функції n аргументів, задаючись набором аргументів і таблицею істинності, в якій визначається значення функції для кожного поєднання аргументів. Для визначення кінцевого числа функцій використовується формула 2^{2n}, отже, для двозначної системи набір функцій двох змінних дорівнює 16, а від однієї змінної - 4. Логічні двійкові функції отримали назву Бульових на ім'я англійського математика XIX віку Дж. Буля. Найменування, позначення і значення всіх шістнадцяти Бульових функцій для двозначної системи двох змінних дано в табл. 1.1 разом з назвами схем логічних елементів.

1.2. Основні закони алгебри логіки

У алгебрі логіки введена наступна система аксіом, що визначає властивості і відносини основних операцій:

$a + b = b + a$

$a(b + c) = ab + ac$

$a + bc = (a + b)(a + c)$

$a + \bar{a} = a$

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

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

На основі цих аксіом виводяться всі теореми, що виражають основні закони алгебри логіки. Їх ще називають системою рівносильних перетворень функції або рівносильностями.

1. Закони нульової безлічі

$0 \cdot a = 0$

$0 + a = a$

$0 \cdot adc...z = 0$

тобто кон'юнкція будь-якого числа змінних обертається в нуль, якщо яка-небудь одна змінна має значення 0, незалежно від значень інших змінних.

Таблиця 1.1 Значення Бульових функцій

№ пп	Значення Бульових функцій у залежності від аргументів x та y.					Позначення функції	Назва функції	Назва або позначення схеми логічного елементу
	x	0	0	1	1			
	y	0	1	0	1			
1	$F_0(x, y)$	0	0	0	0	0	Константа нуль	Генератор нуля
2	$F_1(x, y)$	0	0	0	1	$x \wedge y$	Кон'юнкція, логічне множення, І	Кон'юнктор, І, &
3	$F_2(x, y)$	0	0	1	0	$x \Delta y$	Заборона по x, заперечення імплікації	Схема заборони
4	$F_3(x, y)$	0	0	1	1	x	Змінна x	Повторювач x
5	$F_4(x, y)$	0	1	0	0	$y \Delta x$	Заборона по y, заперечення імплікації	Схема заборони
6	$F_5(x, y)$	0	1	0	1	y	Змінна y	Повторювач y
7	$F_6(x, y)$	0	1	1	0	$x \oplus y$	Сума по модулю 2, логічна нерівнозначність	Додавання по модулю 2, М2
8	$F_7(x, y)$	0	1	1	1	$x \vee y$	Диз'юнкція, логічне додавання, АБО	Диз'юнктор, АБО
9	$F_8(x, y)$	1	0	0	0	$\frac{x \downarrow y}{x \vee y}$	Стрілка Пірса, заперечення диз'юнкції	Елемент Пірса, АБО-НІ
10	$F_9(x, y)$	1	0	0	1	$x \equiv y$	Еквівалентність	Рівнозначність
11	$F_{10}(x, y)$	1	0	1	0	y	Заперечення, інверсія y	Інвертор НІ
12	$F_{11}(x, y)$	1	0	1	1	$y \to x$	Імплікація від y к x	Елемент імплікації
13	$F_{12}(x, y)$	1	1	0	0	$\bar{x}$	Заперечення, інверсія x	Інвертор НІ
14	$F_{13}(x, y)$	1	1	0	1	$x \to y$	Імплікація від x к y	Елемент імплікації
15	$F_{14}(x, y)$	1	1	1	0	$\frac{x / y}{x \wedge y}$	Штрих Шеффера, заперечення кон'юнкції	Елемент Шеффера, І-НІ
16	$F_{15}(x, y)$	1	1	1	1	1	Константа одиниця	Генератор одиниці

2. Закони універсальної множини

$1 \cdot a = a$

$1 + a = 1$

$1 + a + b + \ldots + z = 1$

тобто диз'юнкція будь-якого числа змінних звертається в одиницю, якщо хоч би одна з її змінних має значення 1, незалежно від значень інших змінних.

3. Закони ідемпотентності (повторення, тавтології)

$aa..a = a$

$a + a + \ldots + a = a$

4. Закони подвійної інверсії

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

тобто подвійну інверсію можна зняти.

5. Закони додатковості:

а) логічне протиріччя

$a\bar{a} = 0$,

тобто кон'юнкція будь-якої змінної і її інверсії є 0.

б) закон виключеного третього

$a + \bar{a} = 1$,

тобто диз'юнкція будь-якої змінної і її інверсії є 1.

6. Комутативний (переміщення) закон

$ab = ba$

$a + b = b + a$

тобто результати виконання операцій кон'юнкції і диз'юнкції не залежать від того, в якому порядку слідують змінні.

7. Асоціативні (сполучення) закони

$a(bc) = (ab)c = abc$

$a + (b + c) = (a + b) + c = a + c + b$

тобто для запису кон'юнкції або диз'юнкції дужки можна опустити.

8. Дистрибутивні (розподільні) закони:

а) кон'юнкції відносно диз'юнкції

$a(b + c) = ab + ac$;

б) диз'юнкції відносно кон'юнкції

$a + bc = (a + b)(a + c)$.

9. Закони поглинання

$a(a + b) = a$

$a(a + b)(a + c)...(a + w) = a$

$a + ab = a$

$a + ab + ac + ... + aw = a$

$a(\bar{a} + b) = ab$

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

10. Закони склеювання (поширення)

$ab + a\bar{b} = a$

$(a + b)(a + \bar{b}) = a$

11. Закони узагальненого склеювання

$ab + \bar{a}c + bc = ab + \bar{a}c$

$(a + b)(\bar{a} + c)(b + c) = (a + b)(\bar{a} + c)$

$(a + b)(\bar{a} + c) = ac + \bar{a}b$

12. Закони де Моргана (закони інверсії):

а) для двох змінних

$\overline{ab} = \bar{a} + \bar{b}$,

тобто інверсія кон'юнкції є диз'юнкція інверсій;

$\overline{a + b} = \bar{a}\bar{b}$,

тобто інверсія диз'юнкції є кон'юнкція інверсій;

б) для n змінних

$\overline{abc...w} = \bar{a} + \bar{b} + \bar{c} + ... + \bar{w}$

$\overline{a + b + c + ... + w} = \bar{a}\bar{b}\bar{c}...\bar{w}$

13. Теорема розкладання

$F(a,b,...,w) = aF(1,b,...,w) + \bar{a}F(0,b,...,w)$

$F(a,b,...,w) = [a + F(0,b,...,w)] \times [\bar{a} + F(1,b,...,w)]$

14.

$aF(a,\bar{a},b,c,...,w) = aF(1,0,b,c,...,w)$

$\bar{a}F(a,\bar{a},b,c,...,w) = \bar{a}F(0,1,b,c,...,w)$

$a + F(a,\bar{a},b,c,...,w) = a + F(0,1,b,c,...,w)$

$\bar{a} + F(a,\bar{a},b,c,...,w) = \bar{a} + F(1,0,b,c,...,w)$

1.3. Форми логічних функцій і їх використання для синтезу логічних схем

Залежність вихідних змінних y_i, виражена через сукупність вхідних змінних $x_{n-1},...x_1x_0$ за допомогою операцій алгебри логіки, носить назву функції алгебри логіки. Для n- розрядного двійкового коду $x_{n-1},...x_1x_0$ існує 2^n різних значень y_i.

Функція називається повністю визначеної, якщо задані 2^n її значень. Якщо частина значень функції не задана, то вона називається частково визначеною або недовизначеною.

Іноді відомо, що за умовами роботи пристрою поява деяких вхідних кодів неможлива, і по цьому значення функції алгебри логіки на цих кодах не задаються. При цьому виникають так звані факультативні або необов'язкові значення функції, які можуть задаватися довільними. Вхідні коди, для яких функція алгебри логіки має факультативні значення, називаються забороненими.

Для опису функцій алгебри логіки можуть бути використані різні способи. Основними з них є опис функцій в словесній формі, у вигляді таблиць істинності, алгебраїчних виразів, послідовностей десяткових чисел.

Словесний опис функцій алгебри логіки. Словесний вид опису найчастіше застосовується для первинного, початкового опису поведінки логічного пристрою.

Приклад. Логічна функція трьох змінних дорівнює одиниці, якщо хоч дві вхідні змінні дорівнюють одиниці.

Опис функцій алгебри логіки у вигляді таблиці істинності. Таблиця, що містить всі можливі комбінації вхідних змінних $x_{n-1},\ldots x_1x_0$ і відповідні ним значення вихідних змінних y_i, називається таблицею істинності або комбінаційною таблицею. У загальному випадку таблиця істинності містить 2^n рядків.

Приклад. Складемо таблицю істинності для функцій алгебри логіки з попереднього прикладу.

Таблиця 1.2. Таблиця істинності логічної функції трьох змінних

x_2	x_1	x_0	y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Опис функцій алгебри логіки у вигляді алгебраїчного виразу. Алгебра логіки дозволяє утворювати складні функції, аргументи яких є функціями інших двійкових аргументів. Операція заміни аргументом однієї функції іншими, більш простими функціями носить назву суперпозиції функції. Багаторазове використання принципу суперпозиції дає можливість отримати функції бажаного числа аргументів.

Елементарна кон'юнкція утворюється кон'юнкцією кінцевої безлічі логічних змінних і їх заперечень.

Наприклад: $P(x,y,z)=xy\bar{z}$.

Елементарна диз'юнкція утворюється диз'юнкцією кінцевої безлічі логічних змінних і їх заперечень.

Наприклад: $P(x,y,z)=x+y+\bar{z}$.

Кількість змінних в елементарній кон'юнкції (диз'юнкції) називається її довжиною і визначає її ранг.

Наприклад: $P(x,y,z,w)=x+y+\bar{z}+w$ є диз'юнкцією четвертого рангу.

Мінтермом називають функцію, що приймає одиничне значення при одному з всіх можливих наборів аргументів, а макстермом називають функцію, яка приймає нульове значення при одному з можливих наборів і одиничне значення при всіх інших. Мінтерм алгебраїчно являє собою кон'юнкцію аргументів, а макстерм - диз'юнкцію аргументів. Якщо використовується двійкова система і число наборів аргументів n, то число мінтермів або макстермів $N=2^n$.

Диз'юнкція будь-якого числа елементарних кон'юнкцій називається диз'юнктивною нормальною формою (ДНФ).

Наприклад, $a+bc+\bar{a}bc+a\bar{b}\bar{c}$

Кон'юнкція будь-якого числа елементарних диз'юнкцій називається кон'юнктивною нормальною формою (КНФ).

Наприклад, $a(a+b)(\bar{b}+c)(\bar{a}+b+\bar{c})$

Нормальні форми логічних функцій прийнято називати канонічними. Логічну функцію, задану будь-яким аналітичним вираженням, можна безпосередньо перетворити до нормальної диз'юнктивної (або кон'юнктивної) форми. Для цього необхідно:

- виразити всі операції через операції кон'юнкції, диз'юнкції і інверсій;
- позбутися інверсії над цілими виразами, перейшовши до форми, в якій є інверсії тільки окремих змінних;
- розкрити дужки, застосовуючи закон дистрибутивності;
- привести кон'юнкції (диз'юнкції) до елементарних.

Якщо до складу логічної формули входять набори елементарних кон'юнкцій однакового рангу, зв'язані диз'юнкцією, то така форма представлення логічної функції отримала назву довершеної диз'юнктивної нормальної форми (ДДНФ). Правило утворення ДДНФ функції n аргументів складається в наступному.

По кожному набору двійкових змінних, при якому функція приймає значення одиниці, скласти елементарні кон'юнкції (мінтерми).

В елементарну кон'юнкцію записати неінвертованими змінні, що задані одиницею в таблиці істинності, а інвертованими ті змінні, які в таблиці істинності задані нулем. Отримані сполучення називають конституантами одиниці.

Елементарні кон'юнкції з'єднати знаком диз'юнкції.

Довершеною нормальною кон'юнктивною формою (ДКНФ) логічної функції прийнято називати таке її вираження, яке містить елементарні диз'юнкції одного рангу, пов'язані кон'юнкцією. Правило утворення ДКНФ n аргументів складається в наступному.

По кожному набору двійкових змінних, при якому функція приймає значення нуля, скласти елементарні диз'юнкції (макстерми).

У елементарні диз'юнкції записати неінвертованими змінні, задані нулем в таблиці істинності, а інвертованими ті змінні, які в таблиці істинність задані одиницею. Отримані суми називають конституентами нуля.

Елементарні диз'юнкції з'єднати знаком кон'юнкції.

Приклад. Нехай таблицею істинності задана функція $F(x_2, x_1, x_0)$ і потрібно утворити її ДДНФ і ДКНФ.

Таблиця 1.3 - Таблиця істинності функції $F(x_2, x_1, x_0)$

Значення аргументу			Значення функції F	ДДНФ	ДКНФ
				мінтерм	макстерм
x_2	x_1	x_0			
0	0	0	0	-	$x_2 + x_1 + x_0$
0	0	1	1	$\bar{x}_2\bar{x}_1x_0$	-
0	1	0	1	$\bar{x}_2x_1\bar{x}_0$	-
0	1	1	0	-	$x_2 + \bar{x}_1 + \bar{x}_0$
1	0	0	1	$x_2\bar{x}_1\bar{x}_0$	-
1	0	1	0	-	$\bar{x}_2 + x_1 + \bar{x}_0$
1	1	0	0	-	$\bar{x}_2 + \bar{x}_1 + x_0$
1	1	1	1	$x_2x_1x_0$	-

По таблиці знаходимо, що функція F приймає значення одиниці при чотирьох наборах аргументів, тому функція F в ДДНФ буде складатися з логічної суми чотирьох мінтермів:

$$F(x_2, x_1, x_0) = \bar{x}_2\bar{x}_1x_0 + \bar{x}_2x_1\bar{x}_0 + x_2\bar{x}_1\bar{x}_0 + x_2x_1x_0$$

Функція $F(x_2, x_1, x_0)$ в ДКНФ приймає значення нуля при чотирьох наборах аргументів і буде складатися з логічного сполучення чотирьох макстермів:

$$F(x_2, x_1, x_0) = (x_2 + x_1 + x_0)(x_2 + \bar{x}_1 + \bar{x}_0)(\bar{x}_2 + x_1 + \bar{x}_0)(\bar{x}_2 + \bar{x}_1 + x_0)$$

ДКНФ використовується рідше за ДДНФ в процесі перетворення логічних виразів. Отримані формули функції, якщо не застосовувати ніяких перетворень, можна використати для синтезу функціональних схем логічних пристроїв. Так, для реалізації $F(x_2, x_1, x_0)$ в ДДНФ можна прийняти схему рис. 1.1.

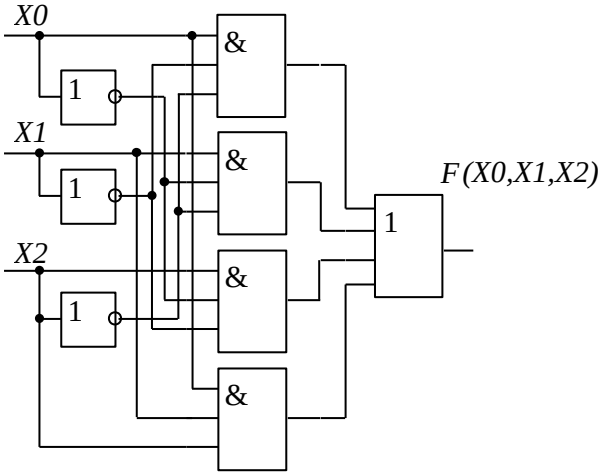


Рис.1.1. Реалізація функції $F(x_0,x_1,x_2)$

Від будь-якої ДНФ можна перейти до ДДНФ функції за допомогою рівносильних перетворень. Такий перехід називається розгортанням. Для цього необхідно:

- ввести бракуючі змінні в кожну кон'юнкцію множенням її на рівносильність вигляду $a + \bar{a} = 1$, де a - бракуюча змінна;
- розкрити дужки, застосовуючи комутативний закон (ab=ba);
- позбутися кон'юнкцій, що повторюються на основі закону ідемпотентності (a + a = a).

Приклад. Нехай ДНФ функції має вигляд: $F = a\bar{b}c + a\bar{c}$.

Слідуючи викладеному правилу переходу до ДДНФ, отримаємо:

$$F = a\bar{b}c + a\bar{c}(b + \bar{b}) = a\bar{b}c + abc + a\bar{b}\bar{c}.$$

Перехід від КНФ до ДКНФ здійснюється аналогічно з переходом від ДНФ до ДДНФ. Для цього необхідно:

- ввести бракуючі змінні в кожну диз'юнкцію, використовуючи закон суперечності $a\bar{a} = 0$ (a- бракуюча змінна);
- виробити перетворення, застосовуючи другий закон дистрибутивності a + bc = (a + b) (a + c) і комутативний закон a + b= b + a.
- позбутися диз'юнкцій, що повторюються, на основі закону ідемпотентності aa = a.

Приклад. Розгортання КНФ вигляду: $F = (a + b)(\bar{b} + c)(\bar{a} + \bar{c})$

в ДКНФ здійснюється таким чином:

$$F = (a + b + 0)(\bar{b} + c + 0)(\bar{a} + \bar{c} + 0) =$$
$$= (a + b + c\bar{c})(\bar{b} + c + a\bar{a})(\bar{a} + \bar{c} + b\bar{b}) =$$
$$= (a + b + c)(\bar{b} + c + a)(\bar{a} + \bar{c} + b)(a + b + \bar{c})(\bar{b} + c + \bar{a})(\bar{a} + \bar{c} + \bar{b})$$

Опис функцій алгебри логіки у вигляді послідовності десяткових чисел. Іноді для скорочення запису функцію алгебри логіки представляють у вигляді послідовності десяткових чисел. При цьому послідовно записують десяткові еквіваленти двійкових кодів відповідних конституент одиниці або нуля.

Приклад. Записати у вигляді послідовності чисел функцію $F(x_2, x_1, x_0)$ з попереднього прикладу.

Рішення. У ДДНФ перша конституента "одиниця" відповідає двійковому коду 001. Десятковий еквівалент цього коду рівний 1. Аналогічно записуються інші конституанти

$F(x_2, x_1, x_0) = \Sigma (1, 2, 4, 7) = \vee(1, 2, 4, 7)$.

$F(x_2, x_1, x_0) = \Pi(0, 3, 5, 6) = \wedge(0, 3, 5, 6)$.

1.4. Логічні елементи і схеми. Принцип подвійності

Відповідно до переліку логічних операцій розрізнюють три основні логічних елемента (ЛЕ): І, АБО, НІ.

Умовні графічні позначення цих ЛЕ показані на рис. 1.2.

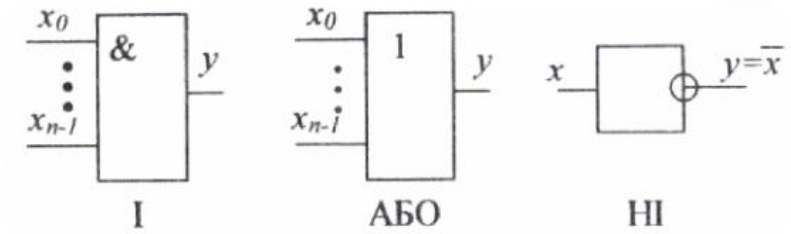


Рисунок 1.2. Умовні графічні позначення логічних елементів

Число входів елементів І, АБО може бути довільним, а елемент НІ має завжди тільки один вхід.

При порівнянні операцій І та АБО, можна помітити, що якщо в умовах, що визначають операцію І, значення всіх змінних і самої функції замінити їх інверсією, а знак логічного множення - знаком логічного додавання, отримаємо постулати, що визначають операцію АБО.

Ця властивість взаємного перетворення постулатів операцій логічного додавання і множення носить назву принципу подвійності. Дві функції алгебри логіки називаються подвійними, якщо одна виходить з іншої заміною кожної операції кон'юнкції на операцію диз'юнкції і навпаки.

Принцип подвійності формулюється так: якщо функції F1 і F2 рівносильні, то рівносильні їм функції F1* та F2*.

Наприклад, для функції $F(a,b) = ab + \bar{a}b$ подвійною є функція $F^*(a,b) = (a+b)(\bar{a}+b)$.

Потрібно відрізняти подвійні форми функції від інверсних функцій, які виходять із вихідних їх інвертуванням. При цьому не тільки всі операції замінюються на подвійні, але і всі змінні замінюються їх інверсіями.

Наприклад, для функції $F(a,b) = ab + \bar{a}b$ інверсною є функція $\bar{F}(a,b) = \overline{ab + \bar{a}b} = (\bar{a}+\bar{b})(a+\bar{b})$.

Важливим практичним слідством принципу подвійності є той факт, що при записі логічних виразів і, отже, побудові логічних схем, можна обійтися тільки двома типами операцій. Наприклад операціями І та НІ або АБО та НІ.

Сукупність ЛЕ, що дозволяє реалізувати логічну схему довільної складності, називається функціонально повною системою. Таким чином, системи двох елементів І та НІ, а також АБО та НІ нарівні з системою з трьох елементів І, АБО, НІ є функціонально повними. На практиці широке застосування знайшли ЛЕ, що суміщають функції елементів вказаних вище функціонально повних систем. Це елементи І - НІ та АБО - НІ, кожний з яких так само утворить функціонально повну систему.

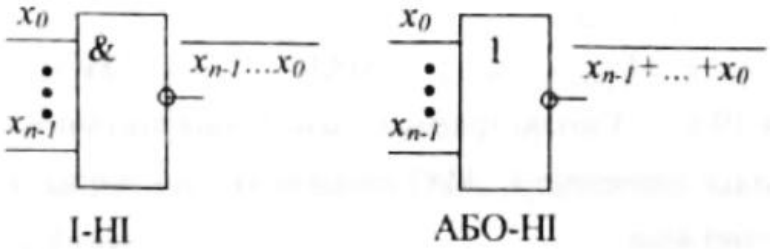


Рис. 1.3. Умовні графічні позначення логічних елементів І - НІ та АБО - НІ

Розглянемо виконання операцій І, АБО, і НІ на елементах АБО - НІ. Згідно з принципом подвійності, якщо $x_1 \cdot x_0 = z$, то $\bar{x}_1 + \bar{x}_0 = \bar{z}$. Інвертуючи праву і ліву частини першого вираження, отримуємо $\bar{x}_1 + \bar{x}_0 = \bar{z} = \overline{x_1 x_0}$, тобто логічна операція І може бути замінена операціями АБО та НІ.

На рис. 1.4 приведені приклади реалізації основних логічних операцій з використанням тільки елементів АБО - НІ.

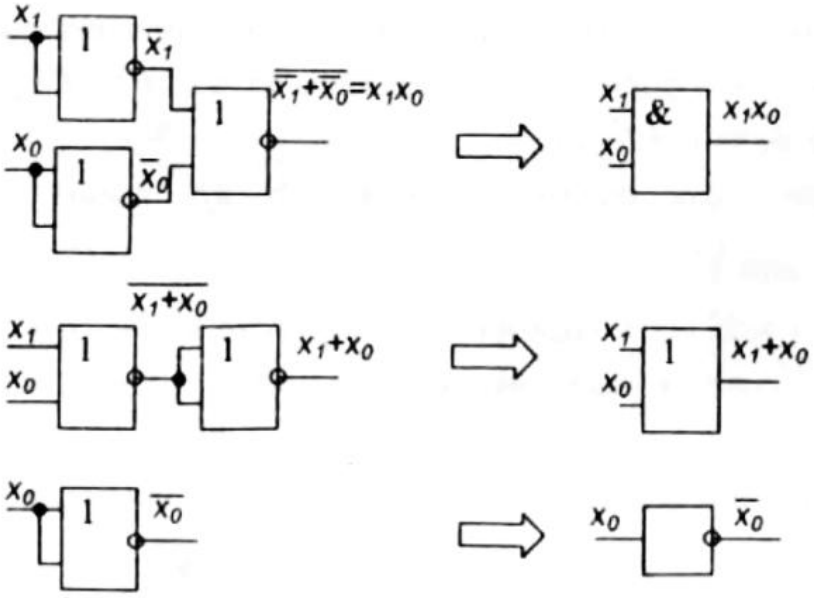


Рис. 1.4. Реалізація логічних операцій І, АБО, НІ на базі елементів 2АБО - НІ

На основі аналогічних міркувань можна показати виконання основних логічних операцій з використанням тільки елемента І - НІ (рис. 1.5).

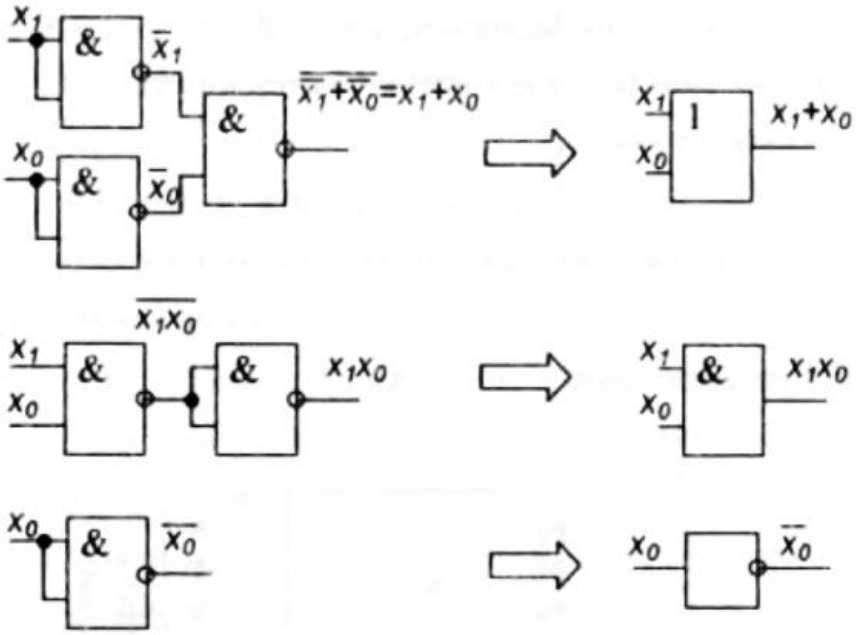


Рис. 1.5. Реалізація логічних операцій на базі елементів 2І - НІ

1.5. Методи мінімізації булевих функцій

Логічну схему, що реалізує заданий алгоритм перетворення сигналів, можна синтезувати безпосередньо по вираженню, поданому у вигляді ДДНФ або ДКНФ. Проте отримана при цьому схема, як правило, не оптимальна з погляду її практичної реалізації. Тому вихідну логічну функцію звичайно мінімізують.

Ціллю мінімізації логічної функції є зменшення вартості її технічної реалізації. Критерій, відповідно до яких виконується мінімізація, далеко не однозначний і залежить як від типу задачі, так і рівня розвитку технології.

Основними вимогами до задачі синтезу є: мінімальне число елементарних кон'юнкцій або диз'юнкцій у логічній формулі й однорідність використовуваних операцій. Крім вимог мінімізації ставиться ряд обмежень і умов на вибір елементної бази для синтезованого пристрою.

Метод безпосередніх перетворень

Безпосереднє спрощення вихідної логічної функції, заданої у виді ДДНФ, виконується в наступному порядку:

Для кожної із можливих пар сусідніх конституєнтів ДДНФ застосовується операція повного склеювання. При цьому з них виключається по одній змінній. Потім виконується приведення подібних членів. Цей процес повторюється доти, поки в отриманому вираженні не буде більше кон'юнкцій, що відрізняються друг від друга значенням однієї змінної. Отримана в такий спосіб форма називається скороченою нормальною формою. Кон'юнкції, що входять у скорочену нормальну форму, називаються простими імплікантами. Кожній логічній функції відповідає лише одна скорочена форма.

Застосовуючи до скороченої нормальної форми операцію узагальненого склеювання, виключають із неї зайві кон'юнкції (імпліканти). Отримана в результаті послідовного ряду таких перетворень форма, що не допускає подальших склеювань, називається тупиковою формою логічної функції. Тупикових форм для однієї функції може бути декілька.

Отримана тупикова форма може випадково виявитися мінімальною. Мінімальною формою є тупикова форма мінімальної довжини. У загальному випадку для пошуку мінімальної форми необхідний перебір тупикових форм, що дозволяє знайти одну або декілька мінімальних форм логічної функції.

Для вихідної функції, заданої у вигляді ДКНФ, мінімізація методом безпосереднього спрощення виконується в такий спосіб:

- Спочатку до членів ДКНФ застосовують операцію повного склеювання.
- Користуючись законом дистрибутивності, розкривають дужки в отриманому вираженні.
- Призводять подібні члени і застосовують операцію поглинання
- Отриману в такий спосіб ДНФ мінімізують у зазначеному вище порядку.

Приклад. Знайти мінімальну форму функції, заданої ДДНФ

$F(a,b,c) = a\bar{b}c + \bar{a}\bar{b}c + \bar{a}\bar{b}\bar{c} + \bar{a}b\bar{c} + ab\bar{c} = abc.$

Застосовуючи операцію повного склеювання до сполучень кожного з конституєнтів із усіма сусідніми і приводячи подібні члени, одержуємо скорочену нормальну форму:

$F(a,b,c) = \bar{a}\bar{b} + \bar{b}c + ac + ab + b\bar{c} + \bar{a}\bar{c}.$

Застосування операції узагальненого склеювання до імплікант можна здійснити в кількох варіантах. Кожному з них відповідає одна з наступних тупикових форм:

$F_1(a,b,c) = ac + \bar{b}c + b\bar{c} + \bar{a}\bar{c};$

$F_2(a,b,c) = ac + b\bar{c} + \bar{a}\bar{b};$

$F_3(a,b,c) = \bar{b}c + ab + \bar{a}\bar{c}.$

Очевидно, що аналізованій функції відповідають дві мінімальні нормальні форми $F_2(a,b,c)$ і $F_3(a,b,c)$.

Метод Карно-Вейча

Метод діаграм Вейча, удосконалений Карно, застосовується в тому випадку, коли число аргументів не більш 5-6. Карти Карно - це графічне представлення таблиць істинності. Кожній комбінації змінних можна поставити у відповідність клітину карти Карно. У клітину записується значення функції (0 або 1) для даної комбінації вхідних змінних. Вхідні змінні розташовуються по зовнішніх сторонах карти навпроти її рядків і стовпчиків. При цьому значення кожної із вхідних змінних відноситься до всього рядка або стовпчика і дорівнює 1, якщо навпроти рядка (стовпчика) стоїть під дужкою позначення цієї змінної; для інших рядків (стовпчиків) значення цієї змінної дорівнює 0.

Кожна з вхідних змінних ділить по-своєму будь-яку карту Карно на дві рівні частини, в одній з яких значення цієї змінної дорівнює 1, а в іншій 0. Кожній клітині карти відповідає певна комбінація значень усіх вхідних змінних, а кожна сторона клітини являє собою межу між значеннями змінних. Число клітин карти Карно визначається величиною 2^n, де n дорівнює числу вхідних змінних.

Приклад. Для функції трьох змінних F(a, b, c), заданою таблицею істинності (табл. 1.4), карта Карно наведена на рис. 1.6.

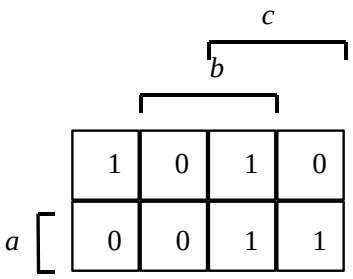


Рис. 1.6. Карта Карно для функції трьох змінних

Карта Карно для логічної функції, заданої алгебраїчним вираженням, може бути складена в наступному порядку:

- по числу змінних, вхідних у вираженні заданої функції, будується карта Карно і розташовуються змінні; задане алгебраїчне вираження приводиться до ДДНФ;
- у карті Карно для кожної конституенти одиниці ДДНФ знаходиться відповідна клітина (із таким же набором змінних), у якій записується 1, в інші клітини карти записуються 0.

Таблиця 1.4. Значення функції F(a, b, c)

a	b	c	F(a,b,c)
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Властивості карти Карно:

Комбінації значень змінних для сусідніх клітин карти Карно розрізняються значенням лише однієї змінної. При переході з однієї клітини в сусідню завжди змінюється значення лише одної змінної від свого прямого значення до його інверсії й навпаки.

Сусідніми між собою є крайні ліві клітини карти Карно з крайніми правими і крайні верхні клітини карти з крайніми нижніми (як якби карти були згорнуті в циліндри по вертикалі і по горизонталі).

Всі клітини, що відрізняються значенням тільки однієї змінної, є сусідніми, незважаючи на те, що іноді вони розташовані не поруч (для функцій п'яти змінних і більше).

Для деякої логічної функції, наданої за допомогою карти Карно, можна записати декілька алгебраїчних виражень різноманітної складності в диз'юнктивній або кон'юнктивній формі.

При цьому варто керуватися рядом правил:

Всі одиниці (при записі функції в диз'юнктивній формі) і всі нулі (при записі в кон'юнктивній формі) повинні бути замкнуті в прямокутні контури. Одиничні контури можуть об'єднувати декілька одиниць, але не повинні містити усередині себе нулів. Нульові контури можуть об'єднувати декілька нулів, але не повинні містити усередині себе одиниць. Однойменні контури можуть накладатися один на одного, тобто та сама одиниця (або нуль) може входити в декілька одиничних (нульових) контурів.

Площа будь-якого контуру повинна бути симетричною щодо меж змінних, пересічних даним контуром. Іншими словами, число клітин у контурі дорівнює 2^n, де n - 0, 1, 2, 3, 4,…, тобто число клітин виражається числами 1,2, 4, 8, 16, 32, ...

Щоб уникнути одержання зайвих контурів їхню побудову варто починати з тих одиниць або нулів, що можуть ввійти в один єдиний контур. Зайвими називаються контури, усі клітини яких увійшли вже в інші контури.

У контури можна об'єднувати тільки сусідні клітини, що містять одиниці або нулі. Дотримання цього правила особливо необхідно перевіряти при числі змінних, більше чотирьох, коли сусідні клітини можуть бути розташовані не поруч і тому контури можуть зазнавати видимий розрив.

Кожній одиничній клітині відповідає кон'юнкція вхідних змінних, що визначають дану клітину. Кожній нульовій клітині відповідає диз'юнкція інверсій вхідних змінних, що визначають дану клітину.

У контурі, що об'єднує дві клітини, одна з змінних змінює своє значення, тому вираження контуру з двох клітин не залежить від цієї змінної, а представляється усіма іншими змінними. Це правило відноситься і до контурів, що охоплюють число клітин більш двох, і має таке формулювання: вираження, що відповідають контурам, не містить тих змінних, чиї межі перетинаються площею, обмеженою даним контуром.

Вираження логічної функції може бути записане по відповідній їй карті Карно в диз'юнктивній або кон'юнктивній формі. Диз'юнктивна форма складається у вигляді диз'юнкції кон'юнкцій, що відповідають одиничним контурам, виділеним на карті для визначення функції; кон'юнктивна - у вигляді кон'юнкції диз'юнкцій, що відповідають нульовим контурам.

Для контурів, що охоплюють різноманітну кількість клітин, утворюються вираження різноманітної складності. Тому для даної логічної функції можна записати по її карті Карно, декілька алгебраїчних виражень, що відрізняються по складності. Найбільше складне вираження відповідає випадку, коли кожній клітині відповідає свій контур. Це вираження являє собою ДДНФ або ДКНФ даної функції.

Для одержання по карті Карно мінімального вираження логічної функції варто керуватися, крім загальних правил, викладених вище, правилом: одиниці або нулі повинні об'єднуватися мінімальним числом найбільших контурів.

Іноді для деяких наборів вхідний змінних значення функції строго не визначенні і її карта Карно містить умовні члени, тобто такі клітини, у яких значення функції можна вважати рівним одиниці або нулю. Такі члени на карті прийнято позначати знаком байдужного значення функції ~, названим "тильда". Порожня клітина відповідає умовному (байдужному) значенню функції.

Наявність знаків умовного значення функції в клітинах карти Карно дозволяє включати ці клітини в контури з одиницями або нулями (по розсуду проектувальника), що сприяє одержанню мінімальних алгебраїчних виражень для даної логічної функції.

Метод мінімізації за допомогою карт Карно дозволяє наочно і достатньо просто здійснювати мінімізацію логічної функції чотирьох - шести змінних, хоча іноді його застосовують і для мінімізації функцій більшого числа змінних. Проте при збільшенні числа змінних застосування карт Карно ускладнюється, тому що з'являються утруднення при визначенні сусідніх станів (клітин).

2. ОСНОВНІ ХАРАКТЕРИСТИКИ ЦИФРОВИХ МІКРОСХЕМ

2.1. Поняття елементів, вузлів і пристроїв комп'ютерної схемотехніки

Технічні засоби комп'ютерної схемотехніки залежно від функцій, які вони виконують, поділяють на елементи, функціональні вузли і пристрої, а також мікропроцесори та комп'ютери (рис. 2.1). Вони призначені для оброблення дискретної інформації і тому називаються цифровими.

Технічні засоби комп'ютерної схемотехніки в даний час основані на інтегральних мікросхемах (ІМС) різного ступеня складності.

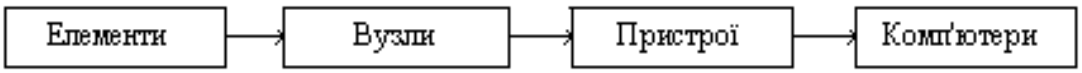


Рис.2.1. Склад технічних засобів комп'ютерної схемотехніки

Елементами в комп'ютерній схемотехніці називаються найменші неподільні мікроелектронні схеми (вироби), призначені для виконання логічних операцій або зберігання біта інформації. До елементів умовно відносяться і допоміжні схеми – підсилювачі, повторювачі, формувачі та ін.

Елементи будуються на основі двопозиційних ключів, що технічно реалізується найпростіше. Елементи з двома станами називаються двійковими.

На входах і виходах двійкового елемента діють напруги, які набувають у сталому режимі двох значень – високого U_H і низького U_L рівнів (індекси від англійських слів High і Low). Ці напруги відображають електричні сигнали. Сигнал з двома станами називається двійковим. Перехід елемента з одного стану в інший називається його перемиканням. На основі елементів будують типові функціональні вузли.

Елементарні дії, які виконуються в комп'ютерах за один машинний такт, називаються мікрооперaціями. Наприклад, інкремент або декремент слова, зсув, інверсія, додавання та ін.

У комп'ютерах команди виконують послідовністю мікрооперацій над двійковими словами (числами). Типовими функціональними вузлами комп'ютерів називаються мікроелектронні схеми, призначені для виконання однієї або декількох мікрооперацій.

За логікою роботи функціональні вузли розподіляються на комбінаційні та послідовнісні схеми.

У комбінаційних схемах логічний стан виходів елементів залежить тільки від комбінації вхідних сигналів у даний момент часу. До функціональних вузлів комбінаційного типу відносяться суматори, дешифратори, шифратори, мультиплексори і демультиплексори, схеми порівняння (компаратори) і контролю за парністю, кодоперетворювачі.

У послідовнісних схемах логічне значення виходів визначають як комбінацією вихідних сигналів, так і станом пам'яті схеми в даний момент часу. До функціональних вузлів послідовнісного типу відносяться регістри, лічильники, генератори чисел і керуючі автомати. На основі типових функціональних вузлів будують різноманітні пристрої комп'ютерів.

Універсальність комп'ютерів забезпечує можливість приймання і видавання інформації, її зберігання та арифметико-логічне опрацьовування, а також керування усім обчислювальним процесом. Ці функції реалізуються відповідними пристроями введення, виведення, запам'ятовування, арифметико-логічними і керування.

В усіх комп'ютерах використовують генератор тактових імпульсів (ГТІ), що виробляє періодичну послідовність прямокутних імпульсів, які називаються тактовими (С). Початок кожного імпульсу С називається тактовим моментом. Часовий інтервал між двома сусідніми імпульсами С називається машинним тактом ТС. На початку кожного імпульсу С відбувається зміна інформації на входах елементів і вузлів машини (рис. 2.2).

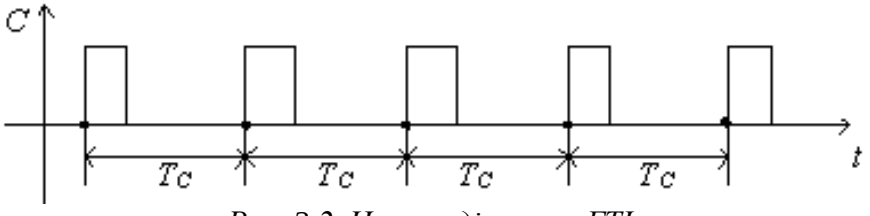


Рис. 2.2. Часова діаграма ГТІ

Частота ГТІ вимірюється десятками і сотнями мегагерц. У літературі ГТІ часто називають генераторами синхронізуючих імпульсів, а самі імпульси називають синхронізуючими або синхроімпульсами. Амплітуда і полярність імпульсу С залежить від фізичних принципів побудови машини. Принцип подачі інформації на входи елементів і вузлів у тактові моменти називається дискретизацією сигналів у часі.

У комп'ютерній схемотехніці застосовуються два основних види двійкових сигналів: потенціальні й імпульсні (рис. 2.3). Сигнал, який змінюється тільки в тактові моменти часу, називається потенціальним. Сигнал, що наростає в тактовий момент, а спадає в границях даного такту, називається імпульсним. Тривалість потенціального сигналу дорівнює або кратна тривалості машинного такту.

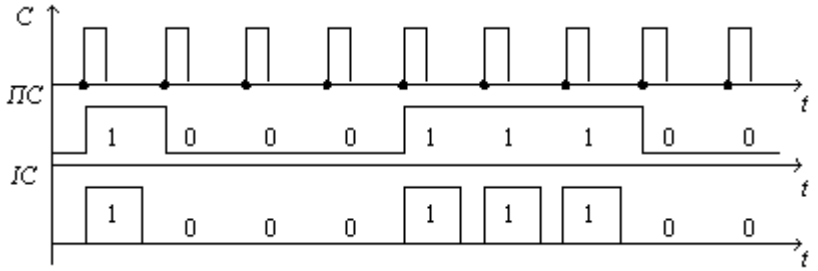


Рис. 2.3. Діаграми потенціальних (ПС) та імпульсних (ІС) сигналів

У логіці значення двійкового сигналу і відповідної змінної X кодуються символами 0 (лог. 0) і 1 (лог. 1). Напругу, що відображає символ 1, позначимо через U_1, а символ 0 – через U_0. Розрізняють два способи кодування логічних сигналів X_i потенціальними сигналами – позитивний та негативний. При позитивному кодуванні (позитивна логіка чи угода) більший рівень напруги U_H з урахуванням знака відображає лог. 1, а менший U_L – лог.0, тобто X = 1, якщо $U_1 = U_H$, та X = 0 при $U_0=U_L$ (рис. 2.4, а). При негативному кодуванні (негативна логіка чи угода) більший рівень напруги U_H з урахуванням знаку відображає лог. 0, а менший U_L – лог. 1, тобто X = 1, якщо $U_1 = U_L$, та X = 0 при $U_0 = U_H$ (рис. 2.4, б).

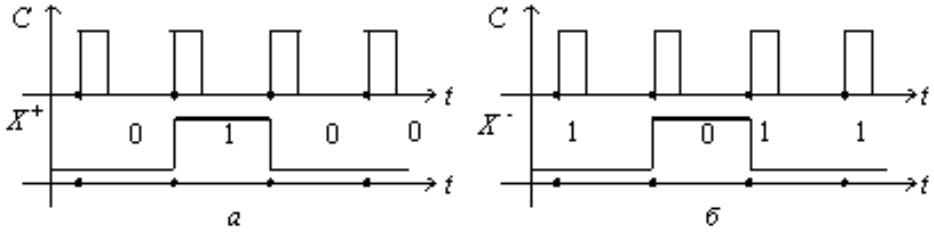


Рис. 2.4. Логічні угоди: а – позитивна X^+; б – негативна X^-

Для імпульсних сигналів розрізняють два роди кодування (рис. 2.5): перший – наявність імпульсу відображає лог. 1, відсутність – лог. 0; другий – наявність імпульсу однієї полярності відображає лог. 1, а іншої полярності – лог. 0.

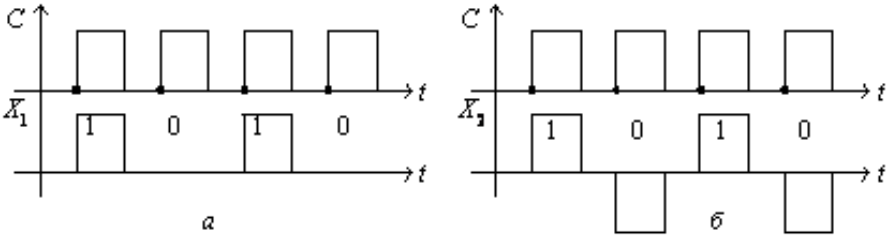


Рис. 2.5. Імпульсне кодування: а – першого роду; б – другого роду

Елементи комп’ютерів класифікуються за такими ознаками:

- використовуваними фізичними приладами;
- видом інформаційних сигналів;
- функціональним призначенням;
- конструкційно-технологічним виготовленням;
- рівнем і ступенем складності.

За типом фізичних приладів розрізняють такі елементи: побудовані на електронних лампах – перше покоління; на транзисторах – друге покоління; на ІМС малого і середнього ступеня інтеграції – третє покоління; на великих і надвеликих ІМС – четверте покоління.

За видом інформаційних сигналів виділяють:

- потенціальні елементи – використовуються тільки потенціальні сигнали;
- імпульсні елементи – використовуються тільки імпульсні сигнали;
- потенціально-імпульсні елементи – використовуються потенціальні й імпульсні сигнали.

За функціональним призначенням елементи комп'ютерної схемотехніки поділяють на такі класи:

- логічні елементи, призначені для виконання логічних операцій – НЕ, І, АБО, НЕ І, НЕ АБО, НЕ І АБО та ін.;
- елементи, які запам'ятовують – тригери, призначені для зберігання значення однієї двійкової змінної – нуля або одиниці, тобто одного біта інформації;
- допоміжні елементи (підсилювачі, формувачі, перетворювачі сигналів, схеми часового узгодження, генератори імпульсів та ін.), призначені для забезпечення роботи елементів перших двох класів.

За конструкційно-технологічним виготовленням елементна база сучасної комп'ютерної схемотехніки складається з інтегральних мікросхем. Це – мікроелектронні вироби з високою щільністю упаковування електрорадіоелементів (резисторів, діодів, транзисторів) і з'єднань між ними. З погляду специфікації, випробування, постачання та експлуатації ІМС розглядаються як єдине ціле.

В ІМС електрорадіоелементи називаються елементами, якщо вони невіддільні від схеми, і компонентами, якщо їх можна використовувати самостійно. Надалі, щоб уникнути плутанини з елементами комп'ютерів, електрорадіоелементи називають просто компонентами.

Мікросхеми класифікують за такими головними ознаками:

- технологією виготовлення – напівпровідникові, гібридні, плівкові;
- конструкційним оформленням – корпусні та безкорпусні;
- формою оброблення інформації – аналогові, цифрові й аналого-цифрові;
- ступенем інтеграції (складності) – малі, середні, великі, надвеликі й ультравеликі;
- типом активних елементів – побудованих на біполярних і МОН-транзисторах;
- областю застосування – широкого застосування, спеціалізовані, у тому числі замовлені і напівзамовлені;
- використовуваними матеріалами – кремнієві, арсенід-галієві;
- перспективними напрямками – кріомікроелектронні, акустоелектронні, оптоелектронні, молекулярної електроніки та ін.

Набір цифрових мікросхем із спільними конструктційно-технологічними і схемотехнічними ознаками утворює серію ІМС. У комп'ютерній схемотехніці широко застосовуються цифрові напівпровідникові корпусні ІМС на основі кремнію і арсеніду галію.

У напівпровідникових ІМС усі компоненти і з'єднання між ними виконані в об'ємі і на поверхні кристала площею від 4 до 100 мм^2. У гібридних ІМС навісні компоненти кріпляться на поверхні діелектричної підкладки. У плівкових ІМС усі компоненти і з'єднання між ними виконані у виді тонких плівок на діелектричній підкладці.

Складність мікросхем характеризується рівнем інтеграції N, ступенем інтеграції K = lgN і ступенем функціональної складності F = lgL, де N – число компонентів, комп.; L – число двовходових логічних елементів (вентилів); значення десяткового логарифма округляється до більшого цілого числа. Промисловість виготовляє ІМС від першого (менше 10 комп.) до шостого (менше 1 млн комп.) і вище ступеня інтеграції.

Можливості інтегральної технології визначає щільність упаковування: відношення числа компонентів до об'єму (іноді до площі) кристала. Щільність упаковування в напівпровідникових ІМС складає 107 комп. /см^3, а для гібридних – 100–200 комп. /см^3.

Мала інтегральна схема (МІС) вміщує до 100 комп. включно, середня мікросхема (СІС) – 100–1000 комп., велика інтегральна схема (ВІС) – до 100000 комп., надвелика інтегральна мікросхема (НВІС) – до 1 млн комп., а ультравелика (ультра-ВІС) – до 10 млн комп. і більше. На МІС бу-

дують елементи, на СІС реалізують типові вузли, на ВІС, НВІС і ультра-ВІС забезпечують побудову мікропроцесорів і мікрокомп'ютерів.

Кожна елементарна логічна функція реалізується відповідно логічним елементом: АБО (диз'юнктор), І (кон'юнктор), НЕ (інвертор). Для реалізації складних функцій логічні елементи об'єднуються у логічну схему.

Функціонально повна система логічних елементів дозволяє побудувати будь-яку складну логічну схему. Такі системи утворюються такими наборами логічних елементів: 1) АБО, НЕ; 2) І, НЕ; 3) НЕ АБО; 4) НЕ І та іншими.

У технічно повній системі елементів забезпечується значення електричних параметрів двійкових сигналів, для цього використовуються допоміжні елементи – підсилювачі, повторювачі, формувачі та ін.

З урахуванням вищевикладеного можна сказати, що система елементів являє собою функціонально і технічно повний набір елементів, який використовує однакові способи представлення інформації, а також має спільні конструктивно-технологічні характеристики.

2.2. Характеристики логічних елементів

Логічні, схемотехнічні й експлуатаційні властивості логічних елементів визначаються сукупністю характеристик і параметрів, до яких відносяться: 1) функції логічних елементів; 2) логічні угоди; 3) коефіцієнти об'єднання за входом і виходом; 4) коефіцієнт розгалуження; 5) швидкодія; 6) потужність споживання; 7) робота перемикання; 8) вхідні й вихідні напруги і струми; 9) статична і динамічна стійкість до перешкод; 10) надійність елементів; 11) допустимі розміри механічних впливів, діапазони тиску і температури навколишнього середовища, стійкість до радіаційних впливів; 12) маса, вартість і конструктивне оформлення. У більшості випадків зазначені характеристики і параметри відносяться і до ІМС, на яких реалізовані логічні елементи.

Коефіцієнт об'єднання за входом N_I характеризує число логічних входів логічного елемента – зазвичай 1, 2, 3, 4 або 8 (рис.2.6).

Коефіцієнт об'єднання за виходом N_o характеризує допустиму кількість з'єднаних між собою виходів логічних елементів з метою утворення нових функцій.

Коефіцієнт розгалуження за виходом N_p характеризує навантажувальну спроможність логічного елемента, тобто максимальне число входів ідентичних схем, яке може бути одночасно залучене до виходу даного елемента без порушення його працездатності (рис. 2.7). До складу серій ІМС зазвичай входять елементи з малою навантажувальною спроможністю (N_p = 3...15) та з великою (N_p = 30...50).

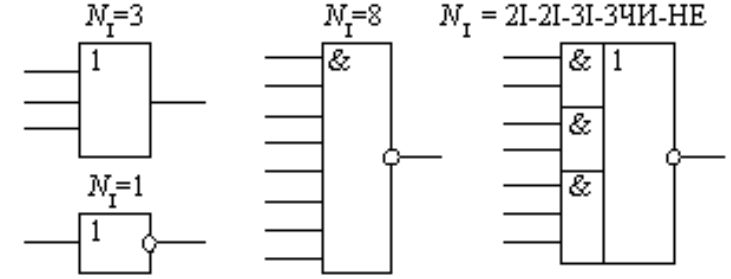


Рис. 2.6. Приклади значень коефіцієнта NI

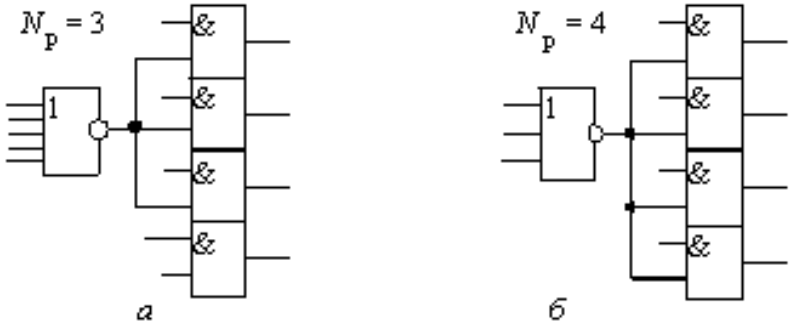


Рис. 2.7. Підключення навантаження: а – Np = 3; б – Np = 4

Прийняті такі визначення і буквені позначення електричних параметрів цифрових мікросхем (ДСТУ 2883-94):

- вхідні U_I і вихідні U_O рівні напруг (індекси – від англійських слів Input і Output);
- вхідні напруги низького U_{IL} і високого U_{IH} рівнів; для них установлені максимальне значення низького рівня U_{ILmax} та мінімальне значення високого рівня U_{IHmin} (рис. 2.8, а);
- вихідні напруги низького U_{OL} і високого U_{OH} рівнів; для них установлені максимальне значення низького рівня U_{OLmax} і мінімальне значення високого рівня U_{OHmin} (рис.2.8,б);
- вхідний I_I і вихідний I_O струми;
- вхідний струм I_{IL} – при низькому рівні напруги на вході, I_{IH} – при високому;
- вихідний струм I_{OL} – при низькому рівні напруги на виході, а I_{OH} – при високому;
- U_{CC} – значення напруги джерела живлення;
- I_{CC} – струм, споживаний ІМС від джерела живлення;
- P_{CC} – потужність, споживана ІМС від джерела живлення;
- вхідні граничні напруги, при яких відбувається перемикання елемента: U_{TIH} – найменше значення для високого рівня і U_{TIL} – найбільше значення для низького рівня.

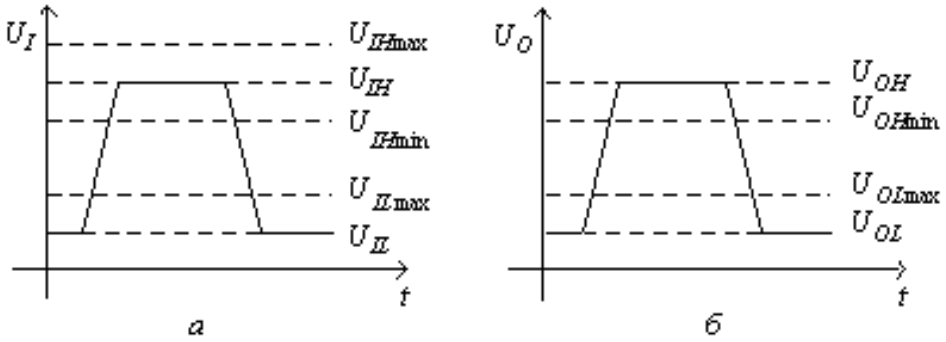


Рис. 2.8. Позначення рівнів напруги: а – вхідних; б – вихідних

Основні параметри логічних елементів визначають за допомогою вхідної, вихідної і передатної характеристик. Типові графіки цих характеристик для інвертувальних елементів транзисторно - транзисторної логіки наведені на рис. 2.9.

Вхідна характеристика логічного елемента $I_I = f(U_I)$ – це залежність вхідного струму від зміни вхідної напруги. Струми, що втікають у схему елемента, вважають додатними, а ті, що витікають – від'ємними (рис. 2.9, а). З цієї характеристики визначають вхідні струми I_{IL} для напруги U_{ILmax} і струм I_{IH} для напруги. U_{IHmin}.

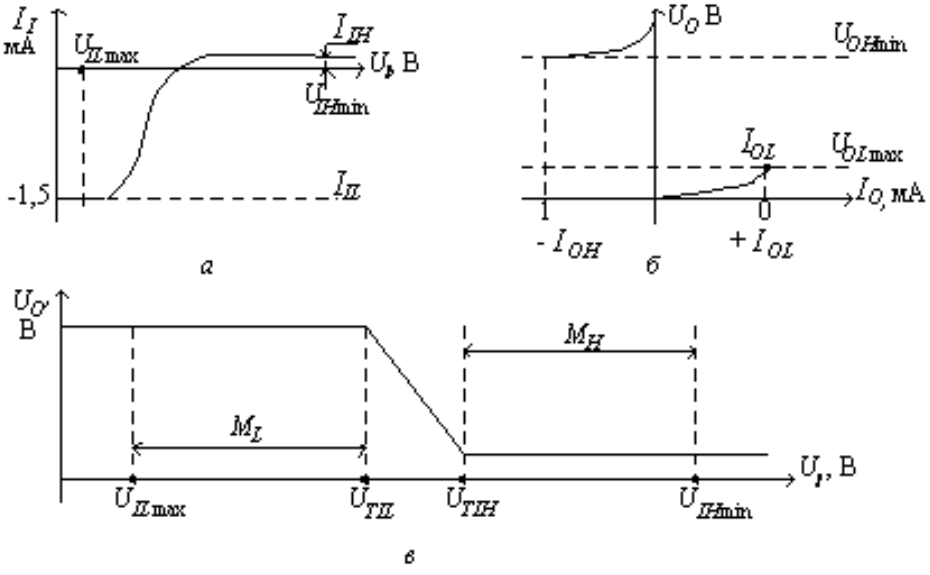


Рис. 2.9. Характеристики логічного транзисторно - транзисторного елемента: а – вхідна; б – вихідна; в – передавальна

Вихідна характеристика логічного елемента $U_O = f(I_O)$ визначає залежність вихідної напруги від струму навантаження для станів високого і низького рівнів (рис. 2.9, б).

Із цієї характеристики визначають допустимі значення струмів: I_{OL} – при низькому рівні вихідної напруги U_{OLmax} і I_{OH} – при високому рівні напруги U_{OHmin} (рис. 2.9, б).

Передатна характеристика $U_O = f(U_I)$ – це залежність вихідної напруги від вхідної (рис. 2.9, в). З цієї характеристики визначають значення завадостійкості для низького рівня на вході M_L (перешкода, що відкриває) і для високого рівня на вході M_H (перешкода, що закриває): $M_L = U_{TIL} - U_{ILmax}$; $M_H = U_{THmin} - U_{TIH}$.

Середня споживана потужність P^*_{CC} елементом від джерела живлення обчислюється за формулою $P^*_{CC} = U_{CC}(I_{CCL} + I_{CCH}) / 2 = U_{CC} I^*_{CC}$, де I_{CCL}, I_{CCH} – струми споживання при низькому і високому рівнях напруги на виході відповідно; I^*_{CC} – середній струм споживання.

Сучасні елементи споживають потужність від мікроват до десятків міліватів.

Потенціальні сигнали характеризуються значенням логічного перепаду (амплітудою) $U_M = U_H - U_L$ і тривалістю позитивного t_{WH} та негативного t_{WL} перепадів (рис. 2.10). Перепади напруг часто називають позитивними і негативними імпульсами.

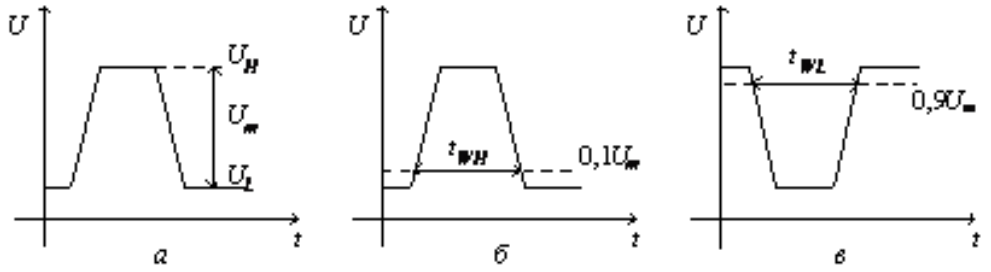


Рис. 2.10. Вимір параметрів сигналу: а – амплітуди; б, в – тривалості перепадів

Для вимірювання часових параметрів сигналу встановлюють умовні рівні в частках від амплітуди – 0,1; 0,5 і 0,9.

Швидкодію мікросхем визначають за значеннями таких тривалостей:

- фронту t_{LH} і спаду t_{HL} (рис. 2.11, а);
- власне вмикання t_{THL} і вимикання t_{TLH} (рис. 2.11, б); та їх затримки відповідно t_{DHL} та t_{DLH};
- затримок поширення сигналу при вмиканні t_{PHL} і вимиканні t_{PLH} (рис. 2.11, в).

Для практичних розрахунків використовують середній час затримки поширення сигналу $t_P = (t_{PHL} + t_{PLH}) / 2$.

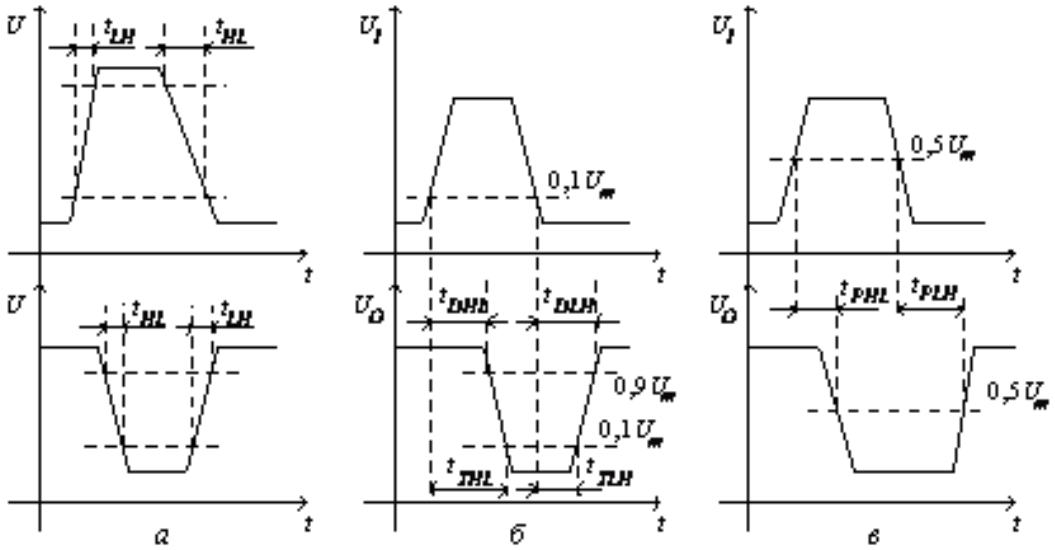


Рис. 2.11. Вимір часових параметрів сигналів:
а – фронту t_{LH} і спаду t_{HL}; б – часу вмикання t_{THL} і вимикання t_{TLH};
в – часу затримок розповсюдження сигналу при вмиканні t_{PHL} та вимиканні t_{TLH}

Для оцінки якості елемента широко використовують узагальнений параметр – роботу перемикання $A_П = P^*_{CC} t_P$.

Якщо потужність P^*_{CC} вимірюється в міліватах, а час затримки – в наносекундах, то робота перемикання $А_П$ виражається в пікоджоулях (пДж). Значення узагальненого параметра $A_П$ знаходиться в границях 0,1–200 пДж. Чим менше значення $A_П$, тим кращі характеристики має логічний елемент.

Надійність ІМС характеризується трьома взаємозалежними показниками: 1) інтенсивністю відмов $l = n / (mt)$, де n – число відмов за час випробування, год; m – загальна кількість випробуваних мікросхем; 2) напрацюванням на відмову $T = 1/l$; 3) можливістю безвідмовної роботи протягом заданого інтервалу часу $P = \exp(-lt)$.

Для сучасних ІМС інтенсивність відмов $l = (10^{-7}…10^{-8})$. Прийнявши, що $l = 10^{-8}$, $t = 15000$, одержимо значення ймовірності безвідмовної роботи $P(t) = 0{,}998$ або 99,8%.

3. ЕЛЕМЕНТИ КОМП'ЮТЕРНОЇ СХЕМОТЕХНІКИ

По типах базових елементів електронних ключів розділяють декілька типів схем, з яких найбільш поширеними є наступні:

- транзисторно-транзисторна логіка (ТТЛ);
- логіка з емітерними зв'язками (ЕЗЛ);
- інжекційна інтегральна логіка ($І^2Л$);
- структури метал – оксид – напівпровідник р-типу (р-МОН);
- структури метал – оксид – напівпровідник n-типу (n-МОН);
- комплементарні МОН- структури (КМОН);
- динамічні МОН- структури.

Більш старі види схемотехніки, такі, як резистивно-транзисторна логіка (РТЛ) і діодно-транзисторна логіка (ДТЛ), зараз практично не застосовуються.

Логічні схеми, що виготовляються на різній конструктивно-технологічній основі, значно розрізнюються за своїми характеристиками, навіть якщо вони реалізовують однакові функції. У кожного з вказаних видів схемотехніки є свої переваги. Так, ЕЗЛ відрізняється високою швидкодією, хоч деякі різновиди ТТЛ наближаються до неї по цьому параметру. Як рМОН, так і пМОН логіка широко застосовуються в мікропроцесорах, а КМОН схемотехніка користується перевагою, коли важливе зменшення споживаної потужності.

Динамічні МОН-структури використовуються для побудови різних запам'ятовуючих пристроїв; вони мають просту організацію, в якій логічний стан визначається зарядом ємності, внутрішньо властивої логічному елементу. $І^2Л$ застосовується в інтегральних схемах.

3.1. Транзисторно-транзисторні логчні елементи

Цей вид схемотехніки має, мабуть, найбільше число різновидів, тому що тривалість його використання дозволила виявити багато з властивих йому обмежень та способи їхнього подолання. Схемотехнічно більшість ІС, що входять до складу серій ТТЛ, виконано на основі комбінації двох базових схем: елемента І-НІ (штрих Шеффера) і розширника по АБО.

Елемент І-НІ (рис. 3.1) може бути представлений послідовним з'єднанням трьох каскадів:

- вхідного багатоемітерного транзистора VT1 з резистором R1 і діодами VD_o—VD_{n-i}, що реалізують логічну операцію І;
- фазорозщепителя на транзисторі VT2, резисторі R2 і кола нелінійної корекції R3, R4 та VT3;
- двотактного вихідного підсилювача на транзисторах VT4 і VT5 , резисторі R5 та діоді VD_n.

Розширник по АБО (рис. 3.2) по суті повторює перші два каскади елемента І-НІ і містить вхідний багатоемітерний транзистор VT1 з резистором R1 і транзистор VT2 фазорозщепителя.

Об'єднання елементів І-НІ і розширника по АБО шляхом з'єднання точок а і б дозволяє одержати ЛЕ, що реалізує послідовність операцій І-АБО-НІ.

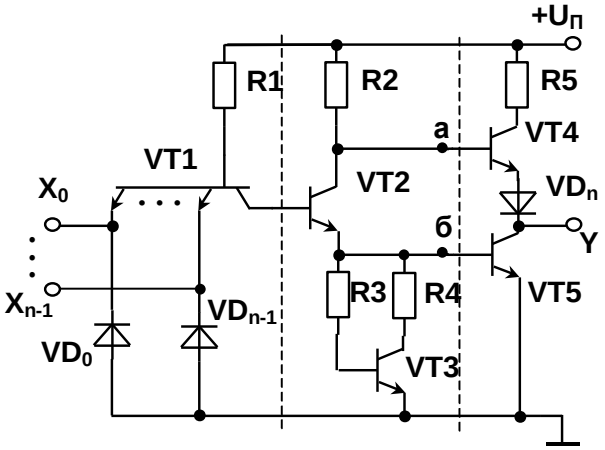


Рис. 3.1. Принципова електрична схема ЛЕ І-НІ ТТЛ (штрих Шеффера)

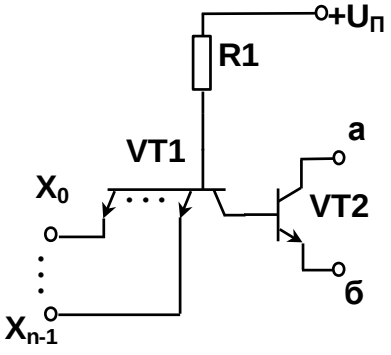


Рис. 3.2. Принципова електрична схема розширника АБО серії ТТЛ

Розглянемо роботу елемента І-НІ (рис. 3.1). Припустимо, що хоча б один вхід елемента X_0,..., X_{n-1} безпосередньо підключений до спільної шини, тобто на нього подана напруга лог. «0». У цьому випадку багатоемітерний транзистор VT1 виявляється насиченим струмом, що протікає від джерела живлення через резистор R1. Напруга на його колекторі буде мало відрізнятися від нульової. Тому транзистор VT2 фазорозщепителя виявляється замкненим.

Так як емітерний струм транзистора VT2 практично дорівнює нулю, буде замкнений і транзистор VT5. Струм, що протікає через резистор R2, втікає в базу транзистора VT4, насичує його. Тому напруга присутня на виході Y, ЛЕ близька до напруги живлення і визначається виразом $U^1_{вих} = U_ж - I_{вих}R_5 - U_{КЭVT4} - U_{VD_n}$.

Таким чином, при наявності на кожному із входів схеми напруги низького рівня, напруга на його виході буде мати високий рівень. Припустимо тепер, що на усі входи ЛЕ поданий високий рівень напруги. У цьому випадку всі емітерні переходи багатоемітерного транзистора VT1 виявляються замкненими. При цьому його колекторний перехід зміщається в прямому напрямку і по колу резистор R1, колекторний перехід транзистора VT1 та послідовно з'єднаним емітерним переходам транзисторів VT2 і VT5 буде протікати струм. Цей струм наситить транзистори VT2 і VT5, і на виході у ЛЕ установиться низька напруга, чисельно рівна напрузі насичення транзистора VT5 $U^0_{вих} = U_{КЭVT5} = I_{вих}r_{вихVT5}$

Так як транзистор VT2 насичений, то присутня на його колекторі напруга виявляється недостатньою для прямого зсуву двох послідовно включених p-n-переходів (емітерний перехід VT4 і діод VD_n). Транзистор VT4 буде замкнений.

Таким чином, якщо на усіх входах схеми присутня висока напруга, на виході БЛЕ будемо мати напругу низького рівня.

Паралельне з'єднання кількох виходів таких елементів неприпустимо, тому що формування на них сигналів різних рівнів призведе до виходу з строю транзисторів вихідного підсилювача потужності. Для позитивної логіки описаний алгоритм роботи відповідає визначенню операції І-НІ $y = \overline{x_0 x_1 x_{n-1}}$.

Виконання вихідного каскаду елемента за двотактною схемою дозволяє одночасно вирішити дві задачі:

1. Підвищити швидкодію елемента. Навантаження БЛЕ, як правило, носить ємнісний характер, і застосування двотактного вихідного каскаду дозволяє збільшити струм перезаряду ємності навантаження;

2. Знизити споживання. У сталому режимі лог. «0» через вихідний каскад протікає тільки струм навантаження.

В вхідному колі багатоемітерного транзистора VT1 включені додаткові діоди VD_0–VD_{n-1}, що захищають елемент від появи на його вході неприпустимих напруг зворотної полярності.

Бажання зменшити час розсмоктування, що при пасивному вимиканні складає значну частину часу вимикання біполярного транзистора, привело до застосування в БЛЕ діодів і транзисторів Шоткі. Діод Шоткі має істотно меншу граничну напругу відкривання, ніж p-n- перехід транзистора. Тому під час дії вхідного імпульсу діод Шоткі відкривається раніш, ніж колекторний перехід транзистора, запобігаючи нагромадження надлишкового заряду в його базовій області. Нагромадження заряду в самому діоді Шоткі не відбувається, тому що струм цього діода обумовлений переносом основних носіїв заряду. Таким чином включення паралельно колекторному переходу транзистора діода Шоткі при подачі вхідної напруги, автоматично фіксує напругу колектор — база на рівні, близькому до нульового. Транзистор при цьому працює поблизу границі режиму насичення.

Робота транзистора з діодом Шоткі в активному режимі приводить до збільшення напруги на його переходах, що трохи знижує споживану елементом потужність і змінює асимптотичні рівні лог. «0» і лог. «1».

Дане схемотехнічне рішення реалізоване в різновиді БЛЕ ТТЛ, іменованої ТТЛШ.

На рис. 3.3 приведена принципова електрична схема БЛЕ ТТЛШ серії 555. Від схеми стандартного БЛЕ вона відрізняється застосуванням у вихідному двотактному підсилювачі потужності складеного транзистора, що сприяє підвищенню швидкодії, і виконанням вхідного каскаду, що реалізує логічну операцію І. Операція І в даному елементі виконується з використанням діодних ключів на елементах VD1, VD2,..., VD_{2n-1}, VD_{2n}. При подачі на усі вхідні виводи елемента напруги

високого рівня, парні діоди VD2,...,VD_{2n} вхідних діодних ключів зміщаються в зворотному напрямку. Тому струм резистора R1 насичує транзистори VT1 і VT5, формуючи на виході елемента сигнал низького рівня.

Якщо хоча б на один вхід схеми подана низька напруга, відповідний діод зміщається в прямому напрямку. Струм резистора R1 замикається на загальну шину, минаючи емітерні переходи транзисторів VT1 і VT5. Останні запираються. При цьому струм резистора R2 насичує складений транзистор VT4', VT4, формуючи на виході напругу високого рівня.

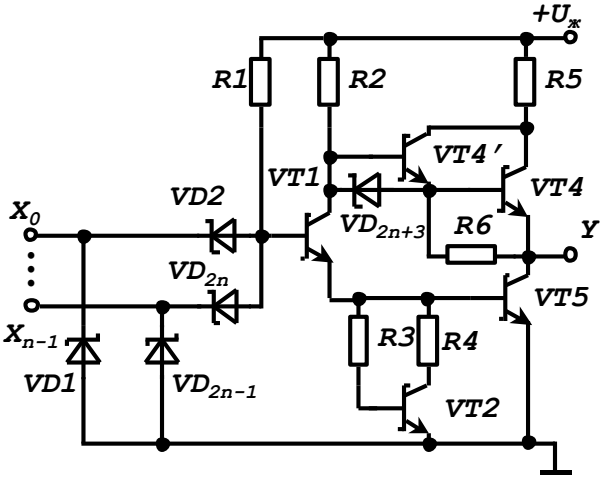


Рис. 3.3. Принципова електрична схема БЛЕ ТТЛШ серії 555

Завдяки такій побудові, вхідні виводи елемента можуть безпосередньо підключатися до шини джерела живлення, що не рекомендується робити для стандартних елементів ТТЛ.

Слід зазначити, що підвищення швидкодії, досягнуте застосуванням у схемі БЛЕ діодів і транзисторів Шоткі, дозволило розробити серію ІС ТТЛ зі зменшеним споживанням при швидкодії, порівнянній зі стандартними елементами. Останнє важливо з погляду підвищення функціональних можливостей ІС.

Дійсно, при використанні заданого корпуса, потужність, що може розсіяти одна ІС при заданих параметрах навколишнього середовища, строго задана. Це значить, що задано і та гранична кількість напівпровідникова приладів, що може бути розміщене в даній ІС не залежно від досконалості використовуваної технології їхнього виготовлення. Отже, обмежена і кінцева функціональна складність схеми, що може бути реалізована на цих елементах.

Потужність, необхідна для роботи схем ТТЛШ зі зменшеним споживанням, приблизно в 5 разів менше, ніж у стандартної схеми ТТЛ. Тому, якщо дозволяє технологія, у тім же корпусі можна розмістити в 5 разів більше напівпровідникових елементів і значно підвищити складність розробленої ІС і коло розв'язуваних з її допомогою задач.

З цієї причини при створенні ВІС і СБІС знаходить практичне застосування тільки даний тип базових схем.

Усі ІС ТТЛ, що випускає вітчизняна промисловість у даний момент, можна розділити на наступні групи:

- стандартні - серія 155;
- швидкодіючі з діодами Шоткі - серії 530, 531, l531;
- малопотужні з діодами Шоткі - серії 533, 555, 1533.

Елементи всіх цих серій практично виконані за єдиною схемою. Основне розходження даних серій полягає в їхній швидкодії і споживаній потужності.

Напруга всіх БЛЕ ТТЛ дорівнює +5В ±5% чи +5В±10% у залежності від типу виконання.

3.2. Логічні елементи емітерно-зв′язної логіки

Причиною появи БЛЕ ЕЗЛ стало бажання підвищити швидкодію цифрових пристроїв. Це бажання привело до використання в них зовсім відмінного від ТТЛ схемотехнічного рішення.

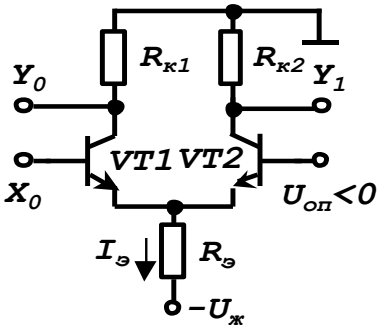


Рис. 3.4. Принципова електрична схема струмового ключа БЛЕ

Основою БЛЕ ЕЗЛ є струмовий ключ, виконаний на двох транзисторах (рис. 3.4). На базу одного з них, наприклад VT2, подана деяка постійна опорна напруга $U_{оп}$. Зміна напруги, поданої на вхід X_o нижче або вище $U_{оп}$ приводить до перерозподілу постійного струму I_e, заданого струмостабілізуючим резистором $R_Э$, між транзисторами VT1 і VT2. При цьому транзистори не попадають у режим насичення і, отже, у ключі принципово відсутній інтервал розсмоктування їхніх неосновних носіїв. Таким чином, особливістю БЛЕ ЕЗЛ є сталість споживаного струму не залежно від значення вихідного сигналу ключа.

Ця особливість вигідно відрізняє БЛЕ ЕЗЛ від БЛЕ ТТЛ, у якому в момент переключення струм, споживаний елементом різко зростає, створюючи внутрішні перешкоди, що погіршують перешкодозахищеність цифрового пристрою.

Загальною шиною для елемента є шина $+U_ж$, у результаті чого всі потенціали точок схеми негативні щодо загальної шини. Однак у схемі струмового ключа так само, як і в схемах ТТЛ, реалізований принцип позитивної логіки, при якому більшій вихідній напрузі відповідає сигнал лог. «1», а меншій — сигнал лог. «0».

Швидкодія струмового ключа дуже велика, тому що, по-перше, транзистори принципово не заходять в область насичення і, по-друге, малий логічний перепад напруг між значеннями лог. «0» та лог. «1». Останнє реалізовано вибором малих опорів резисторів $R_{к1}$ і $R_{к2}$ схеми, що дуже корисно з погляду зменшення постійної часу перезаряду вихідної ємності транзистора.

З струмового ключа знімаються одночасно два сигнали - прямий й інверсний, зв'язані із сигналом X_0 на вході схеми співвідношеннями: $Y_1 = X_0$; $Y_0 = \overline{X}_0$.

Вихідна напруга, що знімається з виходів Y_1 і Y_0 завжди більше $U_{оп}$, тому що транзистори VT1 і VT2 завжди працюють у ненасиченому режимі. Тому безпосереднє послідовне включення декількох таких ключів неможливо. Для цього необхідний каскад узгодження. Як такий каскад узгодження використовуються схеми емітерних повторювачів, включених між виходами струмового ключа і виходами елемента.

Повна схема БЛЕ, виконаного на основі струмового ключа, приведена на рис. 3.5 (елемент серії 500). Базовий елемент отриманий шляхом заміни вхідного транзистора VT1 струмового ключа групою паралельно включених транзисторів VT1—VT_n. Функціонально схему БЛЕ можна розбити на три вузли:

- струмовий ключ на транзисторах VT1—VT_{n+1} і резисторі R_{n+2};
- джерело еталонної напруги, що включає параметричний стабілізатор на елементах R_{n+4},VD1, VD2, R_{n+5} емітерний повторювач на VT_{n+2} і R_{n+3};
- вихідні емітерні повторювачі на транзисторах VT_{n+3} і VT_{n+4}.

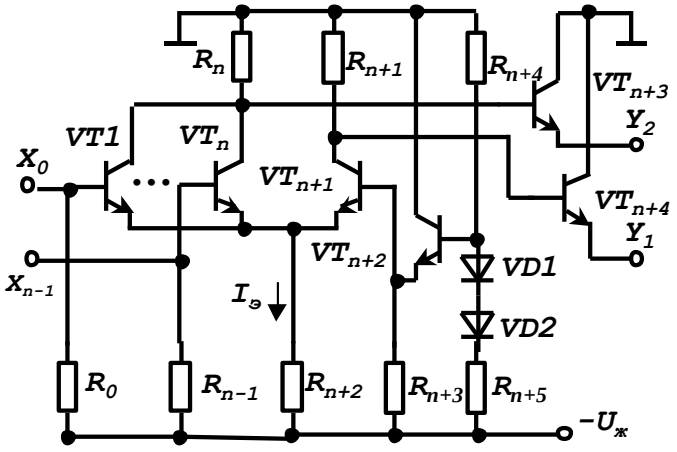


Рис. 3.5. Принципова електрична схема БЛЕ ЕЗЛ

Кола навантажень транзисторів VT_{n+3} і VT_{n+4} винесено з ІС БЛЕ, що сприяє зниженню в ній потужності розсіювання та розширенню функціональних можливостей.

Діоди VD1, VD2 у ланцюзі джерела, що задає, еталонної напруги служать для термокомпенсації напруги $U_{оп}$.

3.3. Логічні елементи на МДН-транзисторах.

Широкому поширенню МДН- схемотехніки сприяли наступні її особливості:

- більш проста технологія виготовлення (коротше технологічний цикл виготовлення), що сприяє підвищенню відсотка виходу придатних виробів;
- менші геометричні розміри приладу і більш прості схемотехнічні рішення, що при однаковій з біполярної ИС площі кристала дозволяє розмістити на ньому більш складну по виконуваних функціях схему чи при однаковій функціональній складності одержати меншу площу кристала, що також сприяє підвищенню виходу придатних виробів (менше вплив дефектів вихідного напівпровідника).

Разом з тим при розробці ІС МДН доводиться враховувати наступні особливості даного класу приладів:

1) МДН-транзистор володіє гіршими ключовими властивостями в порівнянні з біполярним транзистором. Це виражається в більшому вихідному опорі і, отже, у більшій залишковій напрузі у включеному стані;

2) залишкова напруга на включеному МДН- транзисторі значно залежить від керуючої напруги. Стосовно до логічних схем це означає сильну залежність напруги лог. «0» від напруги лог. «1». Послабити цю залежність можна зменшенням абсолютного струму стоку у включеному стані, що вимагає використання в ключі на МДН- транзисторі високоомного навантаження;

3) необхідність застосування власного високоомного навантаження при фіксованій ємності (вхідної ємності аналогічного елемента) збільшує постійну часу, що визначає тривалості фронту і спаду вихідної напруги ЛЕ. Це приводить до падіння його швидкодії;

4) у силу розглянутих вище особливостей (нестабільність вихідних логічних рівнів) для забезпечення достатньої перешкодостійкості ЛЕ на МДН- транзисторах повинні працювати при великих значеннях логічного перепаду. Це також сприяє падінню швидкодії елемента.

Слід зазначити, що необхідність застосування високоомного навантаження має і позитивну властивість, що виражається в зменшенні потужності, яка розсіюється в ключі на МДН- транзисторі.

Схемотехнічні рішення, використовувані при побудові ІС МДН, спрямовані на усунення вищеописаних недоліків елементарного ключа. Тому при побудові ІС схема ключа з навантажувальним резистором не використовується. Широке застосування знайшла схема ключа з навантажувальним МДН-транзистором, що забезпечує збільшення струму перезаряду ємності навантаження, а, отже, і швидкодії ключа. Це додатково дозволяє спростити технологію виготовлення ІС, тому що зі схеми виключаються всі пасивні елементи (резистори) і вона будується тільки на однотипних елементах – МДН- транзисторах.

В залежності від типу використовуваного транзистора розрізняють ІС nМОН і pМОН-типів. Розглянемо побудову БЛЕ з використанням nМОН-транзисторів. На рис. 3.6, а, б приведені принципові електричні схеми двохвходових БЛЕ, що реалізують операції 2І-НІ і 2АБО-НІ.

Обидві схеми містять по три транзистора, з яких VT1 виконує роль активного навантаження, а VT2 і VT3 є власне транзисторними ключами, що реалізують логічні операції. У схемі рис. 3.6 а, що реалізує логічну операцію І-НІ, транзистори VT2 і VT3 включені послідовно. Тому для появи на виході схеми низької напруги на затвори обох транзисторів необхідно подати високу напругу, достатню для протікання струму активного навантаження. У схемі рис. 3.6 б, транзистори VT2 і VT3 включені паралельно. Тому при подачі на затвор кожного з них високої напруги на виході буде сформована напруга низького рівня.

Збільшення числа вхідних змінних елемента вимагає збільшення кількості послідовно або паралельно включених транзисторів.

Тому, використовуючи описаний принцип з мінімальними схемотехнічними витратами, можна легко побудувати логічний елемент із необхідним числом входів.

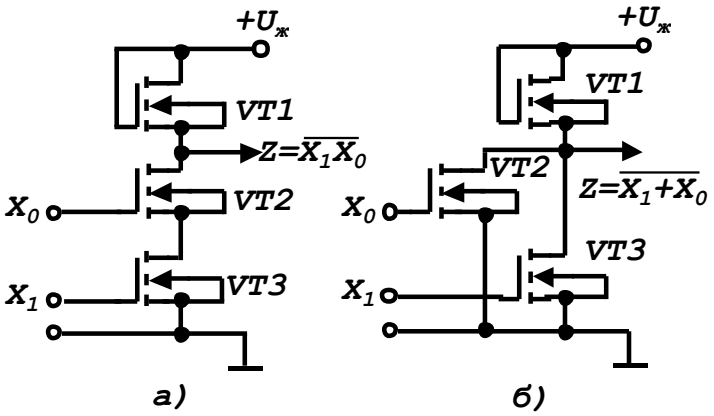


Рис. 3.6. Принципові електричні схеми БЛЕ на МДП- транзисторах: 2І-НІ (а) та 2АБО-НІ (б)

3.4. Логічні елементи інтегрально-інжекційної логіки.

Для підвищення технологічності виготовлення бажано при розробці ІС застосовувати схемотехнічні рішення, що використовують тільки однотипні елементи, наприклад транзистори. Цей шлях, як було показано раніше, реалізований у ІС МДП, що поряд з іншими перевагами є причиною їх широкого поширення. Однак, як уже відзначалося, ключ на біполярних транзисторах на сьогодні має кращі як ключові, так і частотні властивості. Це є передумовою до постійного пошуку нових схемотехнічних рішень для реалізації біполярних ІС. Такий пошук призвів до майже одночасної розробки фірмами Philips і IBM елемента інтегральної інжекціонної логіки ($І^2Л$). Відповідна йому принципова електрична схема БЛЕ $І^2Л$ наведена на рис. 3.7.

Особливістю елементів $І^2Л$ є:

- відсутність резисторів, що різко спрощує технологію виробництва ІС;
- використання токового принципу живлення, при якому в ІС задається не напруга, а струм, який безпосередньо інжектується в область напівпровідника, що утворює структуру одного з транзисторів;
- просторове сполучення в кристалі напівпровідника областей, що функціонально належать різним транзисторам. При цьому структура розташовується як по горизонталі (планарно), так і по вертикалі. Таке рішення дозволяє відмовитися від застосування спеціальних рішень для відділення областей, що належать різним елементам, як це необхідно робити в елементах ТТЛ і ЕЗЛ.
- мале значення логічного перепаду, що дозволяє максимально збільшити швидкодію елемента.

В приведеній схемі (рис. 3.7) багатоколекторний транзистор VT2 виконує функцію інвертування вхідного сигналу, а транзистор VTI — генератора (інжектора) базового струму транзистора VT2, До особливостей елемента варто віднести і сталість струму інжектора у всіх режимах роботи елемента. Струм інжектора задається резистором R, що, як правило, виконується загальним на групу елементів.

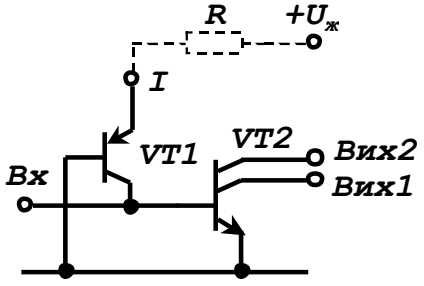


Рис. 3.7. Принципова електрична схема БЛЕ $І^2Л$.

Важливою особливістю елемента $І^2Л$ є можливість варіюючи струм інжектора в широких межах, змінювати його швидкодію. Реально струм інжектора може змінюватися від 1 на до 1 мА, тобто на 6 порядків. Оскільки для заданої схемотехніки енергія переключення елемента - величина

постійна, у таких же межах може змінюватися і швидкодія елемента. Важливо, щоби для цього не потрібно було б ніяких схемотехнічних змін в елементі.

Принцип дії схеми $I^2Л$ полягає в наступному. Допустимо, зовнішній сигнал що відповідає сигналу лог. «1» на вході елемента (база транзистора VT2) відсутній. У цьому випадку струм інжектора, втікаючи в базу транзистора VT2, насичує його. На його колекторах, а, отже, і на вихідних виводах елемента присутня напруга низького рівня, яка дорівнює напрузі насичення транзистора VT2. Реально це 0,1В ...0,2В.

Якщо база транзистора VT2 безпосередньо чи через насичений транзистор підключена до загальної шини, то виконується умова $U_{вх}<U_{бэо}$ транзистор VT2 замкнений, тому що струм інжектора замикається на загальну шину, минаючи його емітерний перехід. У цьому випадку напруга на його колекторах визначається зовнішніми колами. При послідовному включенні декількох інверторів ця напруга дорівнює напрузі емітерного переходу наступного транзистора. Таким чином, для БЛЕ $I^2Л$ справедливі наступні співвідношення: $U^0 = 0,1. ..0.2$ В; $U^1 = 0,6...0,7$ В.

З приведених співвідношень випливає, що логічний перепад для БЛЕ $I^2Л$ складає 0,4...0,6 В.

Особливістю елементів $I^2Л$ є можливість паралельного включення декількох їхніх виходів. З наведеної схеми випливає, що при паралельному включенні декількох виходів у загальній точці щодо вхідних змінних реалізується логічна операція АБО-НІ. Щодо вихідних сигналів елементів реалізується логічна операція І. Таким чином, якщо не потрібний гальванічний поділ між вхідними і вихідними сигналами, то логічна операція І виконується без яких-небудь додаткових схемотехнічних витрат простим об'єднанням відповідних виходів БЛЕ. Після інвертування результату виконаної операції АБО-НІ додатковим елементом щодо вхідних перемінних реалізується логічна операція АБО, а щодо вихідних сигналів перших елементів — операція І-НІ.

БЛЕ $I^2Л$ дозволяє максимально уніфікувати структуру ІС, знизивши площу її кристала, і або зменшити її споживання, або підвищити швидкодію. Типовий час затримки поширення БЛЕ $I^2Л$ при струмі інжектора 0,1мка складає 10нс. При цьому енергія переключення для цього елемента на кілька порядків менше, ніж для елемента ТТЛ.

Через невелику перешкодостійкість, обумовлену малим логічним перепадом, БЛЕ $I^2Л$ використовуються винятково в складі ВІС і СВІС та як окремі ІС малого ступеня інтеграції не випускаються. При цьому вхідні і вихідні кола ІС, виконаних за технологією $I^2Л$, робляться сумісними по логічних рівнях із сигналами ТТЛ.

3.5. Тригери

Тригер – це запам'ятовуючий елемент з двома стійкими станами, зміна яких відбувається під дією вхідних сигналів. Як елемент комп'ютера, тригер призначений для зберігання одного біта інформації, тобто лог. 0 або лог. 1. Схема тригера забезпечує записування, зчитування, стирання та індикацію двійкової інформації, яка зберігається. На основі тригерів будують типові функціональні вузли комп'ютерів – регістри, лічильники, накопичувальні суматори, а також мікропрограмні автомати.

Усі різновиди тригерів являють собою елементарний автомат, який вміщує власне елемент пам'яті (ЕП) та схему керування (СхК), яка утворює вхідну логіку (рис.3.8).

Стан тригера визначається сигналами на прямому Q та інверсному $\overline{Q}$ виходах. При позитивному кодуванні інформації високий рівень напруги на прямому виході відображає значення лог. 1 (стан Q =1), а низький рівень – значення лог. 0 (стан Q = 0).

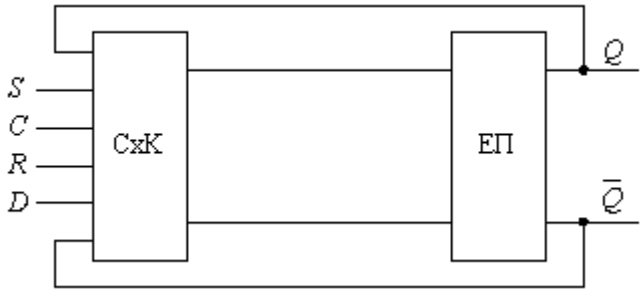


Рис.3.8. Структура тригера у вигляді ЕП і СхК

Зміна стану тригера (його перемикання) забезпечується зовнішніми сигналами й сигналами зворотного зв'язку на виході тригера, які поступають на входи СхК. Звичайно зовнішні сигнали, як і входи тригера, позначають латинськими буквами R, S, T, C, V та іншими. В найпростіших схемах тригерів окрема СхК може бути відсутньою. Оскільки функціональні властивості тригерів визначаються їхньою СхК, то назви основних входів переносяться на всю схему тригера.

3.5.1. Класифікація тригерів

Тригери класифікують за такими ознаками: логікою функціонування (RS, JK, D, T та ін.); способом записування інформації (асинхронні й синхронні); моментом реакції на тактовий сигнал (статичні, динамічні); кількістю тактів синхронізації (одно-, дво- і тритактові); кількістю ступенів (одно- або двоступеневі тригери); складом логічних елементів (тригери на елементах НЕ І, НЕ АБО, НЕ І АБО та ін.).

Відповідно до логіки функціонування розрізняють такі тригери: з роздільною установкою станів "0" і "1" (RS-тригери); з одним інформаційним входом (D-тригери); з лічильним входом (T-тригери); універсальні з роздільною установкою станів "0" і "1" (JK-тригери); комбіновані (RST-, RSJK-тригери); із складною вхідною логікою.

Входи тригерів розділяються на інформаційні (R, S, T та ін.) та керуючі (C, V). Інформаційні (логічні) входи призначені для приймання сигналів інформації, яка запам'ятовується. Назви вхідних сигналів ототожнюють з назвами входів тригера. Керуючі входи служать для керування записуванням інформації. У тригерах може бути два види керуючих сигналів: синхронізуючий (тактовий) сигнал C, який надходить до C-входу (тактового входу) і дозволяючий сигнал V, який надходить до V-входу.

За способом записування (приймання) інформації розрізняють асинхронні й синхронні (тактовні) тригери. Тригери, які не мають C-входу, називаються асинхронними (рис.3.9, а і б). В асинхронних тригерах записування інформації відбувається в будь-який момент часу при надходженні сигналів до інформаційних входів.

Тригери, які мають C-вхід, називаються синхронними. У синхронному тригері записування інформації можливе при збігу сигналів на інформаційному й синхронному входах. Цим пояснюється вища стійкість до перешкод синхронних тригерів порівняно з асинхронними.

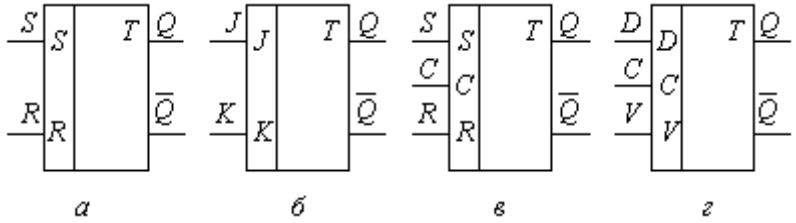


Рис.3.9. Умовні позначення тригерів: а, б – асинхронних; в, г – синхронних

До V-входів тригера надходять сигнали, які дозволяють (V = 1) або забороняють (V = 0) записування інформації. У синхронних тригерах з V-входом записування інформації можливе при збігу сигналів на інформаційному, C- і V- виходах (рис.3.9, г).

Залежно від кількості тактових сигналів, необхідних для формування нового стану, розрізняють однотактові, двотактові та багатотактові тригери.

За способом керування записуванням (моментом реакції на тактовий сигнал) виділяють синхронні тригери зі статичним (за рівнем), динамічним (за фронтами) та двоступеневим керуванням. В асинхронних тригерах записування нуля і одиниці можливе у будь-який момент часу, при цьому вхідний інформаційний сигнал одночасно є й керуючим. У синхронних тригерах з керуванням за рівнем записування інформації можливе тільки впродовж тривалості тактового сигналу. При цьому тактові сигнали можуть бути прямими (змінюватися від нуля до одиниці) або інверсними (змінюватися від одиниці до нуля) (рис.3.10, а і б).

При керуванні фронтами дозвіл на записування інформації дається тільки в момент перепаду тактового сигналу від нуля до одиниці (прямий динамічний вхід) або від одиниці до нуля (інверсний динамічний вхід). В інші моменти часу тригер не реагує на вхідні інформаційні сигнали незалежно від рівня тактового імпульсу (рис.3.10, в і г).

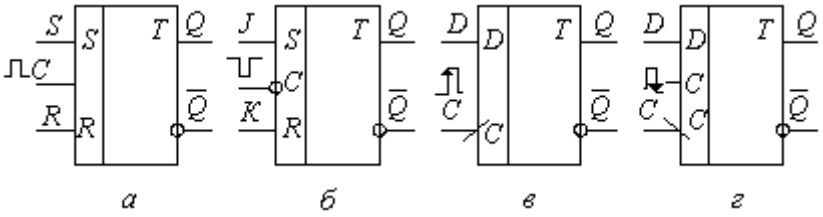


Рис.3.10. Керуючі входи тригера: а – прямий статичний; б – інверсний статичний; в – прямий динамічний; г – інверсний динамічний

3.5.2. Динамічні параметри тригерів

Для тригерів встановлені такі динамічні параметри, які вимірюються на рівні половини амплітуди сигналів (рис.3.11):

• мінімальна тривалість імпульсу t_C на тактовому вході;

• мінімальний час попереднього установлення сигналу на інформаційному вході $t_{уст}$;

• час відновлення (фіксації) $t_{відн}$ - мінімальний час між початком синхросигналу С і закінченням інформаційного сигналу D; для асинхронних тригерів $t_{відн}$ просто тривалість вхідного сигналу;

• час перемикання тригера $t_{п.т}$ – часовий інтервал між фронтом вхідного перемикаючого сигналу та фронтом сигналу на виході Q або $\overline{Q}$ (який пізніше набуває нового положення). Мінімальна тривалість синхросигналу на вході тригера визначається максимальним часом перемикання тригера $t_C \geq t_{п.т\,max}$. У двоступеневому тригері з однотактною синхронізацією час перемикання другого ступеня визначається відносно спаду синхроімпульсу.

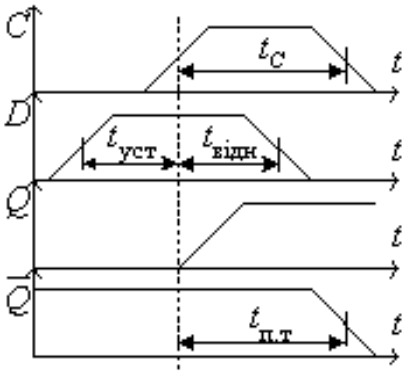


Рис. 3.11. Вимірювання часових параметрів тригерів D-типу.

3.5.3. RS-тригер

RS-тригером називають запам'ятовуючий елемент з роздільними інформаційними входами для установлення його в стан “0” (R-вхід) і в стан “1” (S-вхід). Назва “RS-тригер” утворена від перших літер слів RESET (скинення) і SET (установлення).

У таблиці переходів RS-тригера (рис.3.12) прийняті позначення: R_t, S_t, Q_t – значення логічних змінних у момент часу t на входах R, S і виході Q; Q_{t+1} – стан тригера після перемикання; К6, К7 – невизначені коефіцієнти на тих наборах, де вхідні сигнали R_t і S_t одночасно набувають значення одиниці (заборонена комбінація сигналів).

Таблиці переходів відповідає карта Карно (рис.3.12, а), де значення функції Q_{t+1} для мінтермів R_t, S_t, $\overline{Q}$, і R_t, S_t, Q_t замінені невизначеними коефіцієнтами К6 і К7. Припустивши, що комбінації вхідних сигналів $R_tS_t = 1$ не існує, отримаємо карти Карно для К6 = К7 = 1 (рис.3.12, б) і К6 = К7 = 0 (рис.3.12, в).

Із карт Карно отримуємо логічні рівняння асинхронного RS-тригера:

$$К6 = К7 = 1,\ Q_{t+1} = S_t \vee \overline{R}_t Q_t, \qquad (3.1)$$

$$К6 = К7 = 0,\ Q_{t+1} = \overline{R}_t (S_t \vee Q_t). \qquad (3.2)$$

Логічні вирази (3.1) і (3.2) визначають новий стан тригера Q_{t+1} залежно від старого стану Q_t та вхідних сигналів R_t і S_t. В подальшому для спрощення індекс t у правій частині логічного виразу опускається.

R_t	S_t	Q_t	Q_{t+1}
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	K_6
1	1	1	K_7

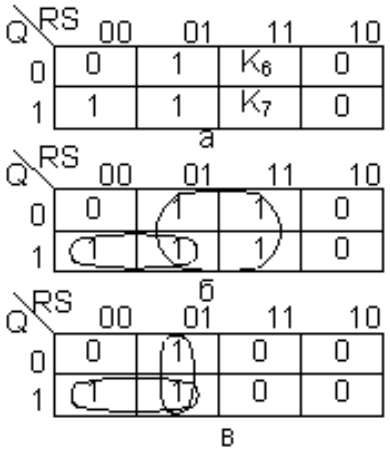


Рис.3.12. Карта Карно для RS-тригерів

Асинхронний RS-тригер на елементах НЕ І. Перетворимо логічний вираз (3.1) до виду, зручного для реалізації на елементах НЕ І:

$$Q_{t+1} = \overline{\overline{S \vee \overline{R}Q}} = \overline{\overline{S} \cdot \overline{\overline{R} \cdot Q}} \quad . (3.3)$$

Схема асинхронного RS-тригера на двох елементах НЕ І з логічними зв'язками на основі виразу (3.3) показана на рис.3.13, а. Особливістю цього тригера є інверсне керування за інформаційними входами, що відображається в умовному графічному позначенні.

Із аналізу діаграм роботи RS-тригера випливає, що елементи НЕ І в схемі перемикаються послідовно. Є інтервал часу, коли на обох виходах встановлюються однакові сигнали Q = 1 і $\overline{Q}$ = 1 (рис.3.13, в, заштриховані області) – явище "ризик".

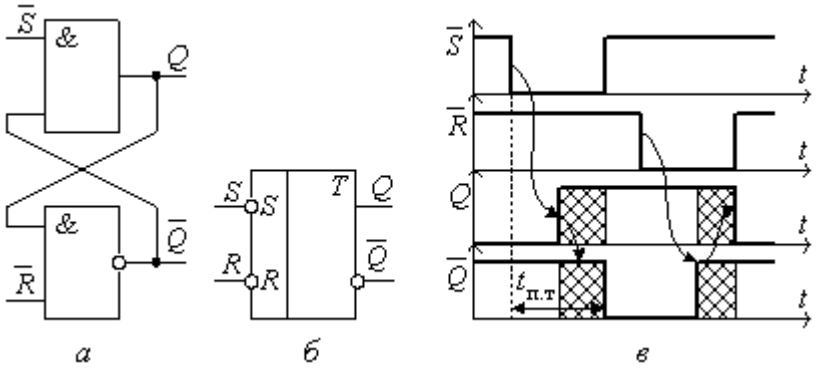


Рис.3.13. Асинхронний RS-тригер на елементах НЕ І:
а – схема; б – умовне позначення; в – часові діаграми

Тривалість перемикання тригера визначається сумою затримок: $t_{п.т} = 2t_p$. Тривалість вхідного сигналу визначається з умови $t_i \geq t_{п.т}$. На практиці для надійності перемикання тригера тривалість вхідного імпульсу збільшують на одну затримку, тобто $t_i = 3t_p$. Максимальна і робоча частоти перемикання тригера відповідно дорівнюють $f_{max} = 1/(2t_p)$ і $f_p = 1/(3t_p)$.

Асинхронний RS-тригер на елементах НЕ АБО. Перетворимо логічний вираз (3.2) до вигляду, зручного для реалізації на елементах НЕ АБО:

$$Q_{t+1} = \overline{\overline{R}(S \vee Q)} = \overline{R \vee \overline{(S \vee Q)}} \quad (3.4)$$

Схема асинхронного RS-тригера на двох елементах НЕ АБО з логічними зв'язками на основі виразу (3.4) показана на рис.3.14, а.

Із аналізу діаграм роботи RS-тригера випливає, що елементи НЕ АБО в схемі перемикаються послідовно. Є інтервал часу, коли на обох виходах установлюються однакові сигнали Q = 0 і $\overline{Q}$ =0, – явище "ризик"(рис.3.14, в). Часові параметри даного тригера аналогічні параметрам тригера, зображеного на схемі рис.3.13, а.

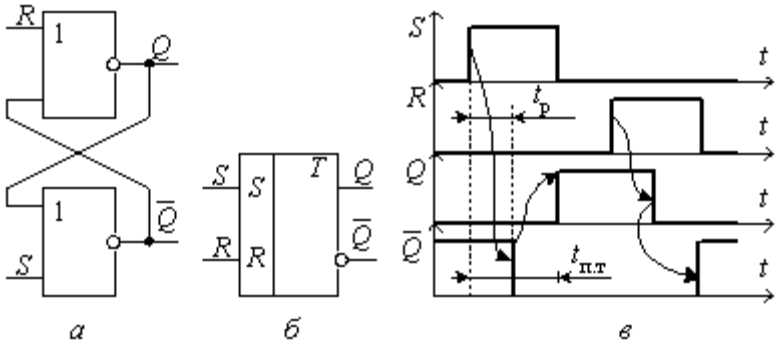


Рис.3.14. Асинхронний RS-тригер на елементах НЕ АБО:
а – схема; б – умовне позначення; в – часові діаграми

Синхронний RS-тригер на елементах НЕ І. Для побудови синхронного RS-тригера на елементах НЕ І треба замінити в логічному виразі (3.3) змінні S і R на сполучення CS і CR, де C – синхро-сигнал:

$$Q_{t+1} = \overline{\overline{CS} \cdot \overline{CR} \cdot Q}. \qquad (3.5)$$

Схема синхронного RS-тригера на чотирьох елементах НЕ І з логічними зв'язками на основі виразу (3.5) показана на рис.3.15, а. Елементи D1 і D2 складають схему керування з прямими входами, а елементи D3 і D4 утворюють фіксатор (асинхронний RS-тригер).

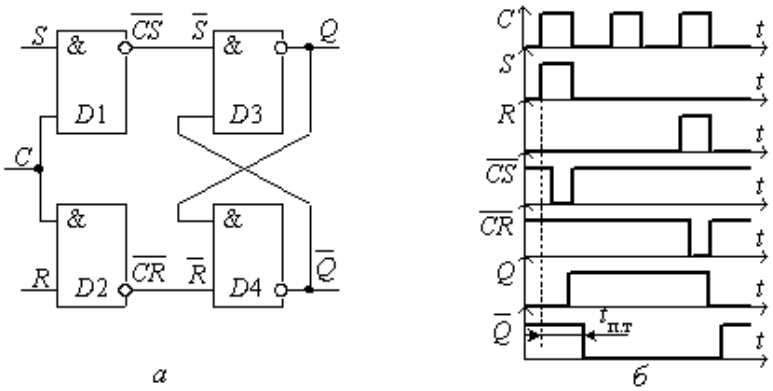


Рис.3.15. Синхронний RS-тригер на елементах НЕ І: а – схема; б – часові діаграми

При значенні сигналів CS = 1 на виході елемента D1 встановлюється лог. 0 і тригер перемикається в стан "1".

При значенні сигналів CR = 1 на виході елемента D2 встановлюється лог. 0 і тригер перемикається в стан "0". Комбінація вхідних сигналів CSR = 1 заборонена, оскільки призводить до невизначеного стану тригера.

Із часової діаграми (рис.3.15, б) випливає, що час перемикання тригера $t_{п.т} = 3t_p$, а тривалість синхросигналу (з урахуванням запасу на одну затримку) визначається з умови $t_C = 4t_p$. Максимальна і робоча частоти перемикання тригера відповідно дорівнюють: $f_{max} = 1/3t_p$ і $f_p = 1/4t_p$.

Синхронний RS-тригер на елементах НЕ АБО. Для побудови синхронного RS-тригера на елементах НЕ АБО належить замінити в логічному виразі (3.4) змінні S і R на сполучення $\overline{\overline{CS}}$ і $\overline{\overline{CR}}$:

$$Q_{t+1} = \overline{\overline{\overline{CR}} \vee \overline{(\overline{\overline{CS}} \vee Q)}} = \overline{\overline{\overline{C} \vee \overline{R}} \vee \overline{(\overline{\overline{C} \vee \overline{S}} \vee Q)}} \qquad (3.6)$$

Схема синхронного RS-тригера на чотирьох елементах НЕ АБО з логічними зв'язками на основі виразу (3.6) показана на рис.3.16. Елементи D1 і D2 складають схему керування з інверсними входами, а елементи D3 і D4 утворюють фіксатор (асинхронний RS-тригер).

При значенні сигналів $\overline{C} = 0$ і $\overline{S} = 0$ на виході елемента D2 встановлюється лог. 1 (тобто CS = 1) і тригер переключається в стан "1". При значенні сигналів $\overline{C} = 0$ і $\overline{R} = 0$ на виході елемента D1

встановлюється лог. 1 (тобто CR = 1) і тригер переключається в стан “0”. Комбінація сигналів $\overline{C} = \overline{S} = \overline{R} = 0$ заборонена, тому що призводить до невизначеного стану тригера.

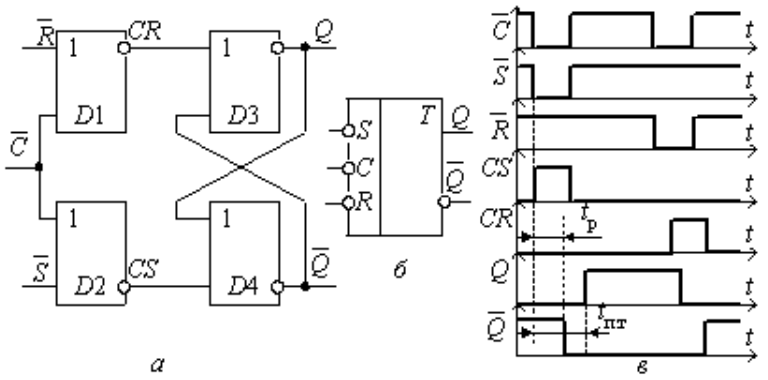


Рис.3.16. Синхронний RS-тригер на елементах НЕ АБО: а – схема; б – умовне позначення; в – часові діаграми

Двоступеневі RS-тригери. Двоступеневі тригери будують за способом “M–S” і забезпечують поєднання двох процесів – одночасного записування нової інформації та зчитування старої. Під час дії синхроімпульсу C перший ступінь “M” (Master – основний) приймає нову вхідну інформацію, а другий ступінь “S” (Slave – допоміжний) в цей же час передає у зовнішні схеми стару інформацію. Після закінчення синхроімпульсу C інформація з першого ступеня переписується у другий ступінь.

При однофазному (однотактному) обміні інформацією зв’язок між ступенями реалізується за допомогою інвертора (рис.3.17, а), забороняючих зв'язків (рис.3.17, б) або різнополярного керування (рис.3.17, в). При двотактному обміні зв'язок між ступенями забезпечується двома серіями синхросигналів – C1 і C2 (рис.3.17, г).

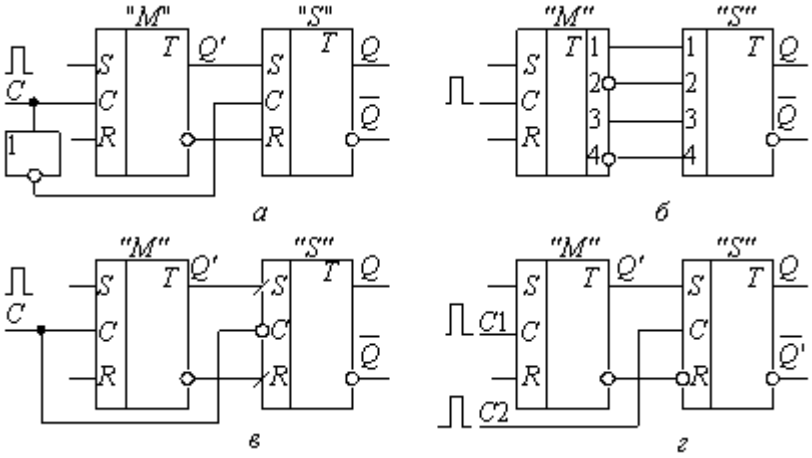


Рис.3.17. Організація зв'язку між ступенями тригера: а – з інвертором; б – із забороняючими зв'язками; в – з різнополярним керуванням; г – з двофазним обміном

3.5.4. Тригери типу JK.

Тригером типу JK називається запам'ятовуючий елемент з двома стійкими станами та інформаційними входами J (аналог S) і K (аналог R), які забезпечують відповідно роздільну установку станів “1” і “0”. Він функціонує подібно до RS-тригера, але при збігу сигналів JK = 1 переключається в протилежний стан, тобто реалізує додавання сигналів за модулем два. Таким чином, JK-тригер не має заборонених комбінацій вхідних сигналів. Тригер типу JK є універсальним, оскільки може виконувати функції RS-тригера (при роздільному надходженні сигналів J і K), T-тригера (при одночасній подачі сигналів J і K), D-тригера (при подачі сигналу від входу J через інвертор на вхід K). Зміна станів JK-тригера наведена на рис. 3.18, і за допомогою карти Карно отримуємо наступне рівняння для тригера:

$$Q_{t+1} = \overline{K}_t \cdot Q_t \vee J_t \cdot \overline{Q}_t . \qquad (3.7)$$

Для побудови одноступеневого синхронного JK-тригера на елементах НЕ І потрібно замінити в рівнянні (3.7) змінні К і J на сполучення СК і JK, після чого виконати перетворення на основі правил подвійної інверсії та правил де Моргана:

$$Q_{t+1} = \overline{\overline{\overline{C \cdot K} \cdot Q \vee C \cdot J \cdot \overline{Q}}} = \overline{\overline{\overline{C \cdot K} \cdot Q} \cdot \overline{C \cdot J \cdot \overline{Q}}}. \qquad (3.8)$$

Схема одноступеневого JK-тригера з логічними зв'язками на основі рівняння (3.8) показана на рис.3.19.

K_t	J_t	Q_t	Q_{t+1}
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

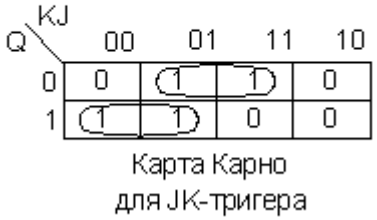


Рис. 3.18. Таблиця зміни станів та карта Карно JK-тригера

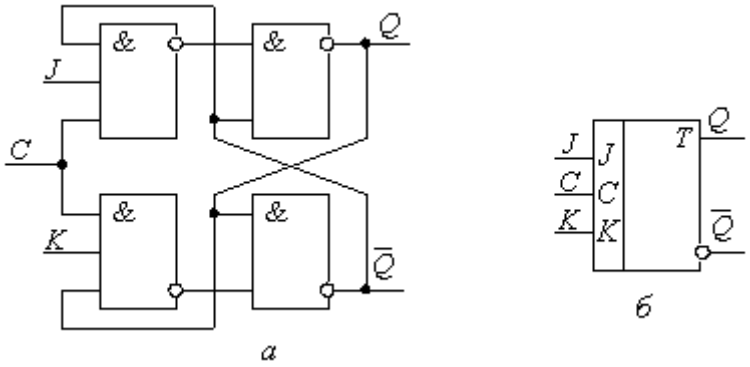


Рис.3.19. Одноступеневий JK-тригер: а – схема; б – умовне позначення

У схемотехніці потенціальних тригерів в основному застосовують двоступеневі синхронні JK-тригери на елементах НЕ І (рис.3.20). Нова інформація знімається з виходів Q основного М-ступеня, а стара – з виходів Q* допоміжного S-ступеня.

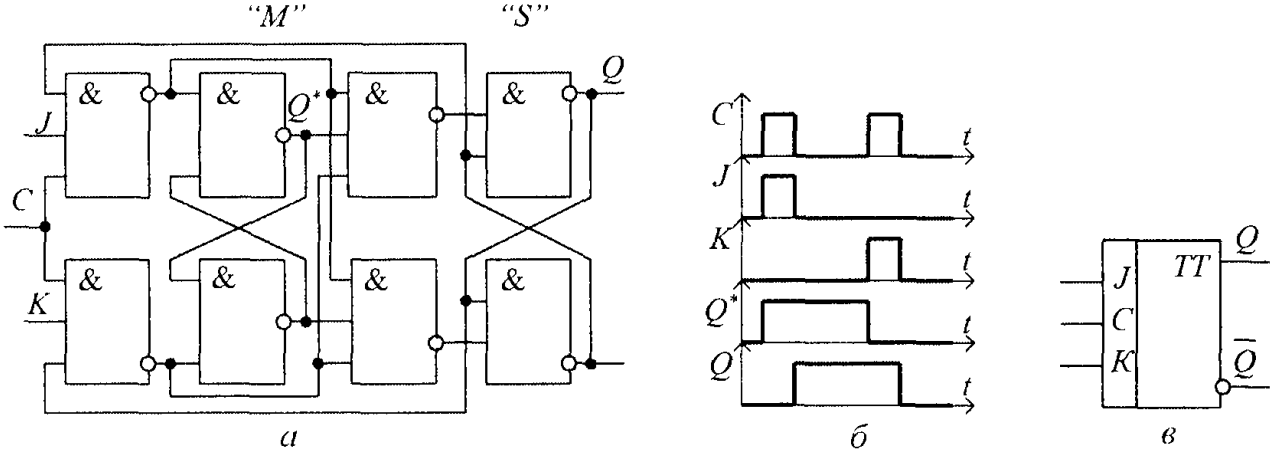


Рис.3.20. Двоступеневий JK-тригер на елементах НЕ І:
а – схема; б – часові діаграми; в – умовне графічне позначення

Із часових діаграм (рис.3.20, б) слідує: при застосуванні JK-тригера для зберігання інформації сигнали на входи J і K подаються почергово; при збігові сигналів на входах J і K реалізується лічильний тригер (на практиці ці входи часто монтажно об'єднуються). Час перемикання JK-тригера визначається сумою затримок першого і другого ступенів і дорівнює $t_{п.тт} = 7t_р$. На виході S-ступеня нова інформація з'являється після спаду сигналу С.

3.5.5. Т-тригер.

Тригером типу Т називається запам'ятовуючий елемент з двома стійкими станами та одним інформаційним Т-входом. Стан Т-тригера змінюється на протилежний після кожного надходження лічильного сигналу на Т-вхід. Логіка функціонування асинхронного лічильного тригера подана таблицею переходів (рис.3.21) і описується логічним рівнянням

$$Q_{t+1} = \overline{T}_t \cdot Q_t \vee T_t \cdot \overline{Q}_t . \qquad (3.9)$$

Для побудови асинхронного Т-тригера на елементах НЕ І рівняння (3.9) перетворюється до вигляду, зручного для реалізації у заданому елементному базисі:

$$Q_{t+1} = \overline{\overline{\overline{T} \cdot Q \vee T \cdot \overline{Q}}} = \overline{\overline{\overline{T \cdot Q} \cdot Q \cdot \overline{T \cdot \overline{Q}}}} . \qquad (3.10)$$

У рівнянні (3.10) для виключення інверсії сигналу Т використовується тотожність $\overline{T} \cdot Q = (\overline{T \cdot Q}) \cdot Q$. Перемикання тригера визначається сумісною дією лічильних сигналів “Т” і зворотного зв'язку виходів Q і $\overline{Q}$. Для виключення “гонок” у тригері сигнали зворотного зв'язку не повинні змінюватися під час дії лічильного сигналу “Т”. Затримка вихідного сигналу може здійснюватися лінією затримки (в імпульсно-потенціальній системі елементів) чи додатковим тригером (в потенціальній системі елементів). Схема одноступеневого асинхронного Т-тригера на елементах НЕ І з логічними зв'язками відповідно до рівняння (3.10) показана на рис.3.21. Сигнали з виходів елементів D1 і D2 затримуються на час Δt, що дорівнює тривалості лічильного сигналу на Т-вході.

T_t	Q_t	Q_{t+1}
0	0	0
0	1	1
1	0	1
1	1	0

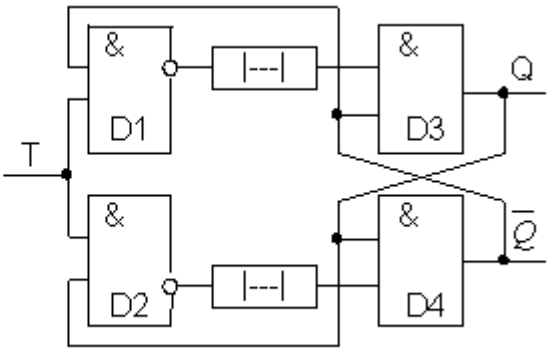


Рис. 3.21. Таблиця переходів та схема одноступеневого асинхронного Т-тригера на елементах НЕ І

Надійні структури лічильних тригерів будують з використанням двоступеневих тригерів або тригерів з динамічним керуванням по фронту сигналу С.

3.5.6. D-тригер.

Тригером типу D називається синхронний запам'ятовуючий елемент з двома стійкими станами і одним інформаційним D-входом. Закон функціонування D-тригера описується логічним рівнянням $Q_{t+1} = C_t D_t$.

Це рівняння показує, що після перемикання стан D-тригера повторює значення сигналу на D-вході в тактові моменти часу. Тому в літературі D-тригери часто називають тригерами затримки (від Delay – затримка).

Схему D-тригера можна побудувати на основі синхронного RS-тригера, якщо сигнал по входу S одночасно подавати через інвертор на вхід R (рис.3.22, а). Схеми D-тригера будують також на основі самостійного логічного рівняння. Перетворимо рівняння (3.5) замінивши сигнал S на D і сигнал R на $\overline{D}$:

$$Q_{t+1} = \overline{\overline{CS} \cdot \overline{CR \cdot Q}} = \overline{\overline{CD} \cdot \overline{C\overline{D} \cdot Q}} . \qquad (3.11)$$

Схема D-тригера на елементах НЕ І з логічними зв'язками згідно з рівнянням (3.11) показана на рис.3.22, б.

D-тригер “слідкує” за зміною сигналу на D-вході під час дії синхросигналу С і зберігає ту інформацію, яка була в момент його закінчення. RS-тригери такої властивості не мають і тому вони менше завадостійкі порівняно з D-тригерами.

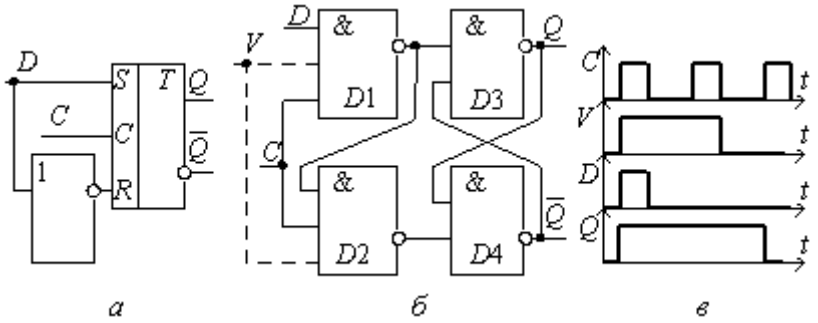


Рис.3.22. D-тригер: а – на основі RS-тригера; б – на елементах НЕ І; в – часові діаграми роботи

Для затримки інформації в D-тригері на довільне число тактів використовується дозволяючий V-вхід, як показано штриховою лінією на рис.3.22, б. Якщо V = 1, то DV-тригер функціонує як звичайний тригер затримки; якщо V = 0, то робота схеми за входами блокується і DV-тригер зберігає попередню інформацію.

Схема двоступеневого однотактного DV-тригера на елементах НЕ І із забороняючими зв'язками між ступенями показана на рис.3.23.

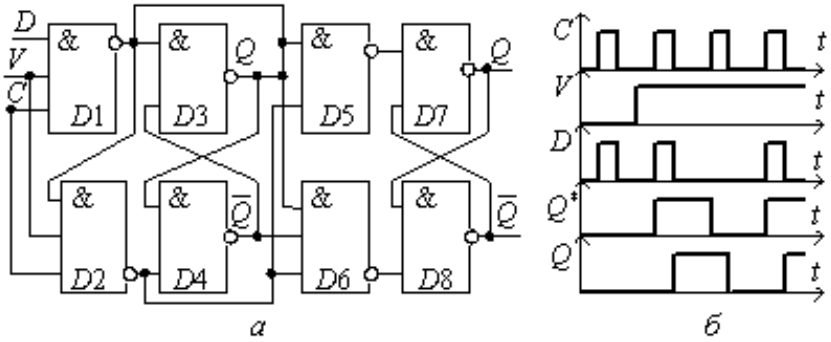


Рис.3.23. Двоступеневий DV-тригер: а – схема; б – часові діаграми

D-тригер з динамічним керуванням

В усіх синхронних тригерах із статичним керуванням (за рівнем) можливе фальшиве перемикання у випадку зміни сигналів на інформаційних входах під час дії синхросигналу С. Наприклад, якщо в D-тригері інформаційний сигнал змінює своє значення від одиниці до нуля до закінчення сигналу “С”, то тригер може знову переключатися з одиничного в нульовий стан. Тому для надійної роботи D-тригера потрібний певний інтервал часу між фронтом синхроімпульсу С і спадом сигналу на D-вході (параметр $t_{відн}$).

У тригерах з динамічним керуванням записуванням інформації синхроімпульс С активний лише на короткому інтервалі часу в області фронту чи спаду. Тому D-тригери з динамічним керуванням мають високу завадостійкість. На практиці широке розповсюдження отримали D-тригери з прямим динамічним керуванням за схемою “трьох тригерів”. Схема такого D-тригера з прямим динамічним керуванням показана на рис.3.24, а, а його умовне графічне зображення – на рис.3.24, б. Тут зберігання інформації здійснює основний вихідний синхронний RS-тригер (елементи D5 і D6) з інверсним керуванням, а приймання тактового і інформаційного сигналів і задання динамічного режиму роботи забезпечують два вихідних комутуючих тригери (елементи D1, D2, D3, D4).

Елемент D4 подає інверсне значення вхідного сигналу D на входи елементів D1 і D3 (рис.3.24, в). Елемент D1 повторює значення сигналу D.

При С = D = 1 вмикається елемент D2 і встановлює основний тригер за входом S1 в стан “1”; одночасно блокується робота елемента D3, в зв'язку з чим схема уже не реагує на зміну вхідного сигналу. При С = 1, D = 0 вмикається елемент D3 і встановлює основний тригер в стан “0”; одночасно відбувається приймання нової інформації елементом D4. При С = 0 забезпечується режим зберігання записаної інформації.

Час перемикання тригера за С-входом $t_{п.т}=3t_р$.

Після закінчення сигналів на D- і C-входах починається етап відновлення, який характеризується переходом комутуючих тригерів у початковий стан за час $3t_p$. Максимальна частота перемикання D-тригера з динамічним керуванням визначається сумарним часом затримок $f_{max} = 1/(6t_p)$.

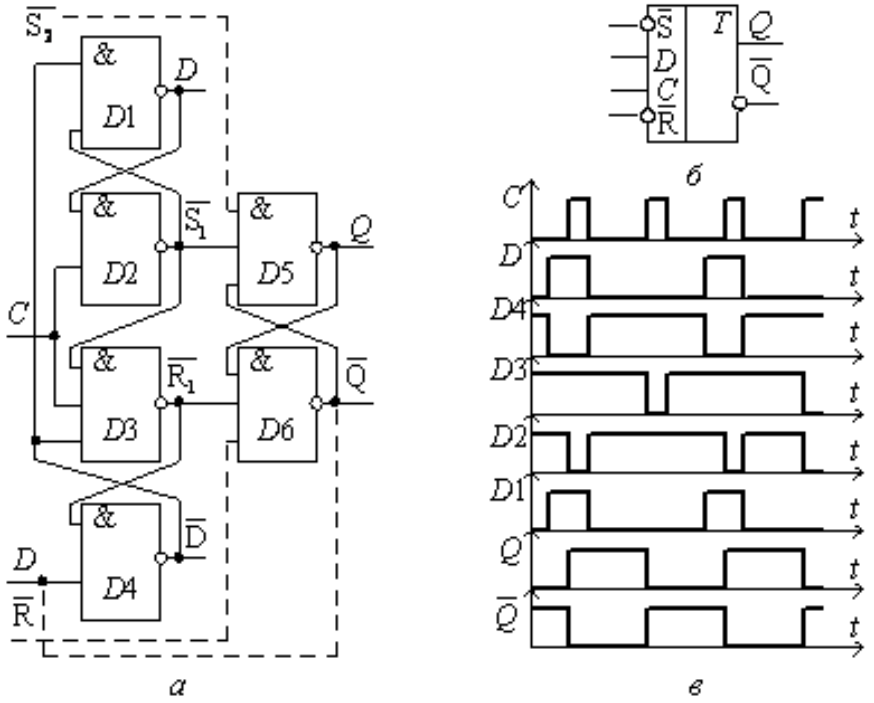


Рис.3.17. D-тригер з динамічним керуванням: а – схема; б – умовне позначення; в – часові діаграми

4. ПОСЛІДОВНІСНІ (НАКОПИЧУВАЛЬНІ) ВУЗЛИ КОМП'ЮТЕРНОЇ СХЕМОТЕХНІКИ

4.1. Регістри

Регістром називається типовий функціональний вузол комп'ютера, призначений для приймання, тимчасового зберігання, перетворення і видачі n-розрядного двійкового слова. Регістр містить регулярний набір однотипних тригерів, в кожному з яких зберігається значення одного двійкового розряду машинного слова. Найчастіше використовують тригери типів RS, JK і D (рис. 4.1).

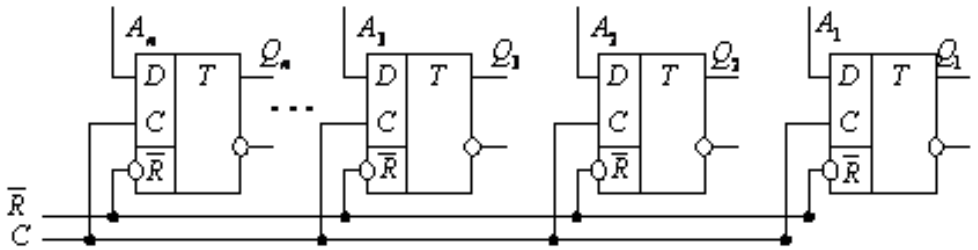


Рис. 4.1 Схема регістра на D-тригерах

Регістри, призначені тільки для приймання (записування), зберігання і передачі інформації, називаються елементарними або фіксаторами. Регістри, в яких зберігання даних поєднується з мікрооперaціями зсуву, називаються зсувовими. Елементарні регістри будують на одноступеневих тригерах, а зсувові – на двоступеневих або D-тригерах з динамічним керуванням. Логічна функція регістра позначається буквами RG (register).

Регістри забезпечують зберігання команд, адреси пам'яті, результатів операцій, індексів та ін.

Регістри класифікують за такими ознаками:

- способом керування записуванням – асинхронні та синхронні;
- способом записування і видачі двійкових слів – паралельні, послідовні й універсальні; у паралельних регістрах записування і видача слів виконується одночасно всіма розрядами, а в послідовних – розряд за розрядом в напрямку від молодших розрядів до старших або навпаки; універсальні регістри забезпечують як паралельний, так і послідовний обмін інформацією;
- числом ліній для представлення значення одного розряду слова (біта інформації) – однофазні й парафазні; при однофазному поданні значення кожного розряду слова передається по одній лінії зв'язку, а при парафазному – по двох лініях (одночасно відображається пряме та інверсне значення розряду);
- числом тактів для записування слова – одно-, дво- і багататактні;
- складом мікрооперацій, які виконуються: установлювальні, записування, читання, порозрядні логічні й зсуву, а також перетворення послідовного коду в паралельний і навпаки;
- напрямом зсуву – односторонні (лівий або правий зсув) і двосторонні (реверсивні);
- типом тригерів, що використовуються;
- елементною структурою – потенціальні, імпульсні й потенціально-імпульсні.

Установлювальні мікрооперації. Однофазний і парафазний спосіб записування інформації. Установлювальні мікрооперації служать для переключення регістрів у певний стан. Наприклад, установлення регістра в стан "0" або "1"; установлення парних розрядів у стан "0", а непарних – у стан "1"; записування в регістр деякої константи або обнуління деяких байтів та ін.

Установлювальні мікрооперації переважно використовують асинхронні входи регістра, наприклад, вхід скидання R для схеми, зображеної на рис. 4.1.

У регістрах на RS- або JK- тригерах можливий однофазний або парафазний спосіб записування інформації. При однофазному записуванні значення кожного розряду слова $A=A_nA_{n-1}\ldots A_i\ldots A_1$ надходить по одній лінії зв'язку на вхід S (або J) відповідних тригерів. Після зчитування записаної інформації регістр має обнулятися по спільному R входу (рис. 4.2). Таким чином, при однофазному записуванні частота обміну інформацією зменшується, оскільки процеси введення і скидання чергуються.

При парафазному записуванні інформації значення кожного розряду слова А передається по двох лініях зв'язку. При цьому пряме значення A_i надходить на вхід S (або J) відповідних тригерів, а інверсне значення $\overline{A}_i$ – на вхід R (або K). У цьому випадку не потрібне попереднє скидання регістра в стан "0", тому що таку функцію виконує сигнал $\overline{A}_i$ (рис.4.3).

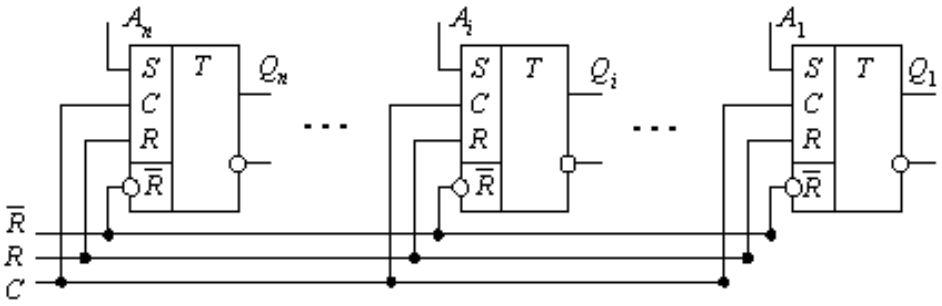


Рис.4.2 Схема регістра з однофазним записом даних

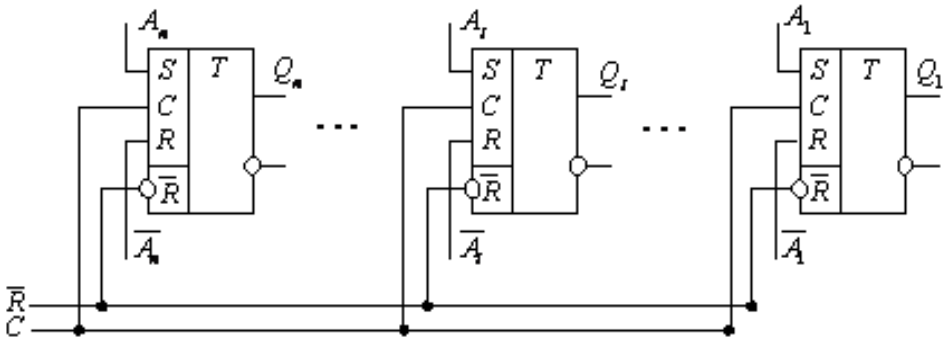


Рис.4.3 Схема регістра з парафазним записом даних

У регістрах на D-тригерах, які мають один інформаційний вхід, можливий тільки однофазний спосіб записування інформації. З приходом чергового синхроімпульсу записується нова інформація або при відсутності даних на вхідній шині стан регістра автоматично обнуляється (див. рис.4.1).

При необхідності збереження інформації на декілька тактів у регістрах на D-тригерах потрібно використати дозволяючий V-вхід, або блокувати проходження синхроімпульсу на C-вхід.

Записування інформації від двох джерел. Для записування інформації від декількох джерел (напрямків) на вході кожного тригера ставлять додаткові комбінаційні схеми, які створюють вхідну логіку регістра. Кожний напрямок має свою сукупність електричних ліній (шину), по яких передаються сигнали, що відображають значення розрядів слова. Якщо n-розрядне слово передається однофазним кодом, то шина має n ліній зв'язку, а якщо парафазним кодом – то 2n ліній. Записування кожного слова ініціюється відповідним керуючим сигналом Y_1, Y_2 та ін.

Для записування в регістр на RS-тригерах однофазним кодом слів А і В потрібно реалізувати такі порозрядні функції збудження входу Si:

$$S_i = Y_1 A_i \vee Y_2 B_i, \qquad (4.1)$$

де A_i і B_i – двійкові розряди слів А і В; Y_1 і Y_2 – сигнали керування приймання слів А і В відповідно. Схема вхідної логіки і-го розряду регістра на основі рівняння (4.1) показана на рис.4.4, а.

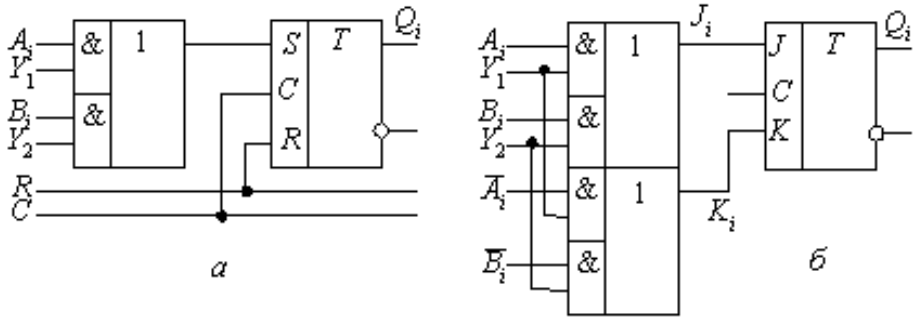


Рис. 4.4 Схеми розряду регістра із записом слів від двох джерел:
а – однофазним кодом; б – парафазним кодом

Для записування в регістр на JK-тригерах парафазним кодом слів A і B потрібно реалізувати такі порозрядні функції збудження входів J_i і K_i:

$$J_i = Y_1 A_i \vee Y_2 B_i \; ; \; K_i = Y_1 \, \overline{A}_i \vee Y_2 \, \overline{B}_i \qquad (4.2)$$

Схема вхідної логіки і-го розряду регістра на основі рівнянь (4.2) показана на рис.4.4,б.

Зчитування інформації. Інформація, яка зберігається в регістрах, може передаватися у зовнішні схеми однофазним або парафазним способом у прямому або оберненому коді. Для реалізації мікрооперацій зчитування до виходів кожного тригера підключаються комбінаційні схеми, які створюють вихідну логіку регістра.

Схеми вихідної логіки будуються на основі таких порозрядних логічних рівнянь:

для зчитування однофазним прямим або оберненим кодом

$$Ш_i = Y_{пр} Q_i \vee Y_{об} \, \overline{Q}_i \; ; \qquad (4.3)$$

для зчитування парафазним прямим або оберненим кодом

$$Ш_i^* = Y_{пр} Q_i \vee Y_{пр} \, \overline{Q}_i \; ; \; \overline{Ш}_i^* = Y_{об} \, \overline{Q}_i \vee Y_{об} Q_i; \qquad (4.4)$$

де $Y_{пр}$ і $Y_{об}$ – керуючі сигнали видачі відповідно прямого або оберненого коду; Q_i і $\overline{Q}_i$ – пряме та інверсне значення виходу і-го розряду регістра; $Ш_i$ – розряд однофазної шини даних; $Ш_i^*$ і $\overline{Ш}_i^*$ – розряди парафазної шини даних.

Очевидно, що керуючі сигнали $Y_{пр}$ і $Y_{об}$ не повинні збігатися в часі. Наприклад, при зчитуванні інформації парафазним оберненим кодом отримаємо:

$$Y_{пр} = 0; \; Y_{об} = 1; \; Ш_i^* = \overline{Q}_i \; ; \; \overline{Ш}_i^* = Q_i.$$

Схеми вихідної логіки для і-го розряду на основі рівнянь (4.3) і (4.4) показані на рис. 4.5.

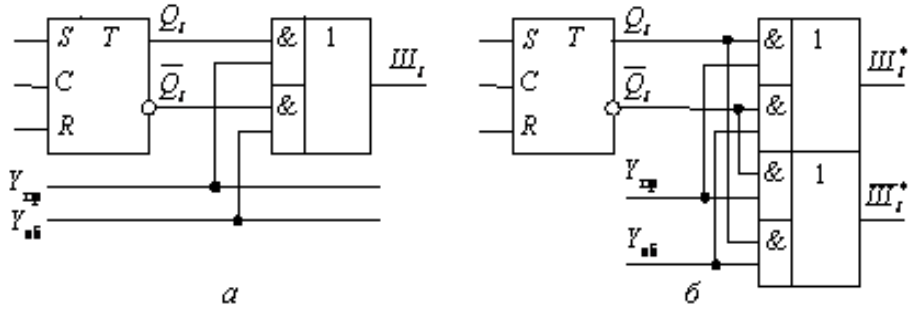


Рис. 4.5 Схеми вихідної логіки і-го розряду регістра для зчитування інформації: а – однофазним кодом; б – парафазним кодом

Логічні мікрооперації в регістрах. У регістрах можуть виконуватися такі порозрядні (без перенесень) логічні мікрооперації над словами A і B: логічне додавання і множення: RG1:=A∨B; RG1:= A∧B; додавання за модулем два і його заперечення: RG1:=A ⊕ B; RG1:= $\overline{A \oplus B}$; інверсія слова: RG1:= $\overline{A}$.

Логічні мікрооперації передбачають наявність першого слова A в регістрі. З урахуванням цього логічне додавання слів A і B в регістрі на RS- або JK-тригерах з однофазним записом виконується введенням слів B без попереднього скидання. Логічне множення реалізується подаванням інверсних значень розрядів слова B на входи R (або K) тригерів регістра. Дійсно, якщо значення $B_i = 0$, то $\overline{B}_i = 1$ і відповідно тригери обнуляються, що і потрібно для порозрядного логічного множення.

Мікрооперації додавання за модулем два і його заперечення реалізуються в регістрах на T-тригерах. Спочатку записується слово A, а потім без попереднього скидання по лічильному входу вводиться слово B. Після цього на прямих виходах тригерів фіксується результат операції Q=A ⊕ B, а на інверсних виходах – $\overline{Q} = \overline{A \oplus B}$. Мікрооперація інвертування складається з подавання імпульсу на всі T-входи тригерів регістра, в яких зберігається слово A. У підсумку на прямих виходах тригерів встановлюється результат згідно із співвідношенням $Q_i = A_i \oplus 1 = \overline{A}_i$.

Мікрооперації зсуву. Зсув – це одночасне просторове переміщення двійкового слова в розрядній сітці із збереженням порядку слідування нулів і одиниць. Регістри, призначені для виконання мікрооперацій зсуву, називаються регістрами зсуву або зсувовими.

Мікрооперації зсуву використовують у процесі виконання команд множення, ділення і нормалізації. Крім того, за допомогою зсуву здійснюється перетворення паралельного коду в послідовний або навпаки (наприклад, при обміні інформацією з магнітними стрічками і дисками).

Зсув слова може виконуватися вправо (у бік молодших розрядів) або вліво (у бік старших розрядів). Позначимо однорозрядні мікрооперації зсуву вправо і вліво символами R і L відповідно. Розрізнюють правий і лівий арифметичний (R_a, L_a), логічний ($R_л$, $L_л$) і циклічний ($R_ц$, $L_ц$) зсуви слова.

Нехай в регістрі А записано слово $A_n\ A_{n-1}\ldots A_2\ A_1$, де A_1 – молодший розряд; A_n – старший розряд. Символічно мікрооперації зсуву записуються таким чином:

- арифметичні зсуви (знаковий розряд не зсувається):

$RGA:=R_a(A)=A_n\ 0\ A_{n-1}\ldots\ A_2$; $RGA:=L_a(A)=A_n\ A_{n-2}\ldots\ A_1\ 0$;

- логічні зсуви (одночасно зсуваються всі розряди):

$RGA:=R_л(A)=0\ A_n\ A_{n-1}\ldots\ A_2$; $RGA:=L_л(A)=\ A_{n-1}\ A_{n-2}\ldots\ A_1\ 0$;

- циклічні зсуви (між старшим і молодшим розрядами є кільцевий зв'язок):

$RGA:=R_ц(A)=A_1\ A_n\ A_{n-1}\ldots\ A_2$; $RGA:=L_ц(A)=\ A_{n-1}\ A_{n-2}\ldots\ A_1\ A_n$.

Арифметичні та циклічні зсуви переважно використовують при виконанні команд в процесорах, а логічні зсуви забезпечують перетворення послідовного коду в паралельний і навпаки в пристроях зв'язку з магнітними стрічками і дисками.

Зсувні регістри проектують на двоступеневих RS- (або JK-) або D-тригерах з динамічним керуванням по фронту . Такі тригери забезпечують розділення під час процесів приймання нової інформації в кожному розряді та видачі (зсуву) старої.

Припустимо, що в реверсивному регістрі мають виконуватися паралельний запис слова А за сигналом керування $Y_{зп}$ і зсуви інформації вліво і вправо під впливом керуючих сигналів $L_л$ і $R_л$. Функції збудження S_i і R_i для кожного розряду регістра зсуву на двоступеневих RS-тригерах мають вигляд:

$$S_i = Y_{зп}\ A_i \vee R_л Q_i + 1 \vee L_л\ Q_{i-1};\ R_i = Y_{зп}\ \overline{A}_i \vee R_л\ \overline{Q}_{i+1} \vee L_л\ \overline{Q}_{i-1} \qquad (4.5)$$

Схема одного розряду регістра зсуву на RS-тригерах згідно з рівнянням (4.5) показана на рис.4.6,а.

Економічна схема розряду регістра, в якій функція на R-входи тригера утвориться інвертуванням сигналу S_i, показана на рис.4.6, б. Однак у цьому випадку час записування даних збільшується за рахунок затримки інвертора.

Для регістра зсуву на D-тригерах функції збудження мають вигляд:

$$D_i=Y_{зп}\ A_i \vee R_л\ Q_{i+1} \vee L_л Q_{i-1}. \qquad (4.6)$$

Схема розряду регістра зсуву на D-тригерах згідно з рівнянням (4.6) показана на рис.4.6, в.

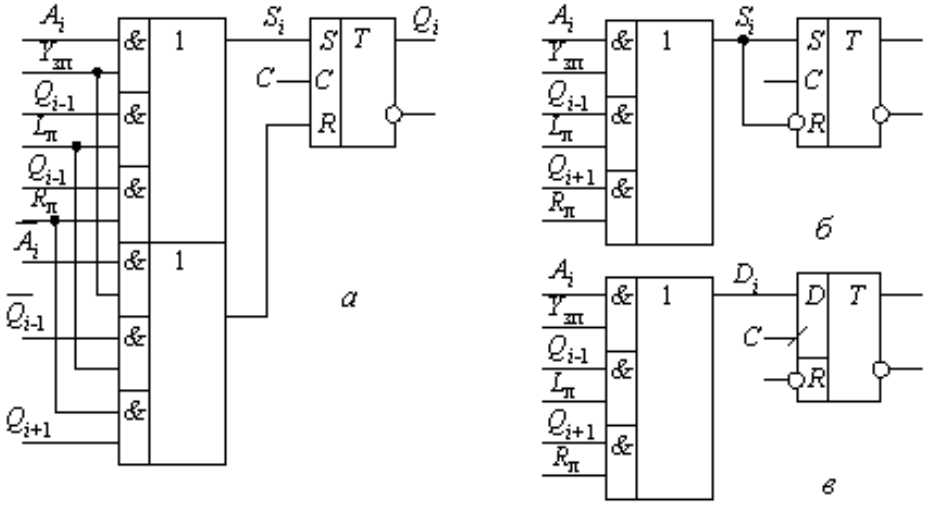


Рис.4.6 Схема розряду регістра зсуву: а – на RS-тригері; б – з інвертором на вході R; в – на D-тригері

Приклад побудови реверсивного трирозрядного регістра зсуву на D-тригерах з динамічним керуванням показаний на рис.4.7.

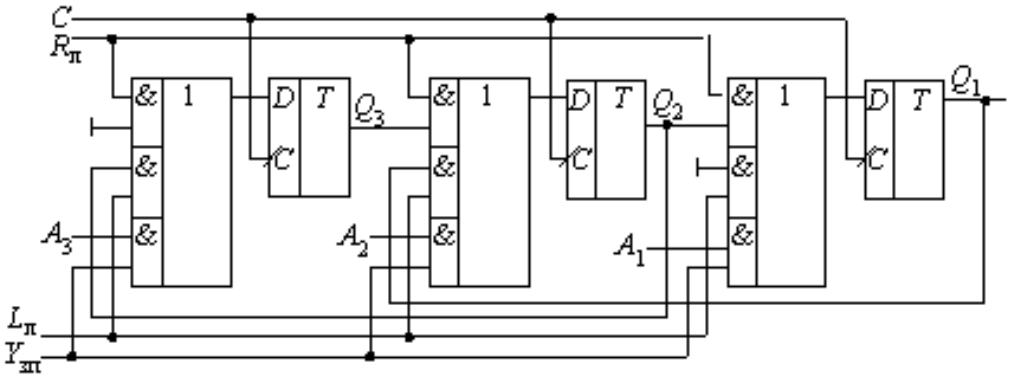


Рис.4.7 Схема реверсивного регістра зсуву на D-тригерах

Реверсивний регістр зсуву працює таким чином. При значенні сигналу $Y_{зп}=1$ в регістр записується інформація паралельним однофазним кодом. При значенні сигналу $R_л=1$ інформація, що зберігається, одночасно зсувається у бік молодших розрядів, при цьому розряд Q_3 обнуляється. При значенні сигналу $L_л=1$ інформація в регістрі одночасно зсувається у бік старших розрядів, при цьому розряді Q_1 обнуляється. Запис і зсув інформації відбуваються по фронту синхроімпульсу.

4.2. Лічильники

Лічильником називається типовий функціональний вузол комп'ютера, призначений для лічби вхідних імпульсів. Лічильник являє собою зв’язаний ланцюг Т-тригерів, які утворюють пам’ять із заданим числом сталих станів (рис. 4.8).

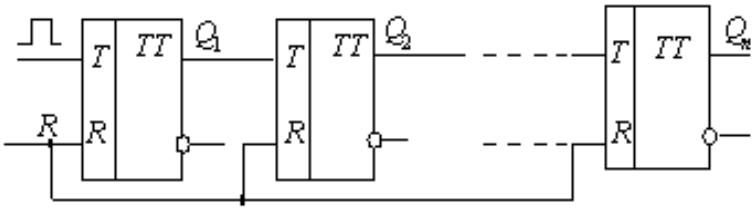


Рис.4.8. Логічна структура лічильника

Розрядність лічильника n дорівнює числу Т-тригерів. Кожний вхідний імпульс змінює стан лічильника, який зберігається до надходження наступного сигналу. Значення виходів тригерів лічильника $Q_n,Q_{n-1},...,Q_1$ відображають результат лічби в прийнятій системі числення. Логічна функція лічильника позначається буквами СТ (counter).

Список мікрооперацій лічильника вміщує попереднє встановлення в початковий стан, інкремент або декремент слова, яке зберігається, видачу слів паралельним кодом та ін.

Вхідні імпульси можуть надходити на лічильник як періодично, так і довільно розподіленими у часі. Амплітуда і тривалість лічильних імпульсів мають задовольняти технічні вимоги для серій мікросхем, які використовуються.

Лічильник є одним з основних функціональних вузлів комп’ютера, а також різних цифрових керуючих та інформаційно-вимірювальних систем. Основне застосування лічильників:

- утворення послідовності адрес команд програми (лічильник команд або програмний лічильник);
- підрахунок числа циклів при виконанні операцій ділення, множення, зсуву (лічильник циклів);
- одержання сигналів мікрооперацій і синхронізації; аналого-цифрові перетворення і побудова електронних таймерів (годинників реального часу).

Лічильник характеризується модулем і ємністю лічби. Модуль лічби $К_{Лч}$ визначає число станів лічильника. Модуль двійкового n-розрядного лічильника визначається цілим степенем двійки $M = 2^n$; в лічильниках інших типів справедлива нерівність $К_{Лч} \le M$. Після лічби числа імпульсів $N_{ВХ} = К_{Лч}$ лічильник повертається в початковий стан. Таким чином, модуль лічби, який часто називають коефіцієнтом перерахунку, визначає цикл роботи лічильника, після чого його стан повто-

рюється. Тому число вхідних імпульсів і стан лічильника однозначно визначені тільки для першого циклу.

Ємність лічби N_{max} визначає максимальну кількість вхідних імпульсів, яку може зафіксувати лічильник при одному циклі роботи. Ємність лічби $N_{max} = К_{Лч} - 1$ за умови, що робота лічильника починається з нульового початкового стану.

У лічильниках використовуються три режими роботи: керування, накопичення і ділення. У режимі керування зчитування інформації виконується після кожного вхідного лічильного імпульсу, наприклад, в лічильнику адреси команд. У режимі накопичення головним є підрахунок заданого числа імпульсів або лічба протягом певного часу. У режимі ділення (перерахунку) основним є зменшення частоти надходження імпульсів в КЛЧ разів. Більшість лічильників може працювати в усіх режимах, проте в спеціальних лічильниках-дільниках стани в процесі лічби можуть змінюватися в довільному порядку, що дозволяє спростити схему вузла.

Лічильники класифікують за такими ознаками:

- способом кодування – позиційні та непозиційні;
- модулем лічби – двійкові, десяткові, з довільним постійним або змінним (програмованим) модулем;
- напрямком лічби – прості (підсумовуючі, віднімальні) і реверсивні;
- способом організації міжрозрядних зв'язків – з послідовним, наскрізним, паралельним і комбінованим переносами (позикою);
- типом використовуваних тригерів – T, JK, D в лічильному режимі;
- елементним базисом – потенціальні, імпульсні та потенціально-імпульсні.

У лічильниках з позиційним кодуванням числовий вираз поточного стану лічильника визначається за формулою:

$$N = \sum_{i=1}^{n} r_i Q_i = r_n Q_n + r_{n-1} Q_{n-1} + ... + r_1 Q_1 ,$$

де r_i – вага і-го розряду; Q_i – значення виходу і-го розряду; n – число розрядів. Нульове значення всіх розрядів звичайно беруть як початковий стан лічильника. Всі інші стани нумерують за числом вхідних імпульсів, що надійшли.

У лічильниках з непозиційним кодуванням (наприклад, у кодах Грея) розряди не мають постійних ваг і кожному набору станів $Q_n, Q_{n-1}, ..., Q_1$ приписується певна кількість вхідних імпульсів. У комп'ютерах переважно використовують лічильники з позиційним кодуванням.

За видом переходів прості лічильники (Лч) розподіляються на підсумовуючі (прямої лічби) і віднімальні (зворотної лічби). У підсумовуючих лічильниках кожний доданий імпульс U+ збільшує стан на одиницю, тобто реалізується мікрооперація інкремента Лч:= Лч+1. Граф переходів підсумовуючого лічильника показаний на рис.4.9, а, де вершини означають стійкі стани. Із М–1-го стану черговий сигнал U+ повертає лічильник у початковий стан і видає сигнал переповнення Р.

У віднімальних лічильниках кожний віднімальний імпульс U – зменшує стан на одиницю, тобто реалізується мікрооперація декремента Лч:= Лч–1. Граф переходів віднімального лічильника (рис. 4.9, б) характеризується наявністю переходів тільки в зворотному напрямку від деякого (наприклад, М–1-го) попередньо встановленого стану. Після віднімання М імпульсів лічильник видає сигнал позики Z і повертається в початковий М–1-й стан.

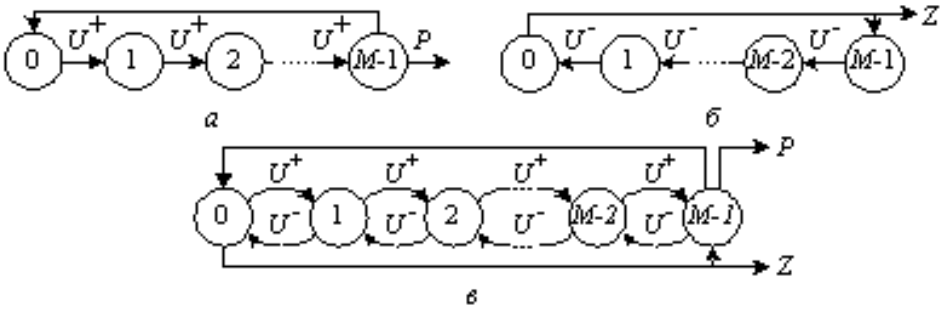


Рис.4.9. Графи лічильників: а – підсумовуючого; б – віднімального; в – реверсивного

Реверсивні лічильники мають переходи в прямому і зворотному напрямках, що дозволяє рахувати підсумовуючі та віднімальні імпульси (рис.4.9, в). У процесі лічби має виконуватися умова $\Sigma U^{+} + N_{п} \geq \Sigma U^{-}$, де $N_{п}$ – попередньо записане число. За поточним станом виходів лічильника визначається результат реверсивної лічби $\Delta N = \Sigma U^{+} + N_{п} - \Sigma U^{-}$.

До часових характеристик лічильників відносяться роздільна здатність, швидкодія і час встановлення (перемикання) коду.

Роздільна здатність $t_{рз}$ визначається мінімальним інтервалом часу між двома вхідними імпульсами, при якому ще зберігається працездатність лічильника. Параметр $t_{рз}$ задають часом перемикання tT першого (молодшого) тригера лічильника, тобто $t_{рз}=t_{Т}$, оскільки він перемикається під дією кожного вхідного імпульсу.

Швидкодія лічильника визначається максимальною частотою F_{m} надходження вхідних імпульсів в режимі ділення й обчислюється за формулою $F_{m} = 1/t_{Т}$. Час встановлення коду $t_{вст}$ відраховується від початку вхідного імпульсу до моменту отримання нового стану. Даний параметр дозволяє обчислювати швидкодію лічильника в режимі керування із співвідношення $F_{m.к}=1/(t_{вст} + t_{зч})$, де $t_{зч}$ – час зчитування інформації.

Міжрозрядні зв’язки забезпечують вироблення сигналів перенесення в старші розряди при додаванні імпульсів і сигналів позики – при відніманні. Від виду реалізації міжрозрядних зв’язків суттєво залежать параметри $t_{вст}$ і $F_{m.к}$.

У лічильниках з послідовними перенесеннями тригери перемикаються почергово після кожного вхідного імпульсу в напрямку від молодших розрядів до старших. Такі лічильники називаються послідовними або асинхронними. У лічильниках з паралельними перенесеннями тригери перемикаються одночасно після кожного вхідного імпульсу, такі лічильники називаються паралельними або синхронними.

5. КОМБІНАЦІЙНІ ФУНКЦІОНАЛЬНІ ВУЗЛИ КОМП'ЮТЕРНОЇ СХЕМОТЕХНІКИ

5.1. Дешифратори

Дешифратором називається функціональний вузол комп'ютера, призначений для перетворення кожної комбінації вхідного двійкового коду в керуючий сигнал лише на одному із своїх виходів. У загальному випадку дешифратор має n однофазних входів (іноді 2n парафазних) і $m=2^n$ виходів, де n – розрядність (довжина) коду, який дешифрується. Дешифратор з максимально можливим числом виходів $m=2^n$ називається повним. Функціонування повного дешифратора описується системою логічних виразів вигляду:

$$F_0 = \overline{X}_n \overline{X}_{n-1} \dots \overline{X}_2 \overline{X}_1;$$
$$F_1 = \overline{X}_n \overline{X}_{n-1} \dots \overline{X}_2 X_1;$$
$$\dots\dots\dots\dots\dots\dots$$
$$F_{m-1} = X_n X_{n-1} \dots X_2 X_1,$$

де $X_1,..., X_n$ – вхідні двійкові змінні; $F_0, F_1,..., F_{m-1}$ – вихідні логічні функції, що являють собою мінтерми (конституенти 1) n змінних.

Індекс функції F_i визначає номер обраного виходу і відповідає десятковому еквіваленту вхідного коду. Вихід, на якому з'являється керуючий сигнал, називається активним. Якщо значення сигналу на активному виході відображається лог.1, то на решті пасивних виходів встановлюється лог.0. Двійковий код, який вміщує завжди тільки одну одиницю, а інші – нулі, називається унітарним. Тому дешифратор є перетворювачем вхідного позиційного коду в унітарний вихідний код.

У дешифраторах в інтегральному виконанні стан активного виходу часто відображається значенням лог.0, а на інших пасивних виходах установлюється лог.1. Функціонування повного дешифратора з інверсними виходами представляється системою виду:

$$L_0 = X_n \vee X_{n-1} \vee \dots \vee X_2 \vee X_1;$$
$$L_1 = X_n \vee X_{n-1} \vee \dots \vee X_2 \vee \overline{X}_1;$$
$$\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots$$
$$L_{m-1} = \overline{X}_n \vee \overline{X}_{n-1} \vee \dots \vee \overline{X}_2 \vee \overline{X}_1,$$

де $L_0, L_1, \dots, L_{m-1}$ – вихідні логічні функції, що є макстермами (конституенти 0) n змінних.

Індекс функції L_i визначає номер вибраного виходу і відповідає десятковому еквіваленту вхідного коду. Між двома видами вихідних функцій існує простий зв'язок: $F_i = \overline{L}_i$.

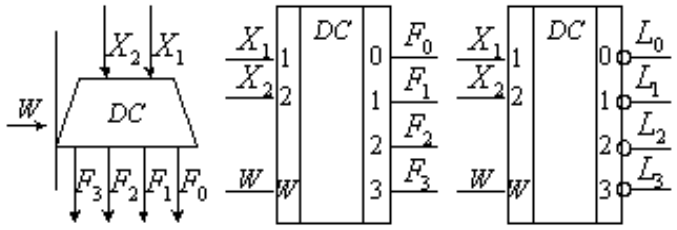
Дешифратори класифікують за такими ознаками:

- способом структурної організації – одноступеневі (лінійні) і багатоступеневі, в тому числі пірамідальні та прямокутні (матричні);
- форматом вхідного коду – двійкові, двійково-десяткові;
- розрядністю коду, який дешифрується – 2, 3, ..., n;
- формою подачі вхідного коду – з однофазними і парафазними входами;
- кількістю виходів – повні й неповні дешифратори;
- видом вхідних стробуючих сигналів – в прямому або інверсному значеннях;
- типом використовуваних логічних елементів – І, НЕ, АБО, НЕ І, НЕ АБО і т.д.

До основних характеристик дешифратора відносять: число ступенів (каскадів) дешифрації, кількість використаних логічних елементів або мікросхем, загальне число входів логічних елементів, час дешифрації і споживану потужність.

Логічна функція дешифратора позначається буквами DC (de-coder). Мітки лівого додаткового поля в умовному позначенні відображають десяткові ваги вхідних змінних, а мітки правого додаткового поля відповідають десятковим еквівалентам вхідних комбінацій двійкових змінних. У схему дешифраторів вбудовуються один або два стробуючих (дозволяючих) входи, наприклад, W (рис.5.1, б). За допомогою сигналу на вході W визначається момент спрацювання дешифратора; крім того, вхід W використовується для нарощування розрядності вхідного коду. На практиці повний дешифратор на n входів і m виходів для стислості називають дешифратором "з n в m" або "n $\rightarrow$ m". Наприклад, дешифратор "з 3 у 8" – активізується одна з восьми вихідних ліній.

Умовні графічні позначення дешифраторів на електричних схемах показані на рис.5.1.



а б в

Рис. 5.1. Умовні графічні позначення дешифратора: а – на функціональних схемах; б, в – на принципіальних схемах

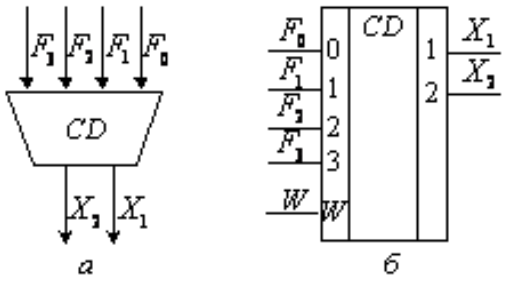
В комп'ютерах дешифратори використовують для виконання таких операцій:

- дешифрації коду операції, записаного в регістр команд процесора, що забезпечує вибір потрібної мікропрограми;
- перетворення коду адреси операнда в команді в керуючі сигнали вибору заданої комірки пам'яті в процесі записування або читання інформації;
- забезпечення візуалізації на зовнішніх пристроях;
- реалізації логічних операцій та побудови мультиплексорів і демультиплексорів.

5.2. Шифратори

Шифратором називається функціональний вузол комп'ютера, призначений для перетворення вхідного m-розрядного унітарного коду у вихідний n-розрядний двійковий позиційний код. Двійкові шифратори виконують функцію, обернену функції дешифратора. При активізації однієї з вхідних ліній дешифратора на його виходах формується код, який відображає номер активного входу. Повний двійковий шифратор має $m=2^n$ входів і n виходів. Умовні графічні позначення шифраторів на схемах показані на рис. 5.2.

Функція шифратора позначається буквами CD (coder). Входи шифратора нумеруються послідовними десятковими цифрами 0, 1, ..., m–1, а позначки виходів відображають ваги вихідних двійкових змінних 1, ..., 2^{n-1}.



а б

Рис. 5.2. Умовні графічні позначення шифратора: а – на функціональних схемах; б – на принципових схемах

У цифрових пристроях шифратори використовуються для таких операцій: перетворення унітарного вхідного коду у вихідний двійковий позиційний код; введення десяткових даних з клавіатури; показання старшої одиниці в слові; передачі інформації між різними пристроями при обмеженому числі ліній зв'язку.

Каскадування шифраторів. Каскадування шифраторів використовується для збільшення розрядності вхідного слова. Схема каскадування двох восьмивходових шифраторів К555ИВ1 для пріоритетного обслуговування 16-розрядного слова $\overline{F}_{15} - \overline{F}_0$ показана на рис. 5.3.

Розряди $\overline{F}_{15} - \overline{F}_8$ старшого байта вхідного слова поступають на перший шифратор CD1, а розряди $\overline{F}_7 - \overline{F}_0$ молодшого байта подаються на другий шифратор CD2. Вхід $\overline{F}_{15}$ має найвищий пріоритет, а $\overline{F}_0$ – найнижчий. Інформаційні виходи обох шифраторів об'єднуються за допомогою логічних елементів АБО, утворюючи трирозрядний інверсний код $\overline{Z}_3\overline{Z}_2\overline{Z}_1$. Значення старшого розряду $\overline{Z}_4$ забезпечується безпосередньо сигналом $\overline{P}_1$.

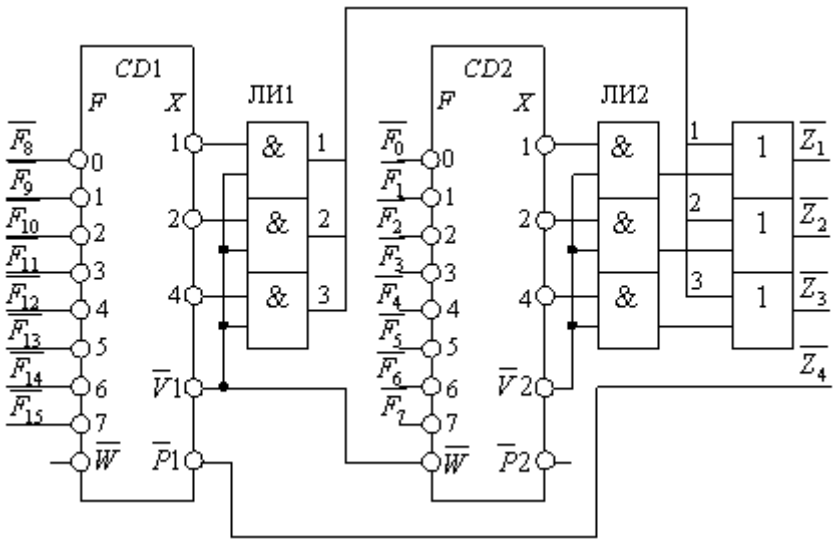


Рис. 5.3. Каскадування двох восьмивходових пріоритетних шифраторів

Інформація з виходів першого шифратора подається на входи елементів АБО за допомогою схем збігу ЛИ1 при $\overline{V}1 = 1$ (мікросхема CD1 сприймає вхідні дані, а CD2 – блокована). Інформація з виходів другого шифратора подається на входи елементів АБО за допомогою схем збігу ЛИ2 при V2 = 1 (мікросхема CD2 сприймає вхідні дані, CD1 – блокується).

Схема працює так: коли на вході є активний сигнал із старшого байта вхідного слова, наприклад, $\overline{F}_{14} = 0$ ($\overline{P}_1 = 0$, $\overline{V}1 = 1$), то працює шифратор CD1 і на виходах елементів АБО формується інверсний код $\overline{Z}_4\overline{Z}_3\overline{Z}_2\overline{Z}_1 = 0001$, що відповідає прямому значенню $Z_4Z_3Z_2Z_1 = 1110 = 14_{10}$. Якщо активний вхідний сигнал відноситься до молодшого байта слова, наприклад, $\overline{F}_6 = 0$, то працює шифратор CD2 ($\overline{P}_1 = 1$, $\overline{V}1 = 0$, $\overline{V}2 = 1$) і на інверсних виходах формується код $\overline{Z}_4\overline{Z}_3\overline{Z}_2\overline{Z}_1 = 1001$, що відповідає прямому числу $Z_4Z_3Z_2Z_1 = 0110 = 6_{10}$.

5.3. Мультиплексори

Мультиплексором називається функціональний вузол комп'ютера, 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 машинного часу, визначається адресним кодом $A_0, A_1, ..., A_{m-1}$. Зв'язок між числом інформаційних n і адресних m входів визначається співвідношенням $n=2^m$. Таким чином, мультиплексор реалізує керовану передачу даних від кількох вхідних ліній в одну вихідну.

Умовне графічне позначення мультиплексорів показане на рис. 5.4. Функція мультиплексорів записується буквами MUX (multiplexor).

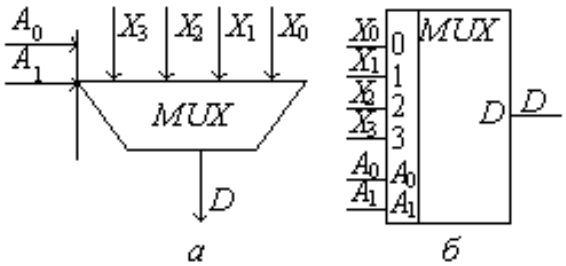


Рис. 5.4. Умовне позначення мультиплексора: а - на функціональних схемах; б - на принципових схемах.

Мультиплексори 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.

Мультиплексор символічно часто позначають: "n–1".

Логіка роботи чотиривходового мультиплексора наведена в табл. 5.1, 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 5.1

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. 5.1. вираз для вихідної функції D можна представити з використанням виходів F_0–F_3 внутрішнього дешифратора у вигляді:

$$D = F_0X_0 \vee F_1X_1 \vee F_2X_2 \vee F_3X_3, \qquad (5.1)$$

або з мінтермами адресного коду:

$$D= \overline{A}_1\ \overline{A}_0\ X_0 \vee\ \overline{A}_1\ A_0\ X_1 \vee A_1\ \overline{A}_0\ X_2 \vee A_1\ A_0\ X_3. \qquad (5.2)$$

Схеми мультиплексорів, відповідні рівнянням (5.1) і (5.2), показані на рис. 5.5.

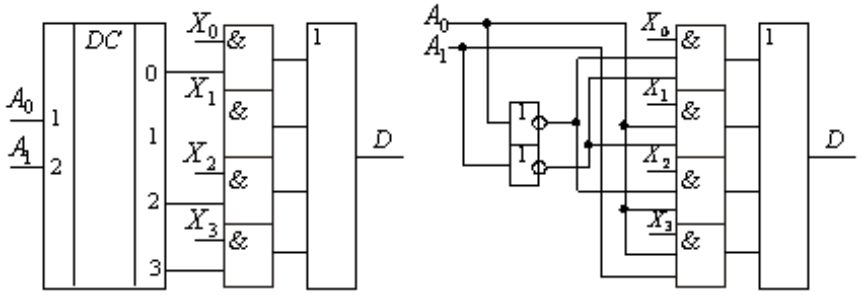


Рис. 5.5. Схеми мультиплексорів: а – з внутрішнім дешифратором; б – з адресними мінтермами

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.

Каскадування мультиплексорів. В інтегральному виконанні мультиплексори випускають на чотири, вісім або шістнадцять 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. 5.6.

Молодші розряди адреси A_1, A_0 підключаються до адресних входів усіх мультиплексорів першого рівня, на виходах яких виробляються такі функції:

$$D'_0 = F_0X_0 \vee F_1X_1 \vee F_2X_2 \vee F_3X_3;$$

$$D'_1 = F_0X_4 \vee F_1X_5 \vee F_2X_6 \vee F_3X_7;$$

$$D'_2 = F_0X_8 \vee F_1X_9 \vee F_2X_{10} \vee F_3X_{11};$$

$$D'_3 = F_0X_{12} \vee F_1X_{13} \vee F_{12}X_{14} \vee F_3X_{15},$$

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.

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

$F'_0 = \overline{A}_3\ \overline{A}_2$; $F'_1 = \overline{A}_3 A_2$; $F'_2 = A_3\ \overline{A}_2$; $F'_3 = A_3A_2$.

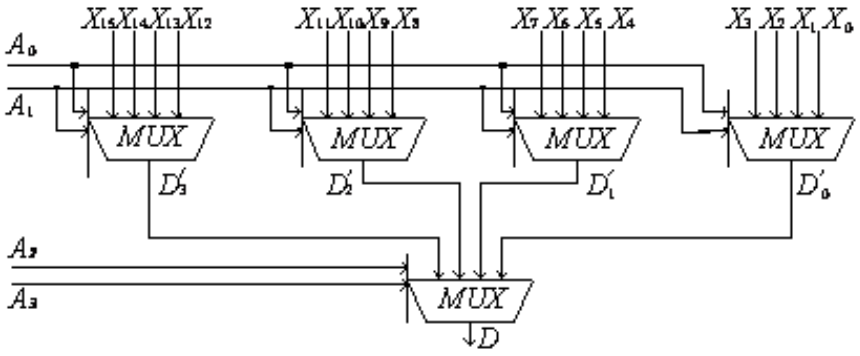


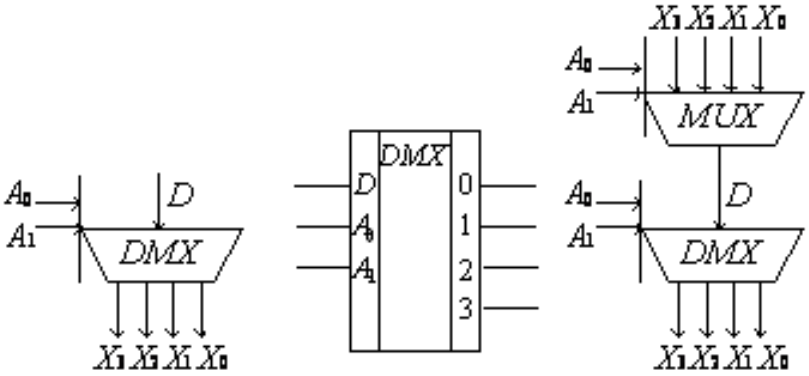
Рис. 5.6. Каскадування мультиплексорів

5.4. Демультиплексори

Демультиплексором 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, ..., A_{m-1}. Адресні входи m та інформаційні виходи n пов'язані співвідношенням $n = 2^m$ або $m = \log_2 n$.

Демультиплексор виконує функцію, обернену функції мультиплексора. Стосовно мультиплексорів і демультиплексорів користуються також терміном "селектори" даних.

В умовних графічних позначеннях (рис. 5.7) функція демультиплексора позначається буквами DMX.



а *б* *в*

Рис. 5.7. Умовні графічні позначення демультиплексорів: а – на функціональних схемах; б – на принципових схемах; в – типове з'єднання з мультиплексором

Демультиплексори використовують для таких операцій:

- комутації як окремих ліній, так і багаторозрядних шин;
- перетворення послідовного коду в паралельний;
- реалізації логічних функцій та інших.

Демультиплексори часто позначають: "1 → n".

Логіка роботи двоадресного демультиплексора на мові мікрооперацій наведена в табл. 5.2, де D – інформаційний вхід; F_0, F_1, F_2 і F_3 – виходи внутрішнього дешифратора адреси.

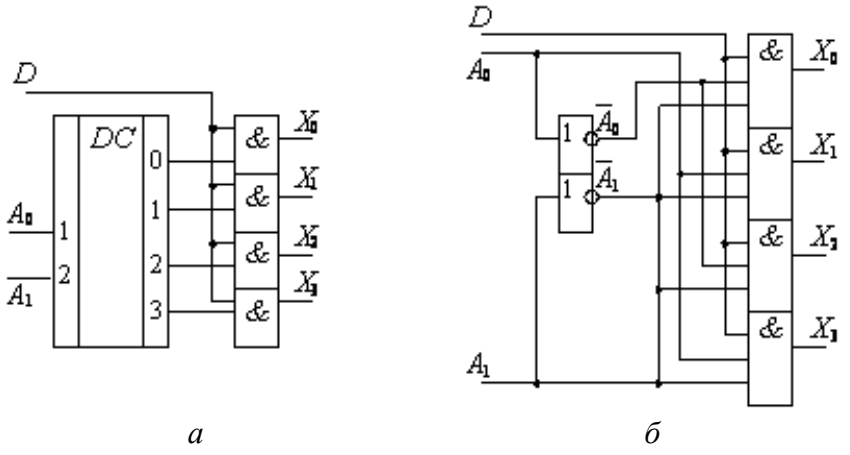
Таблиця 5.2

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_0D	–	–	–
0	1	0	1	0	0	–	F_1D	–	–
1	0	0	0	1	0	–	–	F_2D	–
1	1	0	0	0	1	–	–	–	F_3D

За даними табл. 5.2 записуємо систему рівнянь для інформаційних виходів:

$$X_0 = F_0D = \overline{A}_1\,\overline{A}_0\,D;\ X_1 = F_1D = \overline{A}_1\,A_0D;\ X_2 = F_2D = A_1\,\overline{A}_0\,D;\ X_3 = F_3D = A_1A_0D; \qquad (5.3)$$

На основі рівнянь (5.3) побудовані схеми демультиплексорів із внутрішнім дешифратором (рис. 5.8, а) і з поєднанням адресних і вхідних змінних на тривходових елементах І (рис. 5.8, б).



а *б*

Рис. 5.8. Схема демультиплексорів: а – з внутрішнім дешифратором; б – з поєднанням адресних і вхідних змінних

Схема демультиплексора з поєднанням адресних і вхідних змінних забезпечує високу швидкодію, проте вимагає застосування логічних елементів з більшим числом входів.

Каскадування демультиплексорів. 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 → 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 → 4" для реалізації комутатора "1 → 16" показано на рис. 5.9.

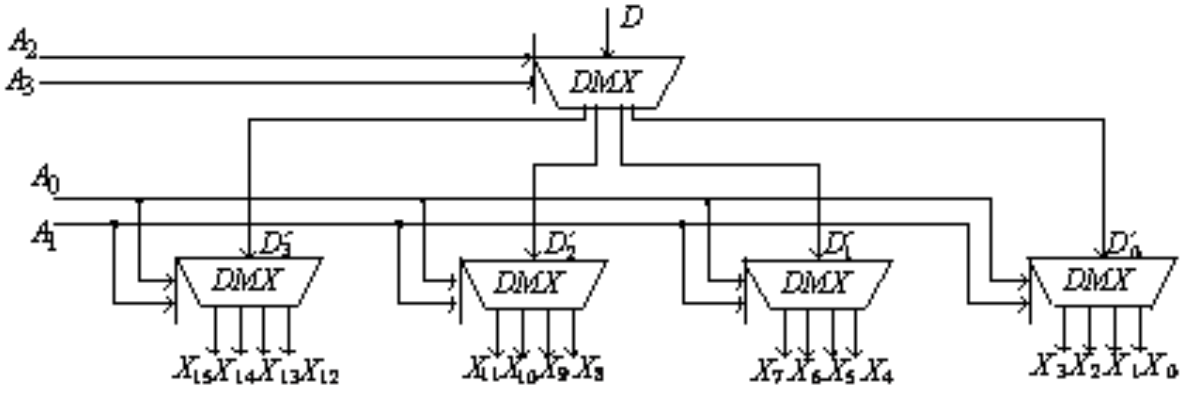


Рис. 5.9. Каскадування демультиплексорів

Нехай для схеми (рис. 5.9) 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. Одиничне значення сигналу D'_2 передається на вихід X_{10} веденого демультиплексора згідно зі співвідношенням $X_{10}=A_1 \overline{A}_0 D'_2 = 1$.

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

5.5. Компаратори.

Схемою порівняння (компаратором) називається функціональний вузол комп'ютера, призначений для вироблення ознак відношень між двійковими словами (числами). Ознаки відношень записуються у вигляді:

$$F_i: = A*K \text{ або } F_{i\,A*K} \text{ або } F_{A*K};$$
$$F_i: = A*B \text{ або } F_{i\,A*B} \text{ або } F_{A*B},$$

де A і B – двійкові або двійково-десяткові числа; K – двійкова константа; i – номер відношення (часто пропускається); * – операція відношення вигляду =, ≠, <, >, ≤, ≥ і т. ін.; F_i – функція, що задає результат відношення: лог.1 – якщо відношення виконується, тобто істинне, і лог.0 – якщо відношення не виконується, тобто помилкове. Функція компаратора позначається буквами COMP (comparator) або знаками = =.

Основними відношеннями вважаються: «рівно» $F_{A=B}$, «більше» $F_{A>B}$ і «менше» $F_{A<B}$. Часто схеми, що реалізують відношення $F_{A>B}$ або $F_{A<B}$, називають схемами порівняння «на більше» або «на менше». Маючи в своєму розпорядженні основні ознаки відношень, можна на їхній основі отримати ряд додаткових ознак, наприклад:

$$F_{A\neq B} = \overline{F}_{A=B}\,;\; F_{A\leq B} = \overline{F}_{A>B}\,;\; F_{A\leq B} = F_{A=}B \vee FA_{<B}.$$

Ознаки відношення використовуються як логічні умови (повідомляючі сигнали) в мікропрограмах, командах передачі керування, а також у пристроях контролю і діагностики. Після виконання кожної команди в машині автоматично формуються ознаки результатів операції. Ці ознаки, які називаються прапорами (прапорцями), вміщуються в спеціальний регістр прапорів. До прапорів звичайно відносять ознаки нульового результату, переповнення розрядної сітки, знак результату, наявність перенесень із старшого розряду суматора, парне або непарне число одиниць в результаті та ін.

Зазначимо, що формування і використання ознак (прапорців) – це основна відмінність комп'ютера від калькулятора. Тільки за допомогою прапорців машина приймає рішення про хід обчислювального процесу, тобто володіє інтелектуальними властивостями.

Контроль (виявлення) і корекція (виправлення) результатів операцій є важливою умовою грамотної експлуатації машин. Контроль може бути програмним або апаратним. До апаратних методів відносяться дублювання операцій і відновлення вхідних сигналів.

Контроль операцій додавання методом дублювання реалізується двома однаковими суматорами (SM), на входи яких одночасно поступають доданки A(n) і B(n). Обидва результати $S_1(n)$ і $S_2(n)$ поступають на входи схеми порівняння (рис. 5.10, а). Якщо обидва результати рівні, то на виході схеми порівняння значення ознаки $F_{S1=S2}$ = 1 і помилок немає. При нульовому значенні ознаки операцію потрібно повторити або зупинити роботу ЕОМ.

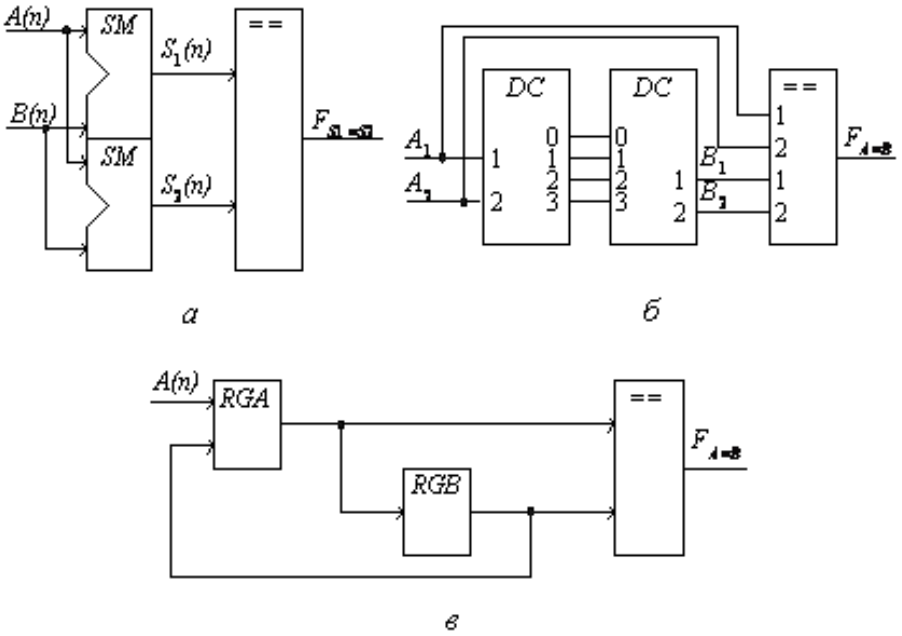


Рис. 5.10. Застосування схеми порівняння для контролю операцій

Схема контролю методом відновлення вхідних сигналів показана на рис. 5.10, б. Дворозрядне слово A_2A_1 декодується і значення унітарного коду з виходів дешифратора поступає на входи шифратора. При правильній роботі дешифратора і шифратора вхідний код A_2A_1 має збігатися з вихідним кодом шифратора B_2B_1. При цьому на виході схеми порівняння встановиться одиничне значення ознаки $F_{A=B}$.

При передачі інформації з одного регістра в інший контроль правильності пересилки може здійснюватися порозрядним порівнянням вмісту цих двох регістрів. На рис. 5.10, в показаний один з варіантів контролю пересилок слів між регістрами. Після передачі інформації з регістра А в регістр В (або навпаки) проводиться порівняння їхнього вмісту. Якщо значення двох слів збігаються, то значення ознаки рівності набуває одиничного значення, інакше – виробляється сигнал помилки.

5.6. Суматори. Арифметико-логічні пристрої

Суматором називається функціональний вузол комп'ютера, призначений для додавання двох n-розрядних слів (чисел). Операція віднімання заміняється додаванням слів в оберненому або доповняльному коді. Операції множення та ділення зводяться до реалізації багаторазового додавання та зсування. Тому суматор є важливою частиною арифметико-логічного пристрою. Функція суматора позначається буквами SM або Σ.

Суматор складається з окремих схем, які називаються однорозрядними суматорами; вони виконують усі дії з додавання значень однойменних розрядів двох чисел (операндів). Суматори класифікуються за такими ознаками:

- способом додавання – паралельні, послідовні та паралельно-послідовні;
- числом входів – напівсуматори, однорозрядні та багаторозрядні суматори;
- організацією зберігання результату додавання - комбінаційні,накопичувальні,комбіновані;
- організацією перенесення між розрядами – з послідовним, наскрізним, паралельним або комбінованим перенесеннями (з груповою структурою);
- системою числення – позиційні (двійкові, двійково-десяткові, трійкові) та непозиційні, наприклад, у системі залишкових класів;
- розрядністю (довжиною) операндів – 8-, 16-, 32-, 64-розрядні;
- способом представлення від'ємних чисел – в оберненому або доповняльному кодах, а також в їхніх модифікаціях;
- часом додавання – синхронні, асинхронні.

У паралельних n-розрядних суматорах значення всіх розрядів операндів поступають одночасно на відповідні входи однорозрядних підсумовуючих схем. У послідовних суматорах значення розрядів операндів та перенесення, що запам'ятовувалися в минулому такті, поступають послідовно в напрямку від молодших розрядів до старших на входи одного однорозрядного суматора. В паралельно-послідовних суматорах числа розбиваються на частини, наприклад, байти, розряди байтів поступають на входи восьмирозрядного суматора паралельно (одночасно), а самі байти – послідовно, в напрямку від молодших до старших байтів з урахуванням запам'ятованого перенесення.

У комбінаційних суматорах результат операції додавання запам'ятовується в регістрі результату. В накопичувальних суматорах процес додавання поєднується із зберіганням результату. Це пояснюється використанням Т-тригерів як однорозрядних схем додавання.

Організація перенесення практично визначає час виконання операції додавання. Послідовні перенесення схемно створюються просто, але є повільнодіючими. Паралельні перенесення схемно організуються значно складніше, але дають високу швидкодію.

Розрядність суматорів знаходиться в широких границях: 4-16 – для мікро- та мініком'ютерів та 32–64 і більше – для універсальних машин.

Суматори з постійним інтервалом часу для додавання називаються синхронними. Суматори, в яких інтервал часу для додавання визначається моментом фактичного закінчення операції, називаються асинхронними. В асинхронних суматорах є спеціальні схеми, які визначають фактичний момент закінчення додавання і повідомляють про це в пристрій керування. На практиці переважно використовуються синхронні суматори. Суматори характеризуються такими параметрами:

- швидкодією – часом виконання операції додавання ta, який відраховується від початку подачі операндів до одержання результату; часто швидкодія характеризується кількістю додавання в секунду Fa=1/ta, тут маються на увазі операції типу регістр–регістр (тобто числа зберігаються в регістрах АЛП);
- апаратурними витратами: вартість однорозрядної схеми додавання визначається загальним числом логічних входів використаних елементів; вартість багаторозрядного суматора визначається загальною кількістю використаних мікросхем;
- споживаною потужністю суматора.

Однорозрядні суматори. Однорозрядним суматором називається логічна схема, яка виконує додавання значень і-х розрядів X_i та Y_i двійкових чисел з урахуванням перенесення Z_i з молодшого сусіднього розряду та виробляє на виходах функції результат S_i і перенесення P_i в старший сусідній розряд. На основі однорозрядних схем додавання на три входи та два виходи будуються багаторозрядні суматори будь-якого типу. Алгоритм роботи однорозрядного суматора відображається таблицею істинності (рис. 5.11).

На основі таблиці істинності записується система логічних функцій для результату S_i та перенесення P_i у ДДНФ:

$$S_i = \overline{X}_i \overline{Y}_i Z_i \vee \overline{X}_i Y_i \overline{Z}_i \vee X_i \overline{Y}_i \overline{Z}_i \vee X_i Y_i Z_i \quad (5.4)$$

$$P_i = \overline{X}_i Y_i Z_i \vee X_i \overline{Y}_i Z_i \vee X_i Y_i \overline{Z}_i \vee X_i Y_i Z_i \quad (5.5)$$

Мінімізація функцій (5.4) та (5.5) за допомогою карт Карно показана на рис. 5.11.

Як видно з карт Карно, функція результату S_i не мінімізується, а функція P_i мінімізується зі зниженням рангу кон'юнкції та використовує тільки прямі значення змінних:

$$P_i = X_i Y_i \vee X_i Z_i \vee Y_i Z_i = X_i Y_i \vee (X_i \vee Y_i) Z_i \quad (5.6)$$

X_i	Y_i	Z_i	S_i	P_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

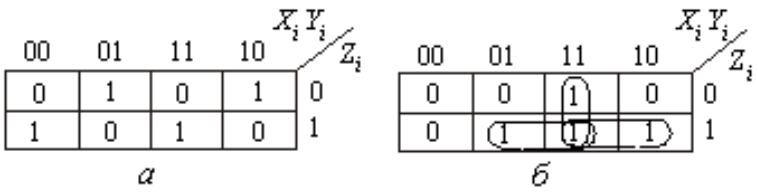


Рис. 5.11. Таблиця істинності і карти Карно для мінімізації функцій: а – S_i; б – P_i

При проектуванні комбінаційних однорозрядних суматорів враховують такі чинники:

- схема має характеризуватися регулярністю (подібністю) структури та мінімальною вартістю, тобто мати по можливості найменше число логічних входів всіх елементів;

- з метою підвищення швидкодії багаторозрядного суматора потрібен мінімальний час одержання функції перенесення $t_П = k\ t_P$, де k – число послідовно увімкнених елементів від входів до виходів P_i або $\overline{P}_i$; t_P – середня затримка розповсюдження сигналу одним логічним елементом в обраній серії інтегральних мікросхем; параметр k часто називають каскадністю (поверховістю) схем. Таким чином, для мінімізації часу одержання перенесення необхідно зменшити каскадність схеми та використати інтегральні мікросхеми з малим часом затримки розповсюдження сигналу;

- для схем однорозрядних суматорів на основі рівнянь (5.4) і (5.5) необхідно виробляти як прямі Pi , так й інверсні $\overline{P}_i$ значення функції перенесення. Така організація перенесень називається парафазною.

Для побудови схеми однорозрядного суматора на універсальних логічних елементах НЕ І рівняння (5.4) і (5.5) перетворюються на основі правил подвійної інверсії та де Моргана до такого вигляду:

$$S_i = \overline{\overline{\overline{X_i}\,\overline{Y_i}Z_i}\cdot\overline{\overline{X_i}Y_i\overline{Z_i}}\cdot\overline{X_i\overline{Y_i}\overline{Z_i}}\cdot\overline{X_iY_iZ_i}} \qquad P_i = \overline{\overline{X_iY_i}\cdot\overline{X_iZ_i}\cdot\overline{Y_iZ_i}} \tag{5.7}$$

Схема однорозрядного суматора, побудована на елементах НЕ І відповідно до рівнянь (5.7), показана на рис. 5.12, а; її вартість, яка вимірюється числом логічних входів всіх елементів, становить 27, каскадність k=3.

Рівняння (5.4) та (5.5) можуть бути виражені через функцію «Виключальне АБО»:

$$S_i = (X_i \oplus Y_i)\overline{Z}_i \vee (\overline{X_i \oplus Y_i})Z_i = X_i \oplus Y_i \oplus Z_i \tag{5.8}$$

$$P_i = X_iY_i \vee (\overline{X}_iY_i \vee X_i\overline{Y}_i)Z_i = X_iY_i \vee (X_i \oplus Y_i)Z_i \tag{5.9}$$

Схема однорозрядного суматора на елементах «виключальне АБО» згідно з рівняннями (5.8) і (5.9) показана на рис. 5.12, б; її вартість становить вісім входів і каскадність k=2.

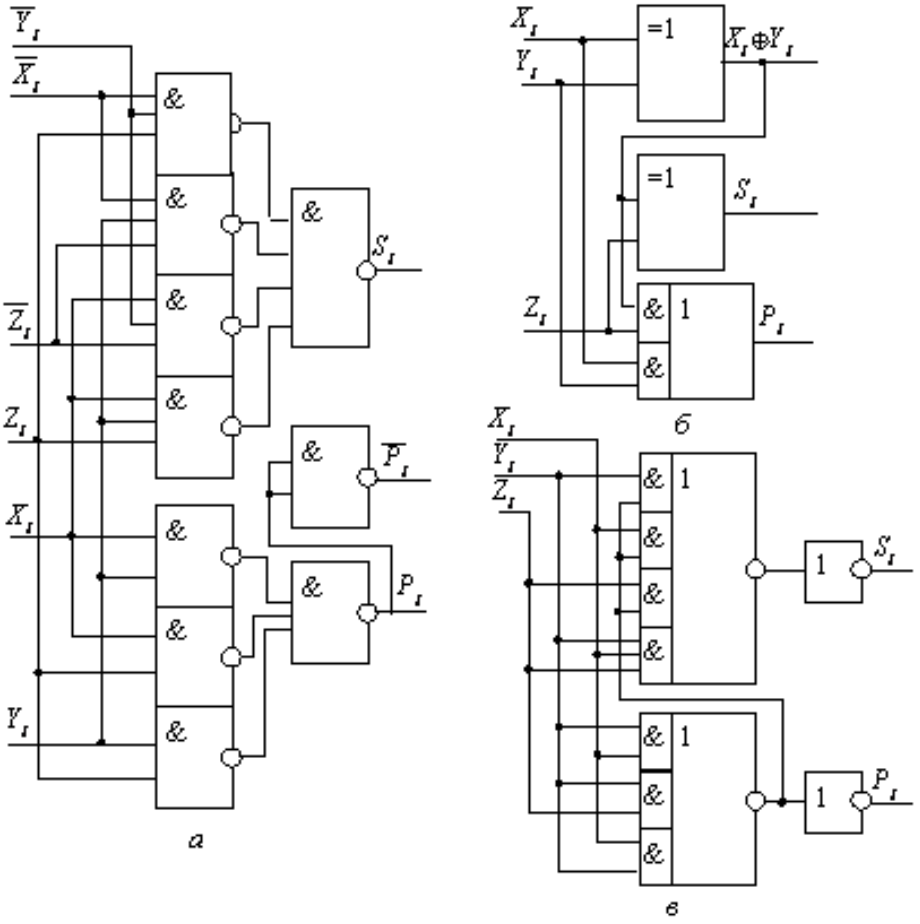


Рис. 5.12. Схеми однорозрядних суматорів: а – на елементах НЕ І;
б – на елементах «виключальне АБО»; в – з використанням власного перенесення

Функції однорозрядного суматора – самоподвійні, тобто їхні інверсії утворюються інвертуванням значень аргументів без зміни місцезнаходження знаків диз'юнкції та кон'юнкції, наприклад, для перенесення з рівняння (5.5):

$$\overline{P}_i = X_i\overline{Y}_i\overline{Z}_i \vee \overline{X}_iY_i\overline{Z}_i \vee \overline{X}_i\overline{Y}_iZ_i \vee \overline{X}_i\overline{Y}_i\overline{Z}_i \tag{5.10}$$

Помножуючи ліві та праві частини співвідношення (5.10) на макстерм $(X_i \vee Y_i \vee Z_i)$, одержують:

$$\overline{P}_iX_i \vee \overline{P}_iY_i \vee \overline{P}_iZ_i = X_i\overline{Y}_i\overline{Z}_i \vee \overline{X}_iY_i\overline{Z}_i \vee \overline{X}_i\overline{Y}_iZ_i \tag{5.11}$$

Після підстановки лівої частини співвідношення (5.11) в праву частину виразу (5.4) одержують рівняння для функції S_i з використанням власного перенесення:

$$S_i = \overline{P}_iX_i \vee \overline{P}_iY_i \vee \overline{P}_iZ_i \vee X_iY_iZ_i \tag{5.12}$$

Схема однорозрядного суматора відповідно до рівнянь (5.12) і (5.6) показана на рис. 5.12, в; її вартість дорівнює 17 входів, каскадність k=2. Важливою властивістю цієї схеми є використання тільки прямих значень вхідних змінних і однофазного ланцюга формування перенесення P_i в старший розряд.

Напівсуматором називається логічна схема, яка виконує додавання значень і-х розрядів X_i і Y_i двійкових чисел X і Y та реалізує на виході значення результату M_i і перенесення в старший сусідній розряд R_i:

$$M_i = \overline{X}_i Y_i \vee X_i \overline{Y}_i = X_i \oplus Y_i; \quad R_i = X_i Y_i \tag{5.13}$$

Таким чином, напівсуматор виконує лише частину завдання підсумовування в і-му розряді, оскільки не враховує перенесення з сусіднього молодшого розряду. Схема напівсуматора, побудована на основі рівнянь (5.13), показана на рис. 5.13. З рівнянь (5.8) і (5.9) виходить, що схема однорозрядного суматора може бути побудована на основі двох напівсуматорів і додаткового логічного елемента АБО, як показано на рис. 5.13, в.

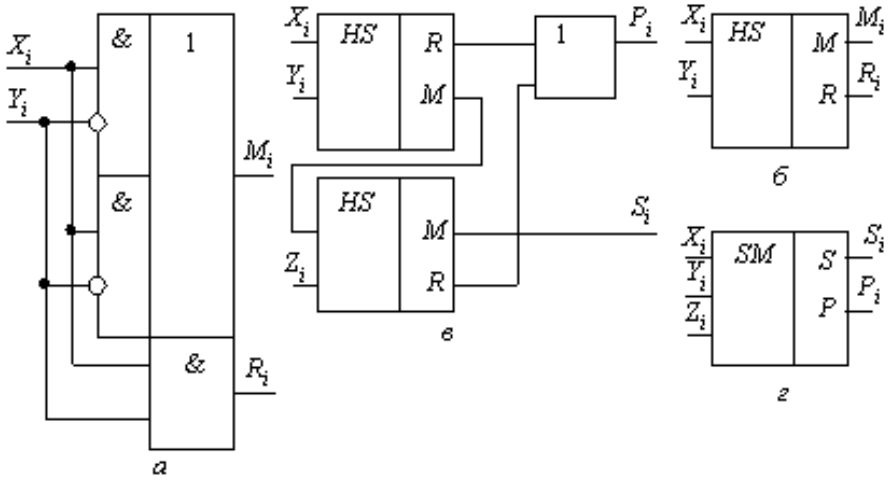


Рис. 5.13. Схеми підсумовування: а, б – напівсуматор і його умовне позначення; в, г – однорозрядний суматор і його умовне позначення

Послідовний багаторозрядний суматор. Послідовний двійковий багаторозрядний суматор містить n-розрядні зсуваючі регістри операндів X і Y, регістр результату S, однорозрядний суматор SM і двоступеневий D-тригер для запам'ятовування перенесення. Усі регістри забезпечують одночасне зсування праворуч, у бік молодших розрядів (рис. 5.14).

У послідовному суматорі попарна подача значень розрядів X_i і Y_i починається з молодших розрядів. Утворюються значення суми S_i і перенесення P_i, які записуються відповідно в регістр результату та в тригер запам'ятовування перенесення на один такт T_c.

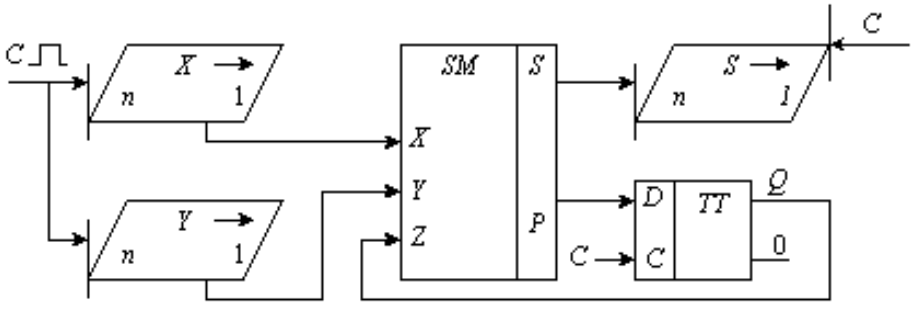


Рис. 5.14. Схема послідовного багаторозрядного суматора

Послідовне додавання виконується за стільки тактів, скільки розрядів у числі. Тому час додавання t_S визначається співвідношенням: $t_S = nT_c$, де T_c – тривалість машинного такту.

Від'ємні числа рекомендується представляти в доповняльному коді.

Послідовний суматор потребує мінімальних апаратних витрат, однак тривалість операції додавання пропорційна розрядності операндів. Тому послідовний суматор можна використовувати у відносно повільнодіючих цифрових пристроях.

Паралельний багаторозрядний суматор містить n однорозрядних схем додавання, наприклад, чотири, як показано на рис. 5.15.

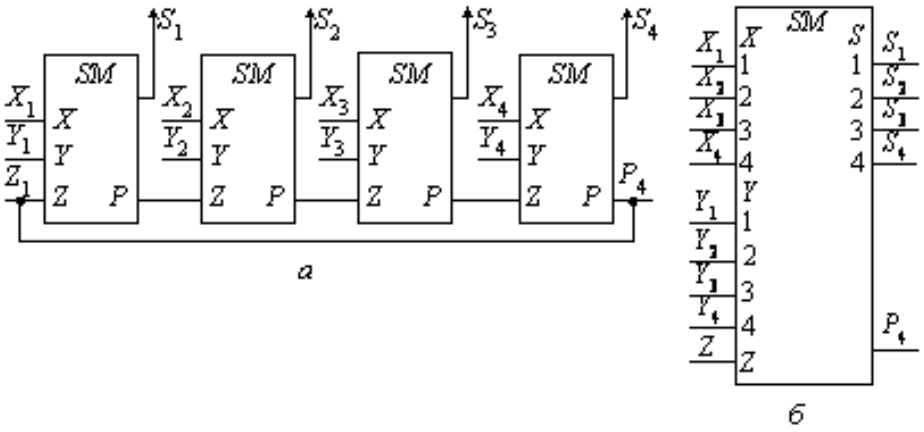


Рис. 5.15. Паралельний чотири розрядний суматор: а – схема; б – умовне позначення

Значення всіх розрядів двох чисел X та Y поступають на входи відповідних однорозрядних суматорів паралельно (одночасно). В паралельних суматорах з послідовним перенесенням значення сигналу перенесення P_i передається від розряду до розряду послідовно в часі (асинхронно). При застосуванні оберненого коду перенесення з найстаршого розряду подається на вхід перенесення молодшого розряду по ланцюзі циклічного перенесення (рис. 5.15, а). При застосуванні доповнялього коду ланцюг циклічного перенесення розривається, а на вхід перенесення молодшого розряду подається логічний нуль.

У паралельних суматорах з послідовним перенесенням час додавання визначається співвідношенням:

$$t_S = (n–1)\, t_П + t_S,$$

де $t_П$ – час формування перенесення в кожному розряді, t_S – час додавання в найстаршому розряді. У гіршому випадку можливий варіант, коли сигнал перенесення послідовно розповсюджується від першого до n-го розряду.

Мікросхеми ALU. Промисловість випускає мікросхеми із символом функції ALU для виконання 16 арифметичних та 16 порозрядних логічних мікрооперацій залежно від вхідних сигналів настройки. У серіях ТТЛШ 530, 531, 533, 555 та 1533 вони мають позначення ИП3; в серіях ЕЗЛ 100, 500 і 700 використовують позначення ИП179.

Мікросхема ALU в серіях ТТЛШ має (рис. 5.16):

- інформаційні входи для подання двох чотирирозрядних операндів X і Y;
- входи настроювання E_3–E_0 для задання номера однієї з мікрооперацій;

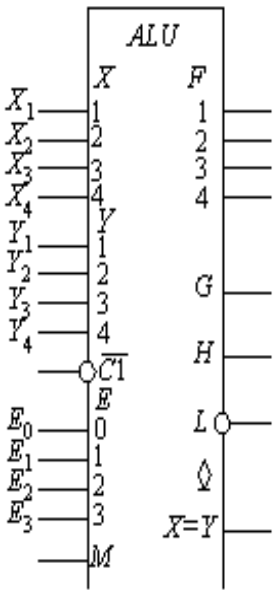


Рис. 5.16. Умовне позначення мікросхеми ALU 1333ИП3

- вхід M для задання типу мікрооперації: M=0 – арифметичні, M=1 – логічні;
- вхід перенесення $\overline{C1}$, необхідний тільки при виконані арифметичних мікрооперацій;
- виходи: результату мікрооперації S_4–S_1, послідовного перенесення L, генерації G, транзиту H, а також вихід з відкритим колектором від внутрішнього компаратора для вироблення ознаки рівності операндів $F_{A=B}$.

Перелік арифметичних і логічних операцій, які виконують ALU, наведений у табл. 5.3. При виконанні логічних операцій перенесення між розрядами не використовується. Арифметичні операції реалізуються з урахуванням перенесень і позик. В арифметичні операції включені фрагменти логічних дій. Наприклад, запис $(X \vee Y) + X\overline{Y}$ означає, що спочатку виконується операція інверсії ($\overline{Y}$), потім – логічного додавання ($X \vee Y$) та логічного множення ($X \cdot \overline{Y}$), а потім одержані таким чином два числа додаються арифметично з урахуванням перенесень.

Таблиця 5.3

E_3	E_2	E_1	E_0	Логіка M=1	Арифметика M=0
0	0	0	0	$\overline{X}$	X
0	0	0	1	$\overline{X \vee Y}$	$X \vee Y$
0	0	1	0	$\overline{X}Y$	$X \vee \overline{Y}$
0	0	1	1	0	-1
0	1	0	0	$\overline{XY}$	$X + X\overline{Y}$
0	1	0	1	$\overline{Y}$	$(X \vee Y) + X\overline{Y}$
0	1	1	0	$X \oplus Y$	$X - \overline{Y} - 1$
0	1	1	1	$X\overline{Y}$	$X\overline{Y} - 1$
1	0	0	0	$\overline{X} \vee Y$	$X + XY$
1	0	0	1	$\overline{X \oplus Y}$	$X + Y$
1	0	1	0	Y	$(X + \overline{Y}) + XY$
1	0	1	1	$X \cdot Y$	$XY - 1$
1	1	0	0	1	$X + Y$
1	1	0	1	$X \vee \overline{Y}$	$(X \vee Y) + X$
1	1	1	0	$X \vee Y$	$(X \vee \overline{Y}) + X$
1	1	1	1	X	$X - 1$

Мікросхема ALU виконує операцію арифметичного додавання двох чотирирозрядних операндів X і Y, якщо на входи настроювання подані сигнали $E_3E_2E_1E_0$=1001 та M=0. В цьому випадку мікросхема ALU виконує функцію суматора.

6. АНАЛОГОВО-ЦИФРОВІ, ЦИФРО-АНАЛОГОВІ ПЕРЕТВОРЮВАЧІ. КОДОПЕРЕТВОРЮВАЧІ

6.1. Цифро-аналогові перетворювачі

Цифро-аналогові перетворювачі (ЦАП) призначені для перетворення цифрової інформації в аналогову форму у вигляді напруги (іноді струму). Їх використовують у системах керування технологічними процесами, в аналогових мікропроцесорах, в дисплеях, графопобудовниках, робототехніці.

Цифро-аналогове перетворення полягає в тому, що для вхідного паралельного n-розрядного коду

$$X = X_1 2^{-1} + X_2 2^{-2} + \ldots + X_i 2^{-i} + \ldots + X_n 2^{-n},$$

де X_i – цифри 0 або 1, а 2^{-i} – вага і-го розряду, спочатку отримують струм I_X, пропорційний значенню числа X, а потім перетворюють його у вихідну напругу. Значення струму I_X визначається сумою еталонних струмів I_i, які створюються для кожного розряду числа

$$I_X = X_1 I_1 + X_2 I_2 + \ldots + X_i I_i + \ldots + X_n I_n,$$

причому підсумовуються струми тільки тих розрядів, для яких $X_i = 1$.Значення еталонів струму I_i пропорційні вазі позиції двійкового числа і зменшуються у два рази при переході від старшого і-го розряду до сусіднього молодшого з номером і+1.

Структура ЦАП вміщує: резистивну або транзисторну матрицю для формування еталонних струмів; ключі для комутації еталонних струмів згідно з вхідним кодом до спільної точки підсумовування; операційний підсилювач (ОП) для перетворення струму I_X у вихідну напругу; допоміжні схеми для узгодження з вхідними рівнями сигналів; стабілізоване джерело опорної напруги $U_{ОП}$.

Резистивні матриці будують або з набору двійково-зважених за номіналами резисторів, або у вигляді сходового (багатоланкового) ланцюжка резисторів лише двох номіналів R - 2R. Схема ЦАП з резистивною матрицею на основі двійково-зважених опорів виду R - 2R … - 2^{n-1}R показана на рис. 6.1. У цій схемі опір резисторів матриці подвоюється при переході від старшого розряду до молодшого, а еталонні струми зменшуються у два рази. Наприклад, якщо для першого, найстаршого розряду взяти значення струму I_1=1 мА, то для другого розряду I_2=0,5 мА, для третього I_3=0,25 мА і т. д.

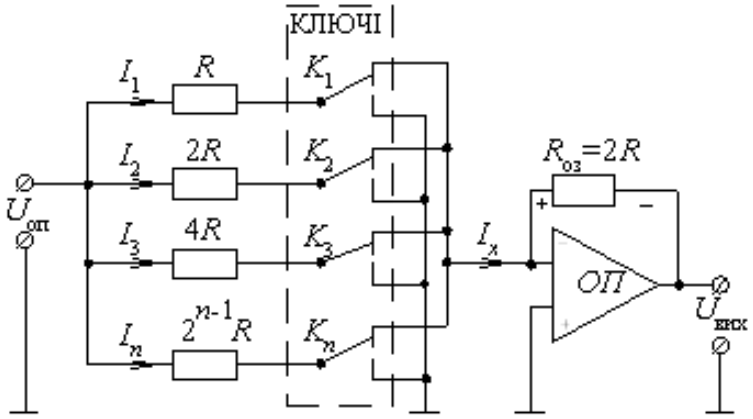


Рис. 6.1. Схема ЦАП із зваженими резисторами

Ключі K_1 – K_n керуються рівнями напруги, які відображають цифри «нуль» і «один» відповідних розрядів вхідного коду. Джерело опорної напруги $U_{ОП}$ найчастіше буває зовнішнім, але у деяких випадках його вбудовують у мікросхему ЦАП. На вході ОП завжди є практично нульовий потенціал, тому додавання розрядних струмів визначається співвідношенням

$$\begin{aligned} I_x &= \frac{U_{ОП}}{R} X_1 + \frac{U_{ОП}}{2R} X_2 + \frac{U_{ОП}}{4R} X_3 + \ldots + \frac{U_{ОП}}{2^{n-1}R} X_n = \\ &= \frac{U_{ОП}}{R}\left(X_1 + 2^{-1} X_2 + 2^{-2} X_3 + \ldots + 2^{-(n-1)} X_n\right) = \frac{U_{ОП}}{R} \sum_{i=1}^{n} X_i 2^{-(i-1)} \end{aligned} \qquad (6.1)$$

Напругу на виході ЦАП розраховують за формулою

$$U_{вих} = -I_x R_{оз} = -U_{ОП} \sum_{i=1}^{n} X_i 2^{-i} \qquad (6.2)$$

де $R_{оз}$ = R/2 – опір у ланцюгу оберненого зв'язку підсилювача.

Основними параметрами ЦАП є число розрядів вхідного цифрового коду, роздільна здатність, похибки перетворення, діапазон вихідних сигналів, динамічні параметри (табл. 6.1).

Таблиця 6.1

Тип мікросхеми	Розрядність, n	Максимальне відхилення, $\delta_{лд}$, %	Час установлення $t_{уст}$, мкс	Технологія
К427ПА4	16	0,0015	20	КМОН
К572ПА2	12	0,025	15	Біполярна
КМ1118ПА1	8	0,75	0,02	Біполярна
К1108ПА3	8	0,5	0,05	Біполярна
КМ1148ПА1	10	0,75	1	Біполярна

Число розрядів n вхідного коду для різних типів ЦАП дорівнює від восьми до вісімнадцяти. Число розрядів визначає максимальну кількість кодових комбінацій на вході ЦАП, що дорівнює 2^n.

Діапазон зміни вихідної напруги (без урахування знака) визначається із співвідношення (6.2) при $X_i = 1$, i = 1, 2, …, n:

$$U_{вих\ max} = U_{оп}(2^{-1}+2^{-2}+\ldots+2^{-n}) = U_{оп}(1-2^{-n}),$$

якщо n = 10, $U_{оп}$ = 10 В, то $U_{вих\ max} \approx 10$ В.

Роздільна здатність h характеризується мінімальним квантом вихідної напруги, який відповідає зміні вхідного коду на одиницю молодшого розряду: $h = U_{оп}/2n \approx 10$ мВ для попереднього прикладу.

Абсолютна похибка перетворення δ_A – відхилення вихідної напруги від розрахункової в кінцевій точці характеристики перетворення. Типова похибка ЦАП не перевищує ±1/2 молодшого розряду.

Нелінійність $\delta_л$ – максимальне відхилення реальної характеристики перетворення від теоретичної (прямої лінії, що з'єднує точку нуля і мінімального вихідного сигналу).

Диференціальна нелінійність $\delta_д$ – максимальне відхилення різниці двох аналогових сигналів сусідніх кодів від значення молодшого розряду.

Параметри δ_A, $\delta_л$ і $\delta_д$ виражаються в частках молодшого розряду або у відсотках від повної шкали вихідної напруги.

Час установлення $t_{уст}$ – інтервал часу від подачі вхідного коду до моменту досягнення вихідним сигналом сталого значення із заданою похибкою (зазвичай ±1/2 молодшого розряду). Цей час визначає загальну швидкодію ЦАП.

Залежно від значень параметрів виділяють прецизійні ($\delta_л$ < 0,1%) і швидкодіючі ($t_{уст} \le 100$ нс) ЦАП.

Розрізняються ЦАП структурою резистивної матриці (зважені резистори або R – 2R), елементною базою, функціональною повнотою, узгоджувальними пристроями, числом джерел живлення та ін.

Реалізуються ЦАП у вигляді гібридних і напівпровідникових мікросхем. Більшість мікросхем ЦАП – напівпровідникові, функціонально закінчені. Ряд ЦАП використовують зовнішні джерела опорної напруги і вихідні операційні підсилювачі.

Живлення ЦАП може здійснюватися від одного або від кількох джерел, що визначається елементною базою.

Перспективами розвитку ЦАП є підвищення швидкодії та точності, зручність узгодження з мікропроцесорами, зниження споживаної потужності.

6.2. Аналого-цифрові перетворювачі

Аналого-цифрові перетворювачі (АЦП) призначені для перетворення аналогової інформації (звичайно у вигляді напруги) у цифровий код. Застосовують АЦП у мікропроцесорних системах, у цифрових вимірювальних приладах. Області застосування їх багато в чому аналогічні ЦАП, оскі-

льки вони часто використовуються спільно, наприклад, в автоматизованих системах керування (АСК) (рис. 6.2).

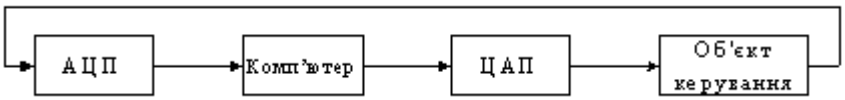


Рис. 6.2. Аналого-цифровий і цифро-аналоговий перетворювачі в контурі керування

Основними параметрами і характеристиками АЦП є:

- число розрядів n вихідного коду;
- роздільна здатність h – мінімальний квант вхідної напруги, за якої вихідний код змінюється на одиницю молодшого розряду;
- нелінійність $\delta_л$ – максимальне відхилення вихідного коду від розрахункового значення у всьому діапазоні шкали;
- абсолютна похибка δ_A – найбільше відхилення вихідного коду від розрахункового в кінцевій точці шкали;
- час перетворення $t_{пр}$ – інтервал від моменту початку перетворення до появи на виході сталого коду; часто замість $t_{пр}$ швидкодія АЦП характеризується частотою перетворення;
- діапазон і полярність вхідної напруги, число джерел живлення, струм споживання, можливість спільної роботи з мікропроцесорами.

У АЦП застосовуються такі методи перетворення:

- послідовної лічби (з використанням ЦАП або з двотактним інтегруванням);
- порозрядного кодування (послідовного двійкового наближення);
- паралельної дії (зчитування);
- паралельно-послідовні (комбіновані).

У АЦП використовують методи порозрядного кодування, послідовної лічби з двійковим інтегруванням і паралельного перетворення. Мікросхеми АЦП виконують за гібридною і напівпровідниковою технологією. У останні роки випускають, в основному, напівпровідникові АЦП. Основні параметри і характеристики деяких напівпровідникових АЦП наведені в табл. 12.2.

Деякі АЦП є функціонально закінченими, але більшість вимагають додаткових зовнішніх елементів: операційних підсилювачів, джерел опорної напруги, генераторів тактових імпульсів, резисторів і конденсаторів.

Таблиця 6.2

Тип мікросхеми	Розрядність, n	Максимальне відхилення, $\delta_{ЛД}$, %	Час перетворення $T_П$, мкс	Технологія	Примітка
КР572ПВ3	8	± 0,75	7,5	КМОН	МК, СМ
К1107ПВ1	6	± 0,5	0,1	Біполярна	—
М1107ПВ6	10	± 1,5	0,06	Біполярна	ФЗ, СМ
К1108ПВ2	12	± 1	0,9	Біполярна	ФЗ

Примітка. У табл. 6.2: СМ – сумісність з мікропроцесорами; ФЗ – функціональна закінченість; МК – багатоканальність.

Мікросхеми АЦП звичайно мають діапазон зміни вхідної напруги 0 – 10 В, а деякі АЦП допускають використання двопо-лярного вхідного сигналу. Розрядність АЦП становить 6 – 12, причому ряд АЦП допускають нарощування розрядності.

Швидкодія АЦП визначається, в основному, методом перетворення і елементною базою (ТТЛШ, ЕЗЛ, КМОН). Найбільшу швидкодію мають АЦП паралельної дії на базі ЕЗЛ-елементів ($t_{пр} \le 20$ нс). Перетворювачі за рівнями вихідних сигналів узгоджуються з ТТЛШ-, ЕЗЛ- і КМОН-мікросхемами. Більшість сучасних АЦП сумісні з мікропроцесорними пристроями. Вихідні ланцюги в таких АЦП мають три стійких стани (лог.0, лог.1 і Z).

Вихідним кодом АЦП найчастіше є двійковий. У АЦП з двійковим інтегруванням застосовується двійково-десятковий код для сполучення з індикаторами і вимірювальними приладами. Тут для представлення кожного десяткового знака використовуються чотири двійкових розряди. Використовуються також обернений і додатковий коди.

Деякі АЦП – це ВІС аналого-цифрової системи збирання даних, що включає в свій склад, крім перетворювача, багатоканальний мультиплексор, оперативний запам'ятовуючий пристрій, схеми буферів і керування.

Основними напрямками вдосконалення АЦП є:

- підвищення швидкодії основних вузлів, особливо компараторів;
- використання АЦП комбінованої дії;
- підвищення точності перетворення;
- зниження споживаної потужності;
- досягнення зручності та гнучкості застосування, особливо узгодження з мікропроцесорними пристроями.

6.3. Перетворювачі кодів

Перетворювачем коду називається функціональний вузол комп'ютера, призначений для перетворення двійкового коду з однієї форми в іншу.

Для подання інформації використовують різноманітні двійкові та двійково-десяткові коди: прямий, обернений, доповняльний і їхні модифікації, циклічний з лишком три та інші. Існує велика кількість кодів, які забезпечують:

- простоту виконання арифметико-логічних операцій;
- зручність переведення чисел з десяткової системи в двійковий код;
- надійність виконання заданих алгоритмів функціонування і ефективний контроль результатів обчислень;
- зменшення апаратних витрат при побудові цифрових пристроїв.

Найбільш поширеними є прямий, обернений і доповняльний коди, які забезпечують представлення знака числа і заміну операції віднімання додаванням. До перетворювачів коду відносяться шифратори і дешифратори, однак за традицією ці функціональні вузли виділені в окремі самостійні класи.

Прямий, обернений і доповняльний коди використовуються для записування знака числа, заміни операції віднімання чисел додаванням їхніх кодів, а також для визначення переповнення розрядної сітки. Для представлення знака числа у них відводиться знаковий розряд, який розташовується зліва від числа і відділяється комою. У знаковий розряд записується нуль – для позитивного числа і одиниця – для негативного.

Перетворювач прямого коду в обернений

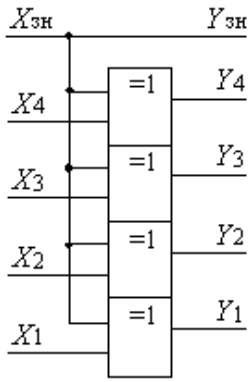


Рис. 6.3. Схема перетворювача прямого коду в обернений

У прямому двійковому коді $X_{ПР} = X_{ЗН}\, X_{n-1},\ldots, X_1$ один розряд, звичайно старший, відображає знак числа, інші – значення цифрових розрядів; при цьому для додатного числа $X_{ЗН} = 0$, а для від'ємного $X_{ЗН} = 1$. Обернений код додатного двійкового числа збігається з прямим кодом, а для від'ємного числа цифрові розряди прямого коду інвертуються.

У процесі перетворення прямого коду в обернений значення знакового розряду $X_{ЗН}$ використовується як керуючий сигнал, що забезпечує отримання такого виразу:

$$Y_i = \overline{X}_{зн} X_i \vee X_{зн} \overline{X}_i = X_{зн} \oplus X_i \qquad (6.3)$$

де Y_i – значення і-го розряду оберненого коду; X_i – значення і-го розряду додатного вхідного числа ($X_{ЗН} = 0$); $\overline{X}_i$ – значення і-го розряду від'ємного вхідного числа ($\overline{X}_{зн} = 1$).

Схема п'ятирозрядного перетворювача прямого коду в обернений, побудована на елементах «виключальне АБО» відповідно до виразу (6.3), показана на рис. 6.3.

Перетворювач прямого коду в доповняльний

Доповняльний код додатного двійкового числа збігається з його прямим і оберненим кодами. Доповняльний код від'ємного двійкового числа утворюється з його оберненого коду додаванням до молодшого розряду одиниці. Таким чином, операція перетворення прямого коду в доповняльний не є порозрядною і виконується значно складніше, ніж отримання оберненого коду.

У загальному вигляді для Y_i справедливе рівняння:

$$Y_i = X_i \oplus (X_{i-1} \vee X_{i-2} \vee ... \vee X_1) X_{зн} \quad (6.4)$$

Схема перетворювача прямого коду в доповняльний на основі виразу (6.4) показана на рис. 6.4, а. Даний перетворювач характеризується високою швидкодією. Час встановлення вихідного коду визначається трьома затримками поширення сигналу, однак в міру зростання номера розряду лінійно зростає й необхідне число входів використовуваних елементів АБО.

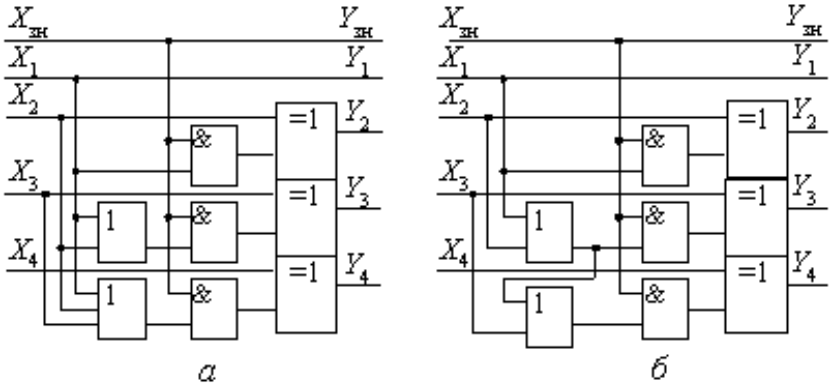


Рис. 6.4. Схеми перетворювачів прямого коду в доповняльний

Другий варіант схеми перетворювача (рис. 12.6, б) використовує тільки двовходові елементи АБО, при цьому диз'юнктивна сума змінних утворюється послідовним способом. У такій реалізації схема перетворювача спрощується, однак час встановлення вихідного коду істотно збільшується.

Практичне правило отримання доповняльного коду полягає в тому, що праворуч від першої одиниці (враховуючи і саму одиницю) в прямому коді числа значення розрядів – незмінні, а зліва від одиниці (крім знакового) – інвертуються. Наприклад, для прямого коду 10100100 доповняльним буде код 11011100.

Для перетворення в доповняльний код багаторозрядних двійкових чисел часто використовують переведення числа в обернений код і подальшого додавання одиниці до його молодшого розряду за допомогою суматора.

Перетворювач двійкових чисел у код Грея

Код Грея утворений послідовністю двійкових чисел, в яких два будь-яких сусідніх числа відрізняються тільки одним розрядом. Перше і останнє числа вважаються сусідніми. Код Грея, який називають циклічним, відноситься до незважених двійкових кодів. Достоїнствами коду Грея є: зручність кодування кутових переміщень; простота кодуючої логіки; скорочення часу перетворення у зв'язку зі зміною значення тільки одного розряду; висока ефективність захисту від збоїв.

Недоліками коду Грея є ускладнення при виконанні арифметичних операцій і цифро-аналогових перетворень. Тому при необхідності код Грея перетворюють у двійковий код.

Вирази для розрядів коду Грея:

$$I_1 = \overline{X}_2 X_1 \vee X_2 \overline{X}_1 = X_1 \oplus X_2;\ I_2 = \overline{X}_3 X_2 \vee X_3 \overline{X}_2 = X_2 \oplus X_3;$$
$$I_3 = \overline{X}_4 X_3 \vee X_4 \overline{X}_3 = X_3 \oplus X_4;\ I_4 = X_4 \quad (6.5)$$

Схема перетворювача прямого коду в код Грея на основі співвідношень (6.5) показана на рис. 6.5.

Вирази для оберненого перетворення коду Грея в прямий код:

$$X_1 = I_1 \oplus I_2 \oplus I_3 \oplus I_4;\ X_2 = I_2 \oplus I_3 \oplus I_4;\ X_3 = I_3 \oplus I_4;\ X_4 = I_4 \quad (6.6)$$

Схема перетворення коду Грея в прямий код на основі співвідношень (6.6) показана на рис. 6.6.

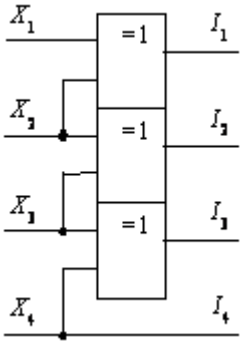


Рис.6.5. Перетворювач прямого коду в код Грея

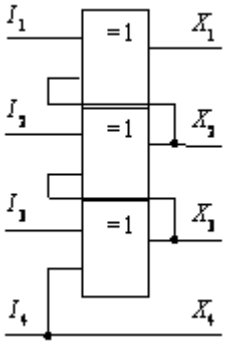


Рис.6.6. Перетворювач коду Грея в прямий код

Перетворювач двійково-десяткових чисел в код семисегментного індикатора

Візуальне відображення двійково-десяткових чисел часто виконується за допомогою семисегментних індикаторів на основі електролюмінісцентних приладів, рідких кристалів або світлодіодних матриць. Кількість семисегментних індикаторів визначається розрядністю чисел, що відображаються на світловому табло – звичайно – шість і більше десяткових цифр.

Десятковий код відображуваної цифри, що виводиться з комп'ютера, поступає на вхід двійково-десяткового перетворювача, виходи якого a, b, c, …, g підключаються до відповідних сегментів індикатора (рис. 6.7, а).

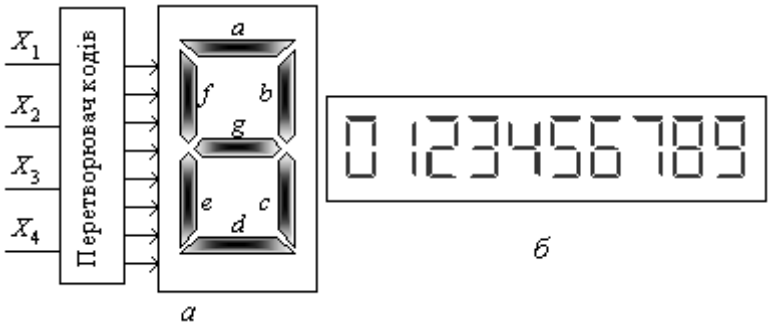


Рис. 6.7. Підключення перетворювача до індикатора (а) і відображення цифр (б)

Одиничне значення вихідного сигналу перетворювача викликає світіння сегмента, підключеного до цього виходу. Комбінації одиничних сигналів на виходах перетворювача утворять зображення десяткової цифри в своєму розряді (рис. 6.7, б).

Система логічних рівнянь для видів перетворювачів кодів має вигляд:

$a = X_2 \vee X_4 \vee X_1X_3 \vee \overline{X}_1\overline{X}_3$; $b = X_1X_2 \vee \overline{X}_1\overline{X}_2 \vee \overline{X}_3 \vee X_4$;

$c = X_1 \vee \overline{X}_2 \vee X_3 \vee X_4$; $d = \overline{X}_1X_2 \vee X_2\overline{X}_3 \vee \overline{X}_1\overline{X}_3 \vee X_1\overline{X}_2X_3 \vee X_4$;

$e = \overline{X}_1X_2 \vee \overline{X}_1\overline{X}_3$; $f = \overline{X}_1\overline{X}_2 \vee \overline{X}_1X_3 \vee \overline{X}_2X_3 \vee X_4$; $g = \overline{X}_1X_2 \vee X_2\overline{X}_3 \vee \overline{X}_2X_3 \vee X_4$.

Схема перетворювача двійково-десяткового коду в керуючі сигнали семисегментного індикатора показана на рис. 6.8.

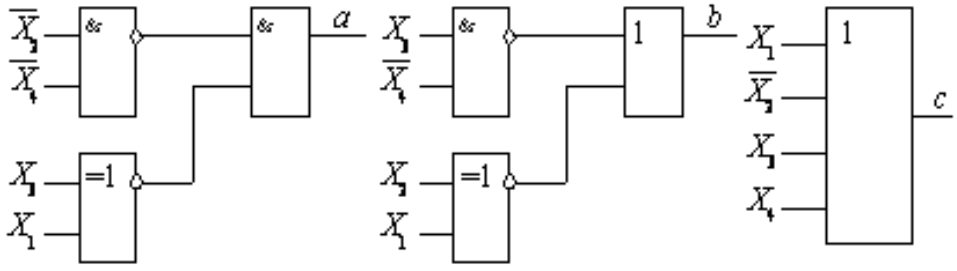


Рис. 6.8. Схема перетворювача коду «8421» в код семисегментного індикатора (для виходів a, b, c)

Двійково-десяткові перетворювачі

У комп'ютерах широко використовується двійково-десяткове кодування, в якому кожна десяткова цифра зображується чотирирозрядним двійковим кодом, тобто тетрадою двійкових символів. Число різних двійково-десяткових кодів визначається числом можливих комбінацій по десять із 16 комбінацій, які допускаються тетрадою. Десяткові числа можуть представлятися в коді «з лишком 3», в коді з вагою «5421» або «2421» (код Айкена) та ін.

Найбільш поширеним є код прямого заміщення «8421», в якому кожна десяткова цифра 0, 1, …, 9 замінюється її двійковим еквівалентом 0000, 0001, …, 1001. Такий код називають також зваженим Д-кодом. Наприклад, число 72910 у двійково-десятковому коді записується у вигляді трьох тетрад: 0111001010012-10.

Для Д-кодів розроблені машинні алгоритми операцій додавання, віднімання, множення, ділення та інші. Операції над десятковими числами (десяткова арифметика) входять до складу команд комп'ютерів різних класів.

Особливістю Д-кодів є наявність десяти дозволених і шести заборонених комбінацій двійкових символів в тетраді. Поява забороненої комбінації при виконанні операцій над числами свідчить про виникнення помилки або ж про необхідність корекції результату.

У розрядній сітці машини двійково-десяткові коди представляються у формі з плаваючою або фіксованою крапкою. При цьому від'ємні числа відображаються в прямому, оберненому або доповняльному кодах. Для Д-кодів не виконується умова отримання оберненого коду інвертуванням розрядів тетради.

Застосування Д-кодів у комп'ютерах не вимагає виконання різного роду перетворень: двійково-десяткових чисел у двійкові та навпаки. Наприклад, за допомогою шифратора забезпечується порівняно простий спосіб введення в машину десяткових цифр двійково-десятковим кодом.

Система рівнянь для розрядів $Y_1 - Y_4$ оберненого Д-коду:

$$Y_1 = \overline{X}_1;\ Y_2 = X_2;\ Y_3 = X_2 \oplus X_3;\ Y_4 = \overline{X}_4\overline{X}_2\overline{X}_2 = \overline{X_4 \vee X_3 \vee X_2} \qquad (6.7)$$

Схема перетворювача прямого Д-коду в обернений на основі співвідношень (6.7) показана на рис. 6.9, а.

Можливий і інший спосіб побудови перетворювача Д-коду в обернений: спочатку інвертуються цифри всіх тетрад (виходить код з лишком шість) і потім відбувається віднімання з кожної тетради числа мінус 0110, що еквівалентне додаванню в доповняльному коді плюс 1010 (без урахування перенесень між тетрадами). Наприклад, А = –148, прямий Д-код $\overline{A}_{ПР}$ = 1 0001 0100 1000; після інвертування маємо: $A_{ПР}$=1 1110 1011 0111; після додавання плюс 1010 одержуємо значення оберненого Д-коду: $A_{ОБ}$ = 1 1000 0101 0001.

Схема перетворювача на основі інвертування і корекції тетрад показана на рис. 6.9, б. В даній схемі інвертування значень розрядів тетрад здійснюється логічними елементами «виключальне АБО», а корекція реалізується типовою мікросхемою чотирирозрядного комбінаційного суматора. Доповняльний Д-код тетради виходить з оберненого додаванням одиниці до молодшого розряду.

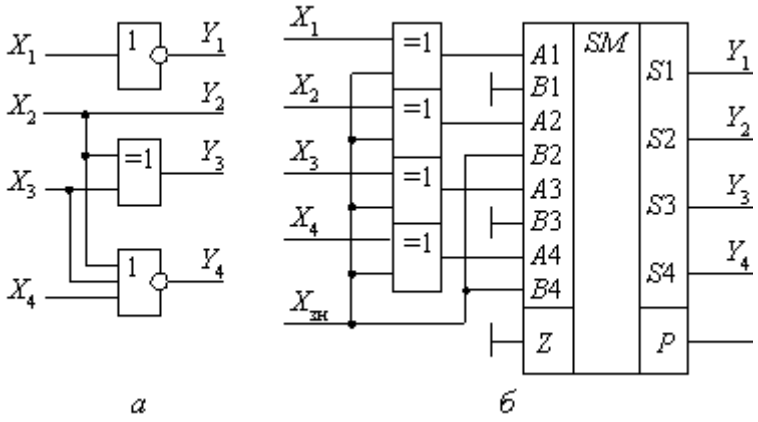


Рис. 6.9. Схеми перетворювачів Д-коду в обернений: а – на основі логічних рівнянь; б – з використанням елемента "виключальне АБО"

ЛІТЕРАТУРА

Основна

1. Схемотехніка електронних систем. Том 2. Цифрова схемотехніка. / Жуйков В.Я., Бойко В.І., Зорі А.А. та ін. – К.: Аверс, 2002. – 772 с.

2. Бабич М.П., Жуков І.А. Комп'ютерна схемотехніка. Навчальний посібник. – К.: НАУ, 2002. – 508 с.

3. Угрюмов Е.П. Цифровая схемотехника. Учебное пособие. – СП.: БХВ Петербург, 2001. – 528 с.

4. Практикум по Electronics Workbench. Под ред. Д.И. Панфилова. – М.: Радио и связь, 2000.

5. Карлащук В.И. Электронная лаборатория на IBM PC. Программа Electronics Workbench и ее применение. – М.: „СОЛОН-Р", 2001. – 726 с.

Додаткова

1. Новиков Ю.В. Основы цифровой схемотехніки. Цифровая и вычислительная техника / Под.ред. проф. Евреинова Э.В. -М.: Радио и связь, 1991.- 326 с.

2. Пухальский Г.И., Новосельцева Т.Я. Проектирование дискретных устройств на интегральных микросхемах: Справочник.-М.: Радио и связь, 1990. – 304 с.

3. Титце У., Шенк К. Полупроводниковая схемотехника: Справочное руководство // Пер. с нем.- М.: Мир, 1983. –512с.

4. Шило В.Л. Популярные цифровые микросхемы: Справочник. – М.:Радио и связь, 1987.- 352 с.

5. Каган Б.М. Электронные вычислительные машины и системы. Учебное пособие для ВУЗов. – М.: Энергоатомиздат. 1985. – 552с.

6. ДСТУ 3212-95. Мікросхеми інтегровані. Класифікація та система умовних позначень.

7. ДСТУ 2399-94. Системи обробки інформації. Логічні пристрої, схеми, сигнали. Терміни та визначення.

8. ДСТУ 2383-94. Мікросхеми інтегровані. Терміни, визначення та літерні позначення електричних параметрів.

9. ДСТУ 2533-94. Системи обробки інформації. Арифметичні та логічні операції. Терміни та визначення.

10. ГОСТ 2.708-81. ЕСКД. Правила выполнения электрических схем цифровой вычислительной техники.

11. ГОСТ 2.7043-91. ЕСКД. Обозначения условные графические в схемах. Элементы цифровой техники.

12. Цифровые интегральные микросхемы: справочник. П.П. Мальцев, Н.С. Долидзе и др. – Радио и связь, 1994. – 240с.

13. Петровский Н.Н. и др. Логические ИС КР1533, КР1554. В 2-х томах. Справочник. – «БИНОМ», 1993. – 550с.

НАВЧАЛЬНЕ ВИДАННЯ

Конспект лекцій з дисципліни «Комп’ютерна схемотехніка (елементи і схеми комп’ютерних систем)» для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Інженерія програмного забезпечення» із спеціальності 121 – «Інженерія програмного забезпечення»

Укладач:
доцент каф. ПЗС, к.т.н.
Бабенко М.В.

Підписано до друку_____
Формат_*А4*_ Обсяг_ др. арк.
Наклад ____ прим. Замовлення ____.
51918, м. Кам'янське
вул. Дніпробудівська, 2.