



Азаров Олексій Дмитрович - д. т. н., професор, Заслужений працівник освіти України, декан Факультету Інформаційних технологій та комп'ютерної інженерії Вінницького національного технічного університету, автор понад 600 наукових праць, серед яких 18 монографій, біля 300 винаходів, 37 навчально-методичних робіт, зокрема, 7 посібників із грифом МОНУ.



Гарнага Володимир Анатолійович - к. т. н., доцент кафедри обчислювальної техніки Вінницького національного технічного університету, автор понад 70 наукових, навчально-методичних праць та винаходів.



Клятченко Ярослав Михайлович - к. т. н., доцент кафедри системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», автор понад 40 наукових, навчально-методичних праць та винаходів.



Тарасенко Володимир Петрович - д. т. н., професор, Дійсний член Академії інженерних наук України, Заслужений діяч науки і техніки України, Лауреат Державної премії України в галузі науки і техніки, голова секції Інформатики та кібернетики НМР МОНУ, завідувач кафедри системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського», автор понад 500 наукових, навчально-методичних праць та винаходів, автор 17 монографій, довідників, підручників та посібників для студентів ЗВО.

КОМП'ЮТЕРНА СХЕМОТЕХНІКА

КОМП'ЮТЕРНА СХЕМОТЕХНІКА

Міністерство освіти і науки України
Вінницький національний технічний університет,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

КОМП'ЮТЕРНА СХЕМОТЕХНІКА

Вінниця
ВНТУ
2018

УДК 004.31
К63

Автори:

О. Д. Азаров, В. А. Гарнага, Я. М. Клятченко, В. П. Тарасенко

Рекомендовано до друку Вченою радою Вінницького національного технічного університету Міністерства освіти і науки України (протокол № 8 від 29.03.2018)

Рецензенти:

І. А. Жуков, доктор технічних наук, професор

С. Д. Погорілий, доктор технічних наук, професор

О. І. Стасюк, доктор технічних наук, професор

Комп'ютерна схемотехніка : підручник / [Азаров О. Д., Гарнага В. А., Клятченко Я. М., Тарасенко В. П.]. – Вінниця : ВНТУ, 2018. – 230 с.
ISBN 978-966-641-736-0

Підручник охоплює двосеместровий курс «Комп'ютерна схемотехніка», який передбачений освітніми стандартами і відповідним навчальними планами для студентів бакалаврського рівня освіти та спеціальності «Комп'ютерна інженерія» в ЗВО України.

Матеріали підручника охоплюють навчальну тематику від фізико-технологічних основ створення електронних комп'ютерних компонент до формально-логічних методів проектування функціонально завершених структурних одиниць обладнання сучасних комп'ютерних систем. Для посилення когнітивного аспекту підручник містить багато прикладів реальних мікроелектронних комп'ютерних засобів і контрольних завдань.

УДК 004.31

ISBN 978-966-641-736-0

©ВНТУ, 2018

ЗМІСТ

ПЕРЕДМОВА	7
РОЗДІЛ 1 ЦИФРОВІ СИГНАЛИ ТА ЛОГІЧНІ ЕЛЕМЕНТИ	9
1.1 Сигнали та їх часові параметри.	9
1.2 Імпульсні та потенціальні сигнали	10
1.3 Скінченні автомати, логічні елементи та логічні схеми	11
1.4 Комбінаційні схеми.....	12
1.5 Автомати з пам'яттю.....	13
1.6 Теорія перемикальних функцій як логіко-математична основа комп'ютерної схемотехніки.....	15
1.6.1 Закони алгебри логіки	16
1.6.2 Алгоритми мінімізації логічних функцій за допомогою діаграм Вейча.....	19
1.7 Контрольні запитання і завдання.....	20
РОЗДІЛ 2 ОСНОВИ СХЕМОТЕХНІКИ НАПІВПРОВІДНИКОВИХ ЛОГІЧНИХ ЕЛЕМЕНТІВ ТА ЇХ КОМПОНЕНТІВ	22
2.1 Базові елементи цифрових пристроїв.....	22
2.2 Резисторно-транзисторна логіка	24
2.3 Діодно-транзисторні логічні елементи (ДТЛ).....	24
2.4 Транзисторно-транзисторні логічні елементи (ТТЛ).	25
2.4.1 ТТЛ із трьома станами виходу	27
2.4.2 ТТЛ із відкритим колектором.....	29
2.4.3 ТТЛ із розширювачами	30
2.5 Логічні елементи на транзисторах Шотткі (ТТЛШ)	31
2.6 Контрольні запитання і завдання.....	33
РОЗДІЛ 3 ЛОГІЧНІ ЕЛЕМЕНТИ НА БІПОЛЯРНИХ І ПОЛЬОВИХ ТРАНЗИСТОРАХ.....	34
3.1 Емітерно-зв'язані логічні елементи (ЕЗЛ).....	34
3.2 Інтегральна інжекційна логіка (ІІЛ)	40
3.3 Логічні елементи на МОН-транзисторах.....	43
3.3.1 Логічні елементи на р-МОН-транзисторах.....	43
3.3.2 Логічні елементи на n-МОН-транзисторах.....	46
3.3.3 Логічні елементи на комплементарних МОН-транзисторах (КМОН-логіка)	47
3.3.4 Динамічні логічні елементи.....	48
3.4 Перетворення рівнів логічних сигналів.	52
3.4.1 Перетворювачі рівнів сигналів.....	52

3.4.2 Транслятори сигналів ВІС і НВІС.....	54
3.5 Контрольні запитання і завдання.....	58
РОЗДІЛ 4 ПЕРЕТВОРЮВАЧІ КОДІВ І КОМБІНАЦІЙНІ	
 ФУНКЦІОНАЛЬНІ ПРИСТРОЇ.....	60
4.1 Перетворення прямого коду двійкових чисел в інверсний та обернений коди.....	61
4.2 Перетворення прямого коду двійкових чисел в доповняльний код.	62
4.3 Різновиди перетворювачів кодів.....	67
4.4 Дешифратори і шифратори	68
4.4.1 Лінійні дешифратори.....	70
4.4.2 Прямокутні дешифратори	72
4.4.3 Пірамідальні дешифратори.....	73
4.4.4 Нарощування розмірності дешифраторів.....	74
4.5 Шифратори.....	75
4.6 Мультиплексори	77
4.7 Демультиплексори.....	80
4.8 Шини передачі даних.....	82
4.9 Пристрої порівняння (цифрові компаратори)	83
4.9.1 Пристрої порівняння на рівність	84
4.9.2 Порівняння на «більше» («менше») однорозрядних слів.....	86
4.9.3 Порівняння багаторозрядних операндів	88
4.9.4 Порівняння слів з константами.	89
4.9.5 Пристрої порівняння двійкових слів	90
4.9.6 Пристрої порівняння двох слів на «більше»	92
4.9.7 Багаторозрядні пристрої порівняння на «більше»	94
4.10 Контрольні запитання і завдання.....	94
РОЗДІЛ 5 КОМП'ЮТЕРНІ ПРИСТРОЇ НА ТРИГЕРАХ	96
5.1 Загальні відомості, характеристики та класифікації тригерів	96
5.2 Тригер як елементарний автомат Мура	102
5.3 Асинхронні та синхронні тригерні пристрої (ТП) зі статичним управлінням	104
5.3.1 RS-тригер та його різновиди.	106
5.3.2 RS-тригер з синхронним записом (СЗ).....	111
5.3.3 JK-тригер	113
5.3.4 JK-тригер з синхронним записом.....	114
5.3.5 Т-тригер	115
5.3.6 Т-тригер з СЗ	116

5.3.7 D-тригер	117
5.3.8 D-тригер з СЗ.....	118
5.4 Синхронні тригери з динамічним записом інформації	121
5.4.1 D-тригер	121
5.4.2 RS-тригер з ДЗ.....	122
5.4.3 JK-тригер з ДЗ	125
5.5 Синхронні двоступеневі тригери за структурою MS	127
5.5.1 ТП типу MS з інвертором	129
5.5.2 ТП із двома інверторами	130
5.5.3 ТП на основі тригерів із заборонними зв'язками	131
5.5.4 ТП на основі тригерів з різнополярним управлінням	132
5.5.5 Схема тригера типу MS з комутуючими транзисторами.....	134
5.5.6 D-тригери типу MS	135
5.5.7 JK- тригер типу MS із заборонним зв'язком.....	136
5.6 Асинхронні входи синхронних тригерів.....	138
5.7 Контрольні запитання і завдання.....	138
РОЗДІЛ 6 РЕГІСТРИ.....	140
6.1 Паралельні реєстри.....	140
6.2 Послідовні реєстри (реєстри зсуву).....	140
6.3 Послідовні реєстри зі зворотними зв'язками (кільцеві лічильники).....	143
6.4 Контрольні запитання і завдання.....	144
РОЗДІЛ 7 ЛІЧИЛЬНИКИ	145
7.1 Двійкові лічильники з послідовним переносом.	145
7.2 Двійкові лічильники з наскрізним і паралельним переносом	146
7.3 Декрементні та реверсивні лічильники.....	147
7.4 Лічильники з довільним коефіцієнтом лічби	148
7.5 Десяткові лічильники.....	151
7.6 Лічильники в коді Фібоначчі	152
7.7 Контрольні запитання і завдання.....	155
РОЗДІЛ 8 СУМАТОРИ.....	156
8.1 Однорозрядні напівсуматори і суматори.	157
8.2 Накопичувальний послідовний суматор на базі лічильного тригера.....	162
8.3 Паралельний багаторозрядний суматор із послідовним переносом	164
8.4 Паралельний багаторозрядний суматор із паралельним переносом	165

8.5 Суматори з груповою структурою.....	168
8.6 Суматор з паралельно-паралельним переносом	170
8.7 Послідовний n-розрядний суматор	171
8.8 Десяткові і двійково-десяткові суматори	172
8.9 Суматори-віднімачі	175
8.10 Контрольні запитання і завдання.....	176
РОЗДІЛ 9 СХЕМОТЕХНІЧНІ КОМП'ЮТЕРНІ ПРИСТРОЇ	
І ВУЗЛИ ВИСОКОЇ ВИРОБНИЧОЇ СЕРІЙНОСТІ.....	177
9.1 Матричні перемножувачі.....	177
9.2 Арифметико-логічні пристрої.....	180
9.3 Пристрої технічної пам'яті.....	183
9.4 Елементи пам'яті на біполярних транзисторах.....	188
9.5 Елементи технічної пам'яті на МОН – транзисторах.....	190
9.6 Стекова і магазинна пам'ять	194
9.6.1 Асоціативна технічна пам'ять	195
9.6.2 Постійні та перепрограмовні засоби ТП	196
9.7 Контрольні запитання і завдання	201
РОЗДІЛ 10 СТРУКТУРНА ОРГАНІЗАЦІЯ ПРОГРАМОВНИХ	
ЛОГІЧНИХ ІНТЕГРАЛЬНИХ СХЕМ І ЇХ ВИКОРИСТАННЯ	
В КОМП'ЮТЕРНІЙ СХЕМОТЕХНІЦІ.....	
10.1 Загальні відомості щодо програмовних логічних	
інтегральних схем	202
10.2 Класифікація ПЛІС за архітектурними властивостями.....	203
10.3 Сучасні тенденції розвитку пристроїв із програмовною	
логікою	219
10.4 Обчислювальні системи на ПЛІС із soft-процесором	221
10.5 Контрольні запитання і завдання.....	227
СПИСОК ЛІТЕРАТУРИ.....	228

У зв'язку з повторюваністю в тексті різносмыслових скорочень і умовних позначень їх переліки, які ще не використовувались попередньо, наведені на початку кожного розділу, де вони вперше використані. Аббревіатури англomовного походження розкриті контекстуально у відповідних розділах.

ПЕРЕДМОВА

У навчальних планах технічних ЗВО України дисципліна «Комп'ютерна схемотехніка» є нормативною професійно-орієнтованою дисципліною підготовки фахівців зі спеціальності «Комп'ютерна інженерія» і має основоположне значення для підготовки фахівців освітніх рівнів бакалавра і магістра як в межах «Комп'ютерної інженерії», так і низки суміжних спеціальностей, пов'язаних із комп'ютерною технікою, цифровим приладобудуванням, цифровими телебаченням і радіозв'язком, інформаційно-вимірювальною технікою, системами телекомунікацій та іншими електронними галузями людської діяльності. Очевидно, що означення «комп'ютерна» є цілком зрозумілим і окремого коментаря не потребує. Водночас термін «схемотехніка», зважаючи на його локальне (щодо географії) використання, вимагає певного попереднього пояснення. В українському науково-технічному лексиконі термін «схемотехніка» свого часу з'явився як пряма калька з російської мови, куди, в свою чергу, він проник із професійного жаргону розробників різних електронних засобів. Така ситуація є наслідком того, що в українській мові (як і в російській) кореневе поняття «схема» набуло двох самотійних значень: перше – це умовне графічне зображення деякого електричного кола (що, до речі, відповідає світовому контексту слова *scheme*); друге – це саме електричне коло деякого призначення. «Жаргонне» походження терміна підтверджується також тим, що у провідних науково-технічних лексиконах (наприклад, в британсько-американському) немає прямого відповідника поняттю «комп'ютерна схемотехніка». Як показує аналіз зарубіжних науково-технічних і офіційних джерел (наприклад, *Computer Engineering Curricula 2016, Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering*), там цьому терміну за змістом відповідають *Discrete Structures*, *Computer Circuitry* або *Digital and Analog Circuits*. Однак, незважаючи на такі «дефекти» в етимології терміна, він «вижив» і зараз досить широко використовується в освіті, в науковій та в проектній діяльності. Тому далі «комп'ютерною схемотехнікою» будемо називати науково-технічну дисципліну, яка вивчає методи аналізу, синтезу і оптимізації апаратних засобів (*hardware*) комп'ютерів і комп'ютерних систем, а також шляхи їх технічної реалізації. Разом з тим, будемо уникати, по можливості, розширеного трактування терміна «схема», замінюючи його вужчими поняттями «пристрій», «структура», «засіб» тощо, які мають прямих відповідників у провідних науково-технічних лексиконах.

Як науково-технічний напрям «Комп'ютерна схемотехніка» спирається на новітні технології виготовлення різноманітних електронних компонентів, що використовуються в цифровій, аналого-цифровій і аналоговій техніці. Причому саме характеристики електронних компонентів вирішальною мірою визначають можливості комп'ютерних систем і мереж. Водно-

час, покращення властивостей інтегральних транзисторів, діодів, резисторів, конденсаторів та іншого дає можливість вдосконалювати цифрові пристрої, мінімізувати їх масо-габаритні показники, збільшувати швидкодію і розширювати функціональні можливості. При цьому з'являються нові схемотехнічні і функціональні пропозиції щодо побудови апаратних блоків комп'ютерних систем і мереж, застосування їх в якісно новій комп'ютерній архітектурі. За останні десятиліття як в Україні, так і за кордоном з'явилися різноманітні підручники, присвячені комп'ютерній схемотехніці. Однак швидкий розвиток комп'ютерно-орієнтованих технологій і вимоги, що постійно зростають, до структурно-функціональних та техніко-економічних характеристик комп'ютерів і комп'ютерних систем вимагають регулярного оновлення тематики навчально-методичної літератури.

Обсяг і зміст підручника відповідають нормативним показникам, що вказані в освітньому стандарті спеціальності «Комп'ютерна інженерія», і тому є мінімально достатніми для засвоєння студентами. За необхідності глибшого вивчення деяких розділів чи тем це варто робити шляхом введення відповідних змін в варіативній частині навчальних планів. Для засвоєння матеріалу підручника необхідно мати підготовку з дисциплін «Основи теорії інформації і кодування», «Теорія електричних і магнітних кіл», «Комп'ютерна логіка» та «Комп'ютерна електроніка» в обсязі, передбаченому освітніми стандартами для студентів спеціальності «Комп'ютерна інженерія». З метою вирівнювання попередніх знань студентів перші два розділи книги містять інформацію «повторювально-втягувального» характеру. Наприкінці підручника наведено список основної сучасної літератури, опублікованої у цій галузі в Україні і за кордоном та рекомендованої для поглибленого вивчення певних розділів.

Матеріали підручника підготовлені викладачами профільних кафедр, що забезпечують спеціальність «Комп'ютерна інженерія» у Вінницькому національному технічному університеті та в Національному технічному університеті «Київський політехнічний інститут імені Ігоря Сікорського». Крім власне програмних положень підручник містить практично всі методичні знахідки, породжені більш ніж двадцятирічним досвідом викладання цієї дисципліни (з назвами, які часом змінювалися) на вказаних кафедрах. Під час підготовки текстового матеріалу книги автори звертали особливу увагу на формування, вибір та застосування лексики, граматики і стилістики професійної української мови. Це дозволило мінімізувати вплив фахового жаргону на вміст підручника. У цьому плані плідно здійснювалися консультації з провідними фахівцями Інституту української мови НАН України. Підручник розраховано на студентів, аспірантів, науковців і фахівців-практиків, які вивчають комп'ютерну схемотехніку чи займаються розробкою комп'ютерних засобів. Автори будуть вдячні за відгуки щодо вдосконалення матеріалів книги.

1 ЦИФРОВІ СИГНАЛИ ТА ЛОГІЧНІ ЕЛЕМЕНТИ

Перелік скорочень і умовних позначень до розділу 1

ДФ – диз'юнктивна форма; ДНФ – диз'юнктивна нормальна форма; ДДНФ – доконана ДНФ; ДКНФ – доконана КНФ; Еп – елемент пам'яті; ІМС – інтегральна мікросхема; КС – комбінаційна схема; КФ – кон'юнктивна форма; КНФ – кон'юнктивна нормальна форма; ЛС – логічна схема; ЛЕ – логічний елемент; ЛФ – логічна функція; МДНФ – мінімальна ДНФ; СІ – ступінь інтеграції.

1.1 Сигнали та їх часові параметри

Для подання інформації в комп'ютерній техніці використовують певні фізичні процеси, наприклад, електричний струм і магнітний потік. Фізичний процес характеризується деякою величиною, що визначає стан процесу. Так, електричний процес може характеризуватися величиною струму або напруги. Абстрагуючись від фізичної суті процесу, що використовується для подання інформації, кажуть, що стан процесу на деякий момент часу є сигналом. Величина, що характеризує сигнал, змінюється безперервно в часі, приймаючи будь-які значення в деякому діапазоні.

Процеси передачі, зберігання і перетворення інформації реалізуються найпростіше, якщо розрізняти тільки два рівні (дві величини) сигналу, які ототожнюють з символами 0 і 1. Сигнал, який може набувати одне з двох можливих значень, називають двійковим. Відповідність між фізичним процесом і двійковим сигналом встановлюють шляхом квантування процесу за рівнем і квантування в часі.

Квантування за рівнем полягає в розбитті сигналу на окремі значення. Нехай сигнал характеризується фізичною величиною v і змінюється в часі, як показано на рис. 1.1, а). З фізичних міркувань призначають два рівні сигналу a та b (тут $a < b$), які виділяють дві зони $v \leq a$ та $v \geq b$ значень. У зоні $a < b < v$ значення двійкового сигналу не визначене. Встановлення відповідності між рівнями фізичного та логічного сигналів називають кодуванням сигналу. Можливі два способи кодування двійкового сигналу s : двійкові сигнали, що відповідають цим двом способам кодування, наведені на рис. 1.1, б), в).

У інтервалах (t_1, t_2) і (t_3, t_4) (рис. 1.1, а)) відбувається зміна значення сигналу. Проміжок $\mathcal{I} = (t_2 - t_1)$ – називають часом перемикання сигналу в 1, проміжок $\mathcal{O} = (t_4 - t_3)$ – часом перемикання сигналу в 0. Сигнали однієї фізичної природи можуть мати різні значення. Для оцінювання часу перемикання за множиною можливих сигналів використовують граничні, середні або середньоквадратичні значення. В розрахунках зручно використовувати середній час перемикання сигналу $\mathcal{S} = (\mathcal{O} + \mathcal{I})/2$. За

необхідності відображення динаміки процесу сигнал подають у вигляді (рис. 1.1, з)).

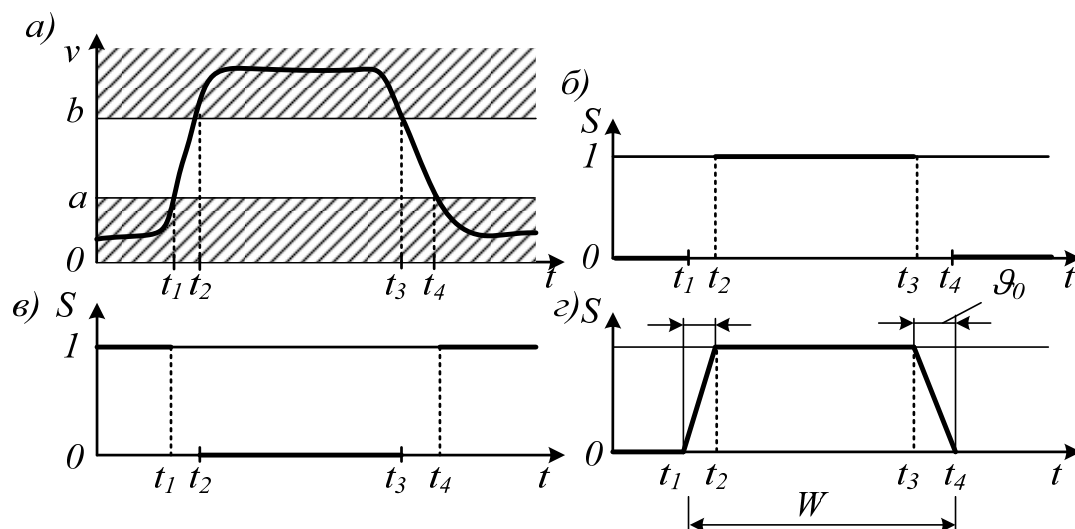


Рисунок 1.1 – Цифровий сигнал: а) узагальнена форма цифрового сигналу; б), в), г) – його подання

Квантування фізичного процесу в часі зумовлено тим, що в багатьох практичних випадках викликає інтерес саме інформація, яку несе сигнал, а часові його характеристики є несуттєвими. Тому в таких ситуаціях можна ігнорувати невизначеність рівня сигналів в моменти їх перемикання. При цьому сигнали розглядають як процеси дискретного часу $t_n = nT$, де $n = 0, 1, \dots$ і T – такт, значення якого відповідає умові $T > \mathcal{Q}$. При цьому вважають, що сигнали перемикаються тільки в межах інтервалу дискретності, тобто перехідні процеси встановлюються між моментами t_n і $t_n + 1$. В такому випадку сигнал вважають визначеним 0 або 1 на будь-який момент t_n . Щоб не посилалися на значення T , вводять поняття безрозмірного дискретного часу $t = t_n / T$, $t = 0, 1, 2, \dots$

Таким чином, процедура квантування фізичного процесу в часі дозволяє уникнути необхідності постійних посилань на фізичну природу сигналів і зосередити увагу на інформаційних аспектах роботи цифрових пристроїв.

1.2 Імпульсні та потенціальні сигнали

Проміжок часу $W = t_4 - t_1$ (рис. 1.1, г)) називають тривалістю сигналу. Залежно від тривалості сигнали прийнято поділяти на два типи: імпульсні та потенціальні.

Імпульсний сигнал (імпульс) носить миттєвий характер (вважають, що ідеальний імпульсний сигнал має нульову тривалість) і приймає одиничні значення тільки в дискретні моменти часу $t = 0, 1, 2, \dots$. Реальний імпульс –

це сигнал з тривалістю до $W < T$. Очевидно, що тривалість імпульсного сигналу не може бути меншою ніж $W \geq (\mathcal{G}_0 + \mathcal{G}_1) = 2\mathcal{G}$. Приклад імпульсного сигналу наведено на рис. 1.2, а)–в). Потенціальний сигнал приймає значення 1 на інтервалі, що не менший тривалості такту T (рис. 1.2, г), д)).

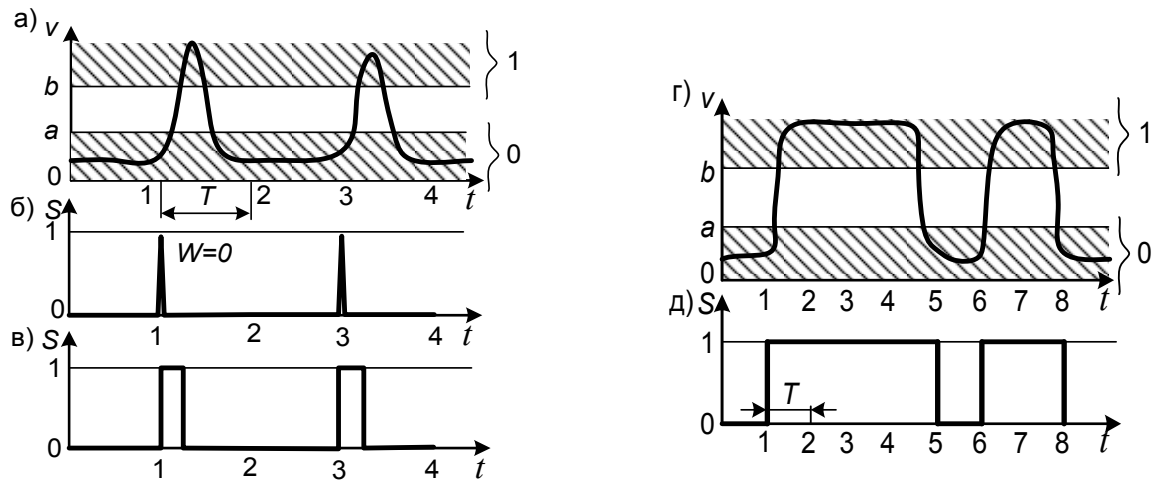


Рисунок 1.2 – Різновиди цифрових сигналів: а) імпульсний сигнал та його ідеалізоване зображення б), в); потенціальний сигнал г) та його ідеалізована форма д)

Якщо моменти перемикання сигналів приведені до моментів дискретного часу $t = 0, 1, 2, \dots$, то тривалість потенціальних сигналів приймається кратною інтервалу дискретності: $\omega W = kT, k = 1, 2, \dots$. Потрібно зазначити, що взагалі поділ сигналів на імпульсні і потенціальні має відносний характер: тип сигналу практично визначається тривалістю такту T .

1.3 Скінченні автомати, логічні елементи та логічні схеми

У цифровій техніці прийнято вважати, що найзагальнішою моделлю будь-якого технічного засобу для перетворення цифрової інформації є так зване автоматне його подання у вигляді «чорного ящика» («black box» – пристрій з невідомою внутрішньою структурою), який має скінченну множину Q внутрішніх станів; на вході якого діє узагальнений сигнал X , що може приймати скінченне число значень; на виході якого з'являється результат перетворення інформації, тобто узагальнений сигнал Y , як функція конкретного внутрішнього його стану з множини Q і конкретного вхідного сигналу з множини X ; при цьому, внаслідок скінченності X і Q , узагальнений вихідний сигнал Y також може приймати тільки скінченне число значень. Тому технічну реалізацію такої моделі (рис. 1.3) називають скінченним (рос. – «конечным», англ. – «finite») або цифровим (англ. – «digital») автоматом.

Узагальнені сигнали X та Y називають також вхідними і вихідними словами, а елементарні сигнали x_i та y_j ($i = 1, 2, \dots, n; j = 1, 2, \dots, m$), з яких

формують X та Y , називають вхідними та вихідними буквами. (Тут і далі поняття «елементарний сигнал, елементарний вхід, елементарний вихід тощо» означають «найпростіший» або такий, що може бути переданий по одному окремому фізичному провіднику). Вхідні слова X викликають переходи автомата з одного внутрішнього стану до іншого, причому кожен такий перехід супроводжується формуванням нового вихідного слова.

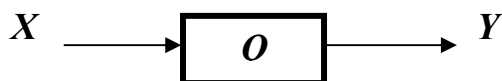


Рисунок 1.3 – Узагальнене автоматне подання комп’ютерних апаратних засобів (hardware)

Логічним елементом називають пристрій, котрий реалізує певну перемикальну функцію (підрозділ 1.6) і призначений для обробки (перетворення) інформації в цифровій формі. Кажуть, що ЛЕ виконують певну логічну функцію (логічну операцію, логічне перетворення) над вхідними аргументами (даними, цифрами операндів, сигналами). В деяких випадках, наприклад, коли відповідно до контексту деякого викладу конкретний тип елементарної перемикальної функції, відтворюваної ЛЕ, не є принципово важливим, вживають практично синонімічний термін «вентиль» або «логічний вентиль».

Як фізичні об’єкти ЛЕ можуть бути виготовлені на основі використання явищ різної фізичної природи (як механічні, електромеханічні, пневматичні, гідравлічні, оптичні і т. п. прилади), однак найбільшого поширення набули саме електронні ЛЕ, в яких носієм інформації є електричний сигнал. Конструктивно ж сучасні ЛЕ найчастіше виконують як інтегральні мікросхеми (ІМС) різного ступеня інтеграції. При цьому ступінь інтеграції (СІ) визначають як $\lceil \lg N \rceil$, де N – кількість внутрішніх активних електронних компонентів (найчастіше транзисторів) ІМС, а квадратні дужки означають округлення в бік найближчого більшого цілого числа. Наприклад, $N \leq 10$ для ІМС першого ступеня інтеграції, $N \leq 10^2$ для ІМС другого ступеня, $N \leq 10^3$ для ІМС третього ступеня і т. д.

Сукупність ЛЕ, призначених для перетворення цифрових сигналів і з’єднаних відповідно до правил суперпозиції, називають логічною схемою (ЛС). Залежно від функцій, виконуваних ЛС, розрізняють два основних їх типи:

- комбінаційні схеми або скінченні автомати без пам’яті;
- скінченні автомати з пам’яттю.

1.4 Комбінаційні схеми

Комбінаційні схеми (КС) складаються тільки з ЛЕ (АБО, І, НЕ, І-НІ, АБО-НІ і т. д.). У загальному випадку окрема КС має n елементарних вхо-

дів і m елементарних виходів. Узагальнений сигнал X для КС у вигляді деякого слова подають як комбінацію елементарних сигналів (окремих букв) x_i на входах КС (рис. 1.4). Узагальнений сигнал Y на виходах КС є, відповідно, комбінацією елементарних сигналів y_j на окремих її виходах. Для будь-якої КС сигнал Y залежить тільки від вигляду функції F , що її реалізує КС, та від вхідного сигналу X , тобто від комбінації елементарних сигналів на входах КС:

$$Y = F(X). \quad (1.1)$$

Очевидно, ця «залежність від комбінації» і зумовила назву даного типу ЛС. Далі будемо вважати, що окремі елементи і КС в цілому передають елементарні сигнали від входу до виходу без їх затримки в часі. У разі, коли це припущення неприйнятне і затримку сигналів слід враховувати, то цей факт буде відзначений і досліджений окремо.

Таким чином, при будь-якій зміні сигналів на вході КС практично миттєво мають змінюватися сигнали на її виходах.

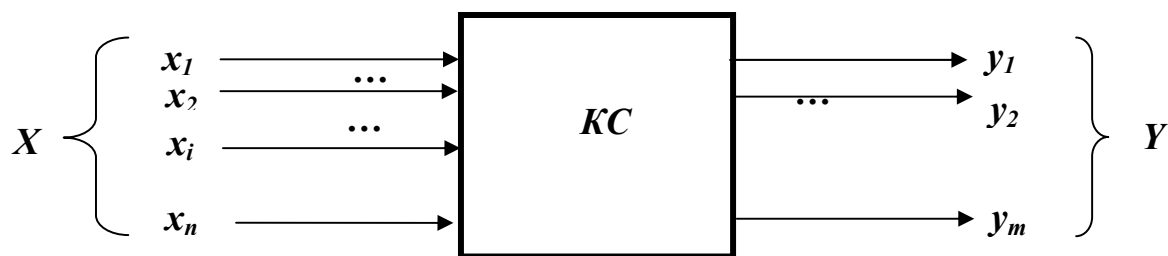


Рисунок 1.4 – Узагальнене подання комбінаційної схеми

При багаторазовому надходженні одного і того ж вхідного сигналу КС видає один і той же вихідний сигнал. Отже, сигнали на виході КС не залежать від того, які сигнали надходили на її вхід до поточного моменту часу. Тому КС використовуються для одномоментного перетворення вхідного слова у вихідне.

1.5 Автомати з пам'яттю

Будь-який скінченний автомат з пам'яттю містить у своїй структурі (рис. 1.5) ЛЕ і елементи пам'яті. На рис. 1.6 прийняті такі позначення: $ЕП_r$ – елементи пам'яті, $r = 1, 2, \dots, m$; X_t – узагальнений вхідний сигнал в поточному такті; x_i – сигнал на i -му вході автомата, $i = 1, 2, \dots, n$; Y_t – узагальнений вихідний сигнал в поточному такті; y_j – сигнал на j -му виході автомата, $j = 1, 2, \dots, k$; Q_t – стан автомата в поточному такті; q_m – стан m -го елемента пам'яті, $m = 1, 2, \dots, m$. Вміст пам'яті автомата в цілому (Q) називають внутрішнім станом автомата. Він визначається станом окремих

елементів пам'яті (q_i) автомата. Сигнал на виході скінченного автомата залежить як від вхідного сигналу, так і від внутрішнього стану автомата. Таким чином, в загальному випадку

$$Y_t = F(X_t, Q_t). \quad (1.2)$$

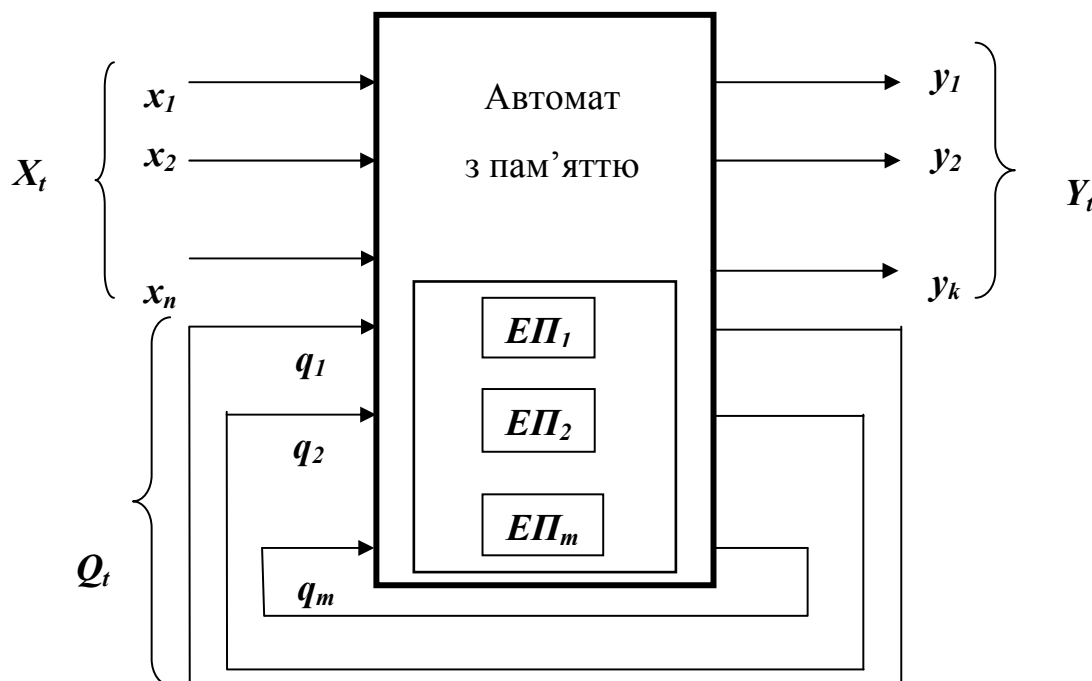


Рисунок 1.5 – Узагальнена структура автомата з пам'яттю

Конкретну залежність вигляду (1.2) називають також функцією виходів автомата. З попереднього викладу зрозуміло, що для одного і того ж вхідного сигналу, залежно від внутрішнього стану, автомат може формувати різні вихідні сигнали. Тому автомат з пам'яттю може використовуватися для виконання певної послідовності неповторюваних дій, тобто певного розгалуженого алгоритму перетворення інформації.

Скінченні автомати можуть бути асинхронними і синхронними. Далі будемо розглядати переважно синхронні автомати. Робота синхронних автоматів відбувається в дискретні моменти часу, які називають тактами. У проміжках між тактами автомат не реагує на вхідні сигнали. Такти задають за допомогою генераторів синхронізувальних (тактових) імпульсів.

При роботі автомата в кожному такті, крім вихідного сигналу, формується також новий стан, який залежить від вхідного сигналу і поточного стану автомата, тобто

$$Q_{t+1} = P(X_t, Q_t) \quad (1.3)$$

Конкретну залежність вигляду (1.3) називають функцією переходів автомата. З функції (1.3) видно, що після початку роботи автомата зі стану Q_0 його наступні стани пов'язані з вхідними сигналами так: $Q_1 = P(X_1, Q_0)$, тобто Q_1 залежить від сигналу X_1 і стану Q_0 ; $Q_2 = P(X_2, Q_1)$, тобто Q_2 зале-

жить від сигналу X_2 і стану Q_1 . Оскільки $Q_1 = P(X_1, Q_0)$, то $Q_2 = P(X_1, X_2, Q_1)$, тобто Q_2 залежить від сигналів X_1, X_2 і стану Q_1 . Аналогічно можна показати, що стан Q_3 залежить від сигналів X_1, X_2, X_3 і стану Q_2 і т. д. У наведених вище міркуваннях індекси при X і Q означають номери тактів. Таким чином, можна зробити висновок, що кожний стан автомата залежить від усіх вхідних сигналів, що надійшли на попередніх тактах, тобто від передісторії вхідних сигналів. Оскільки вихідний сигнал залежить від поточного стану, то він також визначається всією передісторією вхідних сигналів. Саме цим і пояснюються більш широкі можливості перетворення інформації в автоматах з пам'яттю порівняно з комбінаційними схемами.

1.6 Теорія перемикальних функцій як логіко-математична основа комп'ютерної схемотехніки

Перемикальними (рос. – «переключательными», англ. – «switching») називають функції, які, як і їх аргументи, набувають значення з деякої скінченної множини допустимих значень, що звичайно пронумеровані цілими числами. Найчастіше для цього використовують двійкову множину з номерами 0 і 1. Перемикальні функції з двійковими допустимими значеннями вперше досліджував в середині XIX століття ірландський математик Джордж Буль, пізніше такі функції стали називати логічними або булевими. Системи булевих функцій, що задовольняють певні вимоги, назвали алгебрами логіки або булевими алгебрами. В подальшому найсуттєвіших результатів в теорії перемикальних функцій стосовно її використання для проектування ЛС і цифрових пристроїв досягли в 1910 році П. С. Еренфест і в 1938 році К. Шеннон.

Отже, логічною функцією (ЛФ) називають функцію $F(x_1, x_2, \dots, x_n)$, яка, як і її аргументи $x_1, x_2, \dots, x_n$, може приймати тільки два значення: -0 або 1. Для будь-якої логічної функції від n змінних існує $N = 2^n$ різних наборів значень аргументів, на кожному з яких функція має бути визначена нулем або одиницею. Таку функцію називають повністю визначеною. При цьому число всіх можливих повністю визначених логічних функцій від n змінних дорівнює 2^N . ЛФ може бути задана словами природної мови, алгебраїчним виразом або таблично за допомогою так званих таблиць відповідності (таблиць істинності). Якщо логічна функція задана не на всіх наборах аргументів, то її називають неповністю визначеною або частковою функцією. При табличному поданні часткових функцій їх значення на невизначених наборах аргументів задають деяким символом (наприклад, $\emptyset$), який не входить до множини $\{0,1\}$ і може бути, за необхідності, замінений нулем або одиницею.

Систему ЛЕ, що дозволяє реалізувати на їх основі логічні функції будь-якої складності, називають функціонально повною системою або базисом. Наприклад, базис утворюють ЛЕ І, АБО, НІ. Крім того, на практиці широко застосовуються базиси, що складаються з одного ЛЕ і реалізують най-

простіші функції декількох змінних АБО-НІ, І-НІ і деякі інші. Ці функції також називають операторами, а запис складної функції у вигляді суперпозиції операторів ЛЕ називають операторною формою функції.

В теорії перемикальних функцій взагалі важливе значення має задача подання довільних функцій у вигляді суперпозицій функцій заданого базису. Для булевих функцій базис найчастіше утворюють три основні логічні операції: диз'юнкція (логічне додавання, операція АБО, скорочені позначення $\vee$ та $+$), кон'юнкція (логічне множення, операція І, скорочені позначення $\&$ та $\cdot$), інверсія (логічне заперечення, операція НІ, скорочене позначення $-$ у вигляді риски над буквою). На рис. 1.6 наведені таблиці істинності функцій цього базису та умовні графічні позначення відповідних ЛЕ.

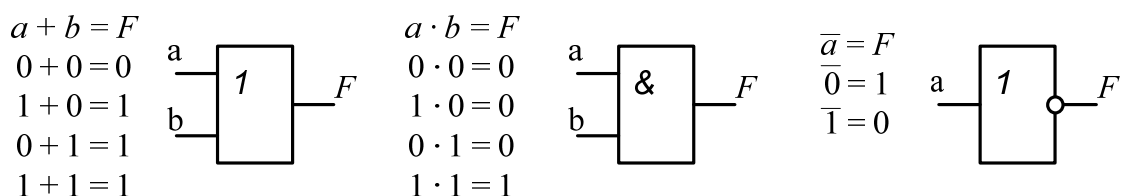


Рисунок 1.6 – Таблиці істинності та умовні позначення ЛЕ типів АБО, І, НІ

Обчислення значень ЛФ проводять за правилами виконання логічних операцій (законами алгебри логіки), які задають тотожні перетворення наборів значень двійкових аргументів $a, b, c \dots$ в значення функцій.

1.6.1 Закони алгебри логіки. Це правила, які дозволяють перетворювати логічні функції, що описують функціонування КС, до виду, зручного для подальшого використання.

1. Переставний закон (закон комутативності) для логічного множення і логічного додавання

$$a \cdot b = b \cdot a; a + b = b + a.$$

2. Сполучний закон (закон асоціативності) для логічного множення і логічного додавання

$$(a \cdot b) \cdot c = a \cdot (b \cdot c); (a + b) + c = a + (b + c).$$

3. Розподільний закон (дистрибутивний закон)

$$\begin{aligned} a + b \cdot c &= (a + b) (a + c); \\ (b + c) \cdot a &= a \cdot b + a \cdot c; \\ a + b \cdot c &= (a \cdot 1 + b \cdot c) = a \cdot (1 + b + c) + b \cdot c = \\ &= a \cdot a + a \cdot b + a \cdot c + b \cdot c = a \cdot (a + b) + c \cdot (a + b) = \\ &= (a + b) \cdot (a + c). \end{aligned}$$

4. Закон поглинання

$$a + a \cdot b = a \cdot (1 + b) = a; a \cdot (a + b) = a \cdot a + a \cdot b = a.$$

5. Закон склеювання

$$(a + b) \cdot (a + \bar{b}) = a; \quad a \cdot b + \bar{a} \cdot b = b.$$

6. Закон де Моргана

$\overline{a + b} = \bar{a} \cdot \bar{b} = a \downarrow b$ – позначення функції в правій частині цієї рівності, називають також «стрілка Пірса» або «функція Вебба»);

$\overline{a \cdot b} = \bar{a} + \bar{b} = a / b$ – запис функції в правій частині цієї рівності називають також «штрих Шеффера»;

7. Закон подвійного заперечення $a = \overline{\bar{a}}$.

8. Закон заперечення констант $\bar{1} = 0, \bar{0} = 1$

9. Закон множення для 1 і 0 $a \cdot 1 = a; a \cdot 0 = 0$;

10. Закон додавання для 1 і 0 $a + 1 = 1; a + 0 = a$;

11. Закон виключення $a + \bar{a} = 1; a \cdot \bar{a} = 0$.

12. Закон ідемпотентності $a \cdot a = a; a + a = a$.

Аналіз законів 1–12 свідчить про властивість двоїстості (дуальності) булевої алгебри щодо операцій диз'юнкції і кон'юнкції f . Двоїстість проявляється в тому, що якщо в деякому виразі для функції f замінити всі знаки операцій І на знаки операцій АБО та всі знаки операцій АБО на знаки операцій І, то тим самим отримаємо інверсію функції f .

Логічну функцію для зручності запису і подальшої її реалізації звичайно виражають у вигляді логічної суми добутоків аргументів або як логічний добуток їх сум. Перший запис називають диз'юнктивною формою (ДФ), другий – кон'юнктивною формою (КФ). Якщо в ДФ кожен логічний добуток утворений від всіх n аргументів (можливо, їх заперечень) логічної функції, то таку ДФ називають нормальною, тобто ДНФ. Аналогічно, якщо в КФ кожна логічна сума утворена від всіх n аргументів (можливо, їх заперечень) логічної функції, то таку КФ називають нормальною, тобто КНФ.

Взагалі для кожної логічної функції може існувати декілька рівносильних диз'юнктивних і кон'юнктивних форм, проте існує тільки одна ДНФ або КНФ, в яких функція може бути записана єдиним способом. Це так звані доконані нормальні форми ДДНФ і ДКНФ. У ДДНФ функція записується у вигляді логічної суми конституент одиниці (мінтермів), а в ДКНФ – у вигляді логічного добутку конституент нуля (макстермів).

Мінтермом Q_i (елементарною кон'юнкцією, конституентою одиниці) називають логічний добуток прямих або інверсних значень аргументів, причому кожна змінна у добутку зустрічається тільки один раз. Тут індекс i відповідає числу, яке є десятковим еквівалентом n -розрядного двійкового числа, складеного зі значень аргументів, при яких відповідний мінтерм дорівнює одиниці. Наприклад, для $n=4$ і аргументів x_1, x_2, x_3, x_4 маємо

$$Q_0 = \overline{x_1 x_2 x_3 x_4}, \quad Q_1 = \overline{x_1 x_2 x_3} x_4, \quad Q_2 = \overline{x_1 x_2} \overline{x_3} x_4, \quad \dots, \quad Q_{15} = x_1 x_2 x_3 x_4.$$

Макстермом D_j (елементарною диз'юнкцією, конституентою нуля) називають логічну суму прямих або інверсних значень аргументів, причому кожна змінна зустрічається у сумі лише один раз. Для макстермів індекс j відповідає десятковому числу, яке є еквівалентом n -розрядного двійкового числа, складеного зі значень аргументів, при яких відповідний макстерм дорівнює нулю. За умов попереднього прикладу маємо

$$D_0 = x_1 + x_2 + x_3 + x_4, \quad D_1 = x_1 + x_2 + x_3 + \overline{x_4}, \quad D_2 = x_1 + x_2 + \overline{x_3} + x_4, \dots, \\ D_{15} = x_1 + x_2 + x_3 + x_4.$$

Кількість мінтермів і макстермів від заданого числа аргументів дорівнює числу різних наборів аргументів $N = 2^n$. Два мінтерми (макстерми), які утворені від одних і тих же аргументів і відрізняються наявністю (або відсутністю) заперечення тільки щодо однієї змінної, називають сусідніми по Квайну або просто сусідніми. В попередніх прикладах є сусідніми пари $(Q_0, Q_1), (Q_0, Q_2), (Q_1, Q_3), (Q_2, Q_3)$ та інші, однак пари $(Q_0, Q_3), (Q_1, Q_2), (Q_0, Q_5)$ і т. п. властивості сусідства не мають.

Основні властивості мінтермів і макстермів полягають у нижчевикладеному.

1. $\overline{Q}_i = D_{N-1-i}, \quad \overline{D}_j = D_{N-1-j}$, де індекси $i, j = 0, 1, \dots, 2^n-1$.

2. Логічна сума всіх мінтермів будь-якого числа аргументів дорівнює одиниці та, за аналогією, логічний добуток всіх макстермів будь-якого числа аргументів дорівнює нулю, тобто,

$$\sum_{i=0}^{N-1} Q_i = 1, \quad \prod_{j=0}^{N-1} D_j = 0.$$

3. Логічний добуток мінтермів, що мають різні індекси, дорівнює нулю, тобто $Q_i \cdot Q_j = 0$, при $i \neq j$.

4. Логічна сума неоднакових макстермів дорівнює одиниці, тобто. $D_i + D_j = 1$, при $i \neq j$.

Для побудови ДДНФ булевої функції F від n змінних, заданої таблицею істинності, необхідно для кожного набору аргументів, де функція приймає значення 1, записати відповідний мінтерм, далі всі такі кон'юнкції потрібно з'єднати знаками диз'юнкції. Для побудови ДКНФ функції F від n змінних, заданої таблицею істинності, необхідно для кожного набору аргументів, де функція приймає значення 0, записати відповідний макстерм, далі всі такі диз'юнкції потрібно з'єднати знаками кон'юнкції.

Для спрощення зображення булевих функцій і зменшення числа логічних елементів, що їх реалізують, застосовують різні методи мінімізації. При цьому під мінімізацією перемикальної функції розуміють процедуру знаходження найпростішого її подання у вигляді суперпозиції операцій базисної системи. Булева функція може бути спрощена безпосередніми алгебраїчними перетвореннями за допомогою законів алгебри логіки (склею-

вання і поглинання). Але, як правило, такі перетворення досить громіздкі та вимагають певних навичок. Для мінімізації функцій великого числа аргументів (не менше трьох) застосовують спеціальні методи. Для функцій 3...5 аргументів найчастіше застосовують методи на основі так званих діаграм Вейча або карт Карно. Діаграми Вейча і карти Карно є спеціальними формами таблиць істинності булевих функцій, які містять 2^n клітинок і побудовані так, що кожна клітина таблиці відповідає окремому мінтерму, причому сусідні в таблиці клітини займають мінтерми, сусідні за Квайном. Кожному рядку (стовпцю) цієї таблиці відповідає певна комбінація значень аргументів, кожна клітина відповідає певному набору значень аргументів так, що при будь-якому переході з однієї клітини в сусідню уздовж рядка або стовпця змінюється значення лише одного аргумента функції. На діаграмах Вейча позначення рядків і стовпців в таблиці виконують прямими та інверсними значеннями аргументів за їх порядковим номером. На картах Карно порядок позначення аргументами рядків і стовпців визначається їх двійковими номерами в рефлексному коді (коді Грея).

1.6.2 Алгоритми мінімізації логічних функцій за допомогою діаграм Вейча

Виконання цих алгоритмів полягає у нижчевикладеному.

1. Логічну функцію потрібно привести до ДДНФ (ДКНФ). Для цього необхідно скористатися законами алгебри логіки та відповідними тотожними співвідношеннями з тим, щоб підсумковий вираз був поданий як диз'юнкція всіх мінтермів, або як кон'юнкція всіх макстермів заданої функції.

2. Заповнити мінтермами (макстермами) прямокутну таблицю, кількість клітин якої дорівнює 2^n . Мінтерми позначають 1 у відповідних клітинах таблиці, а макстерми – 0. Два мінтерми (макстерми), що розміщені в сусідніх клітинах, можна замінити одним логічним добутком (сумою) з меншим на одиницю числом змінних на підставі законів дистрибутивності, склеювання, поглинання і додавання (множення) з 1 і 0. У загальному випадку наявність одиниць (нулів) в 2^k сусідніх клітинах дозволяє вилучити k змінних.

3. У заповненій таблиці обводять прямокутними контурами (покриттями) всі сусідні одиниці (нулі). При цьому потрібно дотримуватися таких правил: контур має бути прямокутним; кількість клітин всередині контуру має бути цілим степенем числа 2, тобто 1, 2, 4, 8 і т. д.; всередині контуру мають бути тільки клітини, позначені 1 (0) та символами невизначеності X; одні й ті ж клітини, заповнені 1 (0) та X, можуть входити в декілька контурів; нижній і верхній рядки таблиці, а також лівий і правий стовпці вважаються сусідніми; контурів має бути якомога менше, а самі контури – якомога більшими.

4. Вираз для мінімальної ДНФ (МДНФ) функції записують таким чином. Кожен контур на діаграмі Вейча ДНФ відповідає диз'юнктивному

(кон'юнктивному) члену, число змінних в якому на k менше загального n – числа аргументів функції, тобто дорівнює $n - k$. Кожен член МДНФ складається лише з тих аргументів (без інверсії або з інверсією), які для відповідного контуру є загальними. У загальному випадку функція може мати кілька мінімальних форм, яким відповідають різні, але рівноцінні за кількістю членів, МДНФ.

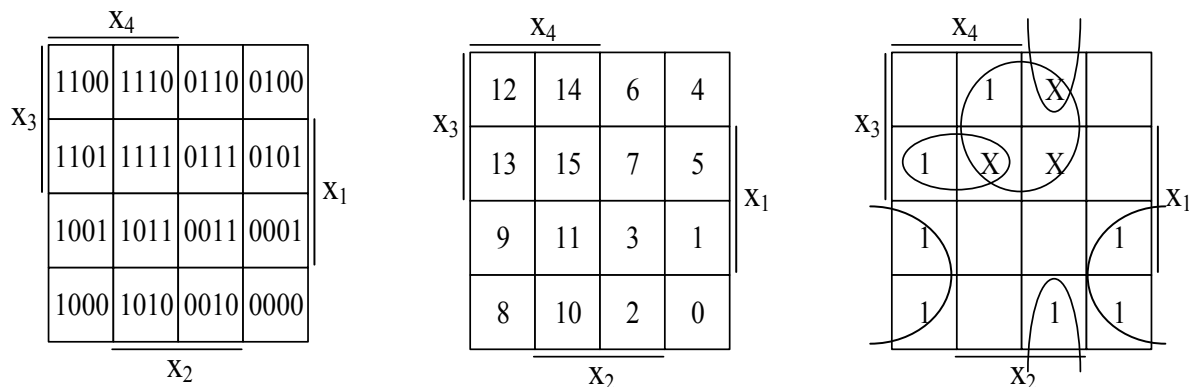


Рисунок 1.7 – Діаграми Вейча до прикладу

Нехай, наприклад, задана неповністю визначена функція чотирьох аргументів $F = 1110\ 00XX\ 1100\ 011X$. Запис цієї функції в ДДНФ має вигляд $F = 0000 + 0001 + 0010 + 1000 + 1001 + 1101 + 1110$. Нанесемо ці мінтерми на карту Вейча, довизначимо функцію одиницями на невизначених наборах аргументів і зробимо покриття сусідніх мінтермів (рис. 1.7). Запишемо результат покриття у вигляді диз'юнкції кон'юнкцій (суми добутків)

$$F = x_3x_2 + \overline{x_3}\overline{x_2} + x_4x_3x_1 + \overline{x_4}\overline{x_2}\overline{x_1}.$$

Функція у вигляді ДДНФ до мінімізації мала 7 мінтермів із чотирьох букв кожен. Після ж мінімізації – 2 кон'юнкції з двох букв та 2 кон'юнкції з трьох букв.

1.7 Контрольні запитання і завдання

1. Чим займається «Комп'ютерна схемотехніка»?
2. Що таке сигнал у комп'ютерній техніці?
3. Які сигнали використовують в комп'ютерній схемотехніці?
4. Чим відрізняється аналоговий сигнал від цифрового?
5. Які цифрові пристрої називають комбінаційними?
6. Які цифрові пристрої називають скінченними автоматами?
7. Побудувати узагальнену структурну схему скінченного автомата.
8. В чому полягає схемотехнічне призначення алгебри логіки?

9. Назвати основні закони алгебри логіки.
10. Що таке перемикальні та логічні функції?
11. Які закони алгебри логіки використовують для приведення логічних функцій до заданого базису?
12. Що таке базис ЛЕ?
13. Для чого виконують мінімізацію логічних функцій?
14. Сформулювати і показати на прикладах різницю між мінтермами і макстермами.
15. Показати на прикладах застосування ДДНФ і ДКНФ.

2 ОСНОВИ СХЕМОТЕХНІКИ НАПІВПРОВІДНИКОВИХ ЛОГІЧНИХ ЕЛЕМЕНТІВ ТА ЇХ КОМПОНЕНТІВ

Перелік скорочень і умовних позначень до розділу 2

БЕТ – багатоемітерний транзистор; ДТЛ – діодно-транзисторна логіка; ЕП – емітерний перехід; ЗБ – загальна база; ЗЕ – загальний емітер; ЗК – загальний колектор; МОН – метал-оксид-напівпровідник; РТЛ – резисторно-транзисторна логіка; ТТЛ – транзисторно-транзисторна логіка; ТТЛШ – ТТЛ Шотткі.

2.1 Базові елементи цифрових пристроїв

Найпростішими електронними компонентами комп'ютерних цифрових пристроїв є діодні і транзисторні ключі (рис. 2.1 та 2.2). Будемо вважати ключ замкненим, якщо існує прямий гальванічний зв'язок від його входу до виходу, та ключ вимкнений (розімкнений), якщо такого зв'язку не існує.

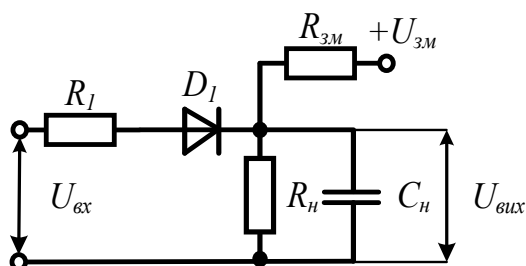


Рисунок 2.1 – Діодний ключ

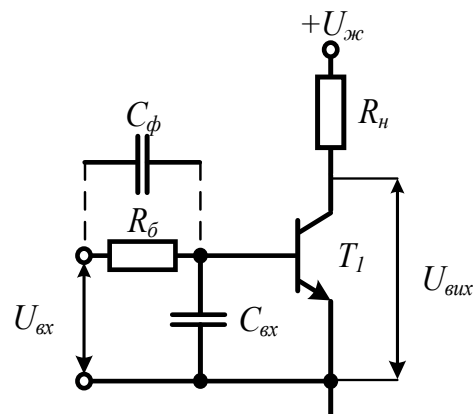
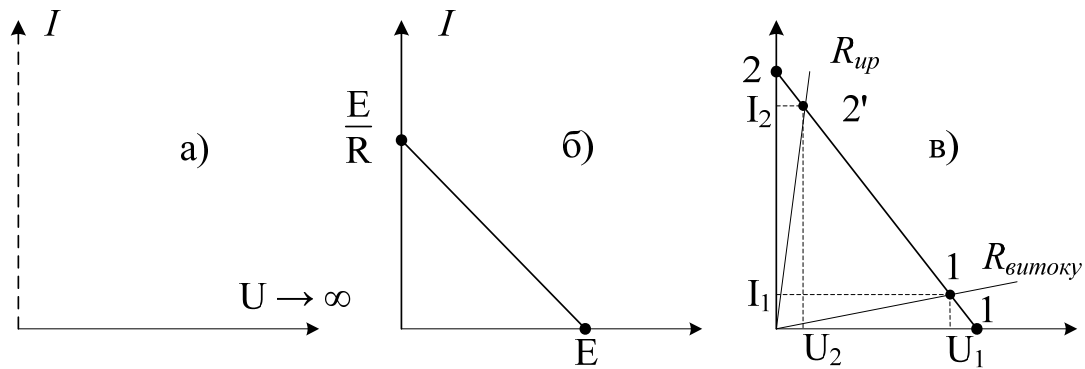


Рисунок 2.2 – Схема транзисторного
ключа

В багатьох випадках практики розрахунків ключів використовують поняття ідеального ключа (рис. 2.3). Замкнений ідеальний ключ визначається його параметрами $R = 0$, $I = ?$, $U_{\text{кл}} = 0$, а вимкнений ідеальний ключ – $R = ?$, $I = 0$, $U_{\text{кл}} = ?$ Для реальних ключів $U_{\text{кл}} = E$, $I = E / R$, $R = R_i$. Для кремнієвих діодів напруга їх вимкнення (розімкнення, відмикання) U (рис. 2.1) становить (0,4...0,6) В, для германієвих – (0,2...0,3) В, а падіння напруги на одному діоді при прямому його включенні складає відповідно $U_{\text{пряме}} = (0,6...0,8)$ В і (0,2...0,4). Тому для забезпечення необхідного рівня заводостійкості діодних ключів використовують додаткове зміщення.



$$U_2 - U_1 = U_m; I_2 - I_1 = I_m$$

Рисунок 2.3 – Вольт-амперні характеристики ідеального а), реального б), діодного в) транзисторних ключів

Для реалізації логічних функцій багатьох змінних найчастіше застосовують кілька паралельно ввімкнених діодних ключів.

В транзисторних ключах на біполярних транзисторах використовують всі основні способи під'єднання транзисторів, а саме: за схемою з загальною базою (ЗБ), загальним колектором (ЗК), загальним емітером (ЗЕ) і ключ-зірку. Однак найчастіше застосовують ключі на основі схеми загальний емітер транзисторів (див. рис. 2.2). Тут транзистор може перебувати в режимі «відсічки» (при цьому $U_k \approx E_k$, $I_k \approx I_{k0}$, а транзистор, насправді, перетворюється на генератор струму I_{k0} причому $I_s \approx I_{k0}$) і в режимі «насичення» ($U_k \approx 0$, $I_k \approx I_{kmax}$, а транзистор перетворюється на еквіпотенціальну точку, що має єдиний потенціал для всіх електродів).

Під впливом сигналу управління ($U_{вх}$) транзистор перемикається не миттєво, а за час, що визначається перехідним процесом, який, в свою чергу, обумовлений дією паразитних ємностей, ємностей $p-n$ переходів і насиченням транзистора. Основні параметри цього перехідного процесу: $t_{зад}$ – час перезарядження $C_{вх}$ від напруги $U_{Бзап}$ до $U_{пор}$; t_{01} – тривалість процесу зміни концентрації носіїв в базі і перезарядження бар'єрної ємності колекторного переходу; $t_{роз}$ – час розсмоктування надлишкових носіїв у базі (поки струм бази не зменшиться до рівня I_{β} насичення); t_{10} – час, пов'язаний зі швидкістю зменшення заряду в базі («інерційністю» транзистора).

Для прискорення перехідних процесів і збільшення швидкодії транзисторних ключів на біполярних транзисторах застосовують форсувальні конденсатори (резистор шунтують паралельним під'єднанням конденсатора C_{Φ} , див. рис. 2.2) і ненасичені ключі. З цією метою фіксують колекторну напругу на рівні E та реалізують нелінійний (діодний) зворотний зв'язок колектор – база.

Поряд з ключами на біполярних транзисторах в комп'ютерній схемотехніці широке застосування знаходять і транзисторні ключі на уніполярних транзисторах (МОН-структурах, розділ 3).

2.2 Резисторно-транзисторна логіка

Схема типового ЛЕ резисторно-транзисторної логіки (РТЛ), (англ. – «Resistor-transistor logic» -RTL) показана на рис. 2.4. Колектор транзистора (ключа) з'єднаний через резистор з джерелом живлення (як правило, +3,15 В), а емітер – із корпусом. До бази під'єднано резистори, які є входами ЛЕ. Цей ЛЕ функціонує таким чином: за відсутності напруги на всіх входах транзистор закритий і на вихід через резистор надходить сигнал, близький до напруги живлення, тобто маємо логічну 1 на виході при нулях на вході (для так званої «позитивної логіки», коли одиниці відповідає високий рівень сигналу, а нулю – низький).

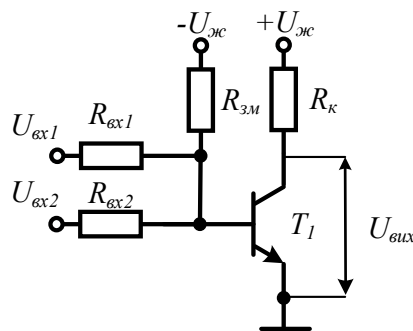


Рисунок 2.4 – Схема резисторно-транзисторного ЛЕ

За наявності сигналу високого рівня хоча б на одному з входів ключ відкривається і замикає вихід на корпус. При цьому він знижує рівень вихідного сигналу майже до нуля. Таким чином, РТЛ-елемент виконує функцію АБО-НІ для позитивної логіки або І-НІ для негативної (одиниці відповідає низький рівень сигналу, а нулю – високий).

Перевагами елементів РТЛ є конструктивна простота і, як наслідок цього, невисока вартість. Недоліки: велика потужність розсіювання (як на замкненому ключі, так і на резисторах); нечіткий рівень сигналів (рівень одиниці – від приблизно 0,9 В до напруги живлення); вкрай низька швидкодія; низька завадостійкість; низька навантажувальна здатність виходів (звичайно не більше трьох входів інших елементів). Після освоєння промисловістю виробництва ІМС високого СІ елементи РТЛ практично зникли з комп'ютерної схемотехніки і застосовуються лише в окремих спеціальних випадках.

2.3 Діодно-транзисторні логічні елементи (ДТЛ)

Діодно-транзисторна логіка (ДТЛ), (англ. DTL – diode–transistor logic) – технологія побудови цифрових інтегральних мікросхем (ЦІМС пристроїв), коли логічні функції реалізують діодні ключі, а для відновлення, підсилення та інверсії сигналів використовують ключі на біполярних транзисторах.

На рис. 2.5 показана схема елемента 2І-НІ типу ДТЛ. Цей елемент працює так: якщо хоча б на один вхід подано рівень логічного нуля, то струм протікає через R_1 і діод у входному колі. В цей час на анодах діодів напруга біля 0,7 В, якої недостатньо для відкривання транзистора. Отже, на виході формується рівень логічної одиниці. Якщо ж на всі входи надходить рівень логічної одиниці, струм протікає через R_1 в базу транзистора, зумовлюючи на анодах діодів напругу біля 1,4 В. Оскільки рівень логічної одиниці більше цієї величини, входи діодів зворотно зміщені і не беруть участь в роботі елемента. Транзистор відкритий і знаходиться в режимі насичення. Через транзистор протікає струм навантаження, за величиною значно більший струму навантаження в стані логічної одиниці.

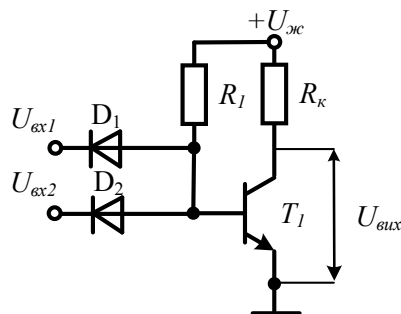


Рисунок 2.5 – Принципова схема базового ДТЛ-елемента 2І-НІ

Основна перевага ДТЛ, порівняно з більш ранньою технологією РТЛ, – можливість створення великого числа входів. Затримка сигналів в ДТЛ-елементах досить велика внаслідок повільного процесу витікання заряду з бази в режимі насичення (коли всі входи мають високий рівень) при подачі на один з входів сигналу низького рівня. Цю затримку можна зменшити підключенням бази транзистора через резистор до шини нульового потенціалу або до джерела негативної напруги. У табл.2.1 наведені найважливіші параметри елементів такого типу.

Таблиця 2.1 – Типові характеристики елементів ДТЛ

Напруга живлення	6 В
Споживана потужність	9 мВт
Затримка сигналів	30 нс
Статична завадостійкість	1,2 В
Вхідна/вихідна напруга високого рівня	3,6 В / 1,4 В
Вхідна/вихідна напруга низького рівня	4 В / 0,5 В

2.4 Транзисторно-транзисторні логічні елементи

Їх особливістю є використання на вході ЛЕ багатоемітерних транзисторів (БЕТ), які реалізують необхідну ЛФ (АБО, І). БЕТ є еквівалентом кількох транзисторів (рис. 2.6, а). Порівняно з ДТЛ в ТТЛ проблема прискорення витікання заряду з бази в режимі насичення вирішена шляхом заміни діодів на багатоемітерні транзистори. Це також зменшує площу кристала

(в разі реалізації ЛЕ методами інтегральної технології) і дозволяє досягти більш високої щільності розміщення елементів в ІМС.

Особливістю БЕТ (рис. 2.6) є те, що безпосередня взаємодія емітерів через бази відсутня. ЛЕ типу ТТЛ можуть функціонувати як в позитивній логіці (реалізується «І»), так і у негативній (реалізується АБО). На рис. 2.6, б показано схему ТТЛ-елемента на базі БЕТ з простим інвертором (І-НІ). Порівняно з ДТЛ тут роль діодів виконують емітерні переходи, а роль діода зміщення – перехід «база–колектор». Хоча це формальний поділ ролей, оскільки емітерний і колекторний переходи в БЕТ не ізольовані. При подачі хоча б на один із входів ($x_1 = 0$) струм через R замикається через перший емітерний перехід (ЕП₁) БЕТ.

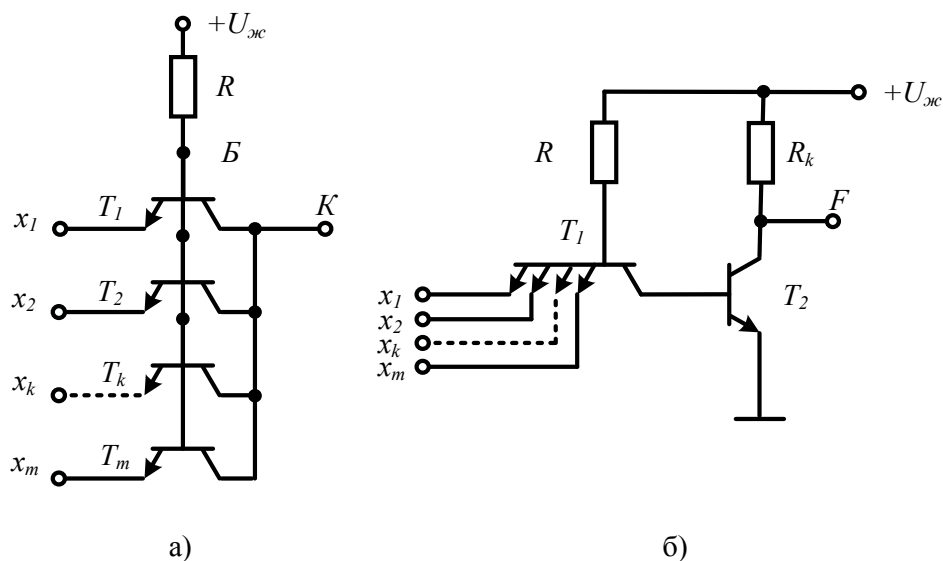


Рисунок 2.6 – ТТЛ-логіка:

а) Еквівалентна схема БЕТ; б) ЛЕ І-НІ з простим інвертором

БЕТ відкритий і працює в інверсному режимі, в базу T_1 струм БЕТ не відгалужується, транзистор T_2 закритий. І лише за наявності на всіх входах «1» ($x_1 = x_2 = \dots = x_m = 1$) усі ЕП₁... ЕП_m БЕТ будуть закриті та буде відкритий T_2 ($U_{kvt} \approx 0$, $F = 0$). Таким чином, ЛЕ виконує операцію І-НІ, $F = x_1 \cdot x_2 \dots x_m$.

Для підвищення навантажувальної здатності ЛЕ і швидкого передзарядження навантажувальної ємності до елемента додають складний вихідний каскад (рис. 2.7). При цьому коло емітера транзистора T_2 з'єднують з резистором R_2 , в результаті утворюється каскад з двома протифазними виходами (приросту напруги на емітері протифазно відповідає зміна напруги на колекторі T_2). Ці виходи забезпечують потрібні для транзисторів T_3 і T_4 сигнали управління. Вихідний каскад не змінює логічної операції, виконуваної ЛЕ, і забезпечує малий вихідний опір елемента для будь-якого його стану. При нульовому стані виходу насичений транзистор T_4 ; при форму-

ванні на виході високого рівня логічної одиниці транзистор T_4 замкнений, а транзистор T_3 вмикається за схемою з ЗК (працює за схемою емітерного повторювача). Таким чином, при обох станах виходу ЛЕ йому забезпечується низький вихідний опір. Діод D введений в схему ЛЕ для узгодження потенціалів при замиканні транзистора T_3 .

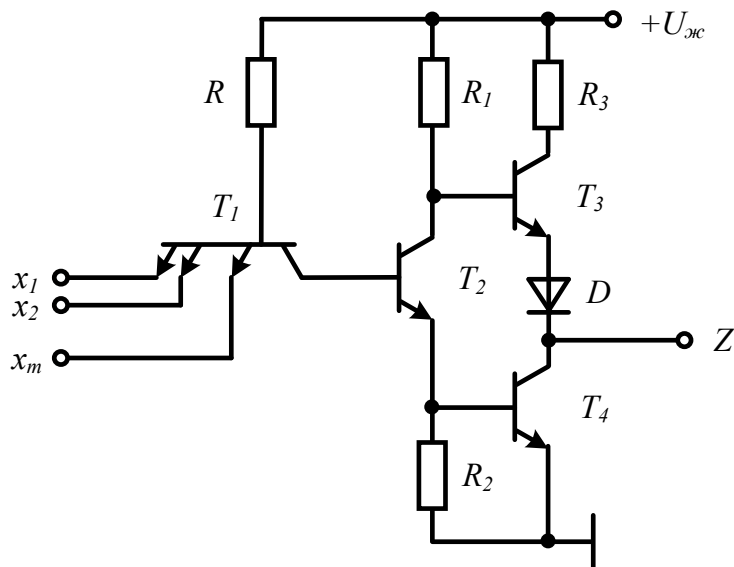


Рисунок 2.7 – Схема ТТЛ-елемента зі складним інвертором

В сучасних швидкодіючих ТТЛ-мікросхемах (серії 74S, 74LS, 74AS, 74ALS, 74F) з діодами Шотткі (ТТЛШ, ТТЛ-Шотткі) фактично сталося повернення до ДТЛ на основі нових схемно-технологічних компонентів – діодів і транзисторів Шотткі. Ці серії не містять БЕ і фактично є ДТЛ. Вони носять назву ТТЛ (ТТЛШ) лише «за традицією», оскільки є подальшим розвитком саме ДТЛ.

2.4.1 Логічні елементи ТТЛ із трьома станами виходу. Такі елементи з'явилися в процесі розвитку методів конструювання цифрових апаратних засобів. Їм, на відміну від інших вже розглянутих типів ЛЕ, властивий третій стан виходу, при якому транзистори (VT) замкнені сигналом з виводу управління. Тому вихідний опір ЛЕ дуже великий, внаслідок чого ЛЕ повністю відмикається від навантаження. Цей стан називається високоімпедансним.

Крім того, розглянуті вище базові ТТЛ-елементи характеризуються досить значним споживанням енергії. Цей недолік також суттєво мінімізується в ЛЕ з трьома станами виходу. Тут (рис. 2.8) в складному інверторі діод D немає, замість нього зсув потенціалу виконує перехід база – емітер транзистора, який вмикають на вході транзистора T_3 і який, до того ж, здійснює додаткове посилення струму. Це збільшує навантажувальну здатність і швидкодію ЛЕ.

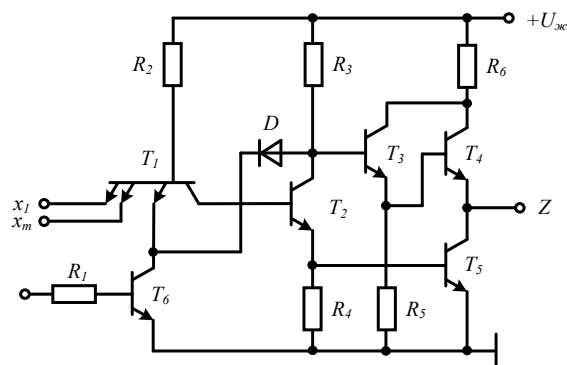


Рисунок 2.8 – Схема елемента ТТЛ із трьома станами виходу

Для забезпечення трьох станів виходу і управління ними тут використовують один з входів БЕТ і діод D . Якщо на вхід управління подати рівень логічної одиниці, то транзистор T_6 насичується, один з входів БЕТ отримає нульовий сигнал, що приводить до замикання транзистора T_2 . Через це замкненим виявиться і транзистор T_5 . При цьому напруга на колекторі T_2 зростатиме, але не досягне номінального рівня, оскільки ввімкнеться діод D , який обмежить напругу на колекторі транзистора T_2 на рівні $U_0 + U_D \approx 1$ В. Така напруга не може відкрити транзистори T_3 і T_4 . Таким чином, транзистори вихідного каскаду вияляться замкненими, струм вихідного кола буде нехтовно малим, а елемент в цілому знаходитиметься в стані «вимкнено». При подачі на вхід управління логічного нуля транзистор T_6 буде замкнений і це забезпечує звичайний режим роботи елемента І-НІ на інформаційних входах $x_1, \dots, x_m$.

В цифрових пристроях, побудованих на таких елементах, управління ІМС організовують так, щоб в будь-який момент часу всі ІМС, крім однієї, знаходились у високоімпедансному стані. Завдяки цьому відбувається «відключення» ІМС від їх навантажень, вдається передавати по одному провіднику в різних напрямках інформацію від декількох джерел і, як результат, скоротити число інформаційних магістралей. Вхід вмикання третього стану має спеціальну мітку EZ, а високоімпедансний вихід позначають Z. Управління EZ може бути як прямим, так і інверсним (рис. 2.9).

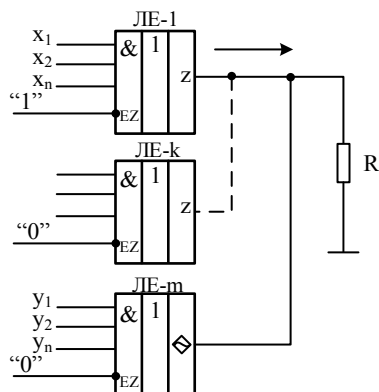


Рисунок 2.9 – Схема ввімкнення ЛЕ з трьома станами виходу

2.4.2 Логічні елементи ТТЛ із відкритим колектором. Мають прямий вивід колектора вихідного транзистора. Хоча базові елементи ТТЛ характеризуються порівняно високою швидкістю, мають малі вхідні і великі вихідні струми, добре узгоджуються з ємнісним навантаженням, однак мають і суттєвий недолік, який полягає в короткочасному зростанні споживаної потужності протягом перехідних процесів при одночасному перемиканні великої кількості ЛЕ. Стрибки струму в колах живлення при цьому досягають кількох одиниць і навіть десятків ампер. Частково цей недолік усувають шляхом установлення в колах живлення окремих конденсаторів великої ємності, які компенсують короткочасні стрибки струму і зменшують взаємовплив ЛЕ у колах живлення.

Крім великих струмів при одночасному перемиканні ТТЛ-елементи мають невеликі значення вихідного опору, що ускладнює об'єднання між собою виходів декількох ЛЕ, оскільки у разі різних вихідних сигналів через вихідні транзистори протікають великі струми. Тому у деяких ЛЕ на виході встановлюють транзистор ТЗ, в колекторному колі якого відсутній резистор (рис. 2.10). Такі ЛЕ і називають елементами з відкритим колектором.

ЛЕ з відкритим колектором (рис. 2.10, а) можуть навантажуватися на нетипові входи (реле, дросель, обмотка трансформатора, індикаторна лампа, що живиться від $U_{\text{ип}}$ та ін.) (рис. 2.10, б) і вмикатися паралельно на виходах (рис. 2.10, в). Паралельне вмикання виходів з відкритим колектором показано на рис. 2.10, в. Стандартні елементи ТТЛ зі складними вихідними інверторами не можна з'єднувати паралельно на виходах, оскільки при різних логічних станах за малих вихідних опорів елементів у вихідних колах розвинулись би неприпустимо великі струми, а логічний стан загального виходу в цілому був би невизначеним.

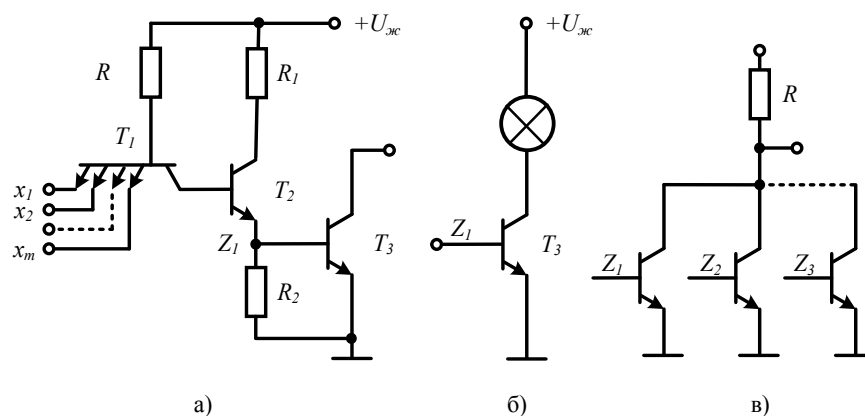


Рисунок 2.10 – Схема елемента ТТЛ з: відкритим колектором а), управлінням нестандартним навантаженням б), паралельним з'єднанням виходів в)

Елементи з відкритим колектором допускають паралельне з'єднання виходів із загальним навантажувальним резистором. У даному випадку

(див. рис. 2.10, в) щодо сигналів на базах вихідних транзисторів виконується операція $Z = \overline{Z_1 + Z_2 + Z_3}$, а елементи з відкритим колектором можуть використовуватися і для почергової роботи на загальну лінію в режимі поділу часу. З цією метою розроблені спеціальні елементи з трьома станами виходу: «1», «0» і «вимкнено» (від'єднання від навантаження), що забезпечує більш високу швидкодію.

У стані «вимкнено» вихід елемента не споживає і не віддає струм у навантаження. Якщо в кожен момент часу лише один з під'єднаних до об'єднувального кола елементів активний, а всі інші відімкнені, то тим самим забезпечується нормальний режим поділу часу.

При використанні такого ЛЕ відкритий колектор (рис. 2.11, а) з'єднується з джерелом живлення через навантажувальний опір R_{ce} (рис. 2.11, б) або через обмотку реле (рис. 2.11, в), світлодіод, лампу розжарювання (рис. 2.11, б) і т. д.

Відкритий колектор дозволяє безпосередньо під'єднувати кілька ЛЕ і утворювати додаткову логічну функцію – монтажне І.

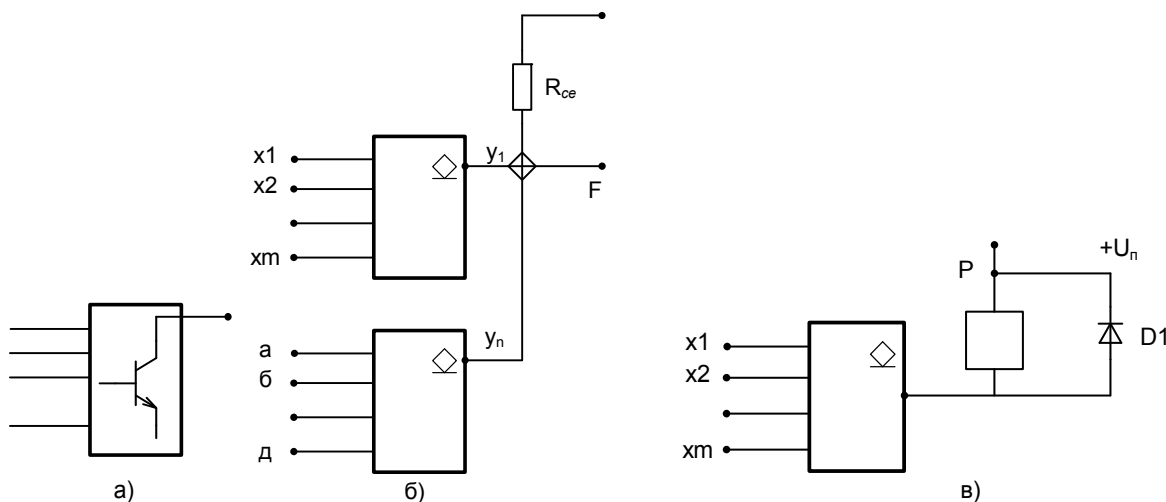


Рисунок 2.11 – Схеми вмикання ТТЛ-елементів з відкритим колектором

2.4.3 Логічні елементи ТТЛ із розширювачами. Застосовують для реалізації ЛФ від багатьох аргументів. В процесі проектування сучасних цифрових пристроїв на базі ТТЛ-елементів часто виникає необхідність збільшення числа входів ЛЕ. Зазвичай це досягається за допомогою елементів-розширювачів, причому елемент, число входів якого потрібно розширити, також повинен мати відповідні можливості. Основою такого елемента звичайно є елемент І-АБО-НІ (рис. 2.12). У ньому замість транзистора T_2 основного елемента І-НІ ввімкнені паралельно два транзистори (T_2 і T_3), якими керують два БЕТ (T_1 і T_4). Тут щодо сигналів Z' і Z'' реалізується операція АБО-НІ, а елемент в цілому реалізує операцію

$$Z = \overline{Z' + Z''} = \overline{X_1 X_2 \dots X_m + X_1^1 X_2^1 \dots X_n^1}.$$

Входи ж 1 і 2 призначені для під'єднання розширювача (рис. 2.12, б), який під'єднують до виходів 1 і 2 паралельно транзистам T_2 і T_3 (рис. 2.12, а) елемента І-АБО-НІ. Група входів, що належать розширювачу, формує їх кон'юнкцію, яка додається до логічної суми кон'юнкцій, що їх формує елемент без розширювача. Такий спосіб збільшення числа входів ЛЕ називають розширенням за АБО. Оскільки розширювач під'єднують до внутрішніх точок елемента, що мають підвищену чутливість до завад, з'єднувальні провідники роблять мінімальної довжини. У деяких серіях ТТЛ передбачають спеціальні елементи для виконання операцій І, АБО та ін. Елемент І містить додатковий транзистор-інвертор перед вихідним каскадом, елемент АБО подібний до елемента І-АБО-НІ, в якому замість БЕТ ввімкнені звичайні транзистори.

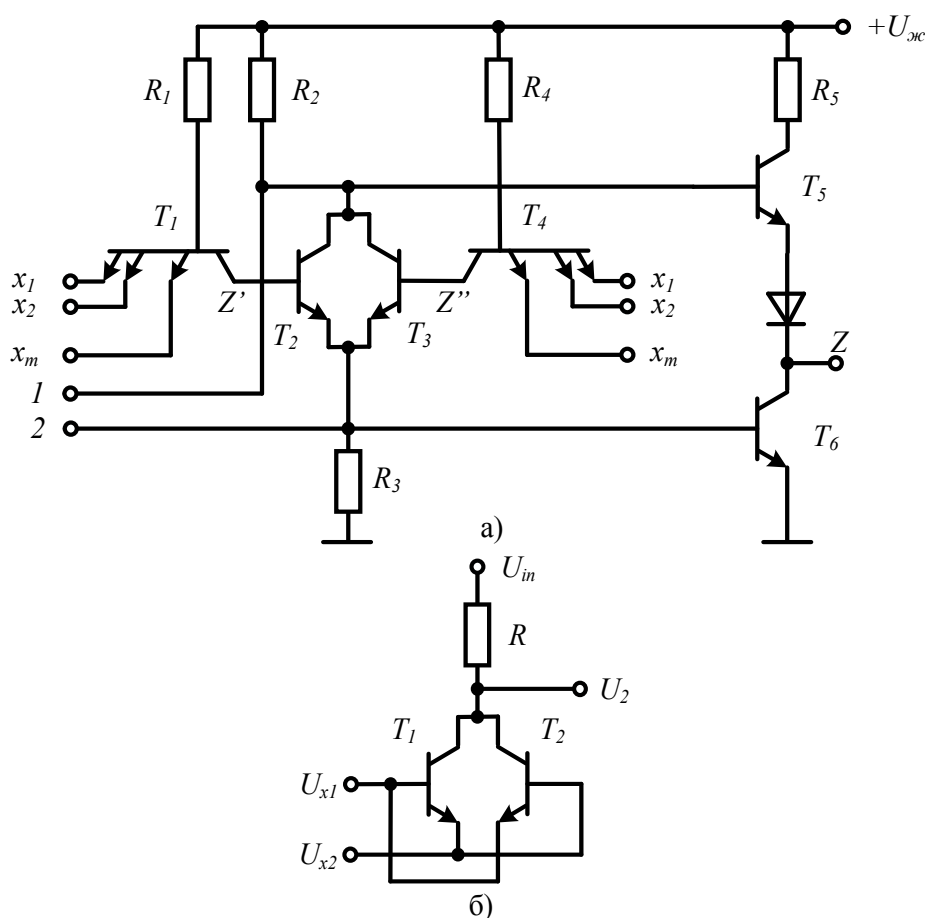


Рисунок 2.12 – Схеми: а) основного елемента І-АБО-НІ типу ТТЛ з можливістю розширення входньої логіки; б) розширювача за АБО

2.5 Логічні елементи ТТЛ на транзисторах Шотткі (ТТЛШ)

Ці ЛЕ, на відміну від елементів ТТЛ, використовують транзистори, в яких паралельно колекторному переходу ввімкнений діод Шотткі. Такі транзистори і отримали назву транзисторів Шотткі. Шунтування колекто-

рного переходу діодом Шотткі унеможливилює появу режиму насичення відкритого транзистора і, тим самим, прискорює його замикання. Застосування транзисторів Шотткі підвищує швидкодію елементів і одночасно покращує електричні параметри режиму перемикання. Оскільки серій мікросхем ТТЛ-елементів існує велика кількість, потрібно навести параметри їх основних різновидів (табл. 2).

Таблиця 2 – Параметри мікросхем ТТЛ-типу

Група мікросхем	ТТЛ	ТТЛ зі зниженим енергоспоживанням	Швидкодіючі ТТЛ	ТТЛШ	ТТЛШ зі зниженим енергоспоживанням	Покращені ТТЛШ
Номер серії	7400	74L00	74H00	74S00	74LS00	74AS00
Напруга живлення, В	5					
Енергоспоживання, мВт	10	1	23	20	2	8
Затримка сигналів, нс	10	33	5	3	9,5	1,7
Максимальна частота перемикання, МГц	40	13	80	130	50	230
Різниця між рівнями 1 і 0, В	1	1	1	0,5	0,6	0,4

В типовому елементі ТТЛШ (рис. 2.13) БЕ також є транзистором Шотткі, але його наявність тут пояснюється іншими міркуваннями, а саме: шунтування колекторного переходу транзистора діодом Шотткі робить дуже малим ефективне значення інверсного коефіцієнта передачі струму, що сприяє кращому функціонуванню елементів ТТЛ.

Особливістю транзисторів Шотткі, пов'язаною з відсутністю у них режиму насичення, є підвищене значення напруги на відкритому транзисторі (близько 0,3 В). Крім описаних вище типових елементів ТТЛ в конкретних серіях мікросхем можуть бути і поліпшені їх модифікації. Завдяки усуненню насичення транзисторів в елементах ТТЛШ їх швидкодія досить висока, і за цим показником елементи ТТЛШ займають серед елементів, реалізованих на основі кремнію, друге місце (після елементів типу ЕЗЛ, далі). Для всіх елементів ТТЛ напруга живлення становить 5 В, а рівні сигналів логічного нуля і одиниці відповідно $U^0 \leq 0,4В$; $U^1 \geq 2,4В$. Для елементів ТТЛШ $U^0 \leq 0,5В$; $U^1 \geq 2,7В$. Типові коефіцієнти розгалуження (на виході і об'єднання (на вході) дорівнюють 10 і 8 відповідно.

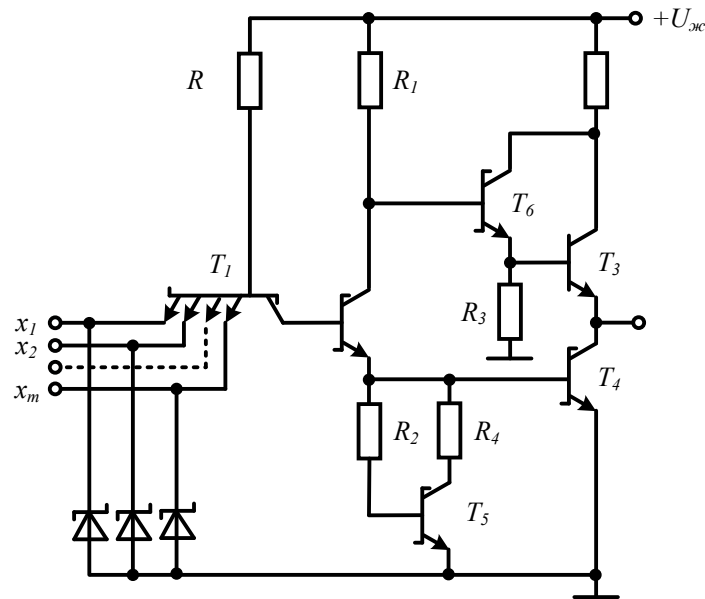


Рисунок 2.13 – Схема типового елемента ТТЛШ

2.6 Контрольні запитання і завдання

1. Що таке логічний елемент?
2. Що означають аббревіатури РТЛ, ДТЛ, ТТЛ, ТТЛШ?
3. Для чого використовуються ЛЕ з трьома станами виходу?
4. Назвіть можливі стани ЛЕ з трьома станами виходу.
5. Яку логічну функцію виконує базовий ЛЕ ТТЛ-типу?
6. Для чого використовуються елементи ТТЛ з відкритим колектором?
7. Для чого використовуються ЛЕ елементи з розширювачами?
8. Яку логічну функцію виконують ЛЕ ТТЛ з розширювачами?
9. Чим відрізняються ЛЕ типів ТТЛ і ТТЛШ?
10. Поясніть принцип роботи ЛЕ типу ТТЛШ.
11. Чим відрізняються ЛЕ ТТЛШ-типу від ЛЕ ТТЛ-типу?
12. Чим зумовлено велике різноманіття серійних ЛЕ ТТЛ-логіки?
13. Вкажіть режими роботи БЕТ в ТТЛ і в ТТЛШ застосуваннях.
14. Побудуйте схему типових ТТЛ і ТТЛШ.
15. Сформулюйте принципові відмінності в роботі базових елементів ТТЛ- і ТТЛШ-типів.
16. В чому полягають особливості структури багатоемітерного транзистора в елементах ТТЛ- і ТТЛШ-типу?
20. Вкажіть основні статичні та динамічні характеристики елементів ТТЛ- і ТТЛШ-типу.
21. Назвіть варіанти модифікацій елементів ТТЛ- і ТТЛШ-типу та поясніть мету, для досягнення якої вони були розроблені.

3 ЛОГІЧНІ ЕЛЕМЕНТИ НА БІПОЛЯРНИХ І ПОЛЬОВИХ ТРАНЗИСТОРАХ

Перелік скорочень і умовних позначень до розділу 3

ЕЗЛ – емітерно-зв’язана логіка; ЕП – емітерний повторювач; І²Л – інтегральна інжекційна логіка.

3.1 Емітерно-зв’язані логічні елементи (ЕЗЛ)

Елементи ЕЗЛ – найшвидкодійні промислово освоєні елементи, що виготовляються на основі кремнію. В елементах ЕЗЛ (ECL англ. Emitter-coupled logic) використовують перемикач струму (рис. 3.1, а), в якому приблизно постійний струм I_e під впливом вхідного сигналу X перемикається з одного «плеча» кола на інше.

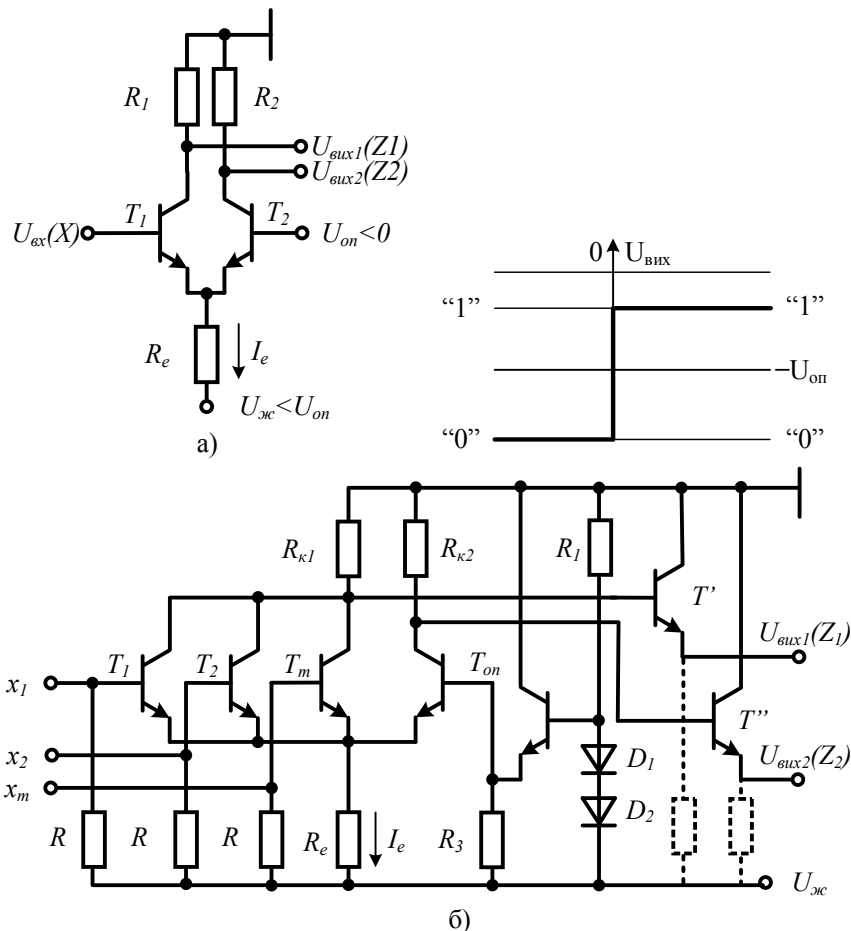


Рисунок 3.1 – Схема перемикача струму а) і базового елемента ЕЗЛ б)

Транзистор правого плеча T_2 має на базі постійну опорну напругу $U_{\text{он}}$, а напруга на базі транзистора T_1 вища опорної при $x = 1$ і нижча за опорну, якщо $x = 0$.

Таким чином, рівні сигналів, що відображають логічну одиницю і логічний нуль, знакозмінні і розташовані симетрично щодо опорної напруги.

Якщо вважати, що напруги на базах обох транзисторів однакові, то перемикач струму виявиться симетричним як за схемною конфігурацією, так і за режимами роботи обох транзисторів, отже, струм I_e розділиться між «плечима» T_1 і T_2 (електричні параметри схемних компонентів вважаємо ідентичними). Якщо далі вважати, що рівень вхідного сигналу дещо збільшився, то це вплине на розподіл струмів між транзисторами. Природно, що транзистор T_1 відкриється сильніше і струм через нього збільшиться. Щодо спільної точки емітерів транзистор T_1 працює як ЕП, і на емітері виникне додатний приріст напруги, яка дещо зменшить струм транзистора T_2 . Таким чином, завдяки зв'язку через спільну точку емітерів здійснюється взаємний вплив транзисторів, а саме: збільшення струму через один з них веде до зменшення струму через інший. Для практично повного перетікання струму з одного «плеча» на інше досить перепаду вхідного сигналу в 350...400 мВ. Для створення необхідного запасу завадостійкості та компенсації впливу змін температури перепад вхідної напруги трохи збільшують, але і в цьому випадку він залишається в межах часток вольт.

Швидкодія перемикача струму досить висока, оскільки транзистори функціонують в умовах, що дозволяють максимально використовувати їх частотні можливості: відсутній режим насичення; перепади напруг при зміні електричного стану транзисторів невеликі, тому можуть бути невеликими і резистори, з яких знімають вихідні сигнали, що зменшує сталі часу електричних кіл перезарядження навантажувальних ємностей.

Таким чином, перемикач струму виконує операції $Z_1 = \bar{x}$, $Z_2 = x$, тобто, його два виходи є взаємно інверсними. При цьому потрібно враховувати, що вищий рівень потенціального вихідного сигналу дорівнює потенціалу спільної точки, що знімається з колектора замкненого транзистора і відповідає логічній одиниці, а низький рівень – з відкритого транзистора і відповідає логічному нулю. За абсолютними значеннями рівні вихідних сигналів відрізняються зміщенням у бік «плюса» (без цього не можна забезпечити ненасичений режим роботи транзисторів). Однак безпосереднє каскадування розглянутих перемикачів струму тут неможливе, оскільки вхідний сигнал наступного за початковим струмового ключа вже не буде симетричним щодо опорної напруги. Для забезпечення можливості каскадування в колі передачі сигналу з виходу одного елемента на вхід іншого слід під'єднати додаткові елементи, що зміщують рівні сигналів. У такому призначенні найчастіше використовують ЕП, напруга на виході яких повторює вхідну, але зі зміщення щодо останньої на відносно постійну напругу база–емітер U_{be} (близько 0,7 В). Поряд з цим ЕП забезпечують малі вихідні опори, що обумовлює швидке перезарядження навантажувальних ємностей.

Базовий логічний елемент ЕЗЛ утворюють заміною вхідного транзистора струмового ключа групою паралельно ввімкнених транзисторів. На рис. 3.1, б) показано схему такого елемента з конкретними особливостями, властивими серіям ЕЗЛ і, зокрема, серії К500. Елемент містить джерело опорної напруги, що складається з дільника напруги R_1-R_2 і ЕП на транзисторі T'' і резисторі R_3 , через який напруга з дільника подається на базу опорного транзистора T_{on} . Діоди D_1 і D_2 забезпечують температурну компенсацію відхилень параметрів елемента.

Входи елемента з'єднані високоомними резисторами (50 кОм) з загальним джерелом живлення, що дозволяє відмовитись від додаткових зовнішніх джерел для забезпечення правильного електричного режиму роботи перемикача струму. В елементах АБО-НІ/АБО на невикористовувані входи треба подавати логічні нулі, це й забезпечує з'єднання входів з джерелом живлення. Якщо ж певний вхід використовується, то вплив його на режим резистора R дуже малий внаслідок його високого опору. Резистори, що входять до вихідних ЕП, виконано зовнішніми для зменшення розсіювання потужності в корпусі мікросхеми і для можливості мати різні варіанти реалізації вихідних кіл. Для звичайного навантаження резистор в 0,3...2,0 кОм вмикають між емітером вихідного транзистора і стандартним джерелом живлення (близько – 5 В), а для передачі сигналів по узгоджених за хвильовим опором лініях, коли потрібно мати невисокий вихідний опір елемента, резистор беруть таким, що дорівнює 50...100 Ом. Щоб при цьому не створювався недопустимо великий струм вихідного ЕП, цей резистор під'єднують до джерела зниженої напруги (близько 2 В).

Електричні кола спільної точки для логічної частини елемента і вихідних ЕП для уникнення завад з боку останніх виконують окремо, як умовно показано на рис. 3.1, б).

Робота базового елемента є такою. Якщо хоча б на одному вході є логічна одиниця, тобто високий рівень сигналу, то відповідний транзистор відкривається, і струм I_e замикається через $R_{\kappa 1}$, знижуючи рівень сигналу на виході $U_{вих1}$. Якщо ж на всіх входах будуть логічні нулі, то відкриється транзистор T_{on} правого «плеча», що одночасно приведе до замикання всіх транзисторів лівого «плеча» та підвищення напруги $U_{вих1}$. Зі сказаного зрозуміло, що $Z_1 = \overline{X_1 + X_2 + \dots + X_n}$.

Сигнали на виходах Z_1 і Z_2 протифазні, тому

$$Z_2 = X_1 + X_2 + \dots + X_n.$$

Оскільки виконуються одночасно операції АБО/НІ і АБО, то можна вважати, що елемент має підвищену функціональну гнучкість.

ЛЕ для більш складних логічних операцій можуть бути реалізовані декількома шляхами: за допомогою відповідного з'єднання базових елементів і спеціальних прийомів, описаних нижче.

Один із таких прийомів полягає в об'єднанні кількох виходів емітерних повторювачів (рис. 3.2). При паралельній роботі декількох ЕП (на рис. 3.2 їх показано лише два) на вихід елемента проходить максимальний з вхідних сигналів, закриваючи одночасно ті транзистори, на базах яких діє більш низький потенціал. Для отримання на виході логічної одиниці достатньо однієї логічної одиниці на входах. При подачі на входи тільки нульових сигналів всі транзистори знаходяться в однакових режимах і спільно працюють на загальне навантаження, створюючи на ньому низький рівень логічного нуля. Таким чином, об'єднання емітерів вихідних каскадів декількох елементів дає додатково монтажну реалізацію операції АБО.

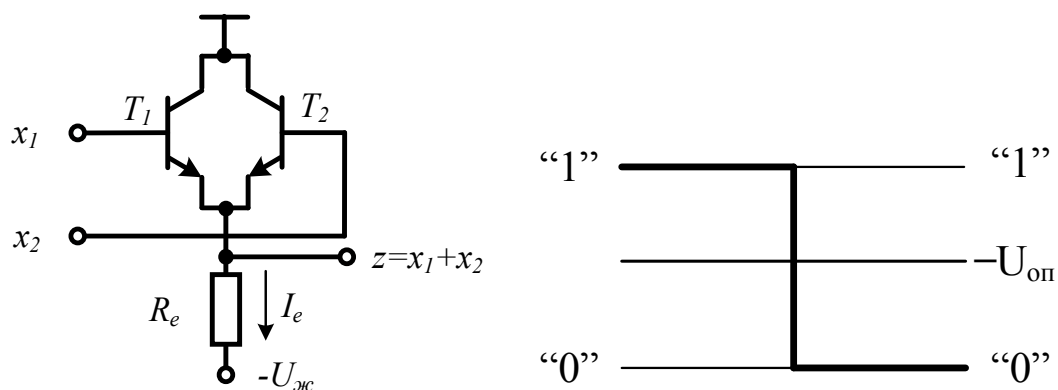


Рисунок 3.2 – Схема логічного елемента з об'єднанням виходів ЕП

Крім об'єднання виходів ЕП широко застосовують багатоярусне вмикання перемикачів струму. На рис. 3.3 показана схема двоярусного перемикача струму з лише одним джерелом струму. Для стабілізації струму тут додано каскад на основі T_I , R_e і $U_{ж}$, де транзистор увімкнений за схемою з ЗБ і забезпечує високий вихідний опір струму через транзистори, що входять до складу власне перемикачів струму. Струм, заданий джерелом струму, перемикається далі на декількох ярусах, у кожному з яких спрямовується у те чи інше коло за допомогою однієї з вхідних змінних (на рис. 3.3 – двох змінних). Для багатоярусного перемикача потрібно кілька опорних напруг і елементів зсуву рівнів вхідних сигналів, що надходять на різні яруси (T_7 , D_I). Схема на рис. 3.3 ілюструє використання багатоярусного перемикача для реалізації функції $z = x_1 \oplus x_2$. Застосування багатоярусних перемикачів при відтворенні складних функцій від двох або трьох змінних дозволяє зменшити споживану елементами потужність і забезпечує високу швидкодію. На сьогодні вже створені й широко застосовуються ЕЗЛ-

[illegible]

Типовий елемент $E^2ЗЛ$ складається з вхідного емітерно-зв'язаного каскаду на транзисторах $T_1 \dots T_m$ і емітерно-зв'язаного ключа на транзисторах T' і T'' (рис. 3.4, *a*). Паралельна робота кількох ЕП, що утворюють вхідний каскад, відповідає, як було показано вище, виконанню операції АБО для сигналів позитивної логіки. Сигнал із елемента АБО надходить на вхід струмового ключа, що має прямий та інверсний виходи. Як і елементи ЕЗЛ, типові елементи типу $E^2ЗЛ$ реалізують функції АБО-НІ/АБО. ЕП вхідної логічної схеми виконує також функцію зміщення сигналів за рівнем, тому сигнали з колекторів транзисторів струмового ключа подаються безпосередньо на входи наступних елементів. В елементах $E^2ЗЛ$ знижена енергія перемикавання порівняно з елементами типу ЕЗЛ, крім того, в них вирішуються затримки сигналів на обох виходах, що спрощує вирішення задачі усунення часових змагань (гонок) сигналів в автоматах з пам'яттю. Це досягнуто завдяки зменшенню вхідної ємності елемента і сумарної ємності переходів транзисторів.

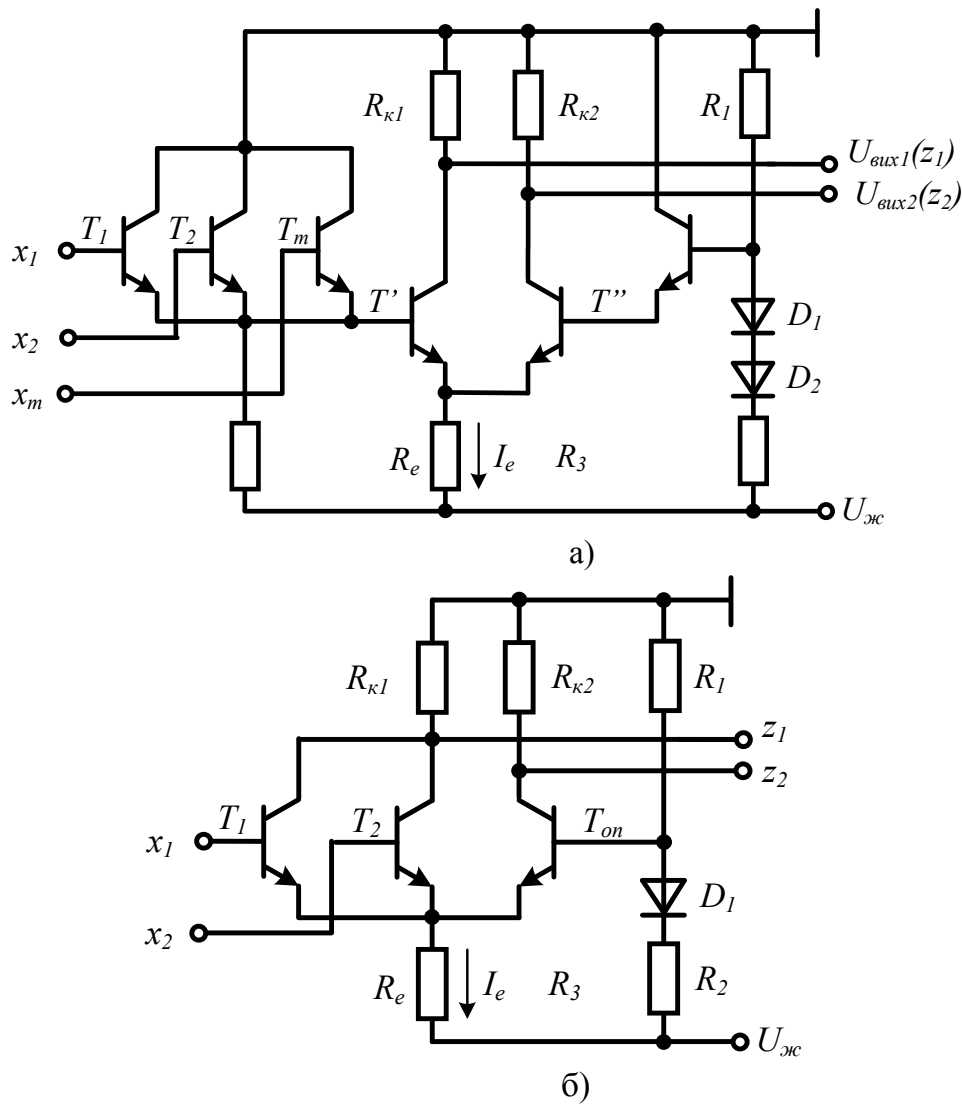


Рисунок 3.4 – Схеми типових елементів E^2ZL а) і $MEZL$ б)

Показники швидкодії і енергії перемикання елементів EZL поліпшуються зі зменшенням логічного перепаду і напруги живлення. Цього можна досягти, якщо вилучити ЕП зі схем сімейства EZL . Однак таке спрощення (рис. 3.4, б) пов'язане з серйозним впливом на режим роботи транзисторів і може здаватися недопустимим, тому що перешкоджає узгодженню потенціалів, необхідних для ненасиченого режиму відкритих транзисторів. Дійсно, в елементах $MEZL$ -типу транзистори входять в режим насичення, але при малих логічних перепадах ступінь насичення невисокий і не спричиняє істотного впливу на перехідні процеси. У той же час позитивний вплив зменшення логічного перепаду на швидкодії елементів позначається повною мірою. Внаслідок малого логічного перепаду елементи $MEZL$ мають невисоку завадостійкість, а відсутність ЕП обумовлює невеликі коефіцієнти розгалуження. Все це ускладнює використання $MEZL$, але при проектуванні ЦІМС високого СІ досить часто виникають умови, коли застосування $MEZL$ є можливим і доцільним.

Одна з переваг елементів сім'ї ЕЗЛ – постійність струму, споживаного ними від джерела живлення, оскільки струми в елементах не формуються або перериваються, а лише комутуються. Нагадаємо, що елементи ТТЛ не мають подібної властивості, а наявність наскрізних струмів при перемиканні – одна з проблем, яку доводиться враховувати при їх застосуванні.

Завадостійкість елементів ЕЗЛ і Е²ЗЛ визначається допустимими завадами амплітудою до 130...160 мВ, МЕЗЛ – 40...50 мВ, перепад рівнів сигналів в елементах ЕЗЛ становить близько 0,8 В, в елементах Е²ЗЛ – близько 0,6 В, в елементах МЕЗЛ – близько 0,4 В, напруга живлення для ЕЗЛ і Е²ЗЛ – 5,2 В, для МЕЗЛ – 2–3 В. Загальний недолік елементів сім'ї ЕЗЛ – підвищене споживання потужності. Найсуттєвіші параметри елементів типу ЕЗЛ показані у табл. 3.1.

Таблиця 3.1 – Параметри ЕЗЛ- елементів

Параметри	Значення
Напруга живлення, В	–5
Потужність споживання, мВт	60
Затримка сигналів, нс	0,5
Максимальна частота пермикання, ГГц	1
Типова завадостійкість, В	0,3
Вхідна/вихідна напруга високого рівня, В	0,1...-1/-0,74...-0,85
Вхідна/вихідна напруга низького рівня, В	–1,4...-5/-1,53...-1,69

Як видно з табл. 3.1, споживана потужність і швидкодія елементів ЕЗЛ набагато вищі, ніж у ТТЛ-елементів.

3.2 Інтегральна інжекційна логіка (І²Л)

ЛЕ типу І²Л в сучасній технічній літературі відомі за декількома назвами, причому найвживанішими є: інтегральна інжекційна логіка (І²Л), поєднана транзисторна логіка (ПТЛ) та транзисторна логіка з інжекційним живленням (ТЛІЖ).

Варіанти назв породжені намаганням їх авторів відобразити в назві певні технологічні чи схемотехнічні особливості виготовлення або функціонування елементів. Наприклад, вищеперелічені назви підкреслюють факт фізичного об'єднання (суміщення) в таких елементах транзисторів T_p з провідністю типу p - n - p і транзисторів T_n з провідністю типу n - p - n (рис. 3.5). При цьому емітерну область T_p називають інжектором і під'єднують до $+E_{ij}$. Загальну для T_p і T_n область з провідністю n -типу (це база T_p і емітер T_n) заземлюють. Колектор T_p і база T_n таким чином утворюють

єдину область p -типу, де обидва транзистори можуть бути багатоклекторними (БКТ).

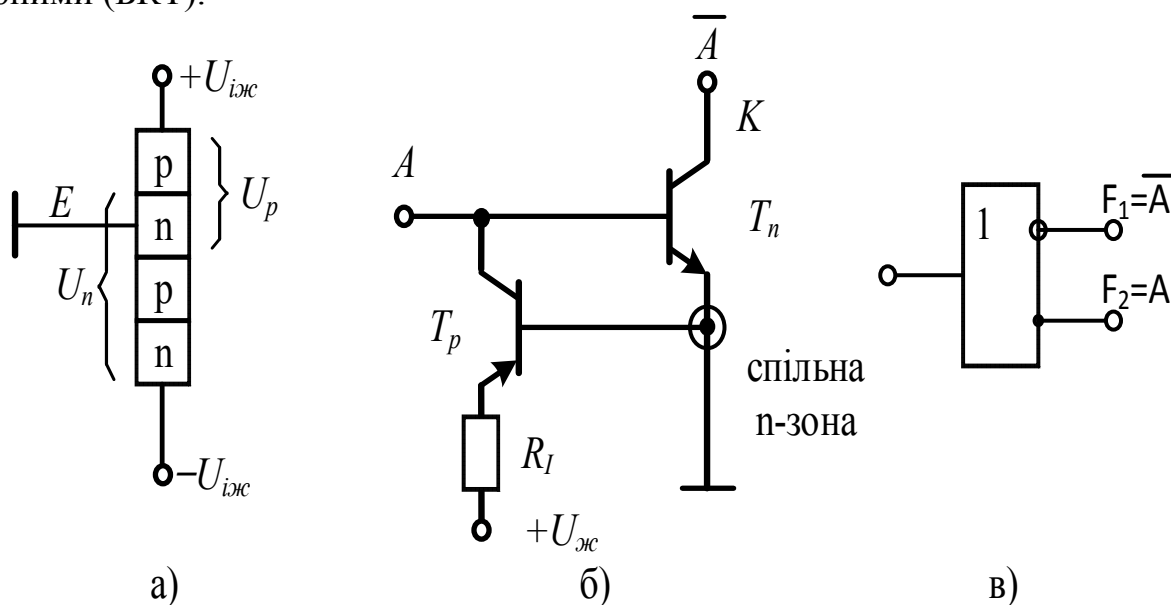


Рисунок 3.5 – Структура а), схема б) та її УГП в) I^2L

Суть інжекційного живлення елементів I^2L полягає в тому, що за допомогою p - n - p транзистора-інжектора утворюють коло генератора струму $I_0 = a_n I_h I_n$ (де n – число колекторів транзистора-інжектора) бази багатоклекторного вертикального n - p - n транзистора (БКТ), що виконує функцію інвертора логічного сигналу $F_{KI} = A$ (рис. 3.6, а). Така конфігурація будуватиметься на безрезисторній структурі, в якій резистори замінюють генераторами постійного струму I_0 .

Робота вентиля I^2L (рис. 3.6, а) ґрунтується на тому, що, залежно від значення вхідного сигналу A , змінюється опір (від високого до низького) на базі БКТ. Якщо опір генератора ($A=0$) наближається до нуля (низький рівень), то струм I_0 інжектора стікає на землю і транзистор БКТ закривається. При цьому на переходах колектор-емітер виникає режим високого опору і на виходах K_1 і K_2 значення вихідного сигналу $A=1$. Якщо ж вхідний сигнал $A=1$ (високий опір), то струм I_0 протікає в базу БКТ, який відкривається і переходить в режим насичення, а опори на переходах колектор-емітер БКТ різко зменшуються і наближаються до нуля. Вихідні рівні сигналів при цьому на колекторах K_1 і K_2 приймають значення $A=0$.

Слід відзначити, що резистор інжектора R_I виготовляється за напівпровідниковою технологією і тому спад напруги на ньому повинен бути не меншим, ніж на відкритому p - n переході $U_{p-n} > 0,6-0,7$ В. Таким чином, значення напруги живлення має бути на рівні $U_{ж} = 1,5$ В.

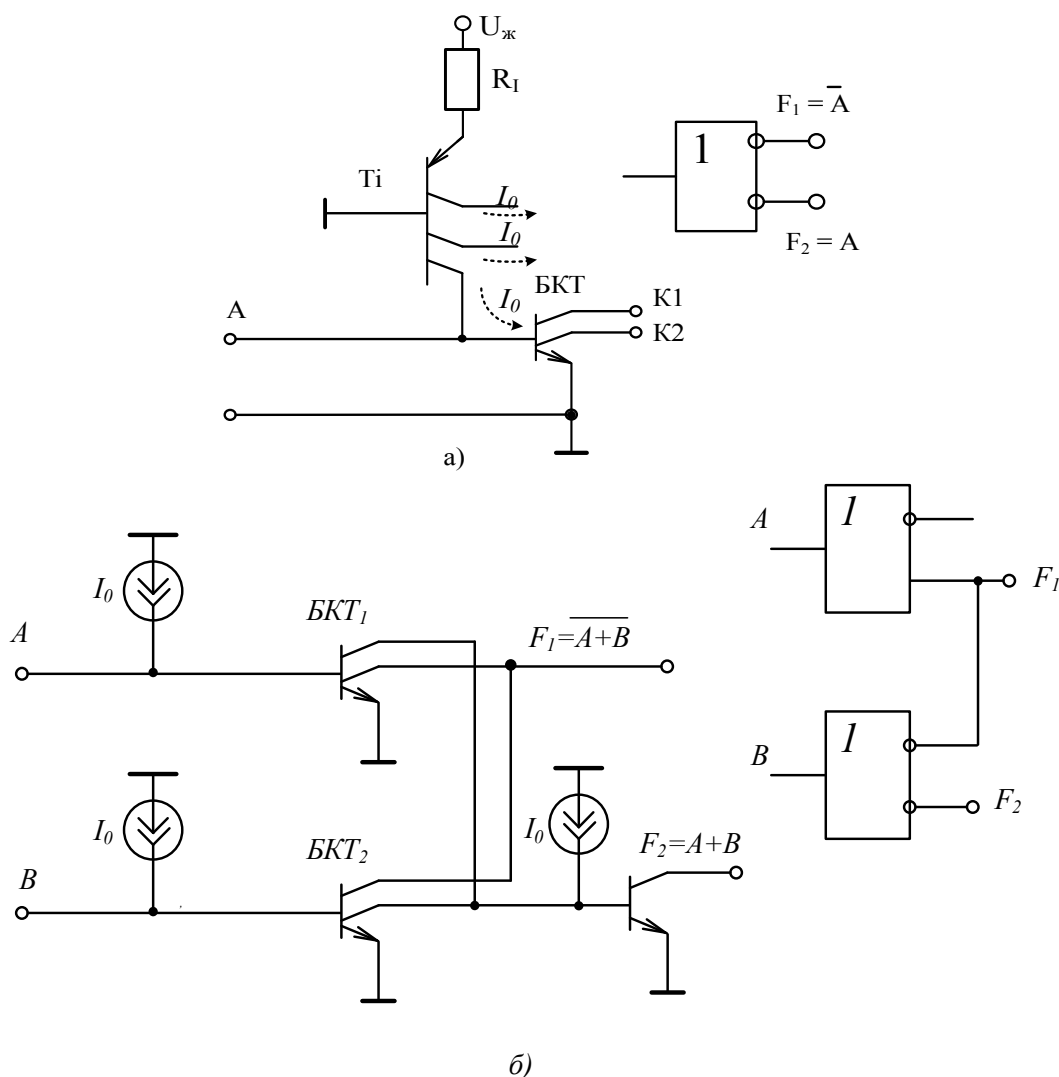


Рисунок 3.6 – Вентиль а) та ЛЕ АБО-НІ і АБО б) типу I^2L

Схемотехніка I^2L має ряд особливостей. Перша: ключовий транзистор T_n потрібно робити багатоколекторним, щоб його колектори могли слугувати генераторами A або $\bar{A}$ для групи інших елементів. Друга: інжектор (генератор струму I_0) має містити багатоколекторний T_p транзистор, щоб його виходи, по яких протікають струми I_0 , можна було під'єднати до баз багатьох транзисторів T_n . Третя: Резистор R_I повинен мати таку вольт-амперну характеристику, яка б при досягненні на переході база-емітер T_p значення $U_{be} \sim U_{p-n} = 0,6-0,7$ В була б лінійною. Це дозволить задавати потрібні значення I_0 , регулюючи рівень $U_{ж}$. Потрібно також зазначити, від значення I_0 залежить швидкодія елементів I^2L . При малих I_0 швидкість перемикання T_n зменшується, і елемент переходить в мікропотужний режим, при збільшенні ж I_0 швидкодія збільшується. Таким чином можна регулювати потужність споживання і швидкодію I^2L .

При виконанні операції АБО-НІ ($F_1 = \overline{A+B}$) два вентилі інвертора BKT_1 і BKT_2 об'єднують по колекторних колах (рис. 3.6, б). Для реалізації

операції АБО ($F_2 = A + B$) застосовують додатковий інжекційний інвертор. Переваги І²Л – відсутність резисторів (і пов'язана з цим економія площі, зменшення потужності, зменшення $U_{iж}$ і часу затримки), мала ємність колектора і мала залишкова напруга на насичених транзисторах.

За допомогою елементів типу І²Л вдалося значною мірою подолати традиційні недоліки ЦІМС на біполярних транзисторах: невисоку щільність компонування і велику розсіювану потужність (в розрахунку на один логічний вентиль). За рівнем інтеграції ЛЕ елементи типу І²Л навіть перевершують елементи на МОН-структурах, за рівнем розсіюваної потужності їх можна порівняти з КМОН-елементами, за швидкодією – з ЦІМС на біполярних транзисторах ($t_{зд.ср}=5$ нс). Мала розсіювана потужність І²Л елементів пояснюється відсутністю резисторів, а велика швидкодія при малих потужностях споживання – незначними паразитними ємностями, відсутністю накопичення заряду і невеликою різницею логічних рівнів.

3.3 Логічні елементи на МОН-транзисторах

Одною з найважливіших особливостей МОН-транзисторів як основи для створення ЛЕ, порівняно з біполярними транзисторами, є високий вхідний опір. Тому на них досить просто можна реалізувати динамічний принцип зберігання інформації, застосування якого дозволяє знизити споживану потужність. Принципи роботи елементів *p*-МОН і *n*-МОН типів зручно простежити на прикладі інверторів. Розрізняють просту схему побудови інверторів (послідовне або паралельне з'єднання ключа і навантажувального резистора) і більш складну (послідовно-паралельне та мостове з'єднання транзисторів і резисторів). При цьому як ключ застосовують активні транзистори з індукованим каналом, а навантаження може бути реалізоване за допомогою транзистора з індукованим або вбудованим каналом. Розглянемо особливості побудови інверторів і логічних елементів на *p*-МОН і *n*-МОН транзисторах.

3.3.1 Логічні елементи на *p*-МОН транзисторах

В історичному аспекті такі ЛЕ є більш ранніми порівняно з іншими типами ЛЕ на МОН транзисторах. Для них характерне застосування навантажувальних транзисторів з індукованим каналом. У цьому випадку навантаження називають нелінійним, якщо транзистор ввімкнений за схемою на рис. 3.7, а, або квазілінійним – за схемою на рис. 3.7, б. Якщо напруга між заслоном і витоком перевищує порогове значення, то в транзисторі виникає канал, що проводить струм між стоком і витоком. В іншому випадку провідність ділянки стік–витік близька до нуля. Логічну одиницю відображають негативним сигналом, абсолютна величина якого перевищує порогове значення, а логічний нуль – негативним сигналом, абсолютна величина якого не досягає порогового значення. Як видно, в елементах типу *p*-МОН використовують негативну логіку.

Коли $x = 1$, то проводить активний транзистор T_a . Навантажувальний транзистор T_n при цьому також вмикається, оскільки між його затвором і витокom діє напруга, що перевищує порогове значення. Отже, рівень вихідного сигналу формується дільником напруги, який утворений опорами каналів замкнених транзисторів T_a і T_n , а саме:

$$U_{\text{вих}} = U_0 = U_{\text{ж}} R_a / (R_a + R_n),$$

де R_a і R_n – опір каналів відповідних транзисторів T_a і T_n .

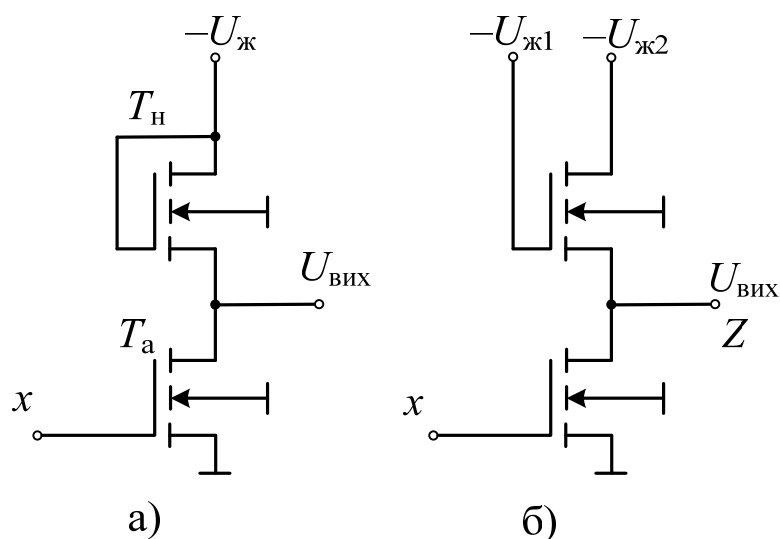


Рисунок 3.7 – Схеми інверторів із навантажувальними транзисторами:
а) нелінійне навантаження; б) квазілінійне навантаження

U_0 має становити малу частку напруги $U_{\text{ж}}$, причому необхідне виконання умови $R_a \ll R_n$. Ця умова забезпечується виготовленням транзисторів з різними розмірами каналів. У навантажувального транзистора збільшують довжину каналу, у активного – його ширину (відношення опорів каналів може досягати 25).

Аналіз статичного стану інвертора при $x = 1$ показує характерні недоліки p -МОН елементів. По-перше, необхідність витримувати певне співвідношення між опорами каналів (елементи цього типу називають іноді елементами «з відношенням») збільшує витрату площі кристала і, отже, вартість елемента. По-друге, через обидва транзистори від джерела живлення до загальної точки протікає статичний струм. Це робить недоцільним застосування низькоомних каналів внаслідок зростання потужності, розсіюваної елементом. Високоомність каналів спричиняє втрату швидкості, оскільки навантажувальні ємності не можуть швидко перезаряджатися через високоомні канали.

Варіант інвертора з квазілінійним навантаженням (див. рис. 3.7, б) відрізняється застосуванням іншого джерела живлення $U_{ж2}$, причому абсолютна величина $U_{ж2}$ має перевищувати абсолютну величину $U_{ж1}$ не менше, ніж на порогову напругу. У цьому випадку властивості навантажувального транзистора наближаються до властивостей лінійного опору, що, до деякої міри, покращує характеристики елемента.

Для елементів типу p -МОН характерні затримки сигналів в сотні або десятки наносекунд, споживана потужність в десятки або одиниці міліват, напруги живлення, що значно перевищують номінали напруги живлення елементів ТТЛ (наприклад, типове значення напруги живлення для p -МОН елементів з нелінійними навантаженнями – 10 В). Очевидно, що такі параметри не сприяють використанню p -МОН елементів у ВІМС або НВІМС. Вони взагалі є програшними порівняно з параметрами елементів інших типів. Однак ще не можна вважати p -МОН елементи морально застарілими. Вони добре відпрацьовані щодо технології, охоплюють широку номенклатуру функціональних різновидів, мають невисоку вартість і досить високий відсоток виходу придатних виробів в умовах масового їх виготовлення. Застосування їх буває доцільним у випадках, коли забезпечувані ними параметри є прийнятними з технічної та економічної точок зору. Розглянуті вище схеми інверторів є основою для побудови базових елементів. Якщо в інверторі замінити активний транзистор групою паралельно або змішано ввімкнених транзисторів, керованих входними змінними, то тим самим можна отримати інші ЛЕ.

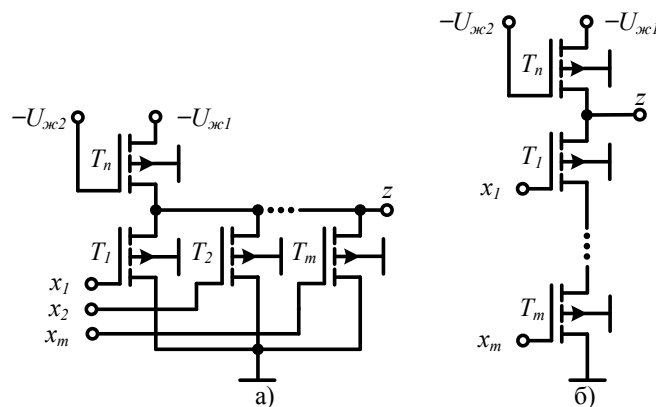


Рисунок 3.8 – ЛЕ на p -МОН транзисторах: АБО-НІ а), І-НІ б)

ЛЕ АБО-НІ (рис. 3.8, а) містить групу паралельно ввімкнених транзисторів $T_1 \dots T_m$, а реалізована функція має вигляд

$$z = \overline{x_1 + x_2 + \dots + x_m}.$$

При послідовному з'єднанні вхідних транзисторів, отримують елемент І-НІ (рис. 3.8, б). Тут логічний нуль на виході формується тільки при стані провідності всіх транзисторів, тобто напруга U_0 є сумою напруг в ланцюжку з m відкритих транзисторів і, порівняно з вихідною напругою елемента АБО-НІ, має більш високе значення. Для отримання того ж значення U_0 в ЛЕ І-НІ потрібно застосовувати транзистори, у яких ширина каналу в m разів більша, ніж у ЛЕ АБО-НІ, тобто за рахунок збільшення геометричних розмірів логічного вентиля. Тому саме з цих міркувань елементу АБО-НІ слід віддати перевагу і розглядати його як базовий, а там, де це можливо, від застосування елементів І-НІ на p -МОН транзисторах потрібно відмовитися.

Швидкодія ЛЕ на МОН транзисторах обмежується часом перезарядження навантажувальних ємностей. Особливо велике ємнісне навантаження дає вихідний каскад ЛЕ, оскільки в ньому додаються ємності декількох входів наступних елементів, під'єднаних до виходу даного. Внаслідок цього розглянуті схемні варіанти ЛЕ часто доповнюють спеціальними буферними каскадами з підвищеною здатністю до навантаження.

3.3.2 Логічні елементи на n -МОН транзисторах

Проектують на базі n -канальних МОН транзисторів, що мають властивість самозакриватися. При цьому застосовують більш досконалу технологію виготовлення n -МОН транзисторів, яка дозволяє досягти більшої щільності пакування електронних компонентів, ніж для p -МОН елементів. Це стає можливим завдяки значно меншим опорам каналів. Їх величина складає приблизно третину від величини опорів каналів p -МОН елементів. Менший опір каналів і менші вхідні ємності в структурі транзисторів обумовлюють підвищення швидкодії і n -МОН елементи працюють приблизно так само швидко, як стандартні ТТЛ-елементи. Їх затримка сигналів складає близько 10 нс.

Менший опір каналів дозволяє також знизити напругу живлення до 5 В. Це робить можливим спільне використання n -МОН вентилів та ТТЛ-вентилів в деяких конструкціях ЛЕ (технічною мовою кажуть, що вони «сумісні»).

Схеми n -МОН елементів за своєю структурою загалом аналогічні схемам p -МОН елементів, відрізняючись лише технологією виготовлення n -канальних МОН транзисторів. На рис. 3.9 наведено дві схеми типових елементів: лівий елемент реалізує логічну операцію І-НІ для позитивної логіки, правий – операцію АБО-НІ.

Відповідні діапазони рівнів сигналів показані на рис. 3.10. На сьогодні за n -канальною МОН-технологією окремі ЛЕ у вигляді ЦІМС практично не виготовляють. З економічної точки зору за такою технологією у вигляді ЦІМС більш доцільно виготовляти досить крупні обчислювальні засоби, наприклад, пристрої для множення, кодувальники, лічильники тощо.

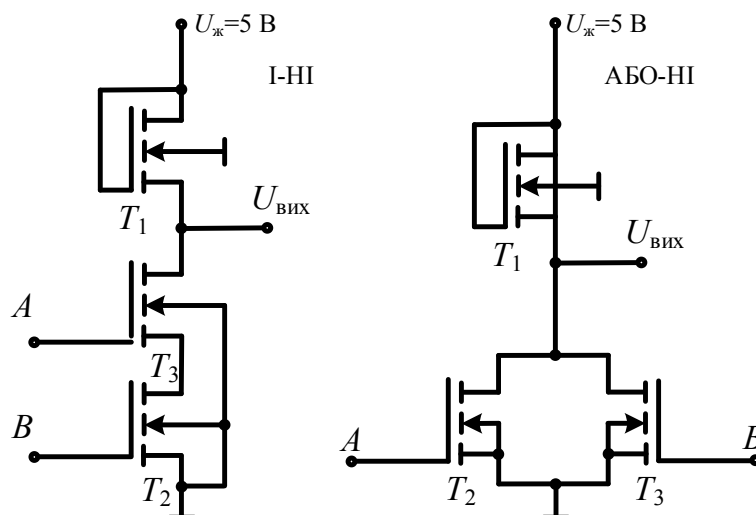


Рисунок 3.9 – Типові *n*-МОН елементи

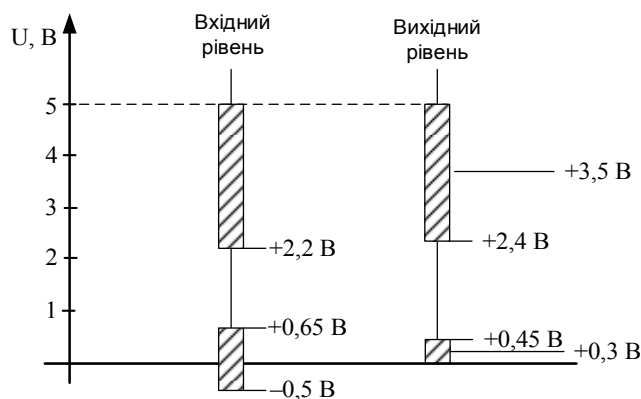


Рисунок 3.10 – Рівні сигналів для *n*-МОН елементів

У табл. 3.2 показано найважливіші характеристики таких типів елементів.

Таблиця 3.2 – Параметри типових *n*-МОН елементів

Напруга живлення	+ 5 В
Енергоспоживання	2 мВт при низькому вихідному рівні 0 мВт при високому вихідному рівні
Затримка сигналів	5 нс
Максимальна частота	80 МГц
Різниця між рівнями сигналів	≈ 2 В

3.3.3 Логічні елементи на комплементарних МОН-транзисторах (КМОН-логіка).

Ці ЛЕ використовують транзистори різних типів провідності. Як і інші ЛЕ на МОН транзисторах їх надбудовують на базі відповідного інвертора (рис. 3.11). З подачею на вхід сигналу високого рівня в кожній парі транзисторів різної провідності *p*-МОН транзистор закривається, а *n*-МОН тран-

зистор відкривається. Низькорівневий вхідний сигнал викликає відкривання p -МОН транзистора і закривання n -МОН транзистора. Отже, за будь-якого рівня вхідного сигналу один з транзисторів знаходиться в закритому стані, і струм через нього не протікає (крім струму витоку). Це приводить до практично нульової споживаної елементом потужності в статичному режимі. Крім того, КМОН-елементи мають надзвичайно високий вхідний опір, який складає величину порядку 10^8 МОм. Як наслідок цього навантажувальна здатність КМОН-елементів може сягати 50 і більше без суттєвого збільшення споживаної потужності.

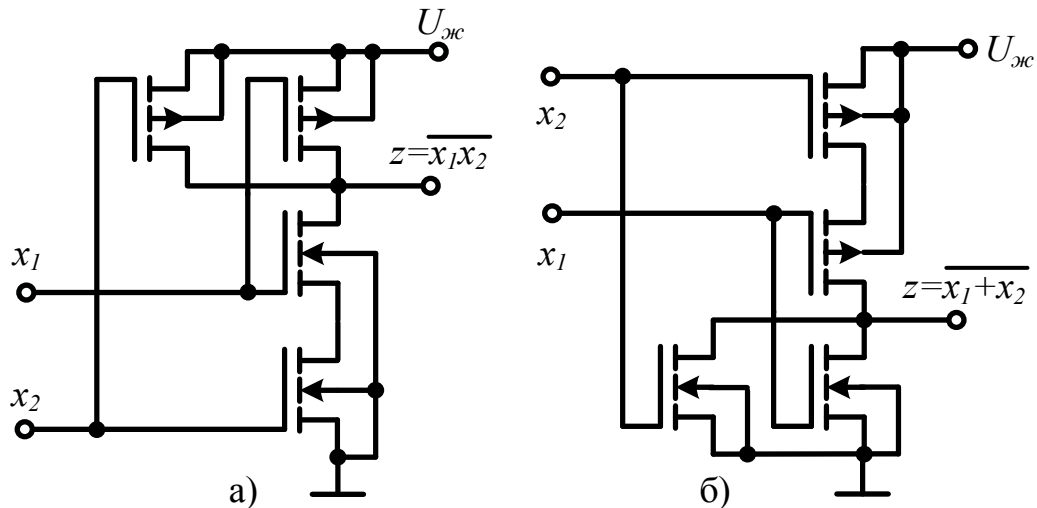


Рисунок 3.11 – Схеми базових елементів І-НІ а) та АБО-НІ б) на МОН транзисторах різних типів провідності

3.3.4 Динамічні логічні елементи

Елементи цього типу свого часу були розроблені для усунення недоліків, властивих потенціальним елементам типу p -МОН. Хронологічно вони з'явилися після елементів типу p -МОН, але до розробки елементів n -МОН і КМОН. Динамічні елементи на транзисторах з p -каналами стали основою створення перших ВІМС, оскільки істотне поліпшення параметрів елементів дозволило різко підвищити ступінь інтеграції ЦІМС на їх основі.

Принципи передачі та обробки сигналів, характерні для динамічних (імпульсних) елементів, можуть застосовуватися не тільки для МОН-структур, але й для біполярних. Хоча ЛЕ на p -МОН за своїми параметрами значно поступаються n -МОН і КМОН-елементам, динамічні ЛЕ все ж отримали розвиток і досить широке застосування, оскільки саме в цих ЛЕ можна найпростіше здійснити тимчасове запам'ятовування сигналів на електричних ємностях. Конденсатори, на яких би запам'ятовувались сигнали в динамічних елементах на МОН-транзисторах, не потрібно спеціально виготовляти, бо для цієї мети використовують паразитні ємності, що неминуче виникають в процесі створення МОН-структур. (Відомо, що для ЦІМС конденсатори – небажані компоненти, які займають велику площу

на кристалі). Запам'ятовування ж сигналів на паразитних ємностях не порушує технологічного процесу. Звичайно ці паразитні ємності сягають декілька пікофарад або навіть їх часток, однак і струми, що викликають їх перезарядження через замкнені МОН-транзистори, також невеликі. В результаті час зберігання вимірюється мілісекундами, а мінімальна частота сигналів синхронізації лежить в кілогерцовому діапазоні. Зверху ця частота обмежена часом перезарядження запам'ятовувальних ємностей через відкриті транзистори і може досягати десятків мегагерц.

Динамічні елементи можна класифікувати за рядом ознак: наявністю або величиною статичного струму (струму витіку); числу фаз сигналів синхронізації і т. п. Розглянемо лише один варіант динамічних елементів – елемент чотирифазної логіки. У цьому випадку використовуються елементи, що управляються лише двома фазами, але в загальному випадку сусідні елементи не можна жити тими ж фазами, тому система, в цілому, вимагає чотирьох фаз управління.

На рис. 3.12, а) показана схема динамічного інвертора з двофазним управлінням і перекриттям фаз та часові інтервали фаз імпульсів тактування (рис. 3.12, б), що пояснюють роботу динамічного інвертора.

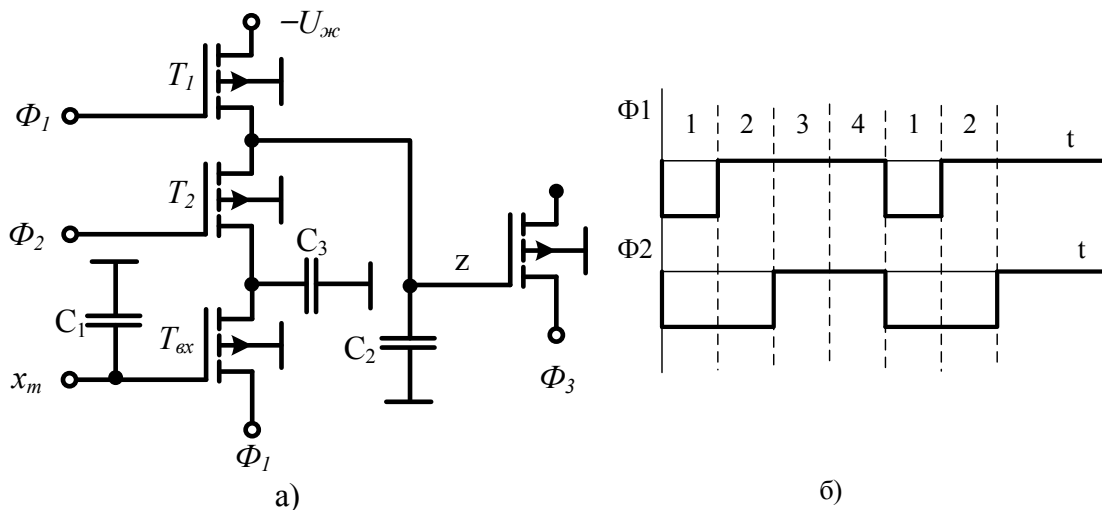


Рисунок 3.12 – Схема інвертора з двома серіями синхросигналів а) і часова діаграма чотирифазного управління динамічною логікою б)

На часовому інтервалі 1 збуджуються серії Φ_1 і Φ_2 , вмикаючи транзистори T_1 і T_2 . Транзистор T_{ex} в цей час замкнений незалежно від значення вхідного сигналу x , оскільки на його витік подається негативний сигнал послідовності Φ_1 . Ємності C_2 і C_3 при цьому заряджаються через ввімкнені транзистори до напруги $-U_{жс}$. Ємність C_2 в такому разі – запам'ятовувальна, а C_3 – паразитна, яка не потрібна для роботи інвертора, але наявність якої змушує з нею рахуватися. Отже, на інтервалі 1 відбувається безумовне за-

ряджання ємностей C_2 і C_3 (безумовне в сенсі незалежності від значення вхідної змінної).

На часовому інтервалі 2 припиняється збудження серії Φ_1 . При цьому замикається транзистор T_1 і знімається негативна напруга з витоку транзистора T_{ex} . Тепер процеси в інверторі вже залежать від значення вхідної змінної. Якщо $x = 1$ (ємність C_1 заряджена), то транзистор T_{ex} проводить і ємності C_2 і C_3 розряджаються до нуля через відкриті транзистори T_2 і T_{ex} (на даному інтервалі напруга на витоку транзистора T_{ex} дорівнює нулю). Якщо $x = 0$ (ємність C_1 розряджена), то транзистор T_{ex} замкнений, і ємності C_2 і C_3 зберігають заряджений стан, оскільки шляху для струму розряду при цьому не утворюється.

Як видно, завдяки чергуванню процесів безумовного заряджання і обумовленого розряджання ємності C_2 елемент виконує операцію інверсії. Інвертування вхідного сигналу (стану ємності C_1) супроводжується часовою затримкою: вихідний сигнал формується в інтервалі 2, тоді як вхідний мав бути сформований на попередньому інтервалі.

Порівняно зі статичними МОН- і КМОН-інверторами динамічний інвертор потребує трьох транзисторів, однак у нього є важливі переваги. Для його нормальної роботи не потрібне виконання певного співвідношення між опором каналів транзисторів (це елемент «без відношень», див. пункт 3.3.1). Дійсно, рівні сигналів одиниці і нуля не залежать від опорів каналів – при достатній тривалості інтервалів фаз ємність C_2 заряджається практично до рівня напруги живлення серії Φ_1 , а розряджається до нуля. Тому всі транзистори можуть виготовлятися однаковими з мінімальними розмірами, що економить площу кристала. Далі, в динамічному інверторі немає статичних струмів, оскільки в послідовному ланцюжку транзисторів T_1 , T_2 , T_{ex} завжди є хоча б один закритий. Відповідно, немає необхідності обмежувати знизу опір каналів і можна застосовувати транзистори з низькоомними каналами. Це підвищує частоту перемикавання елементів, прискорюючи перезарядження ємностей через відкриті транзистори. І нарешті, відсутність статичних струмів і споживання струму тільки для необхідних процесів перезаряджання ємностей зводить до мінімуму споживану елементом потужність.

Таким чином, за перерахованими важливими параметрами динамічний інвертор має суттєві переваги. У той же час він більш складний і потребує системи імпульсного фазового живлення.

Динамічні ЛЕ інших типів будують на основі інвертора, замінюючи вхідний транзистор групою транзисторів, ввімкнених послідовно, паралельно та змішано.

В елементі АБО-НІ (рис. 3.13, а) замість вхідного транзистора ввімкнена паралельна група $T_{ex1} \dots T_{exm}$. Якщо хоча б одна з вхідних змінних має одиничне значення, ємність вихідного вузла на інтервалі 2 розрядиться че-

рез транзистор, керований цією змінною. Одиначний сигнал збережеться на виході схеми тільки при нульових значеннях всіх вхідних змінних.

На рис. 3.13, б показана схема елемента, що виконує операцію $z = \overline{x_1 + x_2 x_3}$.

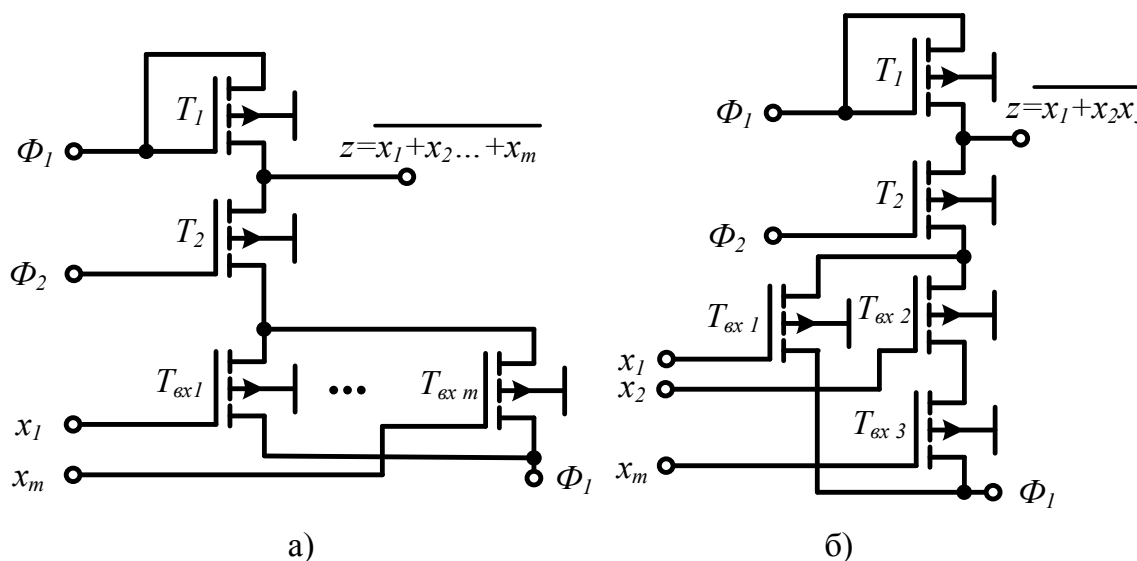


Рисунок 3.13 – Схема динамічного елемента АБО-НІ а)
і елемента, що виконує операцію $z = \overline{x_1 + x_2 x_3}$ б)

При проектуванні пристроїв на динамічних елементах потрібно встановити правила передачі сигналів від одного елемента до іншого з урахуванням характеру вихідних сигналів елементів на різних часових інтервалах чотирифазного циклу. На першому інтервалі відбувається безумовне заряджання вихідної ємності, вихід не має інформативного характеру. В той же час на кінці цього інтервалу необхідно сформувати вхідний сигнал, оскільки він знадобиться в наступному інтервалі. На другому інтервалі відбувається обумовлене розряджання вихідної ємності, вихід знову не має інформативного значення і не може бути використаний для управління іншими елементами. На третьому і четвертому інтервалах вихід переведений в стабільний логічний стан і може бути використаний для управління іншими елементами.

Динамічні елементи на час їх появи стали базою для створення перших ВІМС і до цього часу залишаються однією зі схемно-технологічних основ НВІМС. Після створення елементів на МОН транзисторах з *n*-каналами та удосконалення параметрів елементів типу КМОН стало можливим будувати ЦІМС високого рівня інтеграції і на потенціальних елементах. На сьогодні ще відчутна тенденція подальшого розширення використання потенціальних (статичних) КМОП-елементів в ЦІМС. Разом з тим потрібно зазначити, що в сучасних проектах все частіше зустрічається поєднання використання динамічних і статичних елементів для поліпшення параметрів ЦІМС в цілому.

3.4 Перетворення рівнів логічних сигналів

3.4.1 Перетворювачі рівнів сигналів

У цифровій схемотехніці перетворювачі рівнів сигналів служать для електричного узгодження логічних рівнів сигналів, напруги джерел живлення, передачі інформації між цифровими пристроями, в яких використані ЛЕ різних типів ЦІМС (ТТЛ, ЕЗЛ, І²Л, КМОН і т. д.).

У загальному випадку перетворювач рівня (ПР) – це ЦІМС, призначені для перетворення вихідних сигналів ЦІМС одного типу у вхідні сигнали ЦІМС іншого типу (табл. 3.3).

Таблиця 3.3 – Параметри цифрових сигналів

Параметр	Тип ЦІМС			
	ТТЛ	ЕЗЛ	І ² Л	МОН (КМОН)
$E_{П}, В$	+5	-5	+1...+2	-12...-27, [+9]
$U^0, В$	$\leq 0,4$	-1,6	$\leq 0,05$	-7...-10, [+0,3]
$U^1, В$	+2,4...+4,5	-0,8	+0,6...0,8	-2...-3 [+7]
$I_{вх}^+, мкА$	$\leq 0,1$	$\leq 0,3$	0	0,0015
$I_{вх}, мкА$	$\leq 1,6$	0	0,01..0,05	0,0015
$I_{вих}^+, мА$	≤ 1	$\leq 3...22$	0	$\leq 2,5$
$I_{вих}, мА$	≤ 16	< 3	$< 0,02$	$\leq 2,5$
$U_{ж}, В$	$\leq 0,6$	$\leq 0,2$	$\leq 0,1$	$\leq 1 \leq 0,9$

Очевидно, що для того, щоб вхід ПР (рис. 3.14) можна було з'єднати з виходом ЦІМС1, вхідний каскад П1 потрібно побудувати за принципом вихідного каскаду ЦІМС1, а П3 – за принципом вхідного каскаду ЦІМС2. Прикладом такого узгодження є модифікований ПР ТТЛ-ЕЗЛ (рис. 3.15, а).

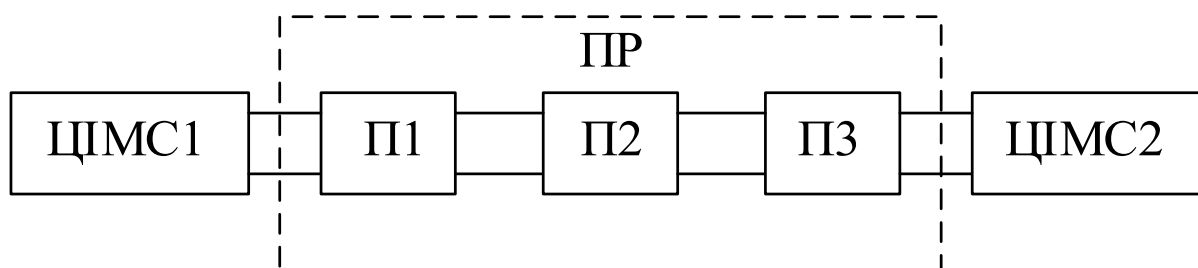


Рисунок 3.14 – Схема узгодження ЦІМС різних типів ЦІМС1 і ЦІМС2, де П100 вхідний каскад ПР; П2 – «власне ПР»

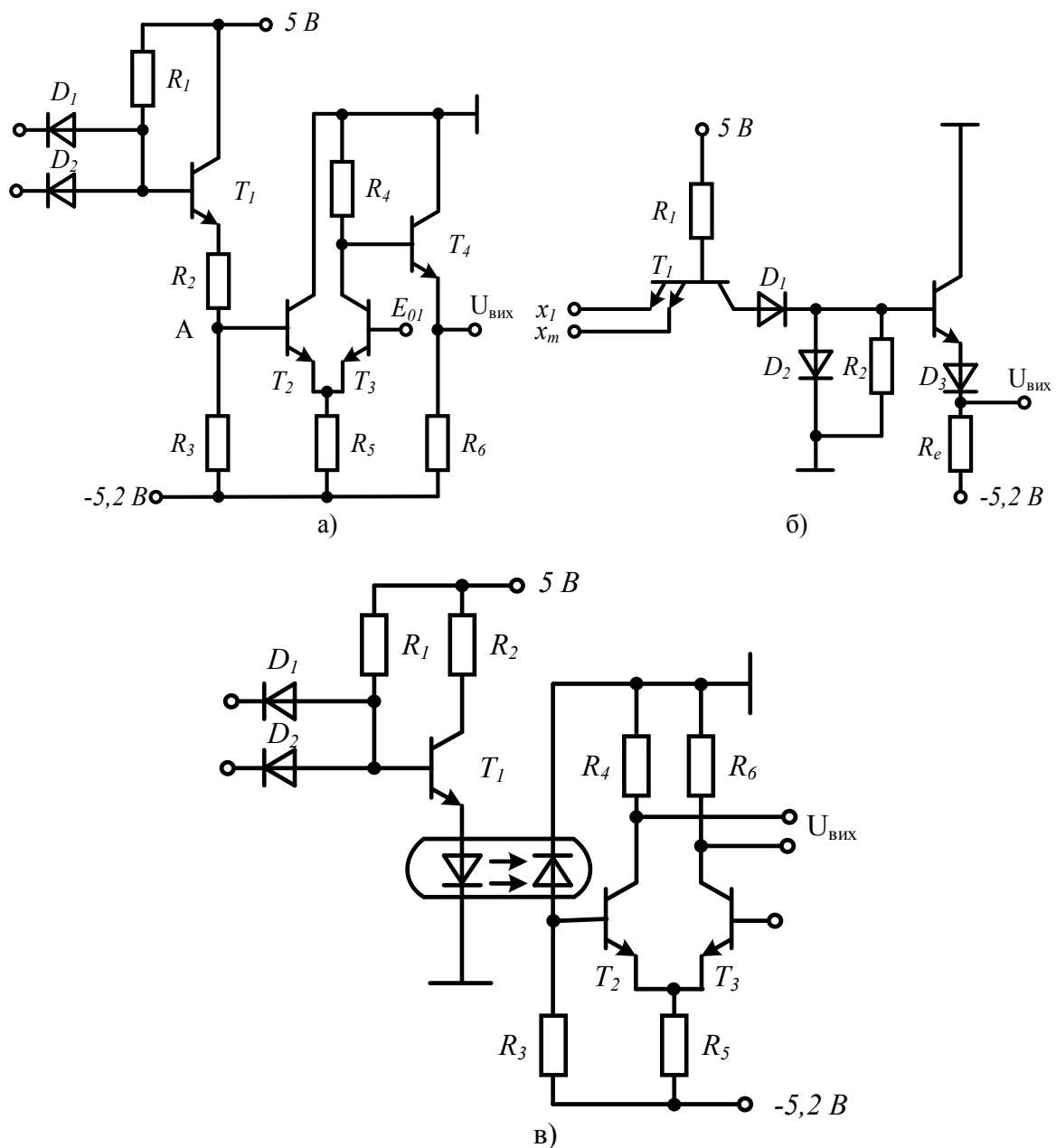


Рисунок 3.15 – Схема ПР ТТЛ-ЕЗЛ: модифікований ПР а);
із застосуванням БЕТ б); з оптронною розв'язкою в)

Як видно в модифікованому варіанті (рис. 3.15, а) вхідний каскад насправді повторює вхідний каскад попередньої ЦІМС з тією різницею, що тут зміщення вхідних рівнів здійснюється не на діодах, як в ТТЛ, а на резисторному дільнику R_2, R_3 . Тому в ПР за цією схемою дещо погіршується крутизна передатної характеристики, тобто розширюється зона перемикання. Для нормальної роботи потенціал в точці А повинен задовольняти такі умови: $U_A > E_{01}$ при $U_{\text{вх}} = U^1$ і $U_A < E_{01}$ при $U_{\text{вх}} = U^0$.

Простіший варіант ПР ТТЛ – ЕЗЛ наведено на рис. 3.15, б. ли $U_{\text{вх}} = U^0$, діод D_1 замкнений, $U_B = -I_B R_2$ де I_B – струм бази транзистора T_2 .

При $I_K = 5 \text{ мА}$ і $\beta = 20$ $I_B = 0,25 \text{ мА}$. Аналіз цього ПР показує, що на його виході забезпечуються ЕЗЛ-рівні ($U_{\text{вих}} = -1,6 \text{ В}$; $U_{\text{вх}} = -0,8 \text{ В}$). Однак ПР має і ряд недоліків: відносно великий вхідний струм; транзистор T_2 працює на межі області насичення і при не найкращому поєднанні параметрів може ввійти в насичення; низька завадостійкість. У ПР (див. рис. 3.15, в) оптрон виконує функцію гальванічної розв'язки. При застосуванні найменш інерційних діодних оптронів тут час затримки сигналів становить близько 200 нс, без оптронів же затримка сигналів на ПР оцінюється приблизно в 50 нс.

У загальному випадку для узгодження взаємодії M різних серій ЦІМС потрібно було б випускати M^2 - M типів ПР. Очевидно, що це число швидко зростає (наприклад, тільки для серій ЦІМС К500; К100; 100; 500; К1500; 1000 потрібно було б випускати 30 типів ПР). Таку широку номенклатуру ПР можна істотно звузити, якщо всі перетворення здійснювати через деякий проміжний стандарт (для цього найчастіше використовують стандарт ТТЛ). При цьому замість M^2 - M типів потрібно лише $2M$ типів ПР сигналів стандарту ТТЛ в інші і для оберненого перетворення.

3.4.2 Транслятори сигналів ВІМС і НВІМС

Ці пристрої цифрової схемотехніки також виконують функції ПР, оскільки в процесі виробництва ВІМС і НВІМС для поліпшення їх параметрів і якості застосовують різні варіанти елементів ТТЛ, І²Л, ЕЗЛ та їх комбінацій.

Вимоги до параметрів ЛЕ в складі ВІМС і НВІМС мають специфіку порівняно з вимогами до параметрів елементів невисокого рівня інтеграції. Ця специфіка визначається умовами, в яких працюють ці елементи. Істотно різні вимоги висуваються до елементів у внутрішній структурі ВІМС (НВІМС) і до елементів на їх входах і виходах. Якщо допустима потужність, що розсіюється кристалом ВІМС (НВІМС), становить $P_{\text{кр}}$, а площа кристала становить $A_{\text{кр}}$, то отримаємо обмеження на споживану потужність і площу кристала, зайняту елементами:

$$P_{\text{ел}} \leq P_{\text{кр}}/N_{\text{ел}}, \quad (3.1)$$

$$A_{\text{зл}} \leq \chi A_{\text{кр}}/N_{\text{ел}}, \quad (3.2)$$

де $N_{\text{ел}}$ – число елементів,

$\chi A_{\text{кр}}$ – частка зайнятої ними площі.

Значення $P_{\text{кр}}$ (2...5 Вт) і $A_{\text{кр}}$ (20...50 мм²) обмежуються конструкторсько-технологічними факторами і зростають відносно повільно. Водночас вимоги до функціональної складності ЦІМС зростають дуже швидкими темпами. Для реалізації функцій сучасних ЦІМС потрібні тисячі ЛЕ, розташованих на одному кристалі. Зростання рівня інтеграції досягають вдос-

коналенням технології, яке дозволяє зменшити розміри й поліпшити параметри компонентів, і розвитком схемотехніки, в результаті чого розробляють нові види елементів, що мають менші споживану потужність та займану площу, вищу швидкодію.

Умову (3.1) найпростіше виконати для елементів типу КМОН, для яких споживана потужність $P = 0$. З інших типів елементів найменші значення P мають елементи I^2L . Умову (3.2) найпростіше виконати для елементів I^2L і МОН з каналом n -типу, що займають найменшу площу. Тому найвищого рівня інтеграції $K_u = 4 \dots 5$ досягають в НВІМС, що їх використовують як елементну основу I^2L та МОН з каналом n -типу. Елементи ТТЛ (також з діодами Шотткі) і ЕЗЛ, що мають більш високі значення P і $A_{кр}$, але меншу величину затримки сигналів t_3 , служать елементною основою швидкодіє-них ЦІМС з меншим рівнем інтеграції ($K_u \approx 3$). Для створення швидкодіє-них ЦІМС з рівнем інтеграції $K_u > 3$ потрібно збільшувати швидкодію елементів I^2L , n -МОН, КМОН і зменшувати споживану потужність і займану площу елементів ТТЛ, ЕЗЛ.

Одним з перспективних способів підвищення швидкодії й зниження споживаної потужності елементів на біполярних транзисторах є зменшення перепаду логічних рівнів сигналу U_L і напруги живлення. Однак при цьому знижується завадостійкість ЦІМС. Тому доцільно використовувати сигнали з малим перепадом U_L всередині мікросхем, де завади відносно малі, а сигнали з підвищеним перепадом U_L – для передачі інформації по зовнішніх колах, де завади значно сильніші. Насправді для відтворення двох рівнів сигнального перепаду в ВІМС (НВІМС) використовуються елементи трьох типів: 1) з малим логічним перепадом U_L у внутрішній структурі ВІМС; 2) вхідні буферні елементи з відносно високим порогом перемикавання, які забезпечують необхідну завадостійкість у зовнішніх колах; 3) потужні вихідні буферні елементи, що формують у зовнішніх колах достатньо великий сигнальний перепад U_L .

В ІМС типу ТТЛ як вхідні транслятори використовують каскади схеми зі складним інвертором або їх спрощені варіанти (рис. 3.16, а, б, в), які займають меншу площу. Транслятор на рис. 3.16, а забезпечує меншу затримку перемикавання, ніж транслятор на рис. 3.16, б, але більш високий рівень $U^0 = U^+ + U_{ост1}$. Тому для перемикавання наступних каскадів з простим інвертором необхідно підвищити їх поріг шляхом вмикання діода D в емітерне коло (рис. 3.16, а). У трансляторі на рис. 3.16, в використовується горизонтальний або вертикальний p - n - p транзистор T_0 , ввімкнений за схемою з ЗК. Завдяки цьому вхідний струм $I_{вх}^0$ зменшується в $B'+1$ разів, де B' – коефіцієнт підсилення струму p - n - p транзистора, а струм $I_{вх}^1 = 0$. Діод D_0 служить для пропускання струму з бази транзистора T_1 при надходженні на вхід низького рівня U^0 , завдяки цьому прискорюються перемикавання ІМС.

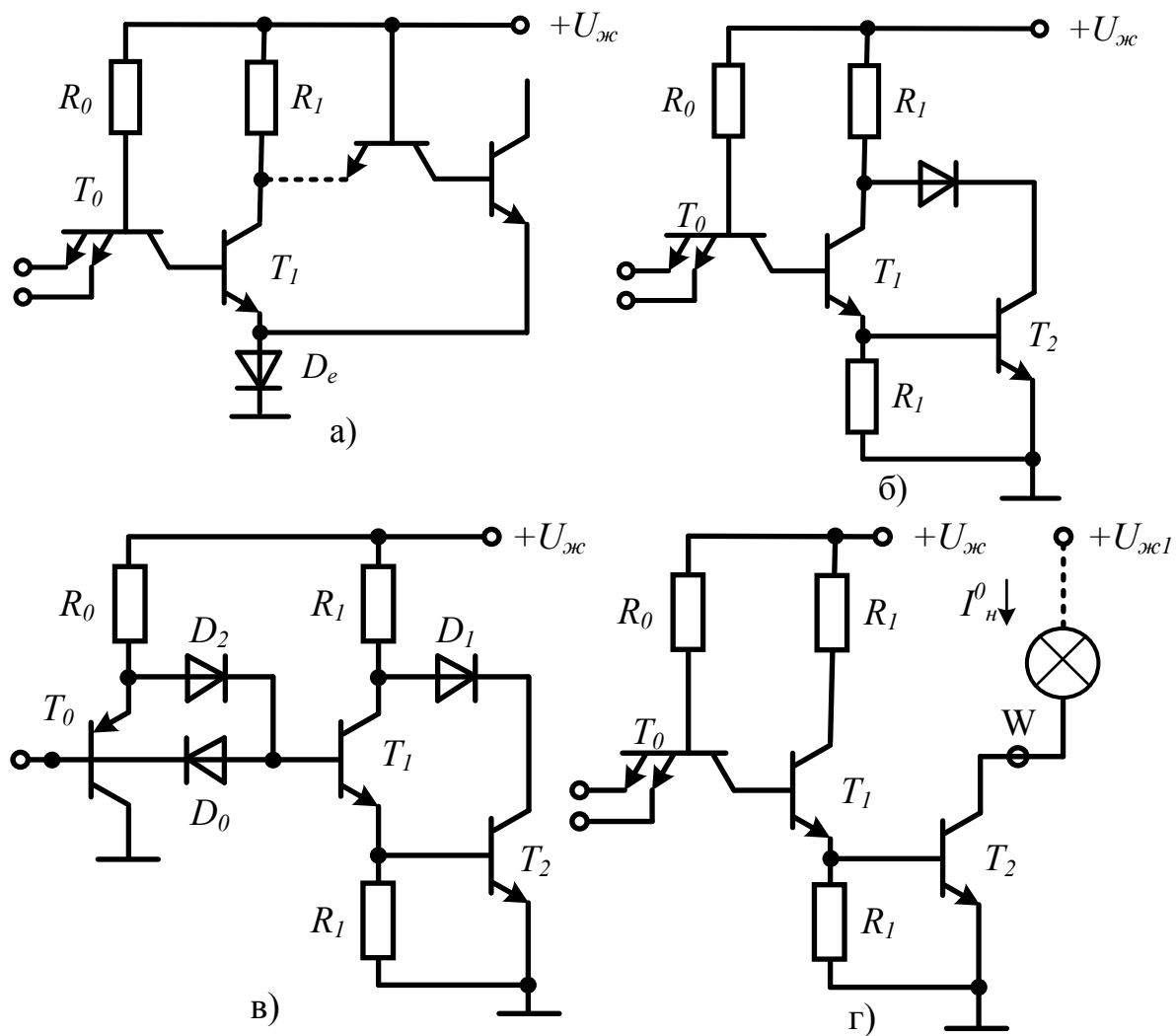


Рисунок 3.16 – Вхідні (а, б, в) та вихідний (г) транслятори ТТЛ-VIMC

Як вихідні транслятори в ІМС ТТЛ найчастіше використовують каскади зі складним інвертором, які забезпечують досить великий сигнальний перепад $U_n = 3$ В, значні вихідні струми $I_{n\max}^0$, $I_{n\min}^1$ і відносно високу швидкодію при великому ємнісному навантаженні C_n (десятки і сотні пФ). Якщо виходи декількох ІМС ТТЛ під'єднують до спільної зовнішньої лінії зв'язку (магістралі), то вихідні транслятори мають бути у третьому, «вимкненому» стані.

Якщо сигнал з виходу ІМС передають на деякий пристрій індикації: лампу, світлодіод, рідиннокристалічний індикатор і т. п., то як вихідний каскад ІМС використовують каскад ТТЛ з «відкритим колектором» (рис. 3.16, г). У стані $U_{вимк} = U$ замикається вихідне коло і вмикається пристрій індикації, наприклад, лампа. Якщо використовують високовольтні пристрої індикації (це, як правило, газонаповнені лампи), що працюють при напрузі $E_l = 50 \dots 100$ В, то вихідний транзистор T_2 має мати високу пробивну напругу $U_{пр.KE} > U_{ж1}$, що досягається завдяки відповідному проектуван-

ню його фізичної структури. Якщо ІМС забезпечує досить високе значення максимального вихідного струму $I_{н\max}^0$, то можна під'єднувати до її виходу світлодіоди, обмотки керування реле та інші виконавчі пристрої, які спрацьовують при протіканні струму.

Для узгодження параметрів елементів ТТЛ і І²Л на входах ІМС типу І²Л як транслятори вмикають буферні інвертори (рис. 3.17, а).

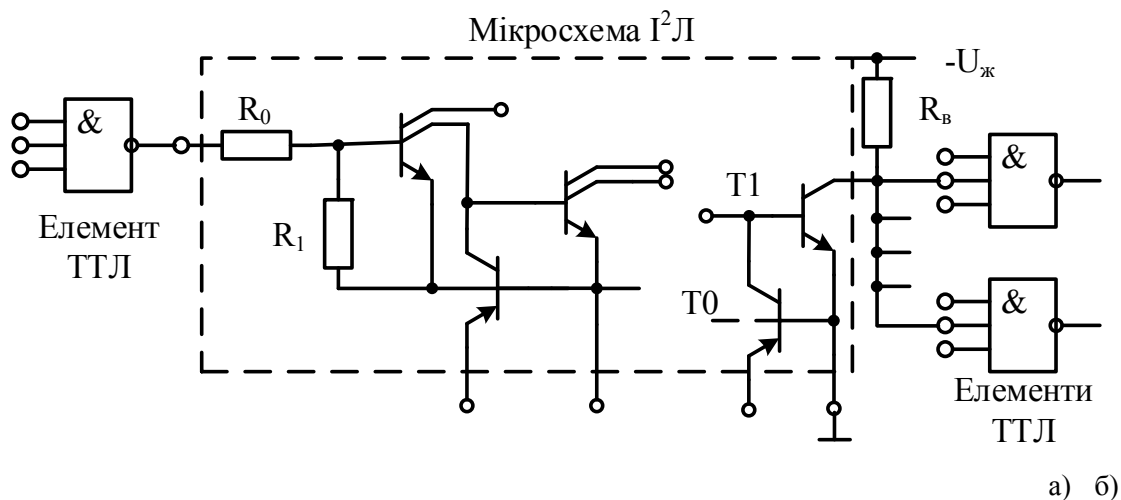


Рисунок 3.17 – Вхідний а) і вихідний б) транслятори І²Л

Опори резисторів R_0 , R_1 вибирають з умови відкриття транзистора Т1 при $U_{вх} = U_{вх}^1$. Вихідним транслятором служить елемент І²Л з під'єднаним зовнішнім резистором $R_в$ (рис. 3.17, б), опір якого вибирають з умови:

$$R_в = (U_ж - U_{вик}^1) / I_{н\max}^1 \quad (3.3)$$

де $U_{вик}^1$ – рівень «1», який потрібно забезпечити в зовнішньому колі; $I_{н\max}^1$ – максимальний струм.

Транзистори Т0 і Т1 вихідного транслятора мають мати достатні значення коефіцієнтів α і β , щоб забезпечувати виконання умови насичення (3.3) при струмі, $I = U_{ж1} / R_в + I_{н\max}^0$, де $I_{н\max}^0$ – максимальна величина струму, що надходить від навантаження при $U_{вих} = U_0$.

При використанні таких трансляторів ІМС типів І²Л і ТТЛ будуть сумісними за рівнями U^0 і U^1 , тобто, вихідні сигнали однієї з них можна безпосередньо використовувати для перемикавання іншої та навпаки. Тому ІМС І²Л і ТТЛ часто використовуються в цифрових пристроях разом чи одночасно.

Більшість трансляторів має істотно більші споживану потужність $P_{кр}$ і площу $A_{кр}$, ніж елементи внутрішньої структури ВІМС. При їх застосуванні розсіювана потужність і площа кристала визначаються як

$$P_{кр} = N_{ел} P_{ел} + n_в (a_p - 1), \quad (3.4)$$

$$A_{кр} = N_{ел}P_{ел} + n_B(a_A - 1). \quad (3.5)$$

де n_B – число логічних входів і виходів ІМС; значення $a_p = P_{кр}/P_{ел}$, $a_A = A_{кр}/P_{ел}$ досягають 3–5. Для зменшення $P_{кр}$ і $A_{кр}$ при проектуванні ВІС прагнуть забезпечити умову $n_B \ll N_{ел}$.

У ЦІМС на елементах МОН і КМОН можливість зниження споживаної потужності обмежується значеннями порогової напруги $U_{ж} = (2-3)U_0$. У таких ІМС зниження $U_{ж}$ і відповідно перепаду $U_{л}$ не приводить до підвищення швидкодії, тому що, поряд із зменшенням перепаду напруги $\Delta U_0 U \cong E$, на паразитній ємності $C_{п}$ зростають постійні часу її перезарядження $\tau \approx 1/b(U_{ж} - U_0)$, де b – питома крутизна МОП-транзисторів. Тому в ВІМС і НВІМС на МОН транзисторах всі електронні елементи звичайно мають однакову напругу живлення E . Вихідні елементи таких ІМС реалізовані за схемами зі складними інверторами, а вхідні елементи – як підсилювачі-інвертори на біполярних транзисторах.

Для внутрішніх елементів НВІМС з метою забезпечення високої щільності їх розміщення застосовують прості схеми без складних вихідних каскадів, оскільки такі елементи працюють при відомих заздалегідь і фіксованих навантаженнях. Для зв'язку з зовнішнім навантаженням у складі НВІС звичайно є спеціальні буферні елементи, що забезпечують вищу вихідну потужність.

3.5 Контрольні запитання і завдання

1. Основні принципи роботи елементів чи ЕЗЛ типу?
2. Чим пояснити високу швидкодію елементів чи ЕЗЛ типу?
3. Які основні частини можна виділити у схемі базового елемента чи ЕЗЛ типу?
4. Провести аналіз статичних характеристик елементів чи ЕЗЛ типу.
5. Принцип роботи елементів МОН- і КМОН-типів?
6. Яка схема є базовою для логічних елементів МОН-типу? Які відомі варіанти її виконання?
7. В чому переваги елементів КМОН-типу порівняно з іншими типами елементів?
8. Принципи роботи елементів I^2L -типу?
9. Переваги та недоліки логічних елементів I^2L -типу?
10. Проведіть порівняльний аналіз логічних елементів за основними експлуатаційними характеристиками.
11. Призначення основних компонент в базовому елементі I^2L .
12. За рахунок чого усунуто перехоплення струму в навантаженні у I^2L ?
13. Як підрахувати ступінь насичення транзистора в I^2L ?
14. Компоненти базової схеми інвертора I^2L .
15. Побудуйте схему ЛЕ АБО-НІ типу I^2L .

16. Як визначається СІ для елементів типу I^2L ?
17. Як визначається потужність I^2L ?
18. Призначення основних компонент в базовому елементі ЕЗЛ.
19. Чому виникає необхідність використання негативної напруги живлення в ЕЗЛ?
20. Чому використовують окреме живлення диференціального каскаду та емітерного повторювача в ЕЗЛ?
21. Визначте порогову напругу ЕЗЛ для додатного живлення.
22. Побудуйте схему джерела опорної напруги ЕЗЛ.
23. Чим викликана необхідність розміщення емітерних повторювачів на входах елементів ЕЗЛ?
24. Поясніть основну ідею побудови «деревоподібних» ЛЕ типу ЕЗЛ.
25. Вкажіть причини високої швидкодії ЕЗЛ.
26. Наведіть основну схему ЛЕ на МОН-транзисторах.
27. Проведіть порівняльний аналіз характеристик ЛЕ різних схемно-технологічних типів.
28. Чому в ЛЕ типу МОН базовою логічною функцією є І-НІ?
29. Вкажіть зв'язок часу перемикання МОН ЛЕ з коефіцієнтом розгалуження за виходом.
30. Вкажіть зв'язок часу перемикання МОН ЛЕ з коефіцієнтом об'єднання за входом.
31. Основні схеми комбінування МОН ЛЕ.
32. Вимоги до значення питомої крутості навантаженого транзистора МОН ЛЕ І-НІ?
33. Вимоги до значення питомої крутості навантаженого транзистора МОН ЛЕ АБО-НІ?
34. Основні схемні варіанти КМОН ЛЕ І-НІ.
35. Основні схемні варіанти КМОН ЛЕ АБО-НІ.
36. Основні схемні варіанти комбінованого КМОН ЛЕ.
37. Вимоги до значення питомої крутості навантаженого транзистора комбінованого МОН ЛЕ.
38. Вкажіть обмеження на час перемикання КМОН ЛЕ.
39. Вкажіть значення порогової напруги КМОН ЛЕ.
40. Вкажіть причини, які визначають вибір значення напруги живлення в МОН ЛЕ.
41. Які чинники треба враховувати при визначенні завадостійкості МОН ЛЕ?

4 ПЕРЕТВОРЮВАЧІ КОДІВ І КОМБІНАЦІЙНІ ФУНКЦІОНАЛЬНІ ПРИСТРОЇ

Перелік скорочень і умовних позначень до розділу 4

КП – комірка пам'яті; ПЗП – постійний запам'ятовувальний пристрій;
УГП – умовне графічне позначення.

У «комп'ютерній інженерії» взагалі і в інженерних жаргонах зокрема термін «код» і похідні від нього є надзвичайно широкоживаними і багатозначними. Лише перелік їх прикладів зайняв би декілька десятків сторінок. Однак більшості таких прикладів бракує лексичної точності і коректності. У цифровій техніці взагалі використовують кілька способів подання інформації. Кодування – це перетворення повідомлень у кодові комбінації. В багатьох випадках термін «код» і похідні від нього є «локально зручними лжесинонімами» інших, менш поширених, але й більш точних понять, або ж є інформаційно надлишковими. Досить часто кодом називають результат виконання якоїсь операції (не обов'язково кодування) або використовують його як заміник слів з явно вираженим функціональним наповненням, наприклад, число, операнд, стан, слово тощо. Однак не можна називати кодом, наприклад, одномоментний зріз стану декількох пристроїв комп'ютера при перериванні програм, адже для цього є інший, більш точний і повний термін – «слово стану програми». Ще декілька прикладів некоректних виразів: «суматор додає коди А і В» –але ж А і В мають точнішу власну назву «операнди»; записувати до регістра адреси пам'яті потрібно не «код базової адреси», а власне «базову адресу»; лічильник формує не «код поточної адреси», а поточну адресу і т. п. Тому далі «код» та «кодування» будемо розуміти так, як це пропонується в підрозділі 1.1, а саме: це спосіб (чи система) встановлення відповідності між елементами двох повідомлень. Для дослідження особливостей проектування перетворювачів кодів їх доцільно розділити на класи за виглядом основних відтворюваних функцій. Найчастіше перетворювачі кодів змінюють спосіб кодування вхідних даних, не змінюючи при цьому їх кількісних еквівалентів. Наприклад, вхідні і вихідні пристрої комп'ютерів оперують із звичними для людини десятковими числами, обчислення ведуться за правилами двійкової системи числення, а проміжним способом подання чисел служить двійково-десятковий код. Для виконання багатьох операцій з урахуванням знаків чисел використовують так звані прямий, обернений та доповняльний коди, де знаки операндів закодовані цифрами. В практичних випадках використовують й інші подання числової інформації, які утворюють відповідним перетворенням кодів. У широкому розумінні слова до перетворювачів кодів можна віднести багато цифрових пристроїв, зокрема шифратори, дешифратори, мультиплексори, селектори, пристрої визначення та форму-

вання пріоритету, пристрої порівняння, пристрої згортання тощо, однак за традицією ці пристрої розглядають як окремі класи цифрових структур.

4.1 Перетворення прямого коду двійкових чисел в інверсний та обернений коди

Загальноприйнято кодувати знаки чисел двійковими символами на спеціально виділених позиціях в запису числа – знакових розрядах. При цьому цифра 0 в знаковому розряді означає «+», а цифра 1 там же означає «-». Далі будемо позначати знакові розряди операндів $A, B, C, \dots$ як $A_0, B_0, C_0, \dots$. В друкованих текстових документах, які містять двійкові числа зі знаками, знаковий розряд звичайно розміщують зліва від старших розрядів числа та відділяють його точкою від цілої частини числа або комою, якщо число – правильний дріб. Запис числа A в прямому коді (прямим кодом) означає його перетворення за правилом

$$A_{ПК} = \begin{cases} 0 |A|, & \text{якщо } A \geq 0, \\ 1 |A|, & \text{якщо } A < 0. \end{cases}$$

Це означає, що прямі коди чисел $+A$ та $-A$ відрізняються лише знаковими розрядами. З наведеного виразу випливає, що в прямому коді нуль можна формально подати двома способами: «додатний» нуль « $+0$ » $\rightarrow 0.00\dots 0$ і «від’ємний» нуль « -0 » $\rightarrow x_0.00\dots 0$ (у випадку двійкової системи числення «від’ємний» нуль зображується як $1.00\dots 0$).

Інверсне кодування деякого двійкового операнда означає заміну всіх цифр вихідного числа на інверсні (тобто «1» на «0» і «0» на «1»). Частіше таке перетворення виконується над абсолютною величиною заданого операнда, проте можливі випадки інверсного перетворення всіх розрядів операнда, охоплюючи й знаковий розряд.

Зображення числа A оберненим кодом визначається як

$$A_{OK} = \begin{cases} 0 |A|, & \text{якщо } A \geq 0, \\ 1 \cdot 2^{s+1} - 2^{-m} - |A|, & \text{якщо } X \leq A. \end{cases}$$

Тут s – номер (позиція) старшого ненульового розряду в двійковому поданні цілої частини числа A , рахуючи як 0, 1, 2, ... справа наліво від положення коми, m – позиція молодшого розряду в двійковому поданні дробової частини числа A , рахуючи як 1, 2, ... зліва направо від положення коми. Зауважимо, що назва коду походить від математичних понять «обернена величина». «обернена функція», тому інші назви, що ще зрідка зустрічаються в неякісній технічній літературі (як от: «оборотний код», «зворотний код», «перевернений код» тощо), є некоректними.

Порівнюючи вирази для $A_{ПК}$ і $A_{ОК}$ легко помітити, що обернений код додатного числа збігається з його прямим кодом. Оскільки

$$2^{s+1} - 2^{-m} = \underbrace{11 \dots 11}_{s+1} , \underbrace{11 \dots 11}_m$$

то для одержання оберненого коду від'ємного числа у двійковій системі числення необхідно в знаковому розряді записати 1, а в інших розрядах одиниці замінити нулями і нулі – одиницями. Для одержання оберненого коду від'ємного числа в системі числення з основою $k > 2$ необхідно в знаковий розряд записати цифру $k-1$, а в інших розрядах замінити цифри їх доповненнями до $k-1$. Наприклад, у десятковій системі цифру 9 потрібно замінити на 0, цифру 8 – на 1, цифру 7 – на 2 і т. д. На відміну від прямого коду, у оберненому коді не можна відкидати нулі після знакового розряду в цілій частині від'ємного числа, переміщуючи при цьому знаковий розряд вправо. Не можна відкидати нулі і в кінці дробової частини від'ємного числа, але можна відкидати записані на цих місцях цифри $k-1$. Нуль у оберненому коді, як і в прямому коді, має два зображення: «додатний» нуль $0.00 \dots 0$ і «від'ємний» нуль $1.11 \dots 1$.

Один з найпростіших перетворювачів кодів, які мають задеклароване в назві підрозділу призначення, може бути реалізований за схемою, зображеною на рис. 4.1, а. Тут в кожному розряді виконують додавання за модулем 2 цифри даного розряду і сигналу управління Упр. При цьому нульове значення сигналу Упр забезпечує проходження на вихід абсолютної величини операнда, а при одиничному значенні сигналу Упр кожна цифра вихідного слова буде інверсією відповідної вхідної цифри. У ситуації, коли як сигнал Упр використовують знаковий розряд, цей пристрій буде виконувати перетворення абсолютної величини двійкового числа в обернений код.

4.2 Перетворення прямого коду двійкового числа в доповняльний код

Назва «доповняльний код» походить від математичного поняття «доповнення» деякої множини чи числа до певного значення, тому термін «додатковий код», або подібні до нього, є некоректним.

Доповняльний код операнда A утворюють відповідно до правила

$$A_{ДК} = \begin{cases} 0 |A|, & \text{якщо } A \geq 0, \\ 1 \ 2^{s+1} - |A|, & \text{якщо } A < 0. \end{cases}$$

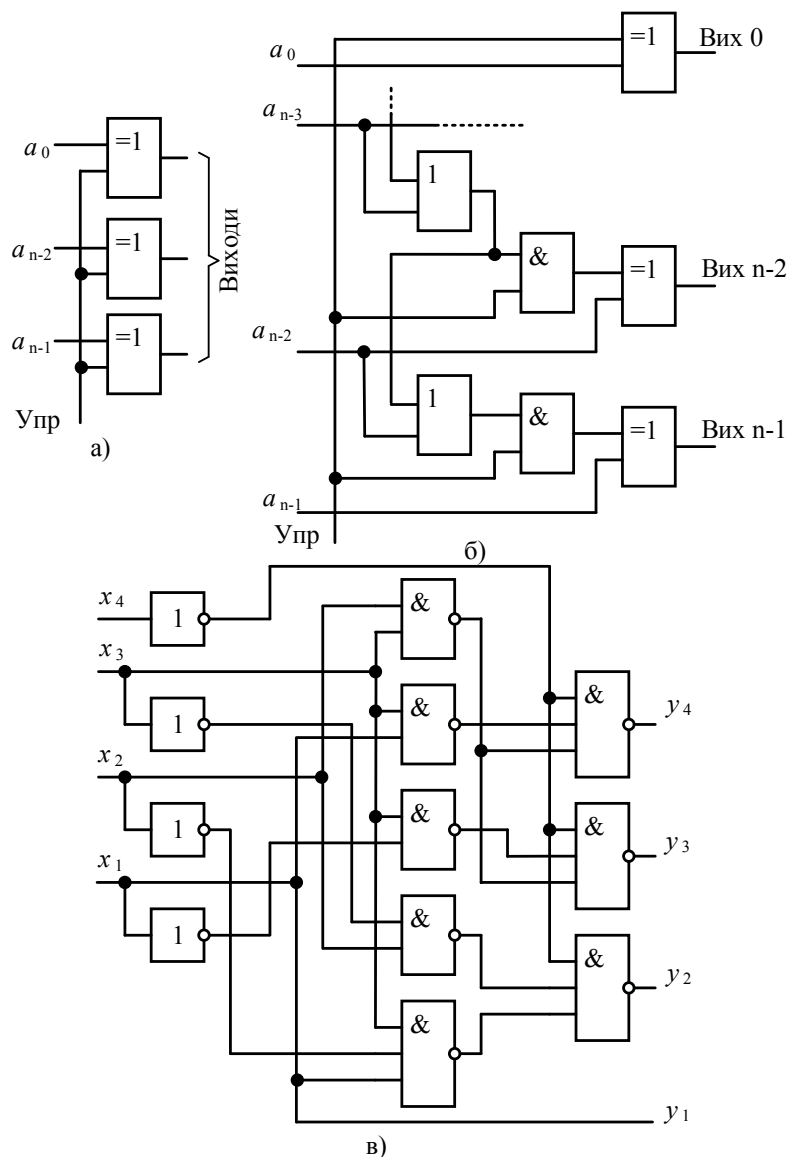


Рисунок 4.1 – Схеми перетворювачів прямого коду в інверсний а) та доповняльний б), а також перетворювач типу «8421» → «2421 в)

Зображення додатного числа в доповняльному коді збігається з його прямим кодом, а також і з його оберненим кодом. Порівнюючи другі рядки виразів для A_{OK} і $A_{ДК}$ легко помітити, що доповняльний код від'ємного числа відрізняється від оберненого коду того ж числа на величину 2^m . На цьому ґрунтується правило формування доповняльного коду від'ємного числа – достатньо одержати його обернений код і додати одиницю в його молодший розряд (рис. 4.1, б). Для двійкової системи числення той же результат можна отримати, якщо в прямому коді від'ємного числа залишити без зміни крайні справа нулі і першу одиницю за ними, а всі цифри лівіше цієї одиниці поміняти на інверсні (крім знакового розряду). Перетворення ж доповняльного коду від'ємного числа в прямий код може бути здійснено або зворотним шляхом (тобто, відніманням одиниці в молодшому розряді і наступним перетворенням оберненого коду), або шляхом формування до-

повняльного коду від доповняльного. Обидва шляхи приводять до однако-
вого результату. Нуль у доповняльному коді має єдине зображення $0.00...0$
(тобто, тут існує тільки «додатний» нуль). Комбінація ж $1.00...0$ є забороненою для доповняльного коду і її поява може свідчити про те, що резуль-
тат деякої операції не вміщується у відведених для цього розрядах (цю си-
туацію називають переповненням розрядної сітки).

Перетворення прямого коду в доповняльний не є порозрядною опера-
цією, і для визначення вихідної цифри в будь-якому розряді потрібен ана-
ліз цифр інших розрядів. Порівняння прямого та доповняльного кодів
від’ємних чисел показує, що доповняльний код відрізняється від прямого
коду інверсією старших розрядів від $n-1$ -го до $i+1$ -го включно, де i – номер
першого праворуч розряду, що містить одиницю. Решта розрядів залиша-
ються незмінними. Наприклад, прямий код цілочислового операнда «-
0100100» має вигляд 1.0100100, а доповняльний код того ж операнда –
1.1011100. Отже, правило формування цифр i -го розряду в загальному ви-
падку можна записати як

$$a_{\text{вих } i} = a_{\text{вх } i} \oplus (a_{\text{вх } i-1} + a_{\text{вх } i-2} + \dots + a_{\text{вх } 0}).$$

Для отримання i -ї цифри доповняльного коду від’ємного операнда A
потрібно до вхідної цифри цього розряду $a_{\text{вх } i}$ додати за модулем 2 ди-
з’юнкцію всіх попередніх (молодших) розрядів. Знаковий розряд вхідного
операнда може використовуватися як сигнал управління (див. рис. 4.1, б).

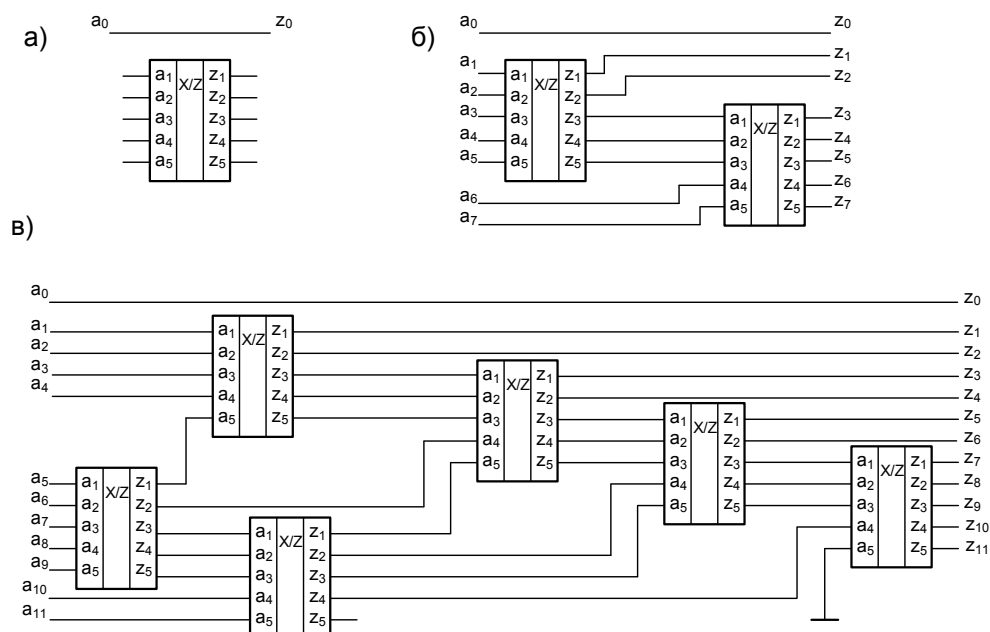


Рисунок 4.2 – Умовне позначення перетворювача кодів а) і схеми перетворювачів для двох б) і трьох в) тетрад двійково-десятькового коду

Наявність ВІМС із внутрішньою структурою постійних запам'ятовувальних пристроїв (ПЗП) в багатьох випадках дозволяє створювати перетворювачі кодів табличним методом. Можливість цього для перетворення в коді «8, 4, 2, 1» двійково-десяткових чисел довжиною в півтори тетради пояснює табл. 4.1.

Таблиця 4.1 – Перетворення двійково-десяткових кодів

Десяткові числа	Входи	Виходи
	a ₅ a ₄ a ₃ a ₂ a ₁ a ₀	z ₅ z ₄ z ₃ z ₂ z ₁ z ₀
0 або 1	0 0 0 0 0/1	0 0 0 0 0 0/1
2 або 3	0 0 0 0 1 0/1	0 0 0 0 1 0/1
4 або 5	0 0 0 1 0 0/1	0 0 0 1 0 0/1
6 або 7	0 0 0 1 1 0/1	0 0 0 1 1 0/1
8 або 9	0 0 1 0 0 0/1	0 0 1 0 0 0/1
10 або 11	0 1 0 0 0 0/1	0 0 1 0 1 0/1
12 або 13	0 1 0 0 1 0/1	0 0 1 1 0 0/1
14 або 15	0 1 0 1 0 0/1	0 0 1 1 1 0/1
....		
38 або 39	1 1 1 0 0 0/1	1 0 0 1 1 0/1
Будь-які	X X X X X X	1 1 1 1 1 1

Зауважимо, що молодші розряди двійкових і двійково-десяткових зображень чисел фактично свідчать про парність чи непарність цих чисел і тому однакові для обох кодів, внаслідок цього в таблицю не внесені. Умовне позначення перетворювача кодів показано на рис. 4.2, а. Для перетворення кодів чисел більшої розрядності шляхом каскадування складають пристрої з розглянутих шестирозрядних перетворювачів. Пристрої для перетворення двох і трьох тетрад двійково-десятьового коду (див. рис. 4.2, б, в) містять відповідно 2 і 6 мікросхем ПЗП і мають логічну глибину 2 і 5. Для аналізу роботи пристрою за схемою на рис. 4.2 потрібно використати таблицю його функціонування, яка, власне, і визначає вміст ПЗП. Вище, для компактності викладу, була наведена лише частина такої таблиці для шестирозрядного перетворювача, проте при необхідності можна легко отримати будь-який рядок цієї таблиці, керуючись такою закономірністю: в п'яти перших рядках таблиці виходи збігаються зі входами; в п'яти наступних рядках вихідне слово є вхідним, від якого за двійковими правилами потрібно відняти 3; в п'яти наступних рядках для отримання вихідного слова потрібно за двійковими правилами відняти 6 від вхідного; нарешті, в п'яти останніх рядках вихідне слово дорівнює вхідному, від якого потрібно відняти 9. Наприклад, число 55 в двійково-десятьовому коді виражається словом 01010101, а в двійковому коді – словом 00110111. На рис. 4.3 показані сигнали, які з'являються на виходах і входах шестирозрядних перетворювачів при подачі на вхід двійково-десятьового коду числа 55, що свід-

чить про правильність функціонування пристрою. За допомогою каскадування будують і перетворювачі більшої розрядності. Наприклад, для перетворювача шести тетрад потрібно 28 мікросхем ПЗП, а логічна глибина пристрою дорівнює 13.

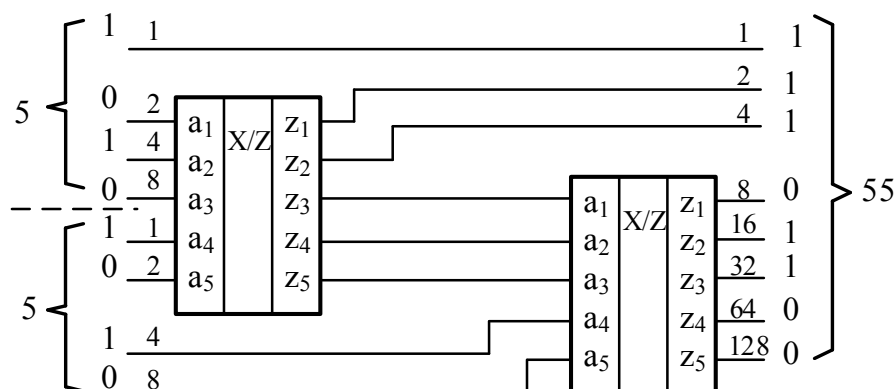


Рисунок 4.3 – Приклад перетворення конкретного коду

Вище показано, що для перетворювача двійково-десятькового коду в двійковий в межах півтори тетради виявляються надлишковими (незадіяними) три виходи ПЗП. Це дозволяє додатково реалізувати на тій же ІМС перетворення двійково-десятькових кодів в обернені або в доповняльні коди в межах однієї тетради. Здавалося б, що відсутність одного виходу не дозволяє такого використання ІМС, однак в таблиці функціонування в обох випадках один з розрядів виявляється однаковим для вхідних і вихідних кодів (табл. 4.2).

Таблиця 4.2 – Перетворення двійково-десятькового коду в коди «доповнення до 9» та «доповнення до 10»

Двійково-десятькове слово	Вхідний код				Доповнення до 9				Доповнення до 10			
	a_4	a_3	a_2	a_1	z_8	z_7	z_6		z_8	z_7	z_6	
0	0	0	0	0	1	0	0	1	0	0	0	0
1	0	0	0	1	1	0	0	0	1	0	0	1
2	0	0	1	0	0	1	1	1	1	0	0	0
3	0	0	1	1	0	1	1	0	0	1	1	1
4	0	1	0	0	0	1	0	1	0	1	1	0
5	0	1	0	1	0	1	0	0	0	1	0	1
6	0	1	1	0	0	0	1	1	0	1	0	0
7	0	1	1	1	0	0	1	0	0	0	1	1
8	1	0	0	0	0	0	0	1	0	0	1	0
9	1	0	0	1	0	0	0	0	0	0	0	1

При перетворенні двійково-десятькового коду в код «доповнення до дев'яти» однаковими для вхідного і вихідного кодів виявляються другі розряди праворуч, а при перетворенні в код «доповнення до десяти» – перші розряди. Це дозволяє застосувати пристрої за схемами на рис. 4.4, де програмування потребує всього три виходи, що не були зайняті в перетворювачі шестирозрядного двійково-десятькового коду в двійковий. Таблиця програмування ПЗП визначається графами, позначеними в табл. 4.2. символами виходів Z_6 , Z_7 , Z_8 .

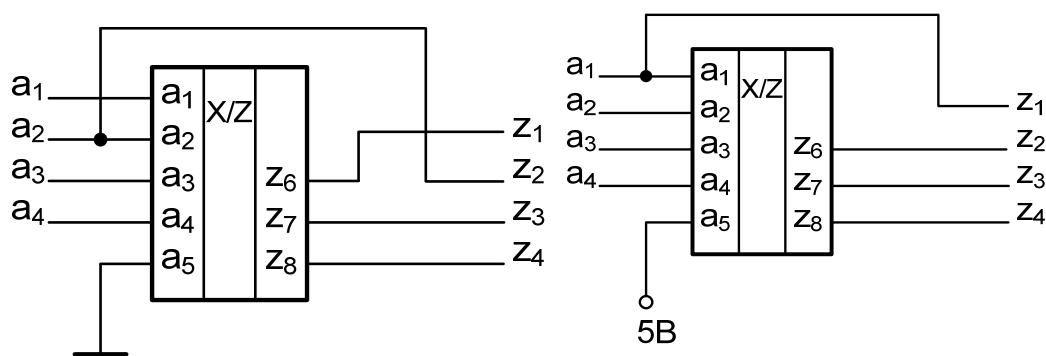


Рисунок 4.4 – Схеми перетворювачів кодів з додатковими можливостями на основі ПЗП

При роботі в режимах перетворення двійково-десятькового коду на доповнення до 9 або до 10 вхід a_5 використовується як сигнал управління. Якщо на нього подати нуль, то отримаємо вхідний набір, за яким буде запрограмований один з типів перетворення (наприклад, на доповнення до 9), а якщо подати одиницю, то отримаємо вхідний набір, за яким буде запрограмований інший тип перетворення (рис. 4.4).

На ПЗП можна реалізувати і перетворювачі двійкових кодів в двійково-десятькові. При цьому на вході можна задавати шестирозрядні коди (п'ять входів має ПЗП, а молодший розряд – спільний для обох кодів і не потребує перетворення) і отримати на виході двійково-десятьковий код в межах півтори тетради. Для перетворення слів більшої розрядності використовують каскадування декількох шестирозрядних перетворювачів.

4.3 Різновиди перетворювачів кодів

Крім вже розглянутих вище перетворювачів кодів, в «комп'ютерній інженерії» широко застосовують й інші їх функціональні типи, як от: перетворювачі двійкових кодів з природним порядком ваги в коди зі штучним порядком ваги, наприклад, двійково-десятькові коди в 2421, код Грея (або рефлексний), доповнення до 9 (10), 5421, 7421, код Джонсона (табл. 4.3.) та інші.

Особливість коду Грея полягає в тому, що при переході до кожної наступної кодової комбінації в коді змінюється значення тільки одного двійкового розряду. У коді Джонсона перехід до наступної кодової комбінації здійснюється шляхом послідовної заміни нулів на одиниці, починаючи справа, до заповнення всіх розрядів одиницями, а потім заміною одиниць на нулі, до заповнення всіх розрядів нулями. Коди з доповненням до 9, 10 з «надлишком 3» та з «надлишком 9» використовують для додавання і віднімання двійково-десяткових чисел.

Взагалі для побудови будь-яких перетворювачів кодів користуються одним з двох методів: 1) перетворення вихідного двійкового коду в десятиковий і подальше перетворення десятикового подання в необхідний двійковий код; 2) синтез логічного пристрою комбінаційного типу, що безпосередньо реалізує потрібне перетворення. Розглянемо побудову перетворювача коду 8421 в код 2421 на основі другого методу. У табл. 4.3 наведено відповідність комбінацій обох кодів.

Таблиця 4.3 – Подання десятикових цифр в деяких двійково-десяткових кодах

Десяткові цифри	8421	7421	5421	Код Айкена 2421	Код Грея	Код з надлишком 3 N+3	Доповнення до 9 9-N	Код Джонсона	Доповнення до 10 10-N
0	0000	0000	0000	0000	0000	0011	1001	00000	1010
1	0001	0001	0001	0001	0001	0100	1000	10000	1001
2	0010	0010	0010	0010	0011	0101	0111	11000	1000
3	0011	0011	0011	0011	0010	0110	0110	11100	0111
4	0100	0100	0100	0100	0110	0111	0101	11110	0110
5	0101	0101	1000	1011	0111	1000	0100	11111	0101
6	0110	0110	1001	1100	0101	1001	0011	01111	0100
7	0111	1000	1010	1101	0100	1010	0010	00111	0011
8	1000	1001	1011	1110	1100	1011	0001	00011	0010
9	1001	1010	1100	1111	1101	1100	0000	00001	0001

Кожну зі змінних y_8, y_4, y_2, y_1 можна розглядати як функцію аргументів x_8, x_4, x_2, x_1 і, отже, можна подати в ДДНФ, яку, в свою чергу, легко мінімізувати за допомогою діаграм Вейча (рис. 4.5), наведена логічна схема перетворювача кодів 8421 в 2421 на елементах І-НІ. (див. рис. 4.1, в).

4.4 Дешифратори і шифратори

Дешифратори і шифратори насправді теж належать до числа перетворювачів кодів.

Дешифратори – пристрої, призначені для перетворення двійкового n -розрядного коду в комбінацію сигналів управління.

Двійкові дешифратори перетворюють двійковий код операнда в код « 1 з N », де $N=2^n$ – число виходів дешифратора, n – число його функціональних входів. Інакше, залежно від вхідного слова, збуджується одиничним сигналом тільки один з його виходів. В технічній документації для умовних позначень дешифраторів і шифраторів використовують букви DC і CD (від decoder і coder відповідно). Оскільки за допомогою n -розрядного двійкового коду можна задати 2^n кодових комбінацій, то число виходів повного дешифратора дорівнює 2^n . Якщо частина можливих вхідних наборів не використовується для кодування, то дешифратор називають неповним і для нього $N_{вих} < 2^n$.

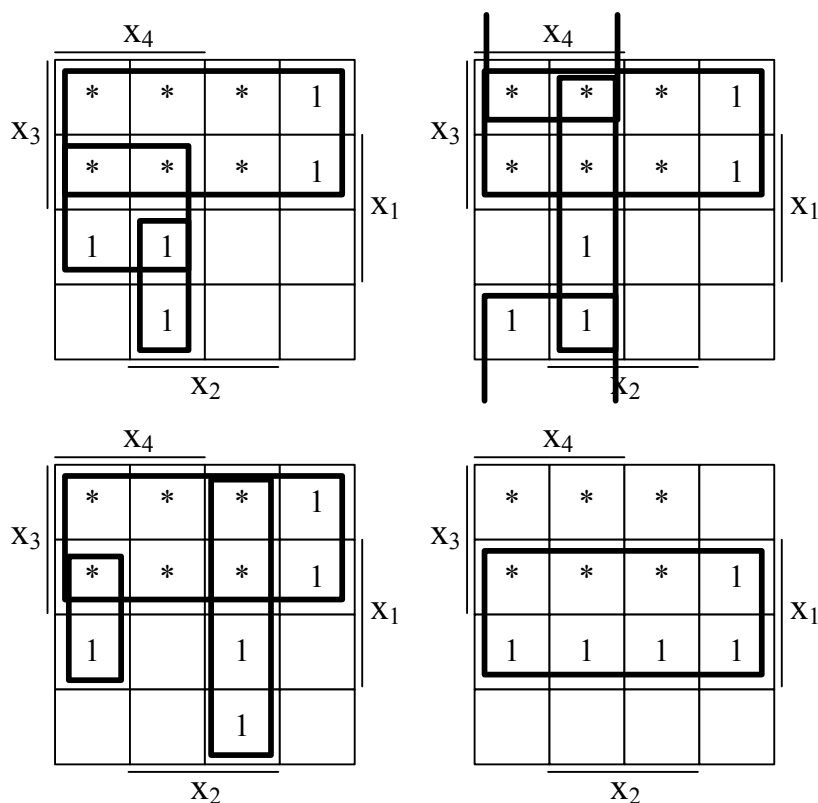


Рисунок 4.5 – Діаграми Вейча функцій перетворювача кодів 8421 в 2421

Функціонування дешифратора описується системою перемикальних функцій:

$$F_i = m_i;$$

$$i = \overline{0, 2^n - 1},$$

де m_i – мінтерми n вхідних змінних. Таким чином, можна дати ще одне означення повного дешифратора: це пристрій, який реалізує всі можливі мінтерми від n вхідних змінних. Залежно від розрядності вхідного слова і функціональних можливостей ЛЕ дешифратор можна виконати як однос-

ступеневий (або лінійний) чи багатоступеневий пристрій. В свою чергу, багатоступеневі дешифратори розділяють на прямокутні (матричні) і пірамідальні структури.

4.4.1 Лінійні дешифратори

Лінійні дешифратори насправді є наборами ЛЕ типу І, які за наявності вхідного слова формують одиничний сигнал тільки на одному з виходів, у той час як на інших виходах сигнал приймає нульове значення (рис. 4.6, 4.7). Функціонування лінійного дешифратора можна описати за допомогою таблиці істинності (табл. 4.4).

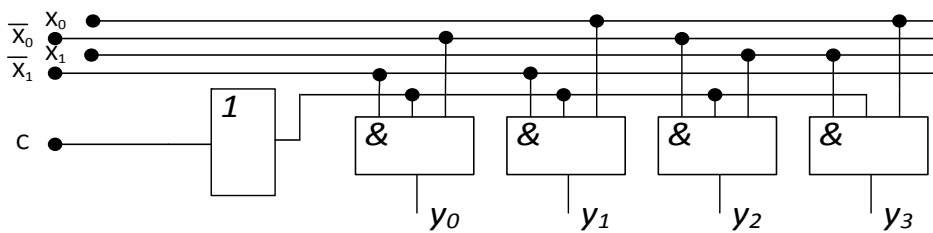


Рисунок 4.6 – Лінійний дешифратор на два входи

Таблиця 4.4 – Таблиця істинності лінійного дешифратора

x_n	x_{n-1}	...	x_2	x_1	y_0	y_1	y_2	...	y_{2^n-1}
0	0	...	0	0	1	0	0	...	0
0	0	...	0	1	0	1	0	...	0
0	0	...	1	0	0	0	1	...	0
...	...	...	...	...	...	...	...	...	...
1	1	...	1	1	0	0	0	...	1

або системи перемикальних функцій:

$$y_0 = \overline{x_n} \cdot \overline{x_{n-1}} \cdot \dots \cdot \overline{x_2} \cdot \overline{x_1},$$

$$y_1 = \overline{x_n} \cdot \overline{x_{n-1}} \cdot \dots \cdot \overline{x_2} \cdot x_1,$$

$$y_2 = \overline{x_n} \cdot \overline{x_{n-1}} \cdot \dots \cdot x_2 \cdot \overline{x_1},$$

.....

$$y_{2^n-1} = x_n x_{n-1} \cdot \dots \cdot x_2 x_1.$$

де $x_1, x_2, \dots, x_n$; $y_0, y_1, \dots, y_{2^n-1}$ – сигнали на входах і виходах дешифратора.

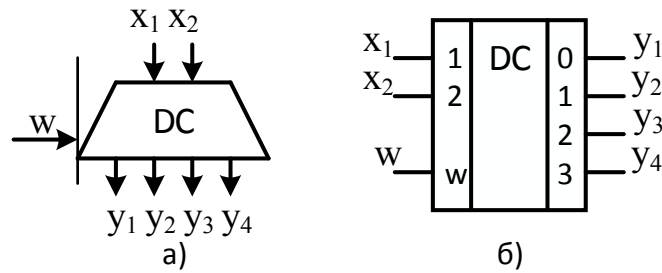


Рисунок 4.7 – УГП лінійного дешифратора: а) на функціональних схемах; б) на принципових схемах

Для реалізації лінійного дешифратора на n входів потрібно n ЛЕ типу І з n входами. Лінійні дешифратори найефективніші, коли розрядність вхідного слова не перевищує числа входів ЛЕ І типового структурного елемента ЦІМС. Швидкодія лінійних дешифраторів найбільша порівняно з іншими структурами дешифраторів, тому що час встановлення $t_{уст}$ одиничного сигналу на виході дешифратора дорівнює середній затримці $t_{з,ср}$ типового елемента ЦІМС ($t_{уст} = t_{з,ср}$). Основний недолік лінійного дешифратора – відносно велике навантаження на елементи, що є джерелами вхідних сигналів (найчастіше це тригери в складі регістрів), оскільки їх навантажувальна здатність має бути не меншою, ніж $m_{mp} = 2^{n-1}$.

Виробники ІМС з функціями дешифраторів часто передбачають реалізацію режиму стробування (тактування), який дозволяє формувати вихідні сигнали тільки в певні моменти часу, що визначаються стробувальними сигналами (стробами). Цього досягають введенням додаткового входу паралельно інформаційним входам в кожному ЛЕ дешифратора, або блокуванням всіх ЛЕ через один із входів дешифратора. В останньому випадку за нульового значення сигналу строба формуються нулі в колах прямих або інверсних значень змінних x_i , внаслідок цього буде наявним хоча б один нуль в наборах вхідних сигналів для кожного ЛЕ дешифратора. При цьому всі виходи приводяться до нульових значень, оскільки будь-яка змінна в прямому чи в інверсному вигляді надходить на всі ЛЕ дешифратора. При одиничному значенні строба відновлюються правильні (безінверсні) кола передачі змінних x_i на входи дешифратора. Описані варіанти називають також стробуванням за входом і виходом.

Час встановлення вихідного сигналу $t_y = \max(t_3^{10}, t_3^{01})$, оскільки при зміні стану дешифратора на одному з виходів одиничний сигнал з'являється, а на іншому – зникає.

При стробуванні за виходом затримка відносно початку дозвільного сигналу становить $t_y = t_3^{01}$, а при стробуванні за входом $t_y = 2t_3 + t_3^{01}$.

Навантаження дешифратора на джерело сигналів обчислюють з урахуванням того, що кожна вхідна змінна з однаковими ймовірностями може приймати пряме чи інверсне значення на вході кожного ЛЕ. Отже, до джерел сигналів a_i і $\bar{a}_i$ буде під'єднано $K_p = 2^{n-1}$ входів.

Одноступеневий дешифратор – найбільш швидкодієний, але його реалізація для великої розрядності вхідного слова ускладнена, оскільки вимагає застосування ЛЕ з великим числом входів (досягає $n+1$ для лінійних дешифраторів зі стробуванням за виходом) і супроводжується великим навантаженням на джерела вхідних сигналів. Одноступеневі дешифратори будують на невелике число входів, яке визначається можливостями ЛЕ застосовуваної серії мікросхем.

Розміщення дешифраторів в одному корпусі ЦІМС обмежується також великим числом зовнішніх виводів, головним чином інформаційних. Наприклад, лінійний дешифратор на p вхідів має мати не менше 40 зовнішніх виводів (для 5 інформаційних входів, 32 виходів, входу стробування і, як мінімум, 2 для кіл живлення і загальної точки) за умови, що інверсії аргументів формуються всередині ІМС. Для подолання відзначених недоліків в дешифраторах на велике число входів використовують багатоступеневі їх структури.

4.4.2 Прямокутні дешифратори

У прямокутному дешифраторі n його входів розбивають на дві групи по $n/2$ змінних в кожній групі для парного n , для непарного n групи повинні містити по $(n+1)/2$ і $(n-1)/2$ змінних (рис. 4.8).

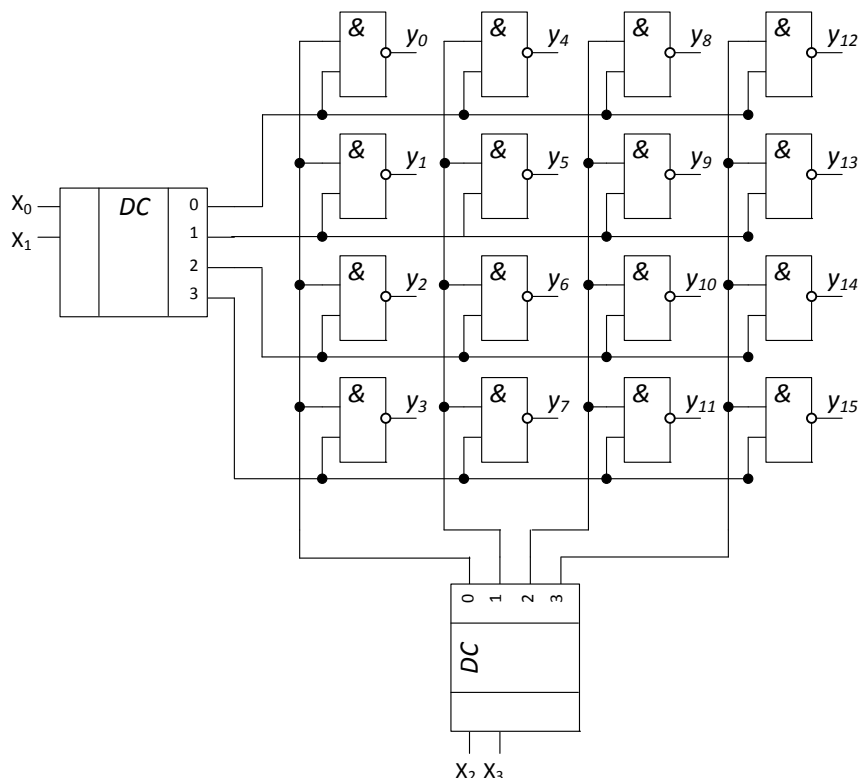


Рисунок 4.8 – Прямокутний дешифратор на 4 входи

Для кожної з цих груп будують лінійний дешифратор. Ці дешифратори є першим ступенем дешифрування. Потім за допомогою ЛЕ І на два входи кожне вихідне коло одного лінійного дешифратора першого ступеня об'єднують з кожним вихідним колом іншого лінійного дешифратора першого ступеня і в такий спосіб утворюють другий ступінь дешифрації.

В прямокутному (матричному) двоступеневому дешифраторі загальна кількість ЛЕ І визначається як $\sum M = 2^n + 2 \cdot 2^{n/2}$ для парного n і як $\sum M = 2^{(n-1)/2} + 2^{(n+1)/2} + 2^n$ для непарного n . Коли $n > 4$, то прямокутні дешифратори приблизно в 2 рази економічніші за пірамідальні (інформація в наступному пункті). Однак слід відзначити, що до навантажувальної здатності ІМС прикінцевого ступеня прямокутних дешифраторів висуваються досить жорсткі вимоги, а саме: $m_{tp} = 2^{n/2}$.

4.4.3 Пірамідальні дешифратори

Пірамідальні дешифратори на n входів мають $K = n - 1$ ступенів, причому на кожному ступені використовують тільки двовходові ЛЕ І (рис. 4.9). Кількість ЛЕ І на i -му ступені становить 2^{i+1} , де i – номер ступеня. Загальна кількість ЛЕ І для пірамідального дешифратора визначається так:

$$N = \sum_{i=1}^K 2^{i+1}.$$

Недолік пірамідальних дешифраторів: різні входи дешифраторів утворюють нерівномірне навантаження на елементи регістрів, а багатоступеневість дешифратора знижує його швидкодію $t_{ycm} = K \cdot t_{з.ср}$.

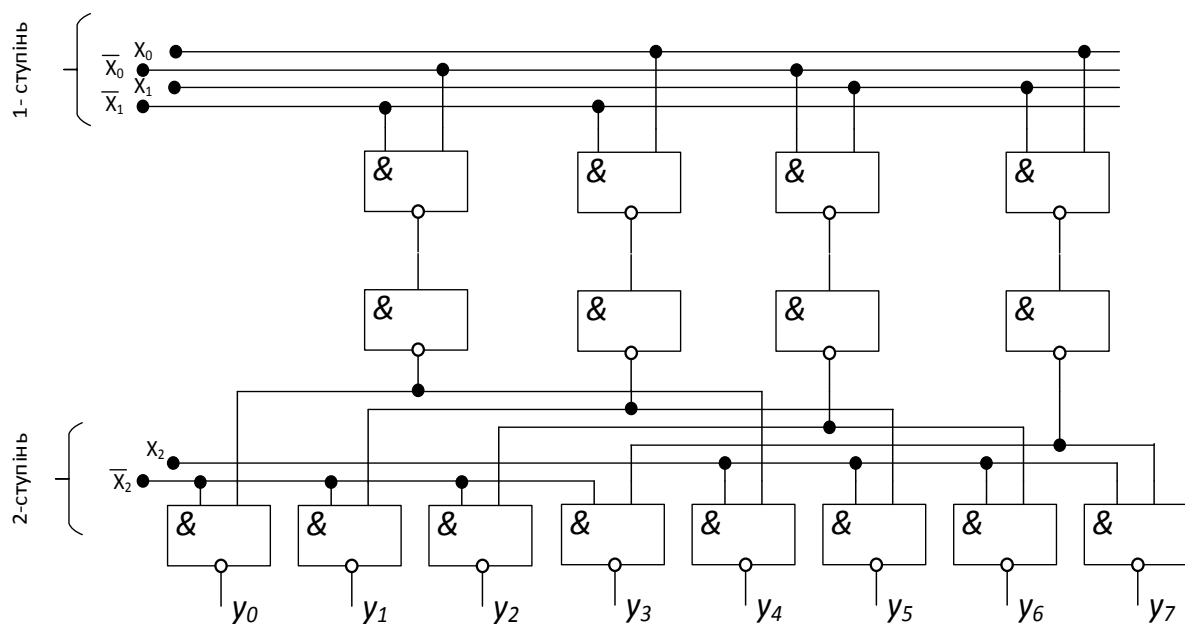


Рисунок 4.9 – Пірамідальний дешифратор

4.4.4 Нарощування розмірності дешифраторів

Серійні ЦІМС дешифраторів звичайно мають обмежене число входів-виходів (наприклад, три входи і два здвоєні виходи в серіях 100, K500, K555). Тому велике практичне значення мають методи побудови дешифраторів заданої (великої) розмірності на основі дешифраторів невеликої розмірності. Розглянемо це на прикладі побудови дешифратора шестирозрядних слів з тривходових стробованих дешифраторів (рис. 4.10). Для цього потрібно дев'ять однакових ЦІМС. Загальне стробування здійснюється за входом С першого дешифратора (при $C = 0$ на всіх виходах першого дешифратора виробляються нульові сигнали, тому нулі будуть формуватися і на всіх виходах дешифраторів другого ступеня).

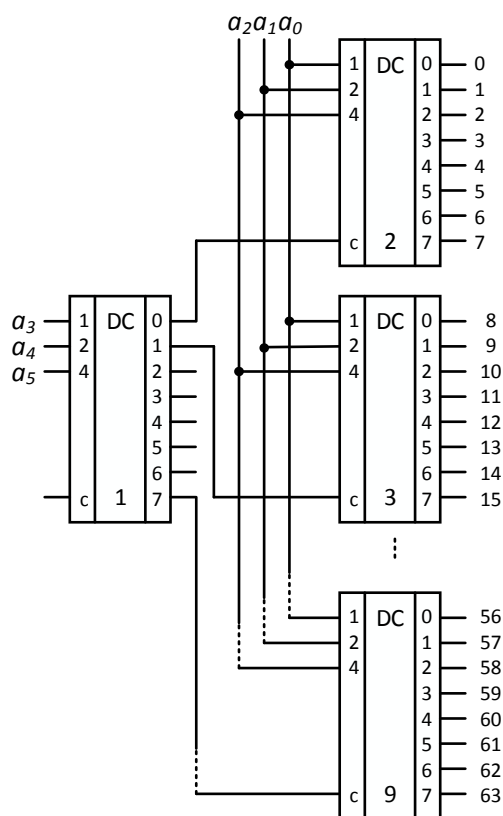


Рисунок 4.10 – Схема нарощування розмірності дешифраторів

На входи першого дешифратора подані три старших розряди вхідного слова, значення яких при $C = 1$ визначають номер вихода з одиничним сигналом. Цей сигнал відкриває один з дешифраторів другого ступеня за його стробувальним входом. Обраний таким чином дешифратор розшифровує три наступних (молодших) розряди слова. Наприклад, при дешифруванні слова 111011 = 59 маємо на вході дешифратора першого ступеня слово 111, яке обумовлює одиничний сигнал на його сьомому виході і вмикає DC9. На вході дешифратора DC9 діє слово 011, тому одиниця з'явиться на його третьому виході, тобто на загальному 59-му виході, що й потрібно.

У неповних дешифраторах частина вхідних наборів є неробочою (не використовується). Цю обставину можна розглядати як можливість спрощення дешифратора – неробочим вхідним наборам можна поставити у відповідність будь-які виходи.

4.5 Шифратори

Це пристрої, які здійснюють перетворення номера входу, на якому діє одиничний сигнал, в його позиційне числове зображення (найчастіше – двійкове). Шифратор має m входів, послідовно пронумерованих десятковими числами $(0, 1, 2, \dots, m-1)$ і n виходів (рис. 4.11). Сигнал на одному з входів веде до появи на виходах n -розрядного двійкового слова (табл. 4.5).

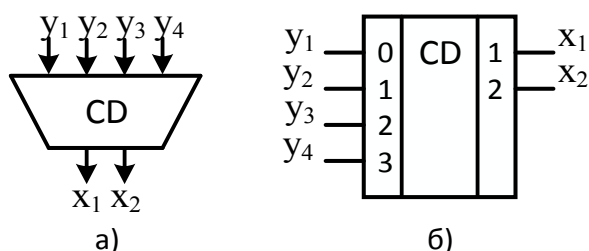


Рисунок 4.11 – УГП шифратора: а) на функціональних схемах;
б) на принципових схемах

Таблиця 4.5 – Таблиця функціонування шифратора

Збуджені входи	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9
Стан виходів $a_3 a_2 a_1 a_0$	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

Шифратори широко використовуються в різноманітних пристроях введення інформації в цифрові системи. Оскільки двійкові шифратори перетворюють код «1 з N» у двійкове число, то можна вважати, що вони виконують мікрооперацію, обернену мікрооперації, виконуваних дешифраторами.

Повний двійковий шифратор має 2^n входів і n виходів. Одне з основних застосувань шифратора – введення даних з клавіатури, при якому натискання клавіші з певною десятковою цифрою повинно приводити до передачі в пристрій двійкового зображення даної цифри (наприклад, тетради двійково-десятькового коду). У цьому випадку потрібен неповний шифратор «10×4» (рис. 4.12, а), на прикладі якого розглянемо принципи будови шифраторів.

З табл. 4.5 випливає, що

$$\begin{aligned}
 a_0 &= F_1 + F_3 + F_5 + F_7 + F_9; \\
 a_1 &= F_2 + F_3 + F_6 + F_7; \\
 a_2 &= F_4 + F_5 + F_6 + F_7; \\
 a_3 &= F_8 + F_9.
 \end{aligned}$$

Для реалізації шифратора на ЛЕ ТТЛ, що часто використовуються в периферійних пристроях, потрібно виразити попередні формули в операторній формі для ЛЕ І-НІ. Тоді

$$\begin{aligned}a_0 &= \overline{F_1 \cdot F_3 \cdot F_5 \cdot F_7 \cdot F_9}; \\a_1 &= \overline{F_2 \cdot F_3 \cdot F_6 \cdot F_7}; \\a_2 &= \overline{F_4 \cdot F_5 \cdot F_6 \cdot F_7}; \\a_3 &= \overline{F_8 \cdot F_9}.\end{aligned}$$

Цим операторним формам і відповідає схема шифратора на рис. 4.12, б).

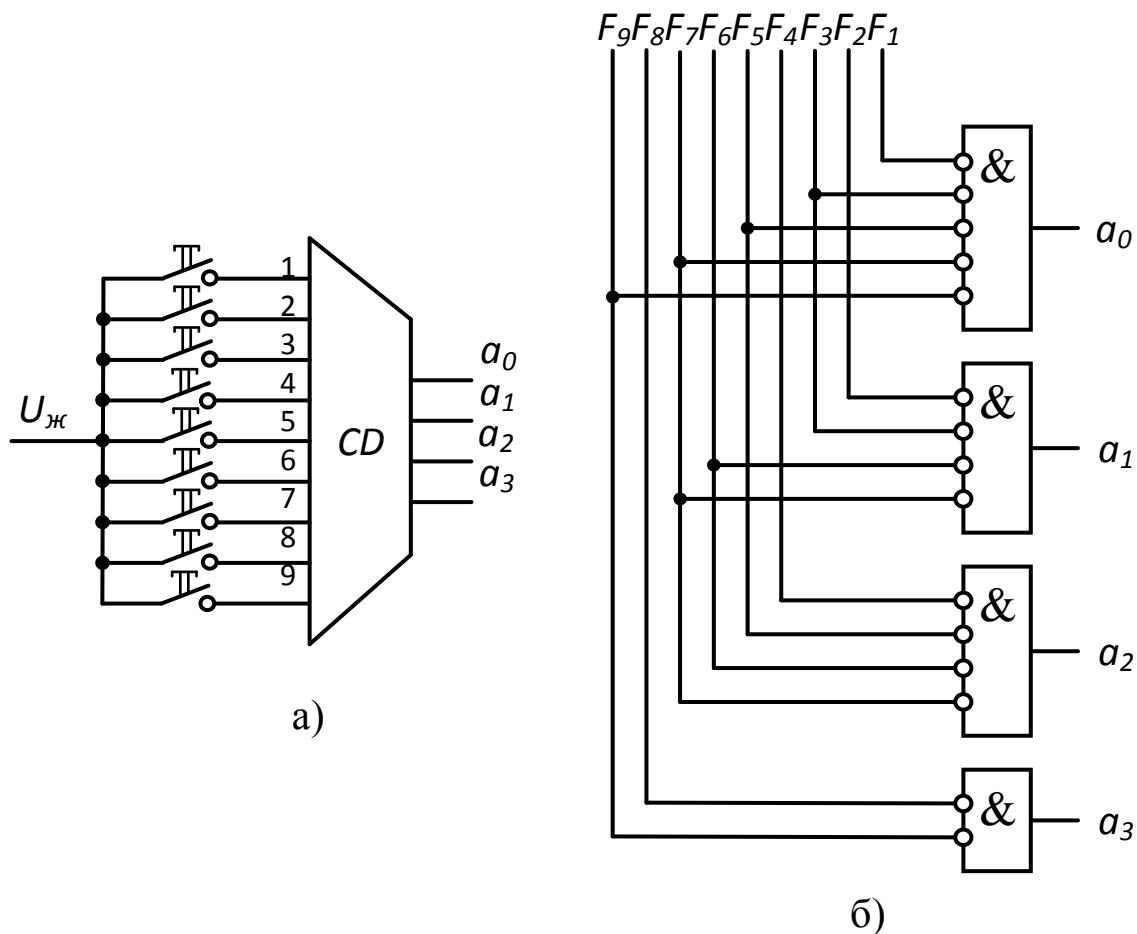


Рисунок 4.12 – Схема введення даних з клавіатури а) і шифратор «10×4» б)

Для визначення старшої за вагою одиниці в деякому слові використовують шифратори пріоритету (рис. 4.13). Пріоритетний шифратор – це пристрій, що виставляє на виході адресу найпріоритетнішого входу, на якому є вхідний сигнал. У цьому шифраторі найпріоритетнішим входом є вхід a_1 тому його під'єднано відразу до входу шифратора. Інші ж входи під'єднуються через додаткові елементи І, які блокують проходження сигналу на вхід шифратора при наявності сигналу на більш пріоритетному вході. Запишемо логічні функції, що утворюються на вході шифратора:

1	a_1
2	$\overline{a_1 \cdot a_2}$
3	$\overline{a_1 + a_2 \cdot a_3}$
4	$\overline{a_1 + a_2 + a_3 \cdot a_4}$
5	$\overline{a_1 + a_2 + a_3 + a_4 \cdot a_5}$

Отже, як видно з наведених логічних функцій, для побудови чи n -розрядної функції потрібно об'єднати через елемент АБО-НІ усі попередні розряди та результат подати на елемент І з n -розрядом.

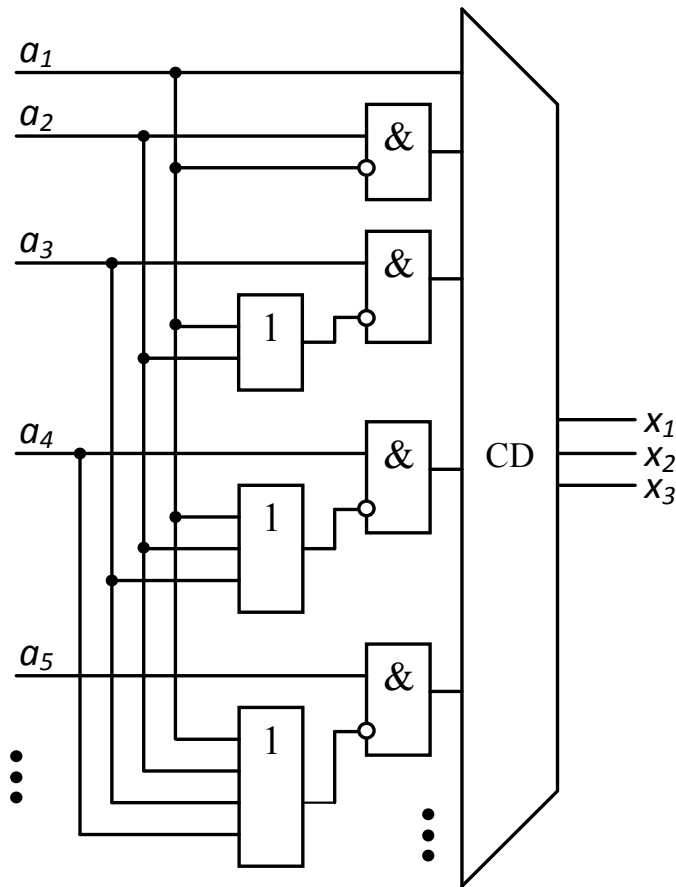


Рисунок 4. 13 – Пріоритетний шифратор

4.6 Мультиплексори

Мультиплексор – комутатор сигналів, які надходять з декількох вхідних ліній зв'язку на одну вихідну лінію (рис. 4.14, а). Вибір вхідної лінії A_i здійснюють, задаючи її адресу. За наявності m адресних входів можна задавати $M = 2^m$ адрес, кожна з яких забезпечує вибір однієї з M вхідних ліній. Мультиплексор складається з дешифратора адреси вхідної лінії, ЛЕ І та пристрою об'єднання виходів на ЛЕ АБО.

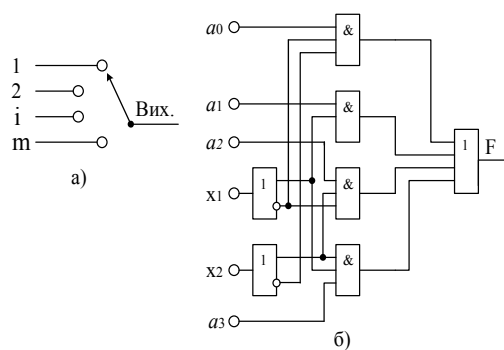


Рисунок 4.14 – Принципи роботи мультиплексора а) та його реалізації б)

Функціональна схема мультиплексора наведена на рисунку 4.14, б. Двійкове слово, що задає входні адреси, відкриває один з ЛЕ I, який і з'єднує з виходом відповідну входну лінію. Інформація на виході визначається станом обраного входного каналу і не залежить від стану інших каналів. Структура мультиплексора «4>1» на ЛЕ лементах I-АБО-НІ показана на рис. 4.15, а.

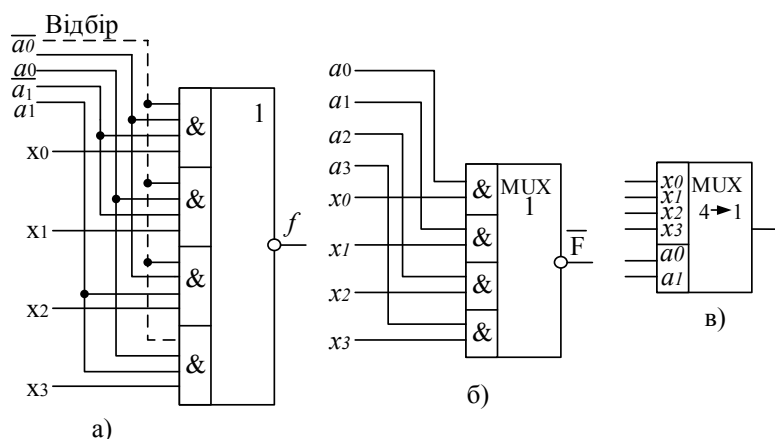


Рисунок 4.15 – Структура мультиплексора «4 > 1» на ЛЕ I-АБО-НІ а), використання мультиплексора як пристрою для формування повідомлень в коді «1 з N» б) та його умовне позначення в)

В графічних позначеннях, використовуваних в технічній документації, функцію мультиплексування називають MUX (від слова multiplexor). Приклад позначення для мультиплексора «4>1» наведено на рис 4.15, в.

Управління мультиплексором може здійснюватися не тільки двійковим кодом, але й кодом «1 з N». У цьому випадку число входів управління збільшується та стає таким, що дорівнює числу інформаційних входів (рис. 4.15, б). Такий режим мультиплексора використовують, зокрема, в мікрозрядних колах реверсивних лічильників і регістрів.

Як серійні ІМС відомі мультиплексори «4>1», «8>1» та «16>1». Мультиплексори на більше число входів, як правило, доводиться будувати з мультиплексорів меншої розмірності. Якщо необхідний мультиплексор

« $N > 1$ », а є ЦІМС з числом входів N_1 , то ще будуть потрібні L ІМС, де $L = N/N_1$, які разом і забезпечать потрібне число входів. Для каскадів звичайного типу потрібен додатково об'єднувальний мультиплексор на виході пристрою (рис. 4.16).

Функціонування пристрою покажемо на конкретних прикладах. Нехай, наприклад, слово управління дорівнює 10101. Це означає, що на виходах мультиплексорів першого ступеня будуть сигнали з їх п'яти інформаційних входів ($y_2y_1y_0 = 101$). На вихідний мультиплексор подається управляльне слово 10, і на вихід пристрою потрапляє сигнал x_2 вихідного мультиплексора, тобто п'ятий вихід третього мультиплексора, номер якого дорівнює 21, що і відповідає двійковому числу 10101.

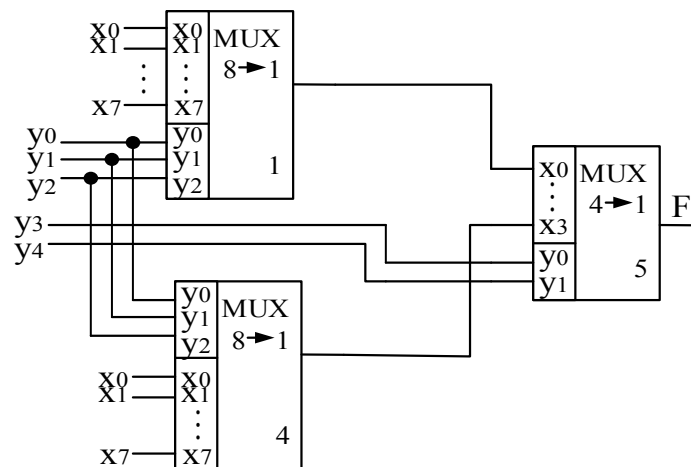


Рисунок 4.16 – Нарощування розмірності мультиплексора

Якщо вихідні каскади ЦІМС мають виходи з трьома станами, то можна безпосередньо об'єднувати ці виходи, а потім під'єднання ЦІМС до вихідного кола здійснити за допомогою дешифратора, який управляє стробувальними входами ЦІМС (рис. 4.17).

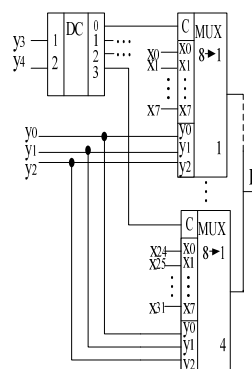


Рисунок 4.17 – Нарощування розмірності мультиплексора, що має вихідні каскади з трьома станами

Недолік такого способу нарощування розмірності – додавання ємностей у вихідних колах, що в ряді випадків (наприклад, для ЦІМС на МОН-транзисторах) може істотно знизити швидкодію мультиплексора.

Мультиплексори як комп’ютерні компоненти є також зручним засобом реалізації перемикальних функцій, особливо від невеликого числа змінних. Якщо число адресних входів мультиплексора m , то це означає, що внутрішня структура мультиплексора вже має засоби для відтворення 2^m мінтермів від m вхідних змінних. Щоб внести потрібні мінтерми до ДДНФ заданої функції, досить подати одиничні сигнали на відповідні інформаційні входи мультиплексора. У випадку, коли загальне число вхідних змінних n більше m , то за допомогою додаткової логічної схеми потрібно утворити мінтерми від $n - m$ змінних, які далі потрібно під’єднати на інформаційні входи мультиплексора подібно до одиничних сигналів в попередньому випадку. Структуру такої додаткової логічної схеми досить просто можна визначити табличним методом або за допомогою діаграм Вейча.

4.7 Демультимплексори

Це теж комутатори сигналів, що забезпечують їх передачу з одного входу на одну з вихідних ліній за її адресою. Для передачі даних по каналу зв’язку з розділенням їх в часі потрібні не тільки мультиплексори, але й демультимплексорів, що розподіляють дані з одного каналу між кількома приймачами інформації. Це завдання і реалізується демультимплексором, функціонування якого можна пояснити за допомогою рис. 4.18, а. Він має один інформаційний вхід, n адресних (управляльних) входів і 2^n виходів.

Побудова демультимплексора «1>4» на ЛЕ І показана на рис 4.18, б. Робота демультимплексора описується логічними виразами:

$$F_i = x m_i; i = \overline{0, 2^n - 1},$$

де m_i – мінтерми, утворені з n адресних змінних. В умовних зображеннях демультимплексор позначають буквами DMX (рис. 4.18, в). В технічній літературі відомі випадки використання також терміна «селектори даних» стосовно мультиплексорів і демультимплексорів.

Функціонування демультимплексора можна описати таблицею (наприклад, табл. 4.6) для демультимплексора з чотирма виходами.

Таблиця 4.6 – Функціонування демультимплексора

Адресні входи		Виходи			
A_1	A_0	0	1	2	3
0	0	Д	0	0	0
0	1	0	Д	0	0
1	0	0	0	Д	0
1	1	0	0	0	Д

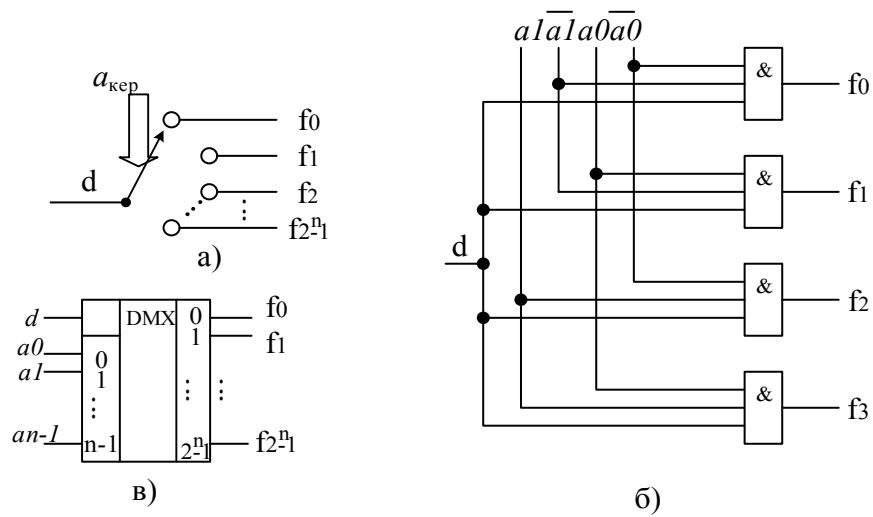


Рисунок 4.18 – Принцип роботи а), реалізація б) та умовне позначення для демультиплексора в)

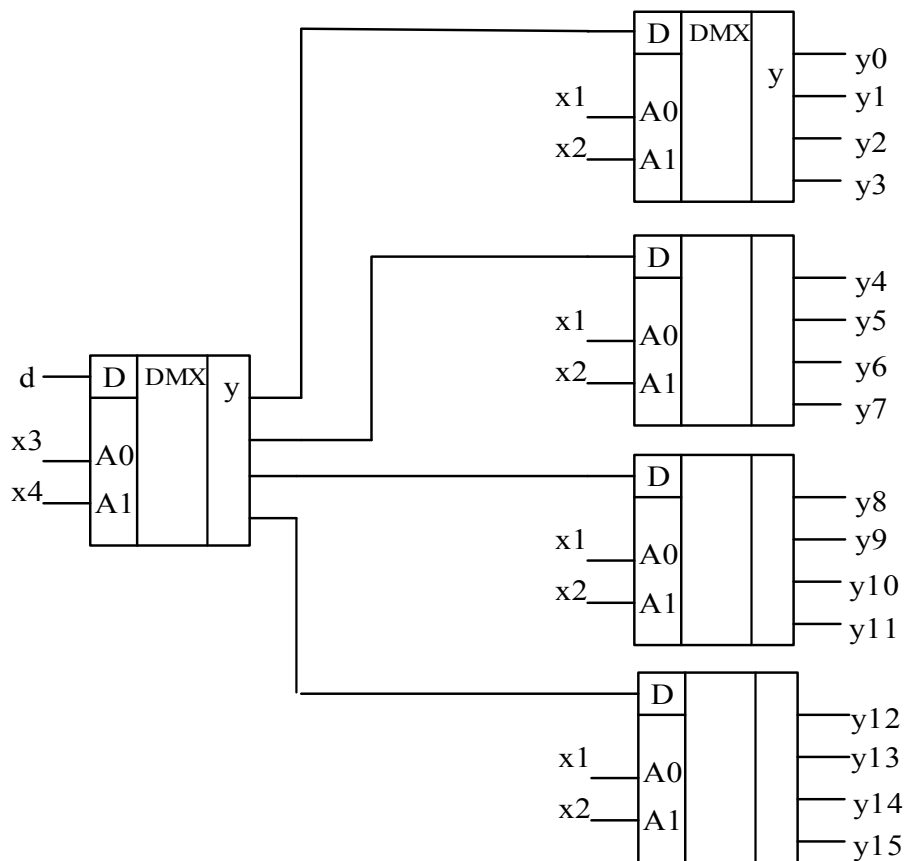


Рисунок 4.19 – Демультиплексорне дерево

Для великого числа входів, за необхідності, може бути побудовано демультимплексорне дерево (див. рис. 4.19). Дешифратори зі стробуванням (див. рис. 4.5, а) можна використовувати як демультимплексор. При цьому стробувальний вхід виконує функції інформаційного входу.

4.8 Шини передачі даних

Комп'ютерна шина (англ. *Computer bus* від латин. *Omni bus* – «всі шини» «для всіх») в архітектурі сучасного комп'ютера – підсистема, яка містить спеціальні апаратні і програмні засоби, що забезпечують обмін даними між функціональними блоками комп'ютера. Насправді є сукупністю сигнальних ліній, які мають певні електричні характеристики і протоколи (набори попередньо прийнятих угод щодо логічних і електричних правил обміну даними між різними програмами). У апаратних пристроях і програмних засобах, які загалом і утворюють поняття шини, можна розрізнити механічний, електричний (фізичний) і логічний (управління) рівні. Шини ранніх комп'ютерних систем являли собою джгути (пучки провідників – інформаційних, управління та живлення – для компактності і зручності обслуговування конструктивно розміщених разом), тобто, насправді реалізовували паралельні електричні кола з декількома відгалуженнями-під'єднаннями. В сучасних комп'ютерних системах термін «шини» використовують для будь-яких фізичних процесів і приладів, що загалом надають таку ж логічну функціональність, як і паралельні комп'ютерні шини. Комп'ютерні шини розрізняють за їх призначенням, пропускну здатністю, розрядністю, кількістю та типами пристроїв підтримки, протоколами роботи та іншим. В реальних реалізаціях шин часто комбінують різні їх змістовно-якісні відмінності. Наприклад, за способом передачі сигналів (послідовні та паралельні, синхронні та асинхронні) відомі використання як власне паралельних, так і послідовних з'єднань, які можуть мати паралельні (англ. *Multidrop*) і ланцюгові (англ. *Daisy chain*) топології. У випадку USB (*Universal Serial Bus*) і деяких інших шин можуть також використовуватися так звані хаби (концентратори).

Деякі типи швидкодіючих шин (Fibre Channel, InfiniBand, швидкісний Ethernet, SDH) для передачі сигналів використовують не електричні з'єднання, а оптичні.

На відміну від звичайного електричного з'єднання двох провідників, яке ще має назву «з'єднання точка-точка», до шини можна під'єднати декілька різних пристроїв, використовуючи один і той же набір провідників. Кожна шина визначає свій набір конекторів (з'єднувачів) для фізичного під'єднання, з'єднувальних карт і кабелів. Конектори до шини, різноманітні роз'єми, як правило, уніфіковані і дозволяють під'єднати різні пристрої до шини.

Більшість комп'ютерів має як внутрішні, так і зовнішні шини. Внутрішня шина під'єднує всі внутрішні компоненти комп'ютера до материнсь-

кої плати (і, отже, до процесора і пам'яті). Такий тип шин також називають локальною шиною, оскільки вона служить для під'єднання локальних пристроїв. Зовнішня шина під'єднує зовнішні периферійні пристрої до материнської плати.

4.9 Пристрої порівняння (цифрові компаратори)

Пристроєм порівняння (цифровим компаратором) називають функціональний вузол комп'ютера, призначений для отримання ознак відношень між двійковими словами (операндами). Ознаки відношень записують у вигляді:

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

$$F_i = A * B \quad \text{або} \quad P_{i,A} * B \quad \text{або} \quad F_{A * B},$$

де A і B – двійкові або двійково-десяткові числа; K – двійкова константа; i – номер відношення (часто пропускають); $*$ – операція відношення вигляду $=, \neq, <, >, \leq, \geq$ і т. п.; F_i – функція, що задає результат відношення: 1 – якщо відношення виконується, тобто істинне, і 0 – якщо відношення не виконується, тобто помилкове. В технічній документації функцію компаратора позначають буквами *COMP* (від англ. comparator) або знаками $=$.

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

$$F_{A \neq B} = \overline{F_{A=B}}; \quad F_{A \leq B} = \overline{F_{A < B}}; \quad F_{A \leq B} = F_{A=B} \cup F_{A < B}.$$

Ознаки відношення використовують як логічні умови в мікропрограмах, командах передачі управління, а також у пристроях контролю і діагностики. Після виконання кожної команди в комп'ютері формуються ознаки результатів операції. Ці ознаки, які ще називають прапорцями, вміщують в спеціальний регістр прапорців. До прапорців зазвичай відносять ознаки нульового результату, переповнення розрядної сітки, знак результату, наявність переносів із старшого розряду суматора, парне чи непарне число одиниць в деякому результаті операції та ін.

Зазначимо, що формування і використання прапорців – це одна з основних відмінностей комп'ютера від калькулятора. Тільки за допомогою прапорців комп'ютер приймає рішення про хід обчислювального процесу, тобто має певні інтелектуальні властивості.

Відношення широко використовуються як логічні умови в мікропрограмах, а також в пристроях контролю та діагностики комп'ютерних засобів.

4.9.1 Пристрої порівняння на рівність

Будують на основі порозрядних операцій над однойменними розрядами обох операндів. Ознака r рівності розрядів має одиничне значення, якщо в обох розрядах містяться або одиниці, або нулі, тобто:

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

У свою чергу ознака нерівності розрядів:

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

Ознака рівності операндів R приймає одиничне значення, якщо всі розряди рівні, тобто:

$$R = r_{n-1} r_{n-2} \dots r_0.$$

Відповідно до вищенаведених виразів для r_i і R будують пристрої (рис. 4.20, а). Переходячи до базису І-НІ, отримаємо схему пристрою на рис. 4.20, б. Затримки формування сигналів рівності R і нерівності $\bar{R}$ операндів A і B складають відповідно 4 і 3 затримки ЛЕ І-НІ.

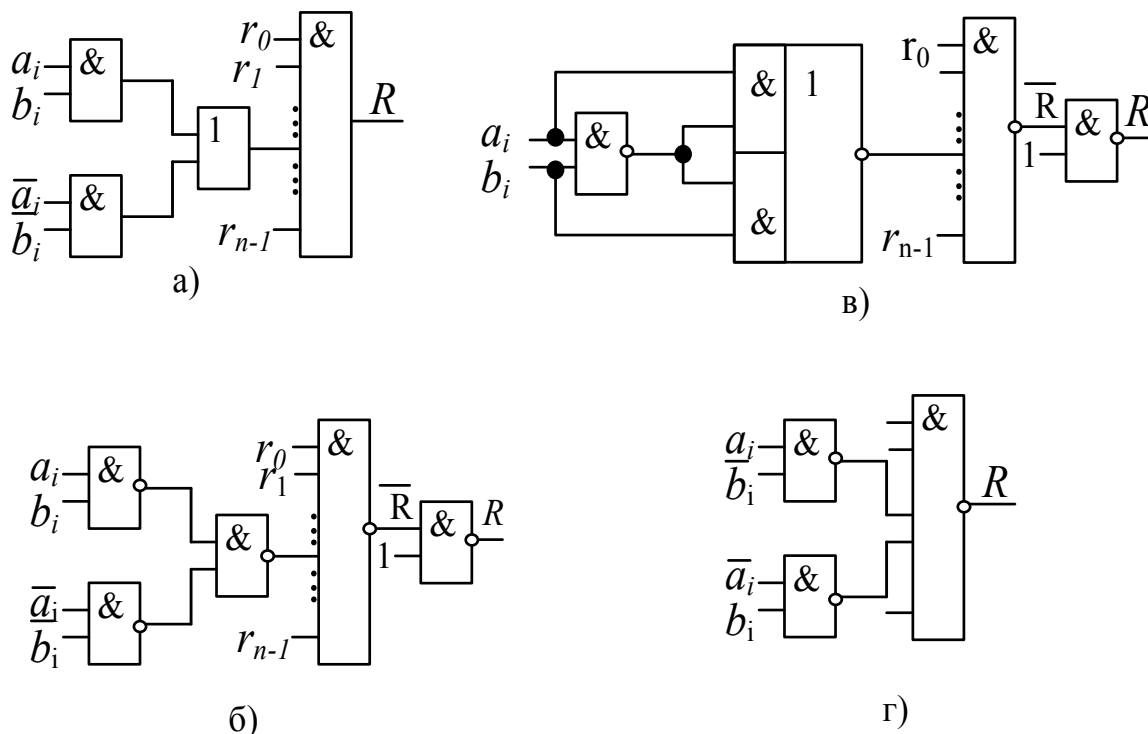


Рисунок 4.20 – Варіанти схем пристроїв порівняння на рівність

Для зменшення числа зовнішніх виводів, як і в інших подібних випадках, доцільно відмовитися від парафазного подання вхідних змінних. Цього можна досягти не тільки звичайним прийомом установаження інверторів у вхідних колах, що збільшує логічну глибину пристрою, але і більш ефективним шляхом, перетворюючи формули для r_i так, щоб вони не містили інверсій аргументів:

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

Цей вираз реалізує схема пристрою на рис. 4.20, в, для якої число входів скоротилося вдвічі, а швидкодія майже не змінилася (для елементів ТТЛ $t_{з.пр.} \approx t_{з.па.}$).

Можлива і двоярусна схема пристрою (рис. 4.20, г), що ґрунтується на реалізації формули:

$$\bar{R} = \overline{r_0 r_1 \dots r_{n-1}} = \overline{a_0 \bar{b}_0 \bar{a}_0 b_0 a_1 \bar{b}_1 \bar{a}_1 b_1 \dots a_{n-1} \bar{b}_{n-1} \bar{a}_{n-1} b_{n-1}}.$$

Варіант пристрою на рис. 4.20, г має підвищену швидкодію і меншу складність, але його застосування обмежене використанням ЛЕ з числом входів $2n$.

Гранично зменшити логічну глибину пристрою (досягти максимальної швидкодії) можна, використовуючи ЛЕ І-АБО-НІ (рис. 4.21), однак така реалізація, як правило, придатна лише для порівняння дворозрядних операндів у зв'язку з обмеженим числом груп І в серійних ЛЕ І-АБО-НІ.

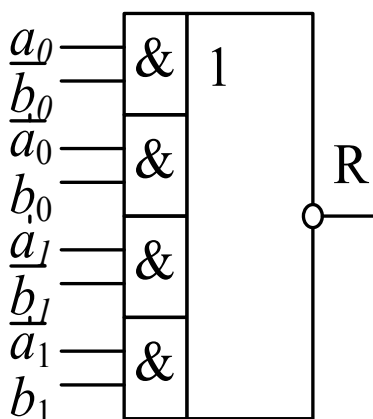


Рисунок 4.21 – Схема пристрою порівняння на рівність з найвищою швидкодією

4.9.2 Порівняння на «більше» («менше») однорозрядних слів

Визначається функцією $F_{A>B}$ в табл. 4.7.

З таблиці витікає, що $F_{A>B} = A\bar{B}$. Функцію $F_{A>B}$ для порівняння двороз-

Таблиця 4.7

A	B	$F_{A>B}$
0	0	0
0	1	0
1	0	1
1	1	0

рядних операндів можна отримати, виходячи з таких міркувань. Якщо в старшому розряді слова A – одиниця, а слова B – нуль, то, незалежно від молодших розрядів, $A > B$ і $F_{A>B} = 1$. Якщо старші розряди однакові, то потрібно переходити до аналізу молодших, застосовуючи до них ту ж умову, що і для старшого розряду ($F_{A>B} = 1$ при $a_0 = 1$ і $b_0 = 0$). Таким чином, має місце співвідношення:

$$F_{A>B} = a_1\bar{b}_1 + a_0\bar{b}_0r_1,$$

де $r_1 = a_1b_1 + \bar{a}_1\bar{b}_1$ – ознака рівності розрядів. Легко бачити, що порівняння багаторозрядних слів може базуватися на таких же міркуваннях, і тому можна записати:

$$F_{A>B} = a_{n-1}\bar{b}_{n-1} + a_{n-2}\bar{b}_{n-2}r_{n-1} + a_{n-3}\bar{b}_{n-3}r_{n-2}r_{n-1} + \dots a_0\bar{b}_0z_1 \dots z_{n-1}.$$

В отриманий вираз входять аргументи і ознаки рівності розрядів. Оскільки пристрій порівняння на «більше» звичайно є частиною більш повного цифрового компаратора, де функції r_i використовують для встановлення інших відношень між операндами, то r_i можна розглядати як такі, що завжди є в наявності. Якщо пристрій порівняння на «більше» призначений для автономного функціонального використання, то можна спростити реалізацію функції $F_{A>B}$. Ця функція для порівняння дворозрядних операндів може бути відображена картою Карно (рис. 4.22).

		a1a0			
b1b0		00	01	11	10
	00	0	1	1	1
	01	0	0	1	1
	11	0	0	0	0
	10	0	0	1	0

Рисунок 4.22 – Карта Карно для функції $F_{A>B}$

Виділення вказаних на карті Карно контурів приводить до формули:

$$F_{A>B} = a_1 \overline{b_1} + a_0 \overline{b_1} \overline{b_0} + a_1 a_0 \overline{b_0} = a_1 \overline{b_1} + a_0 \overline{b_0} (a_1 + \overline{b_1}).$$

Введемо позначення:

$$c_i = a_i \overline{b_i};$$

$$d_i = a_i + \overline{b_i} \text{ або } d_i = r_i.$$

З використанням цих позначень за індукцією можна записати вираз для порівняння $F_{A>B}$ багаторозрядних операндів:

$$F_{A>B} = c_{n-1} + c_{n-2} d_{n-1} + \dots + c_i d_{n-1} \dots d_{i+1} + \dots + c_0 d_{n-1} \dots d_1.$$

Як приклад, побудуємо схему пристрою порівняння на «більше» для чотирирозрядних операндів в базисі І-НІ. У цьому випадку:

$$F_{A>B} = \overline{c_3} \cdot c_2 \overline{d_3} \cdot \overline{c_1} d_3 \overline{d_2} \cdot \overline{c_0} d_3 d_2 d_1.$$

Такий пристрій має затримку $3t_3$ (рис. 4.23) незалежно від числа розрядів порівнюваних слів і потребує застосування ЛЕ з числом входів, що дорівнює $n + 1$. Подібні пристрої використовують для порівняння окремих полів операндів з подальшим їх об'єднанням в один загальний компаратор.

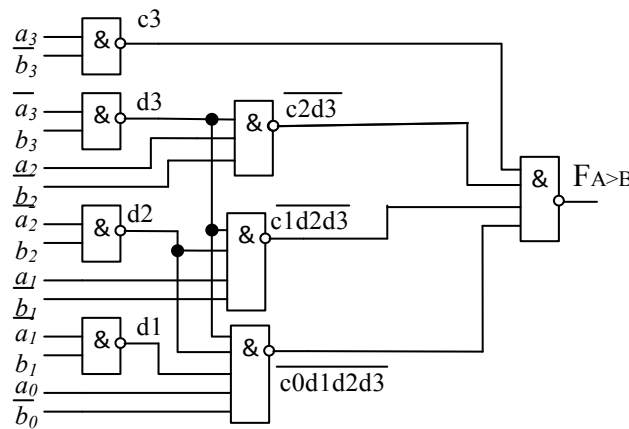


Рисунок 4.23 – Схема пристрою порівняння на «більше» для чотирирозрядних операндів

До формули для $F_{A>B}$ можна застосувати факторизацію (мінімізацію шляхом розміщення дужок)

$$F_{A>B} = c_{n-1} + d_{n-1}(c_{n-2} + d_{n-2}c_{n-3}) + \dots + d_2(c_1 + d_1c_0) \dots).$$

Тому функції $F_{A>B}$ відповідає схема, в якій застосовуються однотипні модулі c і d і двовходові елементи І та АБО. Зі зростанням розрядності порівнюваних операндів не виникає труднощів з числом входів ЛЕ елементів і навантаженням на них, але швидкодія пристрою знижується. Затримка подібним компаратором для багаторозрядних операндів може бути неприпустимо великою

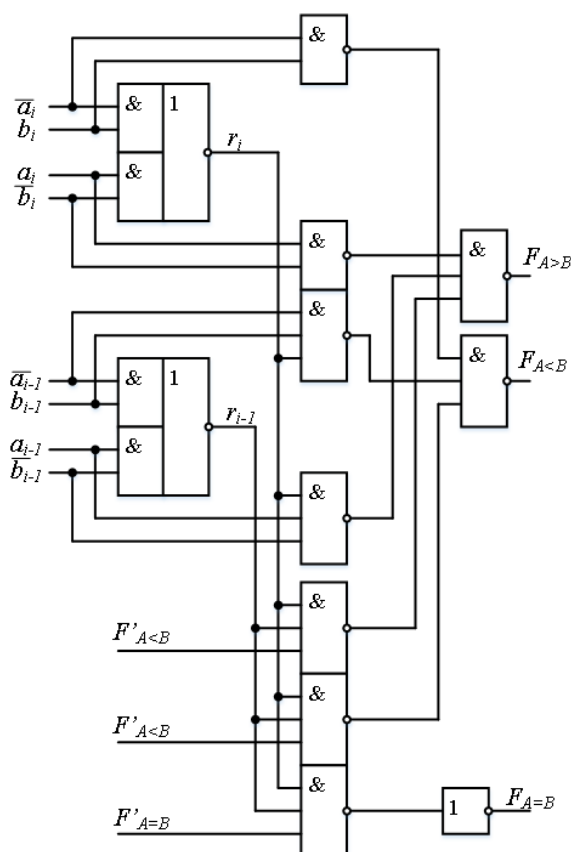
$$t_k = (2n - 1)t_3.$$

4.9.3 Порівняння багаторозрядних операндів

Найчастіше їх виконують за допомогою групових структур компараторів. Кожна група є малорозрядним компаратором, що має входи для розрядів порівнюваних полів і входи ознак $F'_{A=B}$, $F'_{A>B}$ і $F'_{A<B}$ від молодшої групи. Виходи компаратора можуть бути під'єднані до наступної (старшої) групи (рис. 4.24).

Принцип побудови груп покажемо на прикладі дворозрядної групи (рис. 4.24).

Отримання ознаки $F_{A>B}$ тут проводиться за співвідношенням:



$$F_{A>B} = a_1 b_1 a_{i-1} b_{i-1} r_i F'_{A>B} r_{i-1} - 1 = a_1 b_1 + a_{i-1} b_{i-1} r_i + F'_{A>B} r_{i-1} - 1.$$

Групи між собою можна з'єднати послідовно (рис. 4.25, а), тоді затримка компаратора визначається формулою

$$t_k \cong (m - 1)t_n + t_{гр}.$$

Компаратор з груповою структурою можна побудувати інакше, якщо функції $F_{A=B}$, $F_{A>B}$ і $F_{A<B}$, отримані в групах, використовувати як відповідні функції окремих розрядів всередині групи.

Результатом буде паралельно-паралельна структура, вихідний сигнал якої визначається виразом

Рис. 4.24 – Схема дворозрядної групи для побудови компараторів

$$F_{A>B} = F_m + F_{m-1} D_m + F_{m-2} D_{m-1} D_m + \dots + F_1 D_2 D_3 \dots D_m,$$

де F_i ($i = 1, \dots, m$) – функції $F_{A>B}$ i -ї групи (нумерація з молодшої групи);
 D_i – кон'юнкція функцій d всіх розрядів даної групи.

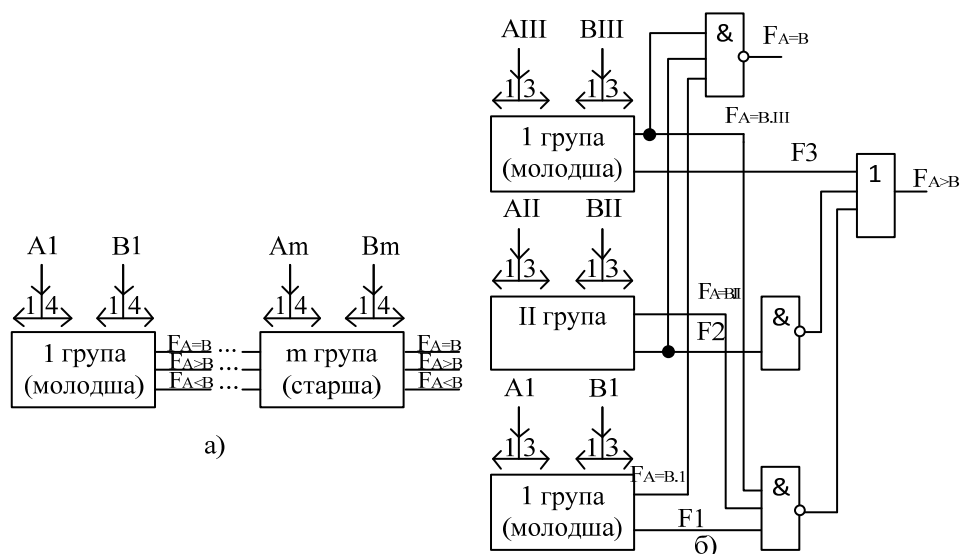


Рисунок 4.25 – Структури групових компараторів послідовного а) та паралельного б) типів

Зокрема, при розділенні дев'ятирозрядного операнда на три групи отримаємо

$$\begin{aligned}
 F_{A>B} &= F_3 + F_2 D_3 + F_1 D_2 D_3, \\
 F_3 &= a_8 \bar{b}_8 + a_7 \bar{b}_7 d_8 + a_6 \bar{b}_6 d_7 d_8, \\
 F_2 &= a_5 \bar{b}_5 + a_4 \bar{b}_4 d_5 + a_3 \bar{b}_3 d_4 d_5, \\
 F_1 &= a_2 \bar{b}_2 + a_1 \bar{b}_1 d_2 + a_0 \bar{b}_0 d_1 d_2.
 \end{aligned}$$

Логічна глибина подібних структур (рис. 4.25, б) всього на 2 перевищує логічну глибину груп.

4.9.4 Порівняння слів з константами

Цю ситуацію розглянемо на прикладі, коли потрібно встановити ознаки відношень двійкового операнда $A=A_2A_1A_0$ до наступних заданих констант

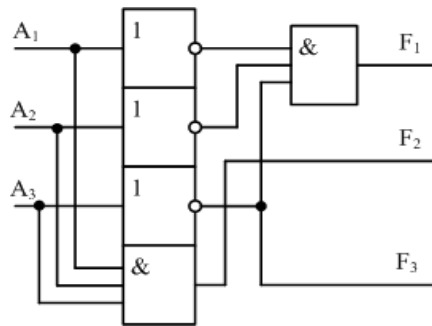
$$F_1 := (A=000); F_2 := (A=111) \text{ і } F_3 := (A \leq 011).$$

На основі табл. 4.8 ознаки відношень слова A до перелічених констант можна подати у вигляді

$$F_1 = \bar{A}_1 \bar{A}_2 \bar{A}_3, F_2 = A_1 A_2 A_3, F_3 = \bar{A}_3. \quad (4.1)$$

Отже, пристрій порівняння слова з константами згідно з виразами (4.1) можна реалізувати за схемою на рис. 4.26.

Таблиця 4.8 – Логічні функції F_1, F_2, F_3



A_2	A_1	A_0	F_1	F_2	F_3
0	0	0	1	0	1
0	0	1	0	0	1
0	1	0	0	0	1
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	0	1	0

Рисунок 4.26 – Пристрій порівняння слова з константами

4.9.5 Пристрої порівняння двійкових слів

Багаторозрядні двійкові слова рівні, коли одночасно попарно рівні всі їхні розряди, тобто $A(n) = B(n)$, якщо $A_i = B_i, i = 1, 2, \dots, n$. На основі табл. 4.9, яка задає умову рівності r_i двох i -х розрядів A і B , отримаємо

$$r_i = \bar{A}_i \cdot \bar{B}_i \vee A_i B_i = \overline{A_i \oplus B_i} = \bar{M}_i, \quad (4.2)$$

де M_i – функція додавання за модулем два. Реалізація функції (4.2) показана на рис. 4.27.

Ознаку рівності двох n -розрядних слів $P_{A=B}$ визначають як логічний добуток порозрядних умов r_i

$$F_{A=B} = r_n \cdot r_{n-1} \cdot \dots \cdot r_1 = \bar{M}_n \cdot \bar{M}_{n-1} \cdot \dots \cdot \bar{M}_1. \quad (4.3)$$

Таблиця 4.9		
A_i	B_i	r_i
0	0	1
0	1	0
1	0	0
1	1	1

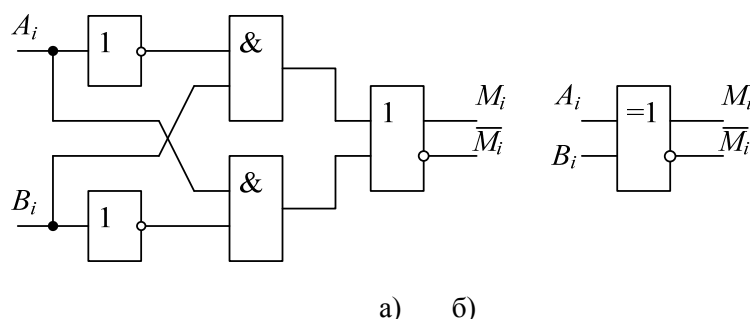


Рисунок 4.27 – Реалізація ЛЕ «Виключне АБО»:
а) – схема; б) – умовне позначення

Пристрій порівняння двох чотирирозрядних слів A і B відповідно до виразу (4.3) показаний на рис. 4.28. Пристрій містить чотири ЛЕ «виключне АБО» і один ЛЕ І.

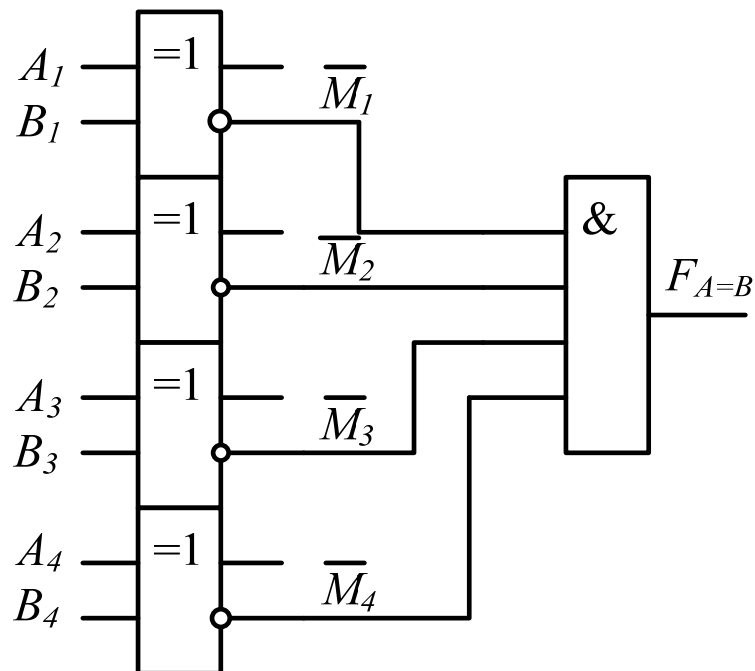


Рисунок 4.28 – Пристрій порівняння двох чотирирозрядних слів A і B

За великої розрядності порівнюваних слів число входів ЛЕ І може стати неприпустимо великим. В такому разі доцільно переходити до ступеневого порівняння: на першому ступені отримати ознаки для чотирирозрядних груп, а на другому – реалізувати загальний прапорець логічним множенням групових ознак. Наприклад, для розрядності порівнюваних слів $n = 16$ маємо чотири групові ознаки порівняння:

$$F_{A=B}^{1,4}, \quad F_{A=B}^{5,8}, \quad F_{A=B}^{9,12}, \quad F_{A=B}^{13,16},$$

де верхні індекси означають номери розрядів у групах. Тоді ознака порівняння двох 16-розрядних слів може бути записана як

$$F_{A=B} = F_{A=B}^{1,4} \cdot F_{A=B}^{5,8} \cdot F_{A=B}^{9,12} \cdot F_{A=B}^{13,16}.$$

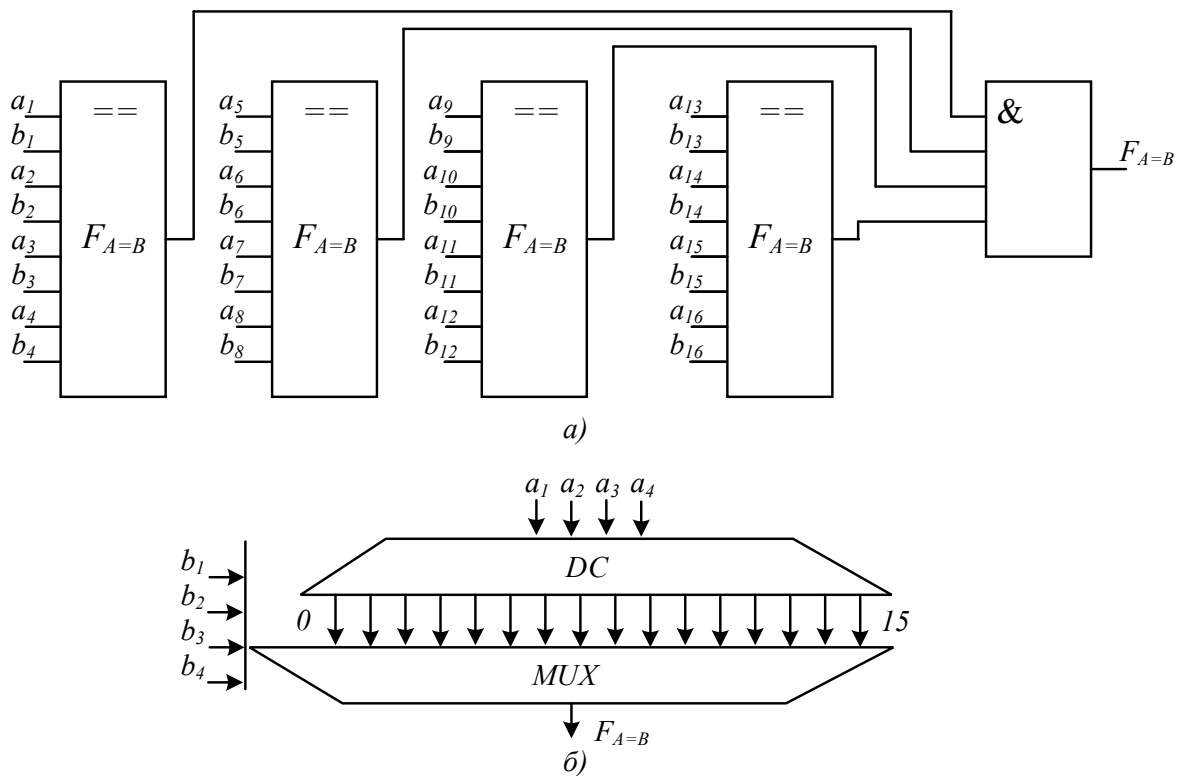


Рисунок 4.29 – Пристрої порівняння двох слів на рівність: а) – групова структура; б) – на основі дешифратора і мультиплексора ДЕ б)

Пристрій порівняння двох 16-розрядних слів показаний на рис. 4.29, а. Порівняння може бути реалізоване і на інших принципах. Порівняння двох чотирирозрядних чисел A і B на основі дешифратора і мультиплексора показано на рис. 4.29, б. Дешифратор формує одиничний сигнал на тому виході, номер якого також визначається десятковим еквівалентом вхідного слова. Наприклад, при $A_4A_3A_2A_1 = 0111$ логічна одиниця з'явиться на виході з номером сім. Мультиплексор під'єднує до виходу той вхід, номер якого також визначається десятковим еквівалентом вхідної комбінації. Якщо $B_4B_3B_2B_1 = 0111$, то дозволяється проходження на вихід сигналу з сьомого входу. Таким чином, якщо слова A і B рівні, то формується прапорець $F_{A=B} = 1$.

4.9.6 Пристрої порівняння двох слів «на більше»

Порівняння двох слів A і B «на більше» за абсолютною величиною виробляє ознаку $F_{A>B}$ і може бути виконано за нижченаведеним алгоритмом:

- аналіз нерівності слів A і B виконують послідовно в напрямку від старших розрядів до молодших;
- молодші розряди задіюють в порівняння в тому випадку, коли старші розряди рівні (еквівалентні);
- для отримання ознаки $P_{A>B}$ будують диз'юнктивну суму порозрядних умов.

Логіка порівняння розрядів A і B наведена в табл. 4.10, де C_i – ознака $A_i > B_i$; r_i – умова використання в аналізі сусідніх молодших розрядів обох слів.

На основі табл. 4.10 отримуємо такі вирази:

$$\begin{aligned} C_i &= A_i \overline{B_i}; \\ r_i &= \overline{A_i} \overline{B_i} + A_i B_i = \overline{A_i \oplus B_i} = \overline{M_i} \end{aligned} \quad (4.4)$$

Таблиця 4.10 –
Логіка порівняння
розрядів A і B

A_i	B_i	C_i	r_i
0	0	0	1
0	1	0	0
1	0	1	0
1	1	0	1

З урахуванням виразу (4.4) і вищенаведеного алгоритму аналізу функцію ознаки $F_{A>B}$ подамо у вигляді:

$$F_{A>B} = C_n + r_n C_{n-1} + \dots + r_n r_{n-1} r_2 C_1 \quad (4.5).$$

Для порівняння двох чотирирозрядних слів «на більше» ознаку нерівності згідно з виразом (4.5) запишемо таким чином

$$\begin{aligned} F_{A>B} &= C_4 + r_4 C_3 + r_4 r_3 C_2 + r_4 r_3 r_2 C_1 = \\ &= A_1 \overline{B_4} + \overline{M_4} A_3 \overline{B_3} + \overline{M_4} \overline{M_3} A_2 \overline{B_2} + \overline{M_4} \overline{M_3} \overline{M_2} A_1 \overline{B_1} \end{aligned} \quad (4.6)$$

Схема пристрою порівняння «на більше» двох чотирирозрядних слів A і B відповідно до формули (4.6) показана на рис. 4.30.

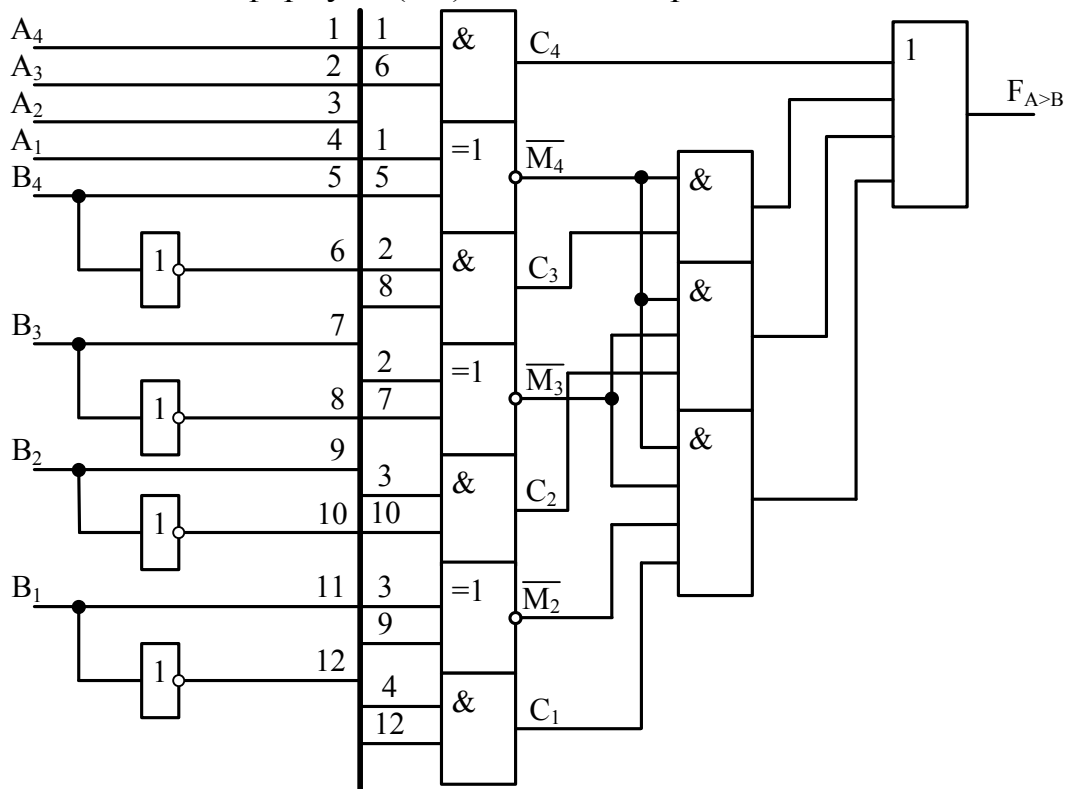
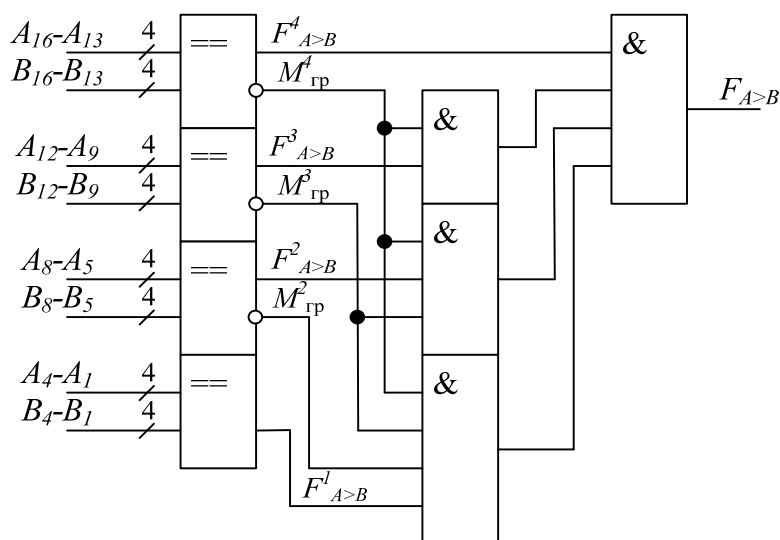


Рисунок 4.30 – Схема пристрою порівняння
двох слів «на більше»

4.9.7 Багаторозрядні пристрої порівняння «на більше». При реалізації схем порівняння багаторозрядних слів «на більше» виникають технічні труднощі, пов'язані з необхідністю використання ЛЕ з великою кількістю входів. Тому слова, що їх порівнюють, розбивають на групи, наприклад, по чотири розряди. Кожна група виробляє свою ознаку нерівності $F_{iA>B}$ і умову задіяння до аналізу молодшої групи згідно з виразом (4.6) і рис. 4.30. Наприклад, для $n = 16$ маємо чотири групи, виходи яких об'єднують відповідно до формули

$$F_{A>B} = F_{A>B}^4 + M_{ep}^4 F_{A>B}^3 + M_{ep}^4 M_{ep}^3 F_{A>B}^2 + M_{ep}^4 M_{ep}^3 M_{ep}^2 F_{A>B}^1, \quad (4.7)$$

Тут $F_{A>B}^4$ – прапорець порівняння «на більше» в найстаршій групі з розрядами $A_{16} - A_{13}$, $B_{16} - B_{13}$ і $M_{гр}^4 = M_{16}M_{15}M_{14}M_{13}$ – умова для задіяння до аналізу сусідньої молодшої групи; $F_{A>B}^3$ – прапорець порівняння «на більше» у групі з розрядами $A_{12} - A_9$, $B_{12} - B_9$ і $M_{гр}^3 = M_{12}M_{11}M_{10}M_9$ – умова аналізу молодшої групи; $F_{A>B}^2$ – прапорець порівняння «на більше» у групі з



розрядами $A_8 - A_5$, $B_8 - B_5$ і $M_{гр}^2 = M_8M_7M_6M_5$ – умова задіяння молодшої групи; $F_{A>B}^1$ – прапорець порівняння «на більше» у групі з розрядами $A_4 - A_1$, $B_4 - B_1$. Схема пристрою порівняння «на більше» двох 16-розрядних слів A і B на основі формули (4.7) показана на рис. 4.31.

Рисунок 4.31 – Схема пристрою порівняння «на більше» двох 16-розрядних слів

4.10 Контрольні запитання і завдання

1. Функції перетворювачів кодів.
2. Основні етапи проектування перетворювачів кодів.
3. Як реалізувати перетворення прямого коду в інверсний?
4. Функції дешифраторів, їх структурні різновиди.
5. Які параметри визначають функціональні і структурні якості дешифраторів?
6. Яка залежність існує між входами і виходами повного дешифратора?

7. Неповні дешифратори.
8. Для чого використовують каскадування дешифраторів?
9. Який тип дешифратора потребує найбільших апаратних витрат?
10. Який тип дешифратора має найвищу швидкодію?
11. Функції шифраторів, їх структурні різновиди.
12. Що таке пріоритетний шифратор?
13. Нарощування структур шифраторів.
14. Що таке мультиплексор і демультиплексор?
15. Основні параметри мультиплексорів і демультиплексорів.
16. Основні застосування мультиплексорів і демультиплексорів.
17. Функції і структури цифрових компараторів.
18. Основні структури для порівняння слів.
19. Як здійснюється порівняння двох слів на «більше»?
20. Як здійснюється порівняння двох слів на «рівність»?

5 КОМП'ЮТЕРНІ ПРИСТРОЇ НА ТРИГЕРАХ

Перелік скорочень і умовних позначень до розділу 5

ДЗ – динамічний запис; КП – комірка пам'яті; ПУ – пристрій управління; СЗ – статичний запис; СС – синхронізувальний сигнал; ТП – тригерні пристрої.

5.1 Загальні відомості, характеристики та класифікації тригерів

У функціональних пристроях комп'ютерних систем для зберігання інформації найчастіше використовують тригери – елементарні скінченні автомати (або елементарні автомати Мура, див. підрозділ 1.5), що складаються з комірки пам'яті (КП) з двома стійкими станами виходів та пристрою управління (ПУ), який перетворює інформацію, яка надходить із зовнішніх джерел, в комбінації сигналів, що діють безпосередньо на входи комірки, забезпечуючи її основні режими роботи, а саме: 1) зберігання певного стану комірки пам'яті, 2) його читання-копіювання, 3) запис – встановлення нового стану. Тригери також широко використовуються в цифрових пристроях формування сигналів, генераторах спеціальних сигналів, в подільниках частоти, лічильниках, лічильних системах, регістрах, у пристроях управління і т. д. Далі розглянуті найрозповсюдженіші типи тригерів з урахуванням особливостей їх функціонального призначення, способів запису і управління процесами передавання інформації. У ряді випадків, наприклад, коли розробнику потрібен тригер з функціями, яких існуючі тригери не мають, то виникає задача проектування спеціального тригерного елемента. Надалі будемо вважати, що узагальнена схема будь-якого тригера відповідає рис. 5.1, де $X_1, \dots, X_n$ – інформаційні входи тригера; $C_1, \dots, C_m$ – тактувальні входи; Q і $\bar{Q}$ – відповідно, прямий та інверсний виходи тригера; f_1 і f_2 – функції збудження КП.

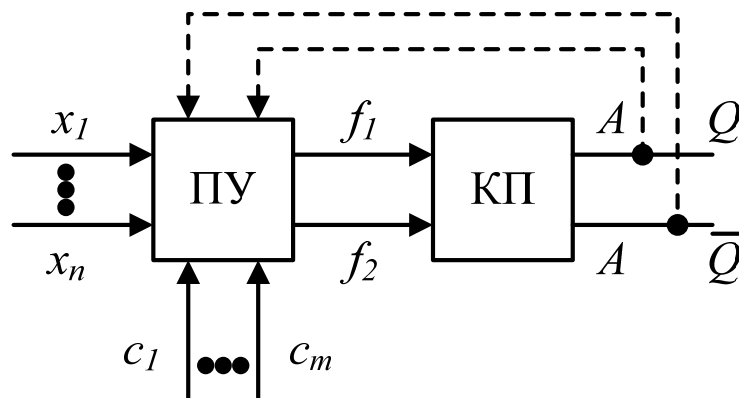


Рисунок 5.1 – Узагальнена схема тригера

Для інформаційних входів тригерів прийняті такі позначення: S (*Set* – встановлення – вхід для роздільного (окремого, не пов’язаного з іншими входами чи сигналами) встановлення тригера у стан «1» ($Q = 1, \bar{Q} = 0$); R (*Reset* – очищення) – вхід для роздільного встановлення тригера в стан «0» ($Q = 0, \bar{Q} = 1$); T (*Toggle* – релаксатор) – лічильний вхід тригера; J (*Jerk* – раптове вмикання) – вхід для роздільного встановлення тригера в стан «1» в універсальному JK -тригері; K (*Kill* – раптове вимикання) – вхід для роздільного встановлення тригера в стан «0» в універсальному JK -тригері; D (*Delay* – затримка, *Drive* – передавання) – інформаційний вхід для встановлення тригера у стан «0» або «1»; V (*Valve* – клапан, вентиль) – вхід управління для дозволу прийому інформаційних або тактових сигналів; тактувальний вхід C (*Clock* – первинне джерело сигналів синхронізації) дозволяє пристрою управління записувати інформацію в тригер.

Найзагальнішими визначальними ознаками тригерів для їх класифікації є функціональна ознака та спосіб запису інформації. Функціональна класифікація базується на характеристиці стану входів і виходів тригера в момент часу до його перемикавання (t) і після його перемикавання ($t+1$). Для цього використовують характеристичні рівняння тригерів, які описують їх поведінку (функціонування). Ці рівняння в загальному вигляді записують так

$$Q^{t+1} = f(Q^n, X_1^n, X_2^n, \dots, X_n^n)$$

На практиці, згідно з функціональною ознакою, розрізняють RS -, JK -, RST -, E -, R -, S -, T -, DV - D - тригери та деякі інші. Функціональну назву тригера звичайно утворюють за позначенням його інформаційних входів, яким замість символів $X_1 - X_n$ на загальній структурній схемі (рис. 5.1) присвоюють символи R, S, J, K, T і т. д. Наприклад, RS -тригер має два інформаційних (логічних) входи – R та S , з яких вхід R еквівалентний входу X_1 , а вхід S – входу X_2 . Проте в деяких випадках тригери позначають однією буквою та мають два і більше інформаційних входи (наприклад, тригер E -типу).

За способом запису інформації тригери поділяють на асинхронні та синхронні (тактовані). При цьому визначальним є час оновлення інформації на тригерах відносно сигналів тактової (синхронізувальної) частоти, що діють в пристрої. До асинхронних відносять тригери, які мають тільки інформаційні входи. Запис інформації в такі тригери здійснюється асинхронно, тобто у довільні моменти часу відносно моментів дії синхронізувальних сигналів.

На відміну від асинхронних синхронні тригери крім інформаційних мають додаткові синхронізувальні (тактові) входи C . Запис інформації в такі тригери здійснюється тільки в момент дії синхросигналу (CC) на вході C . В свою чергу, синхронні тригери поділяють на тригери однократної та багатотактної дії. Багатотактні (n -тактні, $n = 2, 3, 4$) тригери характеризу-

ються тим, що запис інформації в тригер закінчується з надходженням n -го тактувального імпульса. За способом синхронізації розрізняють тригери: синхронні зі статичним і динамічним записом.

У синхронних тригерах зі статичним записом СС починають впливати на стан тригера тільки тоді, коли рівень СС або зростає до рівня «1», або знижується до рівня «0», залежно від елементної бази, на якій виконано тригер. Тригери, стан яких змінюється в інтервалі між рівнями «1» та «0» тактового сигналу, називають тригерами, що спрацьовують за рівнем, а тригери, стан яких змінюється після закінчення дії рівнів «1» або «0» тактового сигналу на тригер, але ще під час його тривалості, називають тригерами з внутрішньою затримкою.

В синхронних тригерах з динамічним записом інформація записується в момент зміни амплітуди тактового сигналу в певному напрямку, тобто під час зростання або спаду його фронту.

Терміни «асинхронні та синхронні тригери» не слід плутати з поняттями асинхронний та синхронний режими роботи тригера, оскільки асинхронні тригери можуть працювати в синхронному, а синхронні – в асинхронному режимах. Так, асинхронний тригер буде працювати в синхронному режимі, якщо інформація, що надходить на його входи, «прив'язана» до синхросигналів, і навпаки, синхронний тригер буде працювати в асинхронному режимі, якщо сигнали на тактовому вході C не «прив'язані» до синхросигналів.

В багатьох конкретних випадках проектування тригерних (тобто, таких, у яких основними структурними елементами є тригери) пристроїв можуть використовуватися інші, менш загальні, класифікаційні ознаки тригерів. Часто таку «локальну» класифікацію здійснюють, покладаючи в її основу спосіб запису інформації на тригер. Ця ознака на етапі вибору тригера для тригерного пристрою дозволяє враховувати важливі «тонкі» моменти в роботі тригера, а саме:

- які частини (фрагменти) сигналу (фронти, рівні, їх комбінації) використовуються для запису (прийому) інформації;
- реакцію тригера на зміну інформації в процесі її запису;
- моменти часу, коли інформація з'являється (фіксується) на виходах тригера.

Можна виділити дві групи тригерів, які розрізняються за способом управління записом інформації:

- тригери, в яких прийом та фіксація інформації суміщені в часі;
- тригери, в яких прийом та фіксація інформації рознесені в часі.

До першої групи належать тригери, в яких для прийому та фіксації інформації використовується тільки один фрагмент сигналу: рівень або фронт. Позначим, як це показано на рис. 5.2, рівні символами L , $\bar{L}$ (від слова *Level* – рівень), а фронти символами F , $\bar{F}$ (від слова *Front* – фронт). Відповідно до прийнятих позначень тригери, які приймають та фіксують

інформацію за рівнями та фронтами, називають тригерами L - та F -типів (видів).

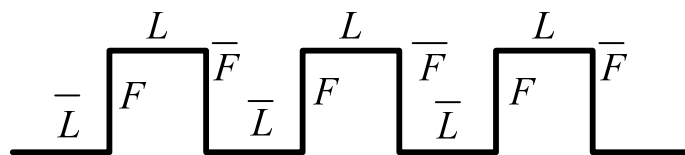


Рисунок 5.2 – Позначення фрагментів сигналів

До другої групи відносять тригери, в яких для прийому та фіксації інформації використовуються як мінімум дві частини сигналу, наприклад рівень і зріз, рівень і фронт, фронт і зріз і т. п. Такі тригери будемо позначати двома символами (перший вказує на інтервал прийому, а другий – на момент фіксації), наприклад, тригер $L\bar{F}$ -типу приймає інформацію за рівнем L з фіксацією на виходах за фронтом $\bar{F}$ (фронт 10). Можливі інші різновиди тригерів цієї групи, наприклад, тригер $\bar{L}F$ -типу, який приймає інформацію у схему за рівнем $\bar{L}$ (рівень 0) з фіксацією на виході за фронтом F (01). Тригер $\bar{L}\bar{F}$ -типу приймає інформацію у пристрій управління за рівнем $\bar{L}$ з фіксацією на виході за фронтом $\bar{F}$; тригер $F\bar{F}$ -типу приймає інформацію за фронтом F з фіксацією за фронтом $\bar{F}$.

Об'єднуючи перераховані класифікаційні ознаки, отримаємо загальне (символьне) позначення тригера, яке дозволяє судити не тільки про виконання тригером функції, але й про особливості окремих фаз роботи їх алгоритму роботи. Нижче на прикладах статичних тригерів RS -, D -, JK -типів наводяться варіанти такого об'єднання при символьному позначенні тригера: $C_L RS$ -тактований RS -тригер типу L , який перемикається на входах R та S сигналами високого рівня (рівень L на рис. 5.3). Враховуючи, що на входах R та S тригером управляють потенційні сигнали рівня L , то такий тригер можна задати як $C_L R_L S_L \equiv (CRS)_L$.

$C_{\bar{L}} \bar{R} \bar{S}$ – тактований RS -тригер типу $\bar{L}$, яким управляють на входах R та S сигнали низького рівня (рівня $\bar{L}$). Цей же тригер можна задати як $(CRS)_{\bar{L}} \equiv C_{\bar{L}} \bar{R}_{\bar{L}} \bar{S}_{\bar{L}} = C_{\bar{L}} \bar{R} \bar{S}$.

$C_F RS$ – тактований RS -тригер типу F , який приймає та фіксує інформацію за фронтом F (тобто протягом деякого інтервалу часу, що практично збігається з тривалістю фронту).

$C_{L\bar{F}} \bar{R} \bar{S}$ – тактований RS -тригер виду $L\bar{F}$, який приймає інформацію за рівнем L з фіксацією на виходах за фронтом $\bar{F}$.

$C_{L\bar{F}} \bar{J} \bar{K}$ – тактований JK -тригер виду $L\bar{F}$, яким управляють на входах JK сигнали з рівнем логічного 0.

$C_{F\bar{F}} D$ – тактований D -тригер типу $\bar{F}$.

$R_L - S_L \equiv (R - S)_L$ – асинхронний RS -тригер типу L .

$R_{\overline{L}} - S_{\overline{L}} \equiv (R - S)_{\overline{L}}$ – асинхронний RS -тригер типу $\overline{L}$.

RS -тригер без класифікаційних ознак, які враховували б особливості запису (аналогічно JK , D та інших типів).

CRS -тактований RS -тригер без класифікаційних ознак, які враховують особливості запису (аналогічно CD , CJK та інших типів).

Наведені приклади не охоплюють всієї множини символьного запису типів тригерних пристроїв, і більш детально це питання буде розглянуто нижче.

Таким чином, індексна класифікація дозволяє спростити процедуру вибору та ефективного застосування тригерних пристроїв, оскільки така класифікація дозволяє насправді не тільки знати алгоритм роботи тригера, але й дає уявлення про його вихідну діаграму, не заглиблюючись в його принципову схему. Останнє є важливою особливістю у проектуванні тригерних пристроїв, оскільки відомо, що тригери одного функціонального типу, але з різними способами запису інформації, мають різні вихідні діаграми при ідентичній вхідній.

На завершення цього розділу наведемо ще декілька класифікаційних ознак, які можуть мати локальні застосування в проектуванні тригерних пристроїв.

За способом організації логічних зв'язків розрізняють тригери з роздільним встановленням станів «0» та «1» (RS -тригери); з лічильним входом (T -тригери); універсальні з роздільним встановленням станів «0» та «1» (JK -тригери); з прийомом інформації за одним входом (D -тригери); універсальні з управлінням прийомом інформації за одним входом (DV -тригери); комбіновані (наприклад, RST -, $JKRS$ -, DRS -тригери і т. п.); зі складною вхідною логікою.

За кількістю інформаційних входів тригери можуть бути одновходовими, двовходовими та багатовходовими. Найбільше поширення отримали одно- та двовходові тригери. Однак слід мати на увазі, що число інформаційних входів тригера не завжди дорівнює числу його фактичних входів, на які надходять інформаційні сигнали, оскільки діючий інформаційний вхід у структурі тригера може бути кон'юнкцією, диз'юнкцією або іншою функцією декількох логічних змінних, які діють на інформаційних входах, наприклад $J = J_1 J_2 J_3$; $K = K_1 K_2 K_3$; $J = \overline{J_1 J_2 J_3}$ і т. п.

За кількістю тактувальних входів розрізняють тригери одноктактні, двоктактні, багатотактні. Іноді до двоктактних відносять двоступеневі тригери (так звані тригери зі структурою типу MS – від англ. *Master-Slave*), однак слід пам'ятати, що основне призначення останніх – отримання ефекту часової затримки інформаційних сигналів у структурі тригера, якщо такий ефект не можна реалізувати з використанням якихось фізичних методів, наприклад за допомогою накопичення зарядів і т. д.

За видом вхідних сигналів тригери поділяють на статичні та динамічні.

Статичні тригери – такі, в яких вихідні сигнали в стійких станах залишаються незмінними в часі. Динамічні тригери – такі, в яких вихідні сигнали в стійких станах змінюються.

За способом реалізації розрізняють такі особливості тригерних пристроїв:

- тип використовуваної комірки пам'яті (статичний, динамічний чи обидва разом);
- вид інформаційного сигналу, з яким може працювати тригер (імпульсний, тобто сигнал обмеженої тривалості; потенціальний, тобто сигнал необмеженої тривалості, або і той, і інший);
- наявність чи відсутність у структурі пристрою вузлів, які перетворюють потенційні вхідні сигнали в імпульсні. Відповідно з цією ознакою всі тригери поділяють на статичні, імпульсно-статичні, динамічні, квазистатичні.

Тригер, як і будь-який інший електротехнічний пристрій, характеризується, крім специфічних функціональних параметрів та вимог, ще й загальноелектротехнічними. До функціональних відносять ті вимоги та параметри, яким має відповідати тригер при його конкретному застосуванні. Вони звичайно містять:

- функціональний тип тригера;
- спосіб запису інформації в тригер;
- число тактувальних, інформаційних, управляючих та установчих входів;
- еквівалент навантаження тригера за тактувальним входом;
- навантажувальна здатність тригера за виходом;
- швидкодія тригера;
- надійність тригера.

Специфічними параметрами тригера є: роздільний час тригера $t_{роз}$ – найменший інтервал часу між вхідними сигналами мінімальної довжини, які викликають безперервне перемикавання тригера. Очевидно, що $t_{роз}$ залежить від того, який зміст вкладається в слова «безперервне перемикавання тригера». Прийнято вважати, що тригер перемикається безперервно, якщо будь-яке значення вихідного сигналу, яке визначається правилами роботи, має тривалість, яка не менша середнього часу затримки розповсюдження $t_{зт.р.сер.}$ одного ЛЕ пристрою; максимальна частота перемикавання тригера

$$f_{max} = 1 / t_{роз}.$$

Оскільки, як визначено вище, вихідні сигнали тригера при дії вхідних сигналів з частотою f_{max} мають тривалість $t_{зт.р.сер.}$, то, враховуючи тривалість фронтів зростання та спаду, можна зробити висновок, що ці вхідні сигнали є недостатніми для надійної передачі інформації в логічні кола, бо рівні «1» та «0» у цьому випадку не будуть фіксуватися. Для забезпечення

їх фіксації прийнято зменшувати частоту $f_{\max}$ в 1,5 раза і вважати її робочою, тобто

$$f_{\text{роб}} = f_{\max} / 1,5.$$

Мінімальна тривалість вхідного сигналу

$$t_i = \sum_{l=1}^k t_{\text{зт.п.сер.}},$$

де k – кількість елементів в колі інформаційного або тактового сигналу від входу до виходу елемента, на якому замикається тригерне коло зворотного зв'язку.

Час затримки перемикання тригера

$$t_{\text{зт.пер.}} = \sum_{l=1}^l t_{\text{зт.п.сер.}},$$

де l – кількість елементів у колі від входу інформаційного або тактового сигналу до виходу елемента, на якому підтверджується стан тригера.

З означення t_i та $t_{\text{зт.пер.}}$ виходить $l = k + 1$.

Тут одиниця характеризує затримку розповсюдження одного з логічних елементів, на яких виконується елементарна КП тригера.

Сукупність параметрів $f_{\text{роб.}}$, t_i , $t_{\text{зт.пер.}}$ визначає швидкодію тригерів і, в кінцевому рахунку, швидкодію цифрових пристроїв, основою яких вони є.

Оскільки тригери в цифрових пристроях взаємодіють з іншими електронними елементами, то, крім функціональних, потрібно знати і їх загальноелектротехнічні параметри. Передусім до цих параметрів слід віднести параметри ЛЕ, на яких побудований тригер: $K_{\text{об}}$ – коефіцієнт об'єднання за входом; $K_{\text{роз}}$ – коефіцієнт розгалуження за виходом; рівні «0» та «1», вхідні й вихідні струми і т. д.

5.2 Тригер як елементарний автомат Мура

В практиці проектування тригерних пристроїв тригери часто розглядають як елементарні автомати 2-го роду (автомати Мура, у яких вихідні сигнали повністю визначаються станом автомата), для яких характерно:

- невелике число інформаційних входів (не більше трьох, реальні ж тригери найчастіше мають два входи);
- число внутрішніх станів дорівнює 2, йому відповідає одна внутрішня змінна, яку прийнято позначати Q ;
- число вихідних змінних не більше однієї, причому її значення збігається зі значенням Q ; як правило, в тригерах разом із значенням Q форму-

ється інверсна змінна $\bar{Q}$;

– повнота функції переходів, яка виражає зв'язок внутрішніх змінних для моменту часу t із значеннями вхідних і вихідних змінних для моменту часу $t + 1$ і яку називають характеристичним рівнянням тригера

$$Q_{t+1} = f(x_b, Q_t). \quad (5.1)$$

Дійсно, для всіх тригерів є стани входів, під дією яких відбуваються зміни стану тригерів $Q_t \rightarrow Q_{t+1}$ всіх чотирьох типів: $0 \rightarrow 0$, $0 \rightarrow 1$, $1 \rightarrow 0$, $1 \rightarrow 1$. Далі такі переходи будемо позначати словом або числом із двох літер, наприклад $Q_t \rightarrow Q_{t+1}$ у вигляді Q_t, Q_{t+1} ; $0 \rightarrow 1$ у вигляді 01 і т. п.

Запис x_t означає, що сигнал x набуває певного конкретного значення «0» або «1» в момент часу t і зберігає його до моменту часу $t+1$. Запис Q_{t+1} означає, що вихідний сигнал набуває певного значення одразу після закінчення моменту часу $t+1$, і зберігає його до моменту часу $t+2$. Іноді визначають x_t та Q_t як значення змінних до приходу інформаційного (тактового) сигналу, а x_{t+1} , Q_{t+1} – після приходу інформаційного (тактового) сигналу.

Оскільки сигнали на виходах Q та $\bar{Q}$ взаємоінверсні, стан тригера визначений, якщо задане значення одного з вихідних сигналів, наприклад, на його прямому виході Q . Стани $Q = 1$, $\bar{Q} = 0$ називають одиничними, $Q = 0$, $\bar{Q} = 1$ – нульовими. За деяких комбінацій вхідних сигналів можуть з'явитися стани $Q = \bar{Q} = 1$ або $Q = \bar{Q} = 0$. Після закінчення дії таких вхідних комбінацій стани 00 або 11 не можуть зберігатись, і тригер перейде або в стан 01 або в стан 10 . Комбінацію вхідних сигналів, після закінчення якої стан тригера невизначений, тобто з рівною ймовірністю може бути одиничним або нульовим, називають забороненою комбінацією. У цьому випадку значення сигналів на виходах Q та $\bar{Q}$ – фіктивні, невизначені та позначаються символами $*$, $\times$, $\emptyset$, або буквами a , b , ...

Таким чином, тригер може мати п'ять логічних станів на виході (0 , 1 , Q , $\bar{Q}$, $\times$), які означають:

«0» – тригер постійно знаходиться в нульовому стані незалежно від зміни сигналів на його вході;

«1» – тригер постійно знаходиться в одиничному стані незалежно від зміни сигналів на його вході;

Q – стан тригера не змінюється при зміні вхідних сигналів, причому може бути або $Q = 0$, або $Q = 1$;

$\bar{Q}$ – стан тригера змінюється на протилежний при зміні вхідних сигналів, причому може бути зміна стану «1» на стан «0» або навпаки;

$\times$ – невизначений (фіктивний) стан тригера.

Число теоретично можливих типів тригерів з n інформаційними входа-

ми дорівнює 5^{2^n} , 5 – кількість можливих станів на виході тригера, 2^n – кількість наборів, які утворюють всі n вхідних змінних i . При $n = 1$ маємо 25, а при $n = 2$ вже 625 типів тригерів, однак значна їх частина або взагалі не має технічного сенсу, або є тривіальним чи дуальним варіантами схемних конфігурацій. Нетривіальних тригерів, з одним інформаційним входом, є тільки два, з двома входами – 24. В техніці найрозповсюдженіші двовходові тригери, але практично використовують тільки вісім їх типів, серед яких три – універсальні. Решта технічно реалізованих двовходових тригерів, серед яких є 10 універсальних, може знайти застосування в майбутньому.

Правила функціонування тригерів можна задавати:

- 1) словесним описом;
- 2) таблицею переходів тригера, тобто таблицею інформаційних значень вхідних сигналів, внутрішніх станів та вихідних сигналів тригера;
- 3) системою характеристичних рівнянь типу (5.1);
- 4) орієнтованим графом, де число вершин графу та їх кодування відповідають можливим станам тригера з урахуванням внутрішніх станів КП, а ребра графу, які починаються і закінчуються на вершинах; відповідають можливим переходам тригера з одного стану в інший, при цьому на ребрах вказується набір вхідних сигналів, під дією яких відбувається перехід з одного стану в інший, або підтверджує даний стан;
- 5) у формі мікропрограмного автомата;
- 6) у вигляді часових діаграм.

5.3 Асинхронні та синхронні тригерні пристрої (ТП) зі статичним управлінням

До статичних відносять тригери, для яких характерні такі ознаки:

1. Управління тригерами здійснюється як імпульсними, так і потенціальними сигналами;
2. Як КП використовуються тільки бістабільні комірки, що дозволяють зберігати інформацію нескінченно довго (статичні КП);
3. Тригери виконуються на потенціальних ЛЕ, їх структурна схема не містить в своєму складі елементів, які перетворюють вхідні потенціальні сигнали в імпульсні (елементів для диференціювання сигналів, для виділення фронтів тощо).

У сучасних цифрових пристроях статичні тригери застосовуються дуже широко. Причиною цього є їх зручність в експлуатації, надійність в роботі, схемна простота та технологічність в інтегральному виконанні.

На рис. 5.3 наведено класифікаційну таблицю статичних тригерів. Важливо відзначити, що всі типи тригерів, які розрізняються за способом запису, реалізують саме як статичні тригери.

Асинхронні тригери з управлінням за рівнем відносять до пристроїв, в яких прийом та фіксація інформації на тригері суміщені в часі. Останнє

означає, що інформація на виходах таких тригерів з'являється практично одночасно з її надходженням на інформаційні входи для асинхронних та з надходженням сигналів на тактувальний вхід С для синхронних тригерів. Поняття управління за рівнем слід пов'язувати з реакцією тригера на зміну інформації в процесі її запису. Щодо синхронних тригерів цей термін означає, що якщо під час дії тактових сигналів буде змінюватись інформація на логічних входах, то ці зміни будуть неперервно фіксуватись і на його виходах протягом всієї дії тактових сигналів.

Реагують на зміну інформації і асинхронні тригери, оскільки тактовані тригери при постійному дозвільному рівні сигналу на вході С практично перетворюються в асинхронні. Іншими словами, тригери з управлінням за рівнем реагують на зміну інформації в процесі її запису, що потрібно враховувати при їх застосуванні.

Розгляд тригерів з управління за рівнем важливий ще з тієї причини, що вони мають достатньо самостійне застосування та є основними (базовими) комірками всіх тригерних пристроїв, виконуючи в них функції тригерів певного функціонального типу. Для такого їх використання важливо, щоб вони задовольняли такі вимоги: мали повну систему переходів та були б якомога простішими.

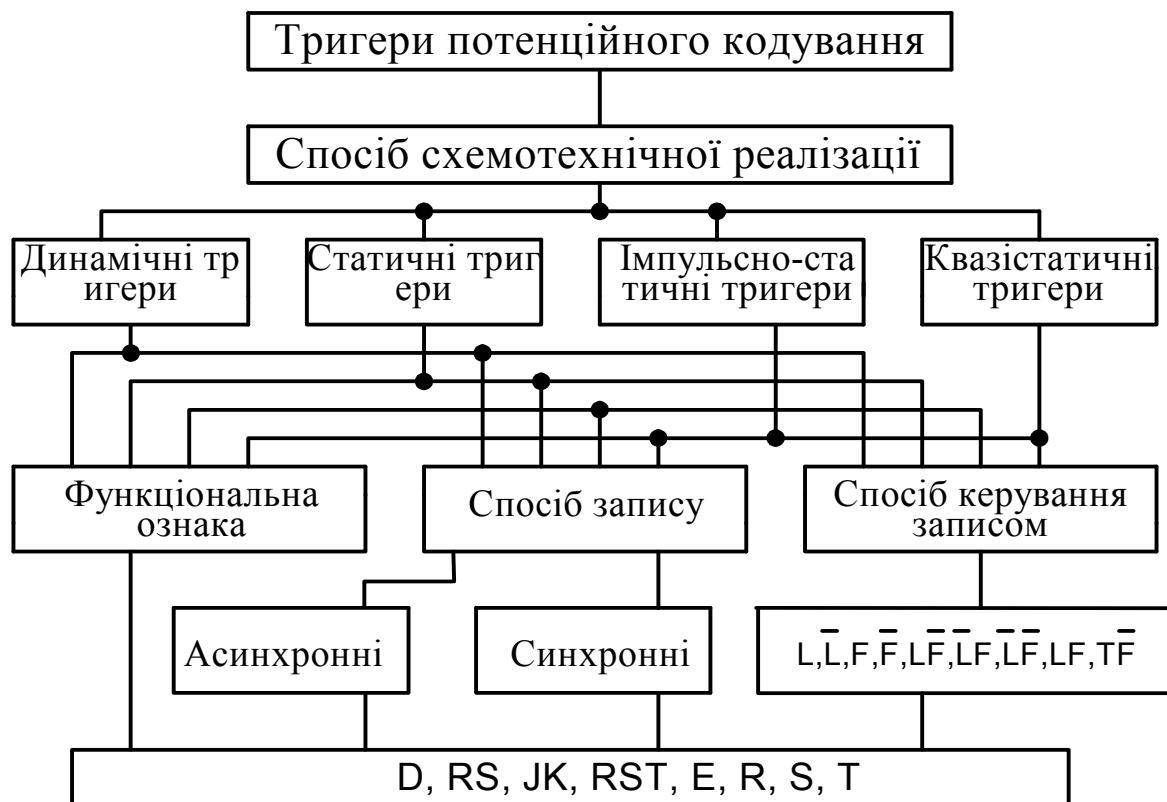


Рисунок 5.3 – Загальна класифікація тригерних приладів

Тригерні пристрої, в яких відсутні пристрої управління, далі будемо називати елементарними або простими. Вимога повноти переходів означає,

що для двох стійких станів тригер має мати такі переходи вхідних сигналів: $0 \rightarrow 0$; $0 \rightarrow 1$; $1 \rightarrow 0$; $1 \rightarrow 1$. Переходи $0 \rightarrow 0$ та $1 \rightarrow 1$ означають, що при дії відповідних вхідних сигналів тригер має зберігати свій попередній стан. Вимога реалізації з мінімальними затратами за кількістю електронних компонентів вказує на те, що тригер має бути асинхронним автоматом з якомога меншою кількістю входів.

Синхронні (тактовані) тригери фактично можна розглядати як тип асинхронних тригерів, для яких введені певні обмеження на можливість дії інформаційних сигналів, що дозволяє суттєво спростити їх синтез та отримання часових діаграм. В синхронних тригерах допускається змінювати інформаційні сигнали тільки впродовж періодів, коли тактові сигнали блокують вхідні кола та зміни стану тригера (синхронний тригер має змінювати свій стан тільки під дією тактового імпульса). Важлива перевага синхронних тригерів – маскування ефектів затримок, які обумовлені затримками логічних елементів та лініями передачі інформації. Остання обставина дозволяє вважати, що логічні елементи та лінії передачі мають нульову затримку, а виходи та стани тригера можна визначати тільки у фіксовані моменти, які викликають інтерес. Ці припущення суттєво спрощують аналіз і синтез синхронних пристроїв.

Далі розглядається реалізація тільки найважливіших типів тригерів, а саме: *RS*-, *D*-, *DV*-, *JK*- та *T*-тригерів.

5.3.1 *RS*-тригер та його різновиди

Асинхронний *RS*-тригер – двовходовий, який при поданні активного сигналу на *S*-вхід і неактивного сигналу на *R*-вхід встановлюється в одиничний стан; при поданні активного сигналу на *R*-вхід і неактивного сигналу на *S*-вхід встановлюється в нульовий стан; одночасна подача двох активних сигналів на *R*- та *S*-входи заборонена, якщо ж така ситуація виникне, то стан тригера невизначений.

Різновидами *RS*-тригера є:

- *S*-тригер – двовходовий, працює як *RS*-тригер, який за одночасного подання на входи двох активних сигналів встановлюється в одиничний стан;
- *R*-тригер – двовходовий, працює як *RS*-тригер, який за одночасного подання на входи двох активних сигналів встановлюється в нульовий стан;
- *E*-тригер (*Exclusive* – особливий) – двовходовий, працює як *RS*-тригер, який за одночасного подання на входи двох активних сигналів встановлюється в попередній стан.

Функціонування *RS*-тригера можна описати таблицею переходів. Якщо за активний сигнал на *R*- та *S*-входах прийняти рівень 1, то отримаємо табл. 5.1.

У графі <Номер набору> записують десятковий еквівалент двійкового числа, яке подано змінними *R*, *S* та *Q_i*. Змінна *R*, яка стоїть в наборі *R S Q_i* зліва, вважається старшим розрядом двійкового числа. Наявність графі

<Номер набору> дозволяє швидко та без помилок заповнювати карти Карно, використовувани далі.

Таблиця 5.1 – Таблиця переходів асинхронного RS -тригера

Номер набору	R	S	Q_t	Q_{t+1}	$\overline{Q}_{t+1}$
0	0	0	0	(0)	1
1	0	0	1	(1)	0
2	0	1	0	(1)	0
3	0	1	1	(1)	0
4	1	0	0	(0)	1
5	1	0	1	0	1
6	1	1	0	×	×
7	1	1	1	×	×

З табл. 5.1 видно, що RS -тригер зберігає один зі стійких станів незалежно від багаторазової зміни інформаційного сигналу на одному вході при нульовому значенні інформаційного сигналу на іншому вході. Ця властивість «блокування» є основною функціональною властивістю RS -тригера, саме вона робить його елементарною КП.

У графі Q_{t+1} (табл. 5.1) записують значення вхідної змінної Q , яких вона набуває в момент

часу $t+1$. Якщо $Q_{t+1} = Q_t$, то такий стан тригера стійкий і в графі Q_{t+1} стан тригера записують в дужках; якщо ж $Q_{t+1} \neq Q_t$, то стан тригера нестійкий і в графі Q_{t+1} стан тригера записують без дужок. Щодо останньої ситуації можливі два випадки:

- тригер за одного й того ж набору вхідних змінних переходить у стійкий стан (перехід позначений стрілкою);
- тригер за одного й того ж набору вхідних змінних буде постійно змінювати свій стан, тобто буде знаходитись в автоколивальному режимі. Останнє вказує на те, що таким тригером не можна управляти потенційними сигналами. Аналіз процесу перемикавання тригера показує, що дві його точки, в яких діють сигнали Q_t та Q_{t+1} , фізично відповідають одній і тій же його точці, де вихідний сигнал тригера після зміни вхідних сигналів встановиться не раніше, ніж через час затримки перемикавання $t_{\text{зат.пер.}}$. Тому тригер буде знаходитись у стійкому стані, якщо через час $t_{\text{зат.пер.}}$ після зміни вхідних сигналів він не змінить свій стан, та буде знаходитись в нестійкому стані, якщо через час $t_{\text{зат.пер.}}$ після зміни вхідних сигналів змінить свій стан на протилежний.

З табл. 5.1 випливає, що на всіх наборах вхідних змінних R - та S -тригери мають стійкий стан, причому за правильної роботи тригерного пристрою набір змінних $RS = 11$ не виникає, тому значення Q_{t+1} на цьому наборі не має значення. В табл. 5.1 таке значення Q_{t+1} позначено знаком $\times$. Відсутність нестійких станів у RS -тригера вказує на те, що його характери-

стичне рівняння повністю відповідає структурі тригера, яким управляють потенційні сигнали.

Визначимо характеристичне рівняння RS -тригера, записуючи (5.1) в ДДНФ та мінімізуючи останню за допомогою методу карт Карно. ДДНФ отримують як логічну суму мінтермів для вхідних наборів, на яких функція приймає одиничне значення, причому символ будь-якої змінної в кожному мінтермі беруть з інверсею, якщо функція на цьому вхідному наборі дорівнює 0. Якщо функція на деяких наборах невизначена, то її довизначають таким чином, щоб підсумкова мінімальна ДНФ функції була якомога простішою. Запишемо формулу $Q_{t+1} = f(x_t, Q_t)$ та зобразимо її на карті Карно (рис. 5.4, б).

Довизначення функції одиницями дає таке характеристичне рівняння RS -тригера:

$$Q_{t+1} = S + \bar{R}Q_t. \quad (5.2)$$

Заборону комбінації інформаційних сигналів $RS = 11$ запишемо у вигляді вимоги

$$RS = 0. \quad (5.3)$$

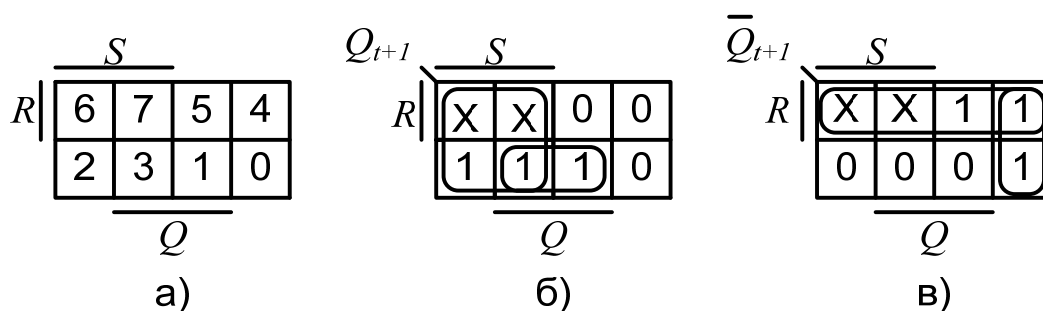


Рисунок 5.4 – Карти Карно RS -тригера: а) – еталонна; б) – для Q_{t+1} ; в) – для $\bar{Q}_{t+1}$

Вибираємо за елементну базу ЛЕ типу АБО-НІ та перетворюємо (5.2), використовуючи правила де Моргана, до вигляду

$$\bar{Q}_{t+1} = \overline{S + R + \bar{Q}_t}. \quad (5.4)$$

На основі табл. 5.1 та рис. 5.3, в маємо

$$\bar{Q}_{t+1} = R + \bar{S}\bar{Q}_t. \quad (5.5)$$

Застосовуючи правила де Моргана ще раз, вираз (5.5) перетворимо як

$$Q_{t+1} = \overline{R + \bar{S}\bar{Q}_t}. \quad (5.6)$$

Отже, з формул (5.4) та (5.6) зрозуміло, що RS -тригер є послідовним з'єднанням двох ЛЕ АБО-НІ, яке замкнене само на себе (рис. 5.5, а).

Вибираючи як елементну базу ЛЕ І-НІ та використовуюючи правила де Моргана перетворимо (5.2) та (5.3)

$$Q_{t+1} = \overline{\overline{S}RQ_t}, \overline{R} + \overline{S} = 1. \quad (5.7)$$

З (5.7) випливає, що $\overline{R}\overline{S}$ -тригер є послідовним з'єднанням двох ЛЕ І-НІ, яке замкнене само на себе (рис. 5.5, в). Отже RS та $\overline{R}\overline{S}$ – дуальні тригери.

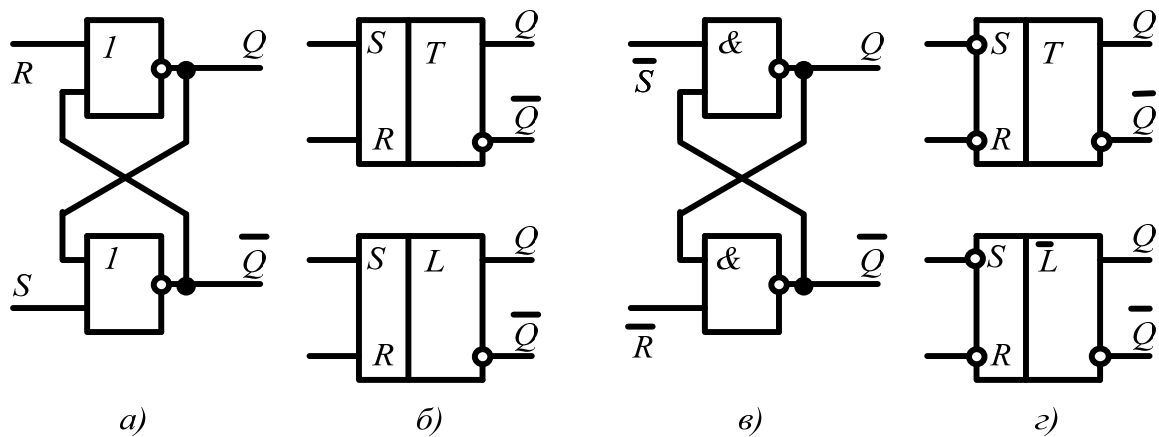


Рисунок 5.5 – Схема RS -тригера на ЛЕ АБО-НІ а) та на ЛЕ І-НІ в) і його умовні позначення б), г)

На рис. 5.6 закон функціонування RS -тригера зображений у вигляді графу. Вершини графу позначають колами, всередині яких записують стани тригерів, а дуги графу визначають переходи тригера, які він здійснює під дією входніх сигналів. Відсутність на графі комбінації входніх сигналів $RS = 11$ говорить про те, що вона заборонена.

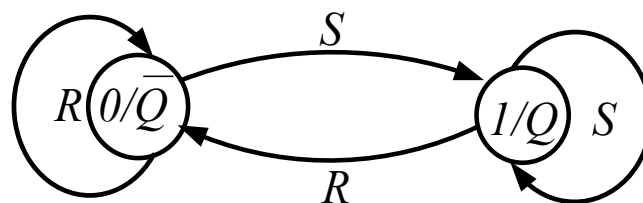


Рисунок 5.6 – Граф асинхронного RS -тригера

За допомогою характеристичних рівнянь (5.2), (5.6), (5.7) можна визначити стан тригера Q_{t+1} , в який він перейде в момент часу $t+1$, якщо відомі

комбінації вхідних сигналів та стан тригера Q_t в попередній момент часу t . При проектуванні тригерних пристроїв часто потрібно розв'язувати обернену задачу, а саме: визначати комбінацію вхідних сигналів для здійснення потрібного переходу з одного стану в інший (другий етап канонічного методу синтезу). Результатом розв'язання цієї задачі буде отримання характеристичної таблиці. Для RS -тригера на ЛЕ АБО-НІ це табл. 5.2, а для ЛЕ І-НІ це табл. 5.3.

Характеристична таблиця може бути отримана також з таблиці переходів, з характеристичного рівняння, з графу.

Нехай закон функціонування RS -тригера заданий таблицею переходів (див. табл. 5.1). Потрібно визначити комбінації вхідних сигналів, при яких тригер здійснить перехід 00. Тригер цей перехід здійснює в рядках, що відповідають наборам 0 та 4. На наборі 0 сигнали $R = 0$, $S = 0$, а на наборі 4 сигнали $R = 1$, $S = 0$. Для того, щоб RS -тригер здійснив потрібний перехід, необхідно на вхід S подати 0, а на вхід R – або 0, або 1 (1-й рядок табл. 5.2). Аналогічно визначаються R та S на переходах 01, 10 і 11.

Таблиця 5.2 – Характеристична таблиця асинхронного RS -тригера на ЛЕ І-НІ

Перехід $Q_t \rightarrow Q_{t+1}$	R^*	S^*
00	×	1
01	1	0
10	0	1
11	1	×
ЛЕ І-НІ		

Таблиця 5.3 – Характеристична таблиця асинхронного RS -тригера на ЛЕ АБО-НІ

Перехід $Q_t \rightarrow Q_{t+1}$	R^*	S^*
00	×	0
01	0	1
10	1	0
11	0	×
ЛЕ АБО-НІ		

Нехай функціонування RS -тригера задано характеристичним рівнянням (5.2). Потрібно визначити комбінації вхідних сигналів, за яких тригер здійснює перехід 01. Після підстановки у (5.2) значень $Q_t = 0$ та $Q_{t+1} = 1$ отримаємо рівняння $1 = S + \bar{R}$, яке буде тотожністю при комбінаціях $RS = 01$ або 11 , але, враховуючи, що ком-

бінація 11 заборонена, перехід 01 може здійснюватись тільки при $R = 0$ та $S = 1$ (2-й рядок табл. 5.2). Аналогічно визначають значення R та S при переходах 00, 10 і 11.

Нехай далі функціонування RS -тригера задано графом (див. рис. 5.5). Потрібно визначити комбінації вхідних сигналів, за яких тригер здійснить перехід 11. Поряд з петлею у вершині зі станом «1» записані дві комбінації RS : 00 та 01. Це означає, що на вхід R потрібно подати 0, а на вхід S – або 0, або 1 (4-й рядок табл. 5.2). Аналогічно визначаються значення R та S при переходах 00, 01 і 10.

Існують різновиди RS -тригерів, для яких комбінація сигналів $R^t = S^t = 1$ не заборонена. R -тригер відрізняється від RS -тригера тим, що при

комбінації вхідних сигналів $S^t = R^t = 1$ він переходить у нульовий стан ($Q^{t+1} = 0$). S -тригер у цьому випадку переходить в одиничний стан ($Q^{t+1} = 1$), а E -тригер не змінює свого стану ($Q^{t+1} = Q^t$).

5.3.2 RS-тригер з синхронним записом (СЗ)

Тактований одиничним рівнем RS -тригер функціонує згідно з табл. 5.4. Нехай синхронний RS -тригер має бути виконаний на ЛЕ І-ІІ. З табл. 5.4 видно, що цей тригер зберігає свій стан при $C = 0$ та працює як асинхронний RS -тригер при $C = 1$. З карти Карно (рис. 5.7, б) маємо

$$Q_{t+1} = SC + \bar{R}Q_t + Q_t\bar{C}, \quad (5.13)$$

Звідси при $C = 1$ отримаємо (5.2), а при $C = 0 - Q_{t+1} = Q_t$.

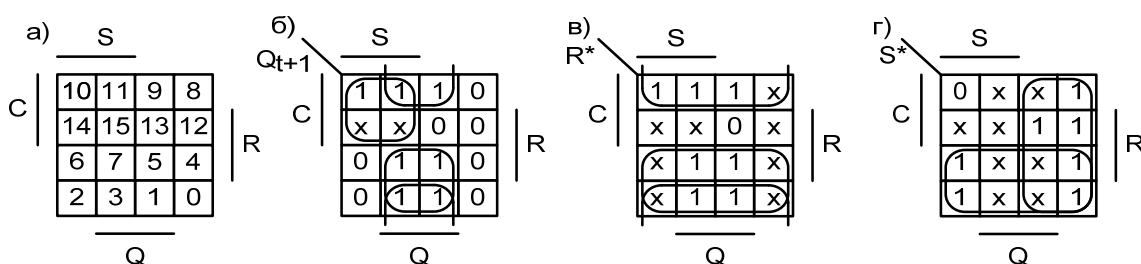


Рисунок 5.7 – Карти Карно тактованого RS -тригера: а) – еталонна; б) – для Q_{t+1} ; в) – R^* ; г) – для S^*

З карти Карно (рис. 5.7, в), виходить, що

$$R^* = \bar{R} + \bar{C} = \overline{R C}; \quad (5.14)$$

$$S^* = \bar{S} + \bar{C} = \overline{S C}. \quad (5.15)$$

Таблиця 5.4 – Таблиця переходів синхронного RS -тригера

Набір	C	R	S	Q_t	Q_{t+1}	Набір	C	R	S	Q_t	Q_{t+1}
0	0	0	0	0	0	8	1	0	0	0	0
1	0	0	0	1	1	9	1	0	0	1	1
2	0	0	1	0	0	10	1	0	1	0	1
3	0	0	1	1	1	11	1	0	1	1	1
4	0	1	0	0	0	12	1	1	0	0	0
5	0	1	0	1	1	13	1	1	0	1	0
6	0	1	1	0	0	14	1	1	1	0	X
7	0	1	1	1	1	15	1	1	1	1	X

Вирази (5.14) і (5.15) визначають значення сигналів для встановлення RS -тригера в «1» стан, який показано на рис. 5.78, а. Синхронний RS -

тригер, як і асинхронний RS -тригер, має заборонену комбінацію $RS = 11$. Такий тригер часто використовують для побудови регістрів.

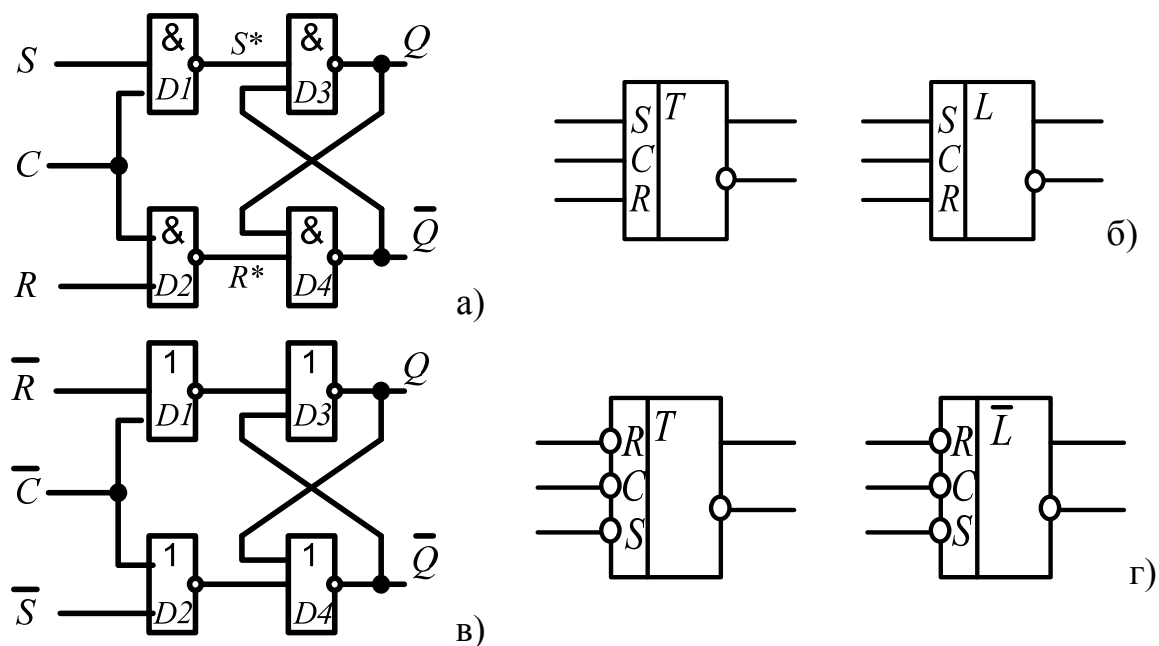


Рисунок 5.8 – Схеми тактованих RS -тригерів на елементах І-НІ а), АБО-НІ в) та їх умовне позначення б), г)

На рис. 5.8, в наведена схема тригера RS -типу, якою управляють сигналом з рівнем логічного «0». Для такого тригера заборонена комбінація двох «0» на його інформаційних входах при наявності тактових сигналів з рівнем 0 ($C = 0$).

Розглянемо принцип роботи синхронного RS -тригера на ЛЕ І-НІ. В такому тригері (рис. 5.8, а) входи S і R – інформаційні, вхід C – синхронізувальний.

1. За відсутності тактового сигналу ($C = 0$) ЛЕ $D1$ і $D2$ закриті незалежно від стану R - і S -входів і стан тригера не зміниться (рис. 5.9 – часові діаграми роботи тригера з урахуванням затримок ЛЕ).

2. При $S = 1$, $R = 0$, $C = 1$ на виході ЛЕ $D1$ формується «0», який встановлює ЛЕ $D3$ в «1». Дві «1» на вході ЛЕ $D4$ встановлюють $Q = 0$.

3. При $S = 0$, $R = 1$, $C = 1$ на виході ЛЕ $D2$ формується «0», який встановлює ЛЕ $D4$ в «1», а потім дві «1» на вході ЛЕ $D3$ створюють $Q = 0$.

З часової діаграми на рис. 5.9 видно, що входні сигнали перемикають тригер з приходом сигналу синхронізації. Зміна сигналів на входах R та S дозволена тільки між синхронізувальними сигналами.

Швидкодія RS -тригера характеризується такими параметрами ($t_{з,ср}$ – середня затримка одного ЛЕ):

а) максимальна затримка перемикання

$$\tau_t = 3 t_{3.ср.};$$

б) тривалість τ_i вхідного сигналу для стійкого функціонування тригера

$$\tau_i > 3 t_{3.ср.}.$$

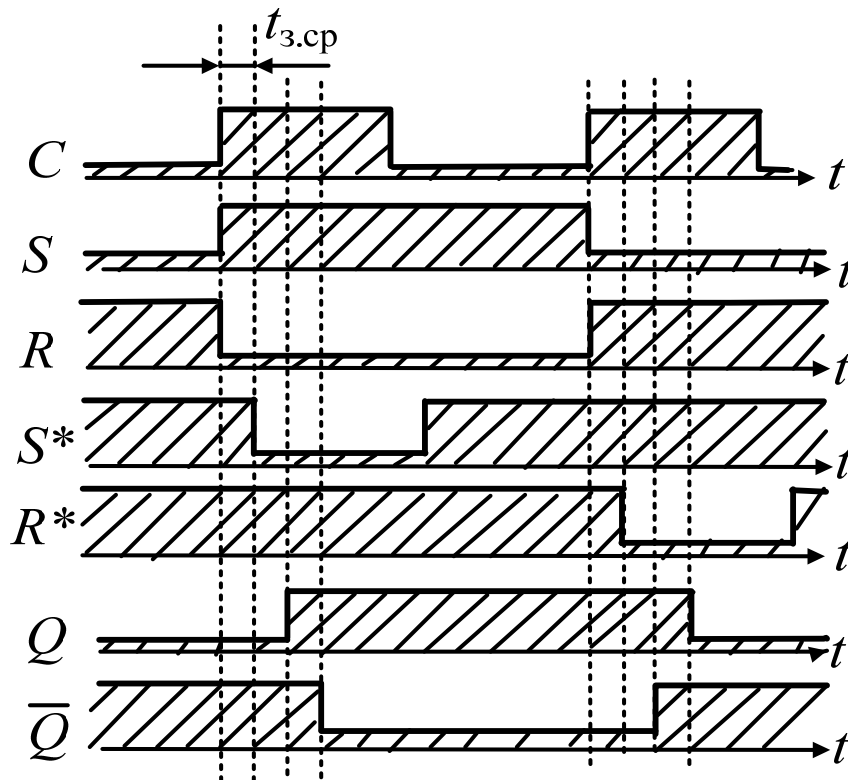


Рисунок 5.9 – Часова діаграма роботи *RS*-тригера на ЛЕ I-НІ

5.3.3 *JK*-тригер

Як видно з табл. 5.5, цей тригер має повну систему переходів і, разом з тим, для нього відсутні заборонені комбінації вхідних сигналів. Характеристичне рівняння тригера має вигляд

$$Q_{t+1} = J_t \overline{Q}_t + \overline{K}_t Q_t. \quad (5.16)$$

Побудова тригера безпосередньо за рівнянням (5.16) приводить до схеми асинхронного *JK*-тригера, показаного на рис. 5.10. Згідно з табл. 5.5 такий тригер при $J = K = 0$ може знаходитися у стані 0 ($Q = 0$) або у стані 1 ($Q = 1$).

Таблиця 5.5 – Таблиця переходів асинхронного JK -тригера

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

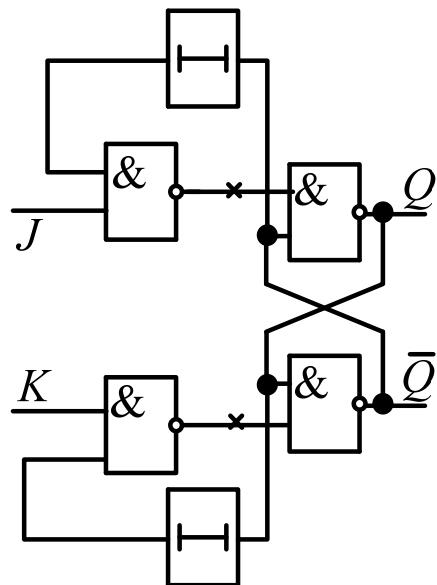


Рисунок 5.10 – Схема асинхронного JK -тригера

При комбінації сигналів $J = K = 1$ тригер, як це видно з табл. 5.5, повинен переключитися в інверсний стан. Тоді, якщо, наприклад, $Q = 1$, тригер встановлюється в 0 ($Q = 0$). Якщо ж тригер знаходився в стані 0 ($Q = 0$), то він прийме стан $Q = 1$. З цього видно, що у випадку, коли на входах J та K рівень «1» діє постійно, то тригер буде знаходитися в коливальному режимі, по чергово перемикаючись з 1 в 0 та з 0 в 1. Останнє означає, що при комбінації сигналів $J = K = 1$ тригером потрібно управляти не потенціальними, а імпульсними сигналами, тобто сигналами обмеженої та, в даному випадку, жорстко фіксованої тривалості. Оскільки для виконання такої вимоги в склад тригера необхідно додатково ввести елемент затримки, який вмикається в коло зворотного зв'язку тригера, то схема, яка наведена на рис. 5.10 не є елементарною.

5.3.4 JK -тригер з синхронним записом

Як видно з табл. 5.6, при комбінації вхідних сигналів $J = K = 0$, $J = 0$ та $K = 1$, $J = 1$ та $K = 0$ тригер функціонує, як RS -тригер (вхід J відповідає входу S , а $K - R$), а при $J = K = 1$ змінює свій стан на протилежний, тобто працює як лічильний тригер.

Таблиця 5.6 – Таблиця переходів синхронного JK -тригера

J_t	K_t	Q_{t+1}
0	0	Q^n
0	1	0
1	0	1
1	1	$\bar{Q}^n$

В JK -тригері момент прийняття вхідної інформації визначається внутрішнім зворотним зв'язком з виходів тригера. Якщо на обидва входи J та K подана «1» та якщо синхронізувальний сигнал довший за тривалістю, ніж час спрацювання тригера (що

звичайно і відбувається), то в цьому випадку на виходах JK -тригера з СЗ буде спостерігатись генерація. Для того, щоб не було генерації на виході тригера, у його зворотні зв'язки вмикають елементи затримки (рис. 5.11). З цієї причини статичне управління записом в JK -тригері не має сенсу.

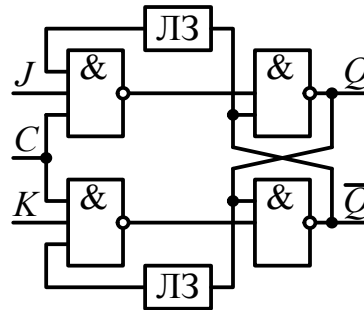


Рисунок 5.11 – Синхронний JK -тригер

Умови правильної роботи тригерів з елементами затримки

$$t_i > t_{лз}, \text{ і } t_{лз} > t_{снр}.$$

5.3.5 Т-тригер

В табл. 5.7 наведено таблицю переходів ще одного розповсюдженого тригера T -типу. Цей тригер має один інформаційний вхід T . З табл. 5.7 видно, що при відсутності інформаційного сигналу ($T = 0$) тригер T -типу зберігає свій попередній стан 0 або 1, а з подачею інформаційного сигналу завжди переходить в інверсний стан. Оскільки зміна станів тригера під дією вхідних сигналів еквівалентна підрахунку вхідних сигналів за модулем 2, то його часто називають лічильним тригером або тригером з лічильним входом. Характеристичне рівняння T -тригера має вигляд

$$Q_{t+1} = T_t \bar{Q}_t + \bar{T}_t Q_t \quad (5.17)$$

Таблиця 5.7 – Таблиця переходів асинхронного T -тригера

Q_t	T_t	Q_{t+1}
0	0	0
0	1	1
1	0	1
1	1	$\bar{Q}_t$

Безпосередня побудова тригера за рівнянням (5.17) дає схему T -тригера, показану на рис. 5.13, а граф його переходів – на рис. 5.12.

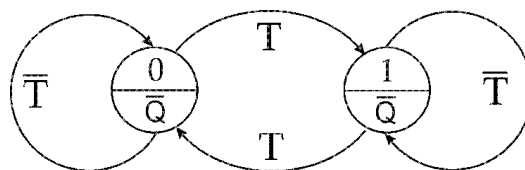


Рисунок 5.12 – Граф асинхронного T -тригера

З табл. 5.7 видно, що при $T = 1$ тригер має нестійкий стан ($Q_{t+1} = \bar{Q}_t$), тобто знаходиться в автоколивальному режимі. Це вказує на те, що асинхронний Т-тригер в принципі може бути реалізований на одній елементарній КП.

Аналіз роботи Т-тригера з імпульсним управлінням є нетривіальним, оскільки при $T = 0$ тригер має стійкий стан.

Аналіз часових діаграм показує, що для стійкої роботи Т-тригера необхідно, щоб $t_{\text{ім}} \geq 2t_{\text{зат.роз.сер.}}$, а для запобігання режиму генерації – щоб

$t_{\text{ім}} \geq 3t_{\text{зат.роз.сер.}}$. Труднощі забезпечення цих досить жорстких вимог в інтегральній схемотехніці унеможливають реалізацію асинхронного Т-тригера за схемою на рис. 5.13.

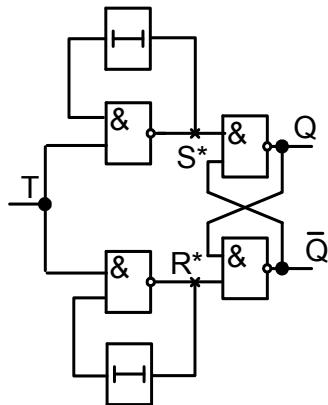


Рисунок 5.13 – Схема асинхронного Т-тригера

Порівнюючи схеми тригерів Т- (рис. 5.13) та JK-типів (див. рис. 5.10), неважко помітити, що тригер Т-типу можна отримати з тригера JK-типу за допомогою з'єднання його інформаційних входів. Отже, цей тригер, як і тригер JK-типу на рис. 5.10, належить до тригерів, де управління реалізовано імпульсними сигналами. Однак необхідно відзначити, що розглянуті вище елементарні тригери реалізуються з мінімальними затратами щодо кількості типових елементів та саме вони використовуються при синтезі більш складніших тригерних пристроїв.

5.3.6 Т-тригер з СЗ також має один логічний вхід – T (*trigger*). Якщо на цей вхід подати «1», то з кожним СС тригер переходить в протилежний стан; якщо ж на вході знаходиться «0», то тригер залишається в попередньому стані (табл. 5.8).

Характеристичне рівняння тригера:

$$Q_{t+1} = T_t Q_t + \bar{T}_t \bar{Q}_t.$$

Схема Т-тригера на ЛЕ І-НІ показана на рис. 5.14.

Таблиця 5.8 – Таблиця переходів синхронного JK-тригера

J_t	K_t	Q_{t+1}
0	0	Q_t
0	1	0
1	0	1
1	1	$\bar{Q}_t$

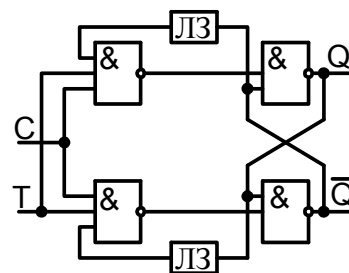


Рисунок 5.14 – Синхронний Т-тригер зі статичним управлінням записом

5.3.7 D-тригер Асинхронний D -тригер функціонує згідно з табл. 5.9 (ЛЕ АБО-НІ). На рис. 5.15 закон функціонування D -тригера зображений у вигляді графу.

Таблиця 5.9– Таблиця переходів асинхронного D -тригера

Набір	D	Q_t	Q_{t+1}
0	0	0	0
1	0	1	0
2	1	0	1
3	1	1	1

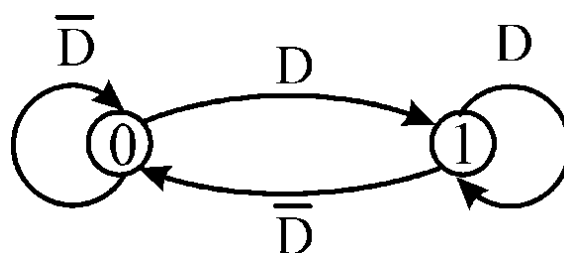


Рисунок 5.15 – Граф асинхронного D -тригера

На рис. 5.16 наведені діаграми Вейча D -тригера. За аналогії з розглянутим вище RS -тригером з рис. 5.17 маємо

$$Q_{t+1} = D. \quad (5.18)$$

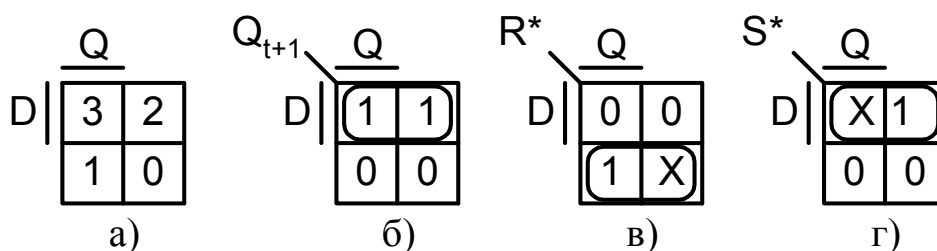


Рисунок 5.16 – Діаграми Вейча D -тригера а) – еталонна; б) – для Q_{t+1} ; в) – для R^* ; г) – для S^*

$$R^* = \bar{D} \quad (5.19)$$

$$S^* = D \quad (5.20)$$

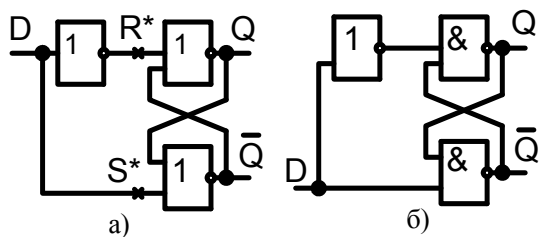


Рисунок 5.17 – Схема D -тригера в базисі а) АБО-НІ; б) І-НІ

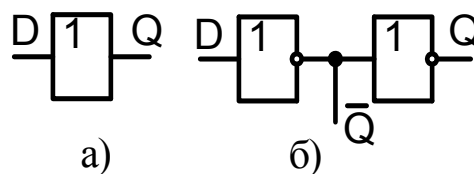


Рисунок 5.18 – Тривіальна реалізація схеми D -тригера з однофазним а) та парафазним б) виходами

Формула (5.18) вказує на те, що отриманий тригер (рис. 5.17) є тривіальним, оскільки його можна отримати з одного повторювача або двох послідовно ввімкнених інверторів (рис. 5.18), якщо необхідно парафазне подання вихідних сигналів. Пристрій на рис. 5.17 іноді називають *RS*-тригером з примусовим парафазним встановленням.

Таблиця 5.10 – Таблиця переходів синхронного *D*-тригера

Набір	C	D	Q_t	Q_{t+1}
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
7	1	1	1	1

5.3.8 *D*-тригер з СЗ Тригером *D*-типу, відомим у літературі як тригер затримки, називають елементарний автомат з одним інформаційним входом. В процесі роботи *D*-тригер з СЗ синхронізують рівнем сигналу, тобто потенціалом тактувального сигналу. *D*-тригер із статичним управлінням називають ще «тригер-защипка». Цей тригер функціонує згідно з табл. 5.10, звідки видно, що при $C = 0$ він зберігає свій стан, а при $C = 1$ працює як асинхронний *D*-тригер.

З карти Карно (рис. 5.19, б) маємо $Q_{t+1} = DC + Q_t \bar{C}$, що для $C=1$ відповідає формулі (5.18), а для $C=0$ маємо $Q_{t+1} = Q_t$.

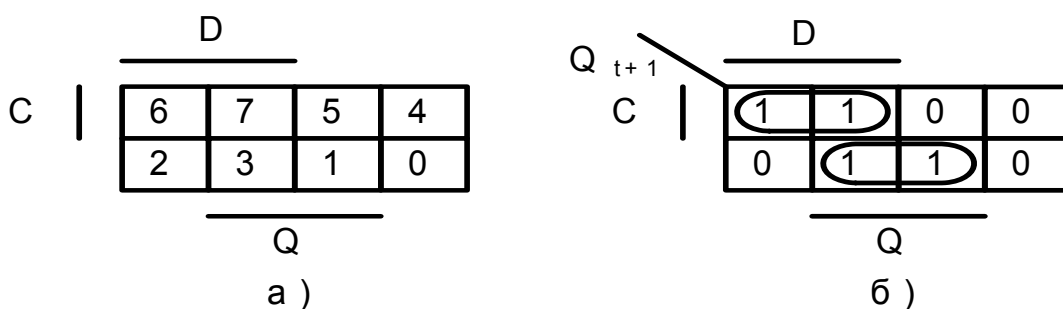


Рисунок. 5.19 – Карти Карно для тактованого *D*-тригера: а) -еталонна; б) -для Q_{t+1}

Якщо вхід *D* з'єднати з виходом $\bar{Q}$ -тригера, *D*-тригер буде працювати як лічильний тригер, але, оскільки він містить тільки одну КП, лічильний тригер на базі такого *D*-тригера може управлятися тільки імпульсним сигналом, тому в такому режимі цей пристрій і відповідну схему не використовують.

На рис. 5.20 показана структура тактованого *D*-тригера, який широко використовують для побудови регістрів та в пристроях управління. Синхронний *D*-тригер зі статичним записом можна отримати з *RS*-тригера, якщо на вхід *R* подати інверсію сигналу на вході *S*. Але доцільніше для цього використовувати елементи, які вже є в тригері.

Розглянемо принцип роботи D -тригера.

1. За відсутності тактового сигналу ($C = 0$) ЛЕ 1 та 2 знаходяться в одиничному стані ($S^* = 1$, $R^* = 1$) незалежно від інформації на D -вході, стани тригера не змінюються.

2. При $C = 1$, $D = 1$ на виході ЛЕ 1 формується рівень «0» ($S^* = 0$), який встановлює елементи 2 та 3 в «1» ($R^* = 1$, $Q = 1$), потім ці «1» підтримують «0» на виході ЛЕ 4.

3. При $C = 1$, $D = 0$ на виході ЛЕ 1 буде «1» ($S^* = 1$), яка разом з $C = 1$ формує «0» на виході ЛЕ 2 ($R^* = 0$), рівень «0» встановлює ЛЕ 4 в «1». Дві логічні «1» на вході ЛЕ 3 підтримують «0» на його виході ($Q = 0$).

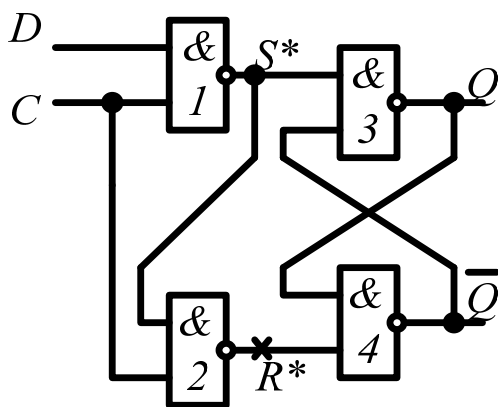


Рисунок 5.20 – Схема тактованого D -тригера на ЛЕ I-НІ

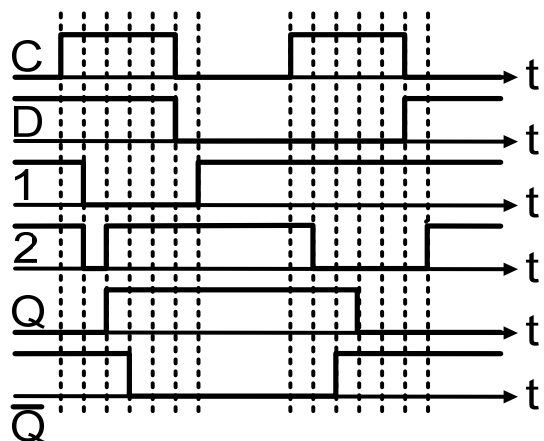


Рисунок 5.21 – Часова діаграма D -тригера

Часова діаграма D -тригера наведена на рис. 5.21. Швидкодія D -тригера характеризується параметрами:

а) затримкою перемикання $\tau_t = 3t_{з.сер.}$;

б) тривалістю τ_i вхідного сигналу; для стійкого функціонування тригера $\tau_i > 3t_{з.сер.}$. Відомі також інші різновиди D -тригерів зі статичним управлінням записом на ЛЕ I-АБО-НІ та на ЛЕ АБО-НІ.

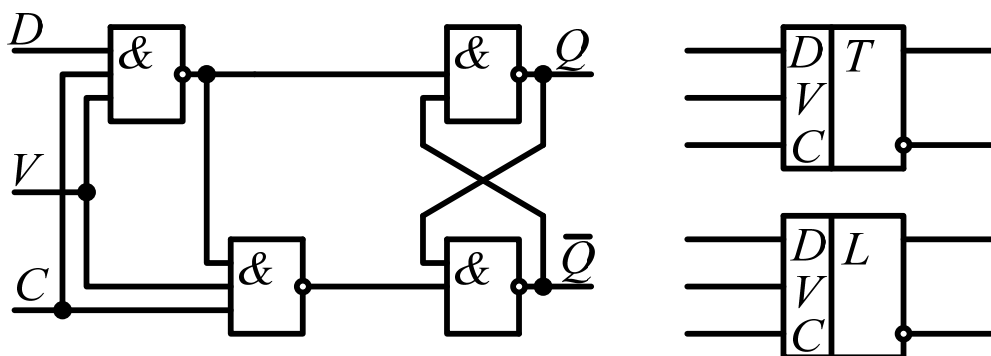


Рисунок 5.22 – Синхронний DV -тригер зі статичним записом та його УГП

Таблиця 5.11 –
Таблиця переходів
синхронного
T-тригера

T_t	Q_{t+1}
0	$Q(t)$
1	$\bar{Q}(t)$

інформацію, що надходить на вхід.

Характеристичне рівняння цього тригера має вигляд

$$Q^{t+1} = (VD + QV)^t.$$

Тригери *D*- та *DV*-типу знаходять застосування при побудові лічильних пристроїв, регістрів зсуву, а також в пристроях управління.

5.3.9 *R*-, *S*- та *E*-тригери

Тактовані рівнем «1» *R*-, *S*- та *E*-тригери при $C = 0$ зберігають свій попередній стан, а при $C = 1$ працюють так, як асинхронні *R*-, *S*- та *E*-тригери. Процедура синтезу цих тригерів аналогічна процедурі синтезу асинхронного *RS*-тригера. Схеми цих тригерів наведені, відповідно, на рис. 5.23, а)–в).

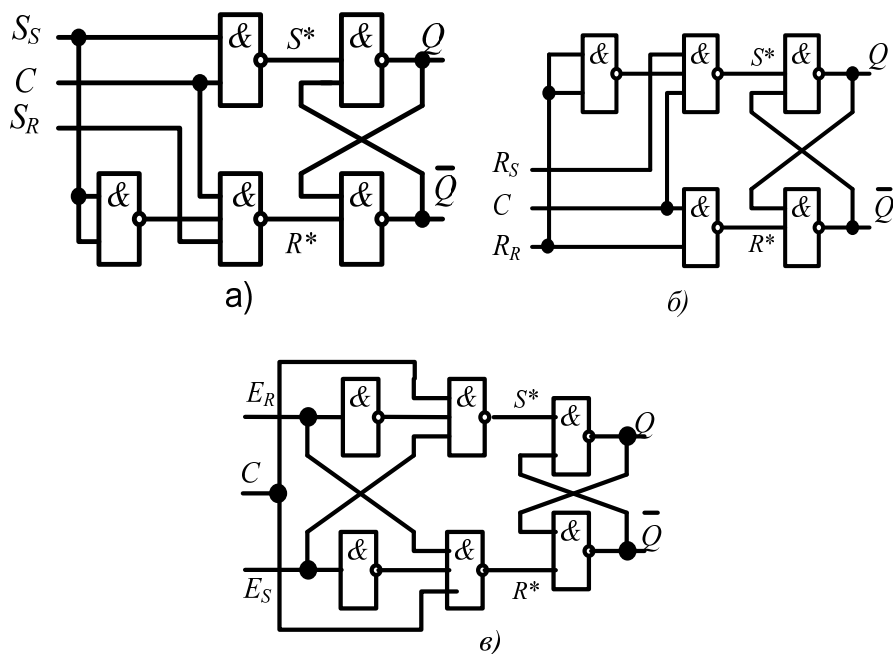


Рисунок 5.23 – Схеми синхронних *S*-, *R*- та *E*-тригерів

Тактовані *R*-, *S*- та *E*-тригери також широко використовуються у пристроях управління різних цифрових систем.

5.4 Синхронні тригери з динамічним записом (ДЗ) інформації

Тригери, де управління записом інформації здійснюється за фронтом $[F]$, як і тригери, в яких управління записом відбувається за рівнем $[L]$, відносять до пристроїв, де прийом та фіксація інформації суміщені в часі. Такі тригери можуть бути синхронними та асинхронними.

Розглянемо синхронні тригери, RS -, D -, JK -типу, які отримали найширше використання в інтегральній схемотехніці.

У тригерах з ДЗ для запису інформації використовується не весь сигнал, а лише його фронт. За відсутності фронту перемикання тактового сигналу інформація на логічних входах може приймати будь-які значення. При цьому стан виходів тригера не змінюється. При надходженні фронту тактового сигналу відбувається запам'ятовування інформації внутрішньою пам'яттю та її фіксація на виходах тригера. Одночасно пристрій управління генерує сигнал блокування (його тривалість дорівнює тривалості тактового сигналу) прийому інформації, який сприймається вхідними колами протягом всього тактового сигналу.

Необхідно мати на увазі, що блокування входів може бути повним або частковим (останнє тільки у тактованих тригерів). Тактовані тригери з повним блокуванням входів характеризуються тим, що при надходженні фронту тактового сигналу забороняється прийом інформації для будь-яких комбінацій вхідних сигналів. У тригерах з неповним блокуванням входів можливі комбінації вхідних сигналів, при яких сигнал блокування не формується. В результаті при таких комбінаціях тригери реагують на зміну інформації не тільки під час дії фронту тактового сигналу, але й під час дії його рівня. Тригери, які спрацьовують за фронтом «0»/«1» та «1»/«0», позначимо індексами $F, \bar{F}$, якщо вони мають повне блокування входів, та індексами $f, \bar{f}$, якщо вони цього не мають. Прикладами індексного позначення такого типу тригерів є: $C_F RS$, $C_F D$; $C_f JK$, $C_f RS$ і т. п.

Тригери з записом за фронтом $[F]$ складніші за своєю структурою на відміну від тригерів з управлінням за рівнем $[L]$, але більш завадостійкі, оскільки сигнал завади в такі тригери може записуватись лише в дуже короткий інтервал часу, який практично збігається з тривалістю фронту, тоді як в тригерах з управлінням за рівнем завада може прийматися тригером протягом всієї тривалості тактового сигналу.

5.4.1 D -тригер

Тут є три асинхронних тригери R - S -типу, один з яких основний ($D5$ - $D6$) та два допоміжних ($D1$ - $D4$) чи комутувальних. Короткочасне запам'ятовування вхідної інформації виконується ЛЕ $D1$ та $D4$ разом з $D2$ та $D3$. ЛЕ $D2$ та $D3$ служать також для управління роботою основного тригера. Розглянемо роботу D -тригера за умови $\bar{S} = \bar{R} = 1$.

1. ЛЕ $D2$ підтримує «0» на виході ЛЕ $D1$. Якщо при цьому надходить

тактовий сигнал $C = 1$, то в початковий момент на всіх входах ЛЕ $D3$ є «1», які встановлюють вихід ЛЕ $D3$ в «0».

2. Рівень логічного «0» з виходу ЛЕ $D3$ веде до появи «1» на виході ЛЕ $D6$, яка разом з «1» з ЛЕ $D2$ встановлює ЛЕ $D5$ в «0» ($Q = 0$). Стан верхнього комутувального ($D1$ та $D2$) тригера при цьому не змінюється.

3. При $D = 1$ на виході ЛЕ $D4$ буде «0», тоді відповідно на виході ЛЕ $D1$ підтримується «1». Якщо далі тактовий сигнал $C = 1$, то в початковий момент часу на всіх входах ЛЕ $D2$ присутні «1», які встановлюють ЛЕ $D2$ в «0». Рівень «0» з ЛЕ $D2$ приводить до появи «1» на виході ЛЕ $D5$ ($Q = 1$). На всіх входах ЛЕ $D6$ присутні «1», тому на його виході встановлюється «0». Стан нижнього (на елементах $D3$ та $D4$) комутувального тригера при цьому не зміниться.

Таким чином, тригер працює відповідно до рівняння $Q^{t+1} = D^t$ та є $C_F D$ -тригером. Важливою особливістю роботи тригера є те, що при зміні рівня CC , тобто при його тактуванні сигналами логічного 0, він також буде фіксувати інформацію по фронту CC . Такий режим називають роботою з внутрішньою затримкою, оскільки інформація на виході тригера з'являється після закінчення CC .

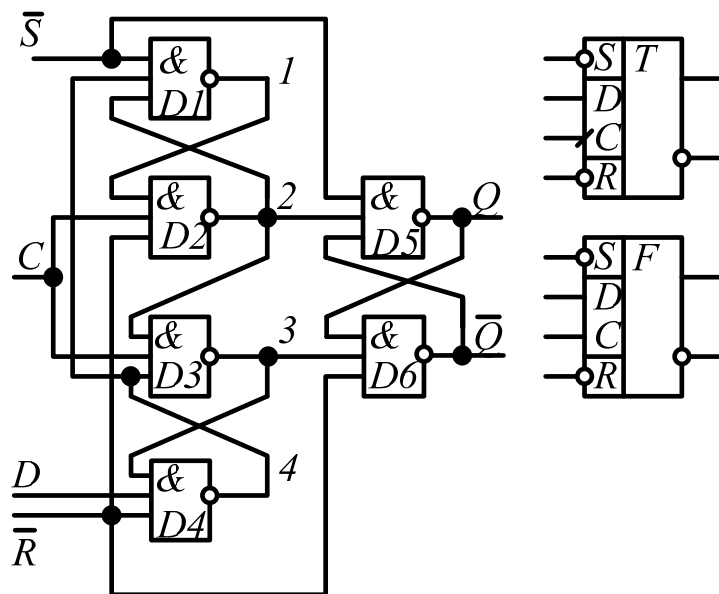


Рисунок 5.24 – Схема D -тригера ($C_F D$) та його умовне позначення

Швидкодія D -тригера за схемою «трьох тригерів» визначається:

а) затримкою перемикавання за інформаційним входом (в гіршому випадку $\tau_i = 5t_{3, \text{ср.}}$);

б) тривалістю вхідного синхронізувального сигналу, необхідною для стійкого функціонування D -тригера: $t_i > 3t_{3, \text{ср.}}$.

D -тригер на рис. 5.24 має також асинхронні (встановлювальні) входи $\overline{RS}$.

5.4.2 RS-тригер з ДЗ

На рис. 5.25 зображена схема CRS -тригера з ДЗ та його умовне позначення. Нехай тригер знаходиться у стані 1 ($Q = 1$), а на його входах діє комбінація сигналів $S = 0$, $R = 1$. У цьому випадку має місце такий розподіл рівнів сигналів на виходах ЛЕ $D3$ - $D6$: $D6 = 0$, $D3$ - $D5 = 1$. В результаті ЛЕ $D4$ буде підготовленим до перемикавання, бо на двох його входах з трьох діє рівень 1. При надходженні CC (момент t_1 на діаграмі рис. 5.26) на виході ЛЕ $D4$ сформується сигнал з рівнем 0, під дією якого тригер встановить-

ся у стан $Q = 0$, та одночасно цей сигнал буде утримувати ЛЕ D3 та D5 у стані 1, забороняючи тим самим прийом інформації у пристрій управління. Останнє означає, що з моменту появи сигналу на виходах ЛЕ D4 інформація на входах S- та R-тригера може набувати будь-яких значень (моменти t_1, t_3), але такі зміни не будуть впливати на стан тригера в цілому, тобто він залишиться у стані $Q = 0$. В момент t_5 сигнал з рівнем 0 сформується на виході ЛЕ D5, бо до моменту фронту F чергового CC саме цей ЛЕ буде підготовлений до перемикавання. В результаті тригер встановиться у стан $Q = 1$ та буде заборонений прийом інформації у пристрій управління, тобто тригер залишиться в стані $Q = 1$, хоча на його входах інформація буде змінена (момент t_6, t_7).

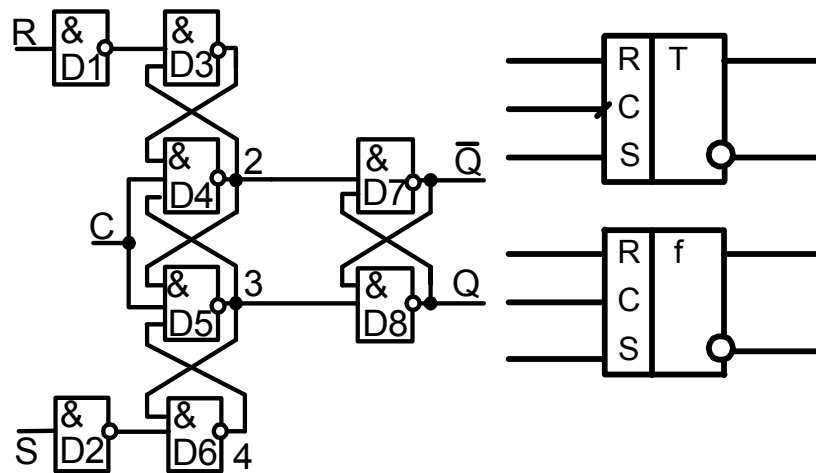


Рисунок 5.25 – Схема RS-тригера з ДЗ а), його УГП

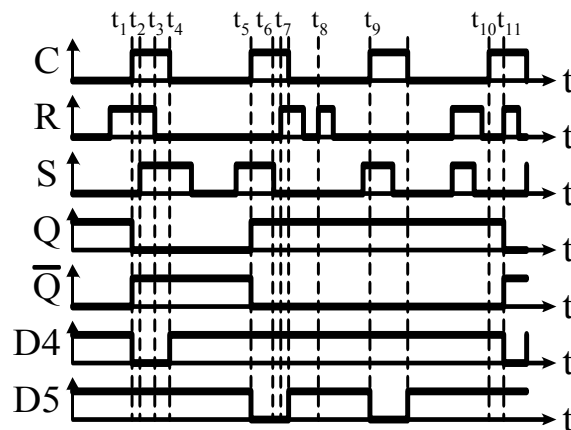
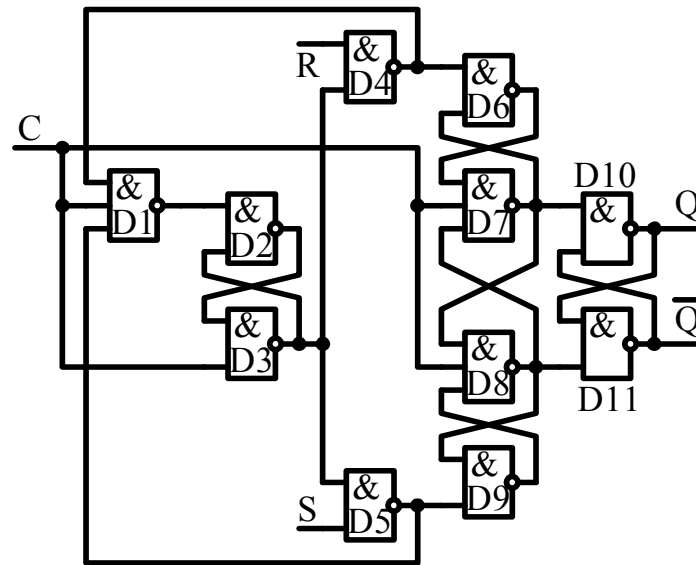


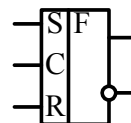
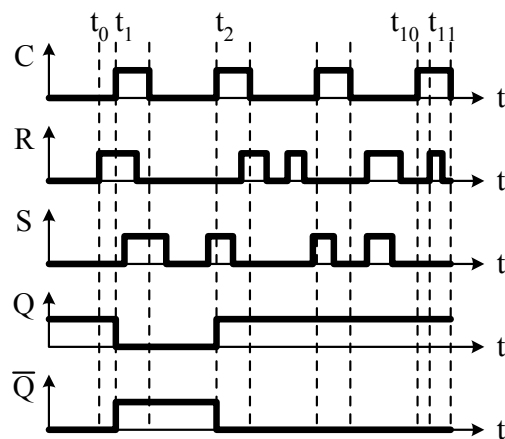
Рисунок 5.26 – Часові діаграми RS-тригера

В момент t_8 надходить сигнал завади на вхід R , але тригер не спрацьовує, бо відсутній CC на вході C . В момент t_9 знову формується сигнал на виході вентилля $D5$, а тригер підтвердить свій одиничний стан. До моменту t_{10} на входах R та S буде діяти рівень 0 і за логікою роботи тригера F він

повинен реагувати на зміну інформації на його входах після приходу фронту F . Як видно з діаграми, в момент t_{11} на вхід R надходить сигнал завади, проте тригер переходить у стан, при якому $Q = 0$, хоча вхідний сигнал надійшов після надходження фронту CC . Останнє означає, що цей тригер є тригером C_FRS -типу, бо не має повного блокування входів, оскільки є одна комбінація вхідних сигналів $S = R = 0$, за якої тригер реагує на зміну інформації на його входах після надходження фронту CC . Отже, цей тригер, як і тригер C_FD -типу, може працювати від CC з рівнем логічного 0, тобто в режимі з внутрішньою затримкою.



а)



в)

б)

Рисунок 5.27 – Схема RS -тригера а), його УГП в) і діаграма роботи б)

Варіант схеми тригера, яка має повне блокування входів, тобто тригера C_FRS -типу, зображений на рис. 5.27. Як видно з діаграми (див. рис. 5.27, б),

такий тригер після надходження на вхід C фронту CC не реагує на зміну інформації на його входах за будь-яких комбінацій вхідних сигналів, також і при $R = S = 0$ (момент t_{11}). Цього досягають за рахунок введення до складу тригера додаткового пристрою, виконаного на ЛЕ $D1-D3$, і зв'язків з виходів ЛЕ $D4, D5$ на вхід ЛЕ $D1$, і з виходу $D3$ на входи ЛЕ $D4, D5$. При всіх комбінаціях вхідних сигналів (крім комбінації $R = S = 0$) такий тригер працює аналогічно тригеру на рис. 5.25.

Нехай тригер знаходиться в стані $Q = 1$, на його вході R діє рівень «1», а на вході C – рівень «0» (момент t_0 на рис. 5.27, б). У цьому випадку має місце такий розподіл рівнів сигналів на виходах ЛЕ: $D1 = D3 = D5 = D6 = D7 = D8 = 1, D2 = D4 = D9 = 0$, тобто ЛЕ $D7$ підготовлений до перемикання. В результаті з надходженням CC (момент t_1) сформується блокувальний сигнал на виході ЛЕ $D7$, тригер переходить у стан $Q = 0$ і не буде реагувати на зміну інформації на його входах. Аналогічно при комбінації сигналів $S = 1, R = 0$ (момент t_2) тригер переходить в стан $Q = 1$. При комбінації сигналів $R = S = 0$ і $C = 1$ на виході ЛЕ $D3$ сформується сигнал з рівнем 0, який блокує прийом інформації через ЛЕ $D4$ і $D5$. В результаті зміна інформації на входах тригера не буде впливати на його роботу (момент t_{11}).

5.4.3 JK-тригер з ДЗ

В комп'ютерній схемотехніці крім розглянутих тригерів CRS- та CD-типів широко використовуються тригери CJK -типу. Варіант схеми такого тригера наведений на рис. 5.28.

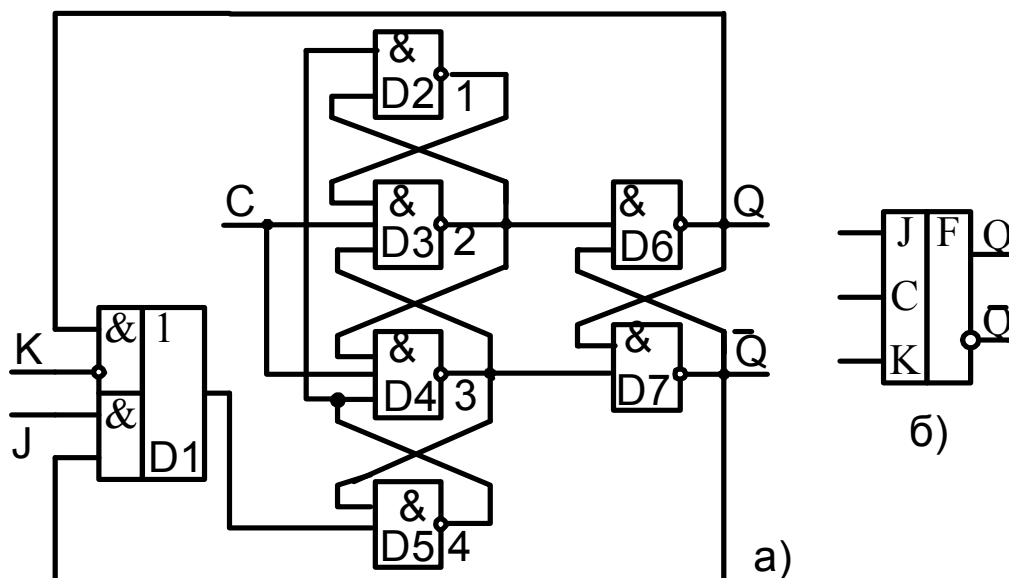


Рисунок 5.28 – Схема JK-тригера а) та його УГП б)

Він виконаний на основі CD-тригера і працює таким чином. Нехай тригер знаходиться у стані $Q = 0$, а на входах J та K діє комбінація сигнала-

лів $J = 1$, $K = 0$. В цьому випадку на виході ЛЕ D1 діє рівень 1; з появою фронту CC тригер переходить у стан $Q = 1$. За комбінації сигналів $J = K = 0$ на виході ЛЕ D1 діє рівень 0 і при надходженні фронту CC тригер перейде у стан $Q = 0$, якщо до того він знаходився у стані $Q = 1$. За комбінації сигналів $J = K = 1$ тригер переходить в інверсний стан, тобто функціонує згідно з таблицею істинності JK-тригера. При всіх комбінаціях вхідних сигналів зміна інформації на входах тригера після надходження фронту CC не буде впливати на його стан, тобто така схема забезпечує повне блокування входів.

Як вже вказувалося раніше, тригери, що працюють за фронтом, мають фіксувати на своїх входах тільки ту інформацію, яка діє на його входах в момент надходження фронту тактового імпульсу. Тоді тригери повинні запам'ятовувати інформацію, яка діє на його входах при відсутності ТІ. Тобто, при відсутності CC інформація на входах тригера може змінюватись за будь-яким законом, та якщо до моменту надходження фронту CC на його входах буде нейтральна комбінація сигналів, при якій тригер знаходиться в режимі зберігання (для JK-тригера це $J = K = 0$), то з надходженням фронту CC тригер повинен підтвердити свій стан, тобто залишитися в стані Q^t . Для тригера на рис. 5.28, а ці умови виконуються автоматично, оскільки ЛЕ D2 не є тригером і не запам'ятовує інформацію, що діє на його входах J і K . Тригер рис. 5.28, а задовольняє обидві умови виду для динамічних тригерів і є CJK-тригером.

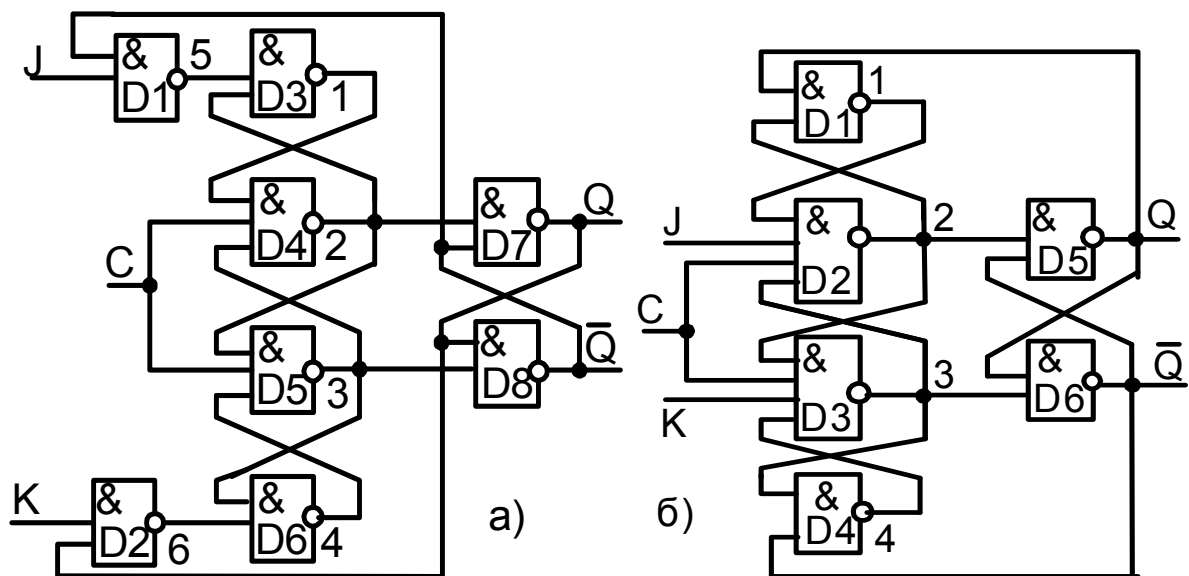


Рисунок 5.29 – Схеми CJK-тригерів

Тригери з внутрішньою затримкою на ЛЕ І-НІ, виконані за схемою «трьох тригерів», зображені на рис. 5.29, а, б. В цих тригерах сигнали, які відповідають новому стану, встановлюються при переході сигналу CC із 0 в 1. Розглянемо принцип роботи JK тригера, який наведений на рис. 5.29, а.

При $C = 0$ на виходах ЛЕ $D4$ і $D5$ присутній одиничний сигнал, тобто тригер на ЛЕ $D7$ і $D8$ не змінює свого стану, а ЛЕ $D3$ і $D4$ виконують функцію інверторів. Перемикання тригера здійснюється за таблицею переходів JK -тригера. Наприклад, якщо $J = 1$, $Q = 1$ і $K = 0$, $Q = 0$, то при переході сигналу CC із «0» в «1» на виході ЛЕ $D4$ встановлюється нульовий сигнал, який перемикає в одиничний стан тригер на ЛЕ $D7$, $D8$ та підтверджує одиничний сигнал на виході вентилі $D3$. Після цього сигнали J і K можуть змінюватися, що не впливає на стан основного тригера (вентилі $D7$ і $D8$), до того часу, поки не звідбудеться черговий перехід сигналів C із «0» в «1». За аналогічним принципом функціонує і JK -тригер, який зображений на рис. 5.29, б. На рис. 5.30 наведена часова діаграма роботи JK -тригера (рис. 5.29, а) з урахуванням затримок ЛЕ. Поряд з JK -тригерами відомі й інші типи тригерів з управлінням інформаційними входами, для яких відсутні комбінації вхідних сигналів, при яких тригер приймає невизначений стан. Це, наприклад, тригери CR -, CS -, $\overline{CR}$ -, $\overline{CS}$ – типів, тобто R - та S -тригери з прямим та інверсним входами.

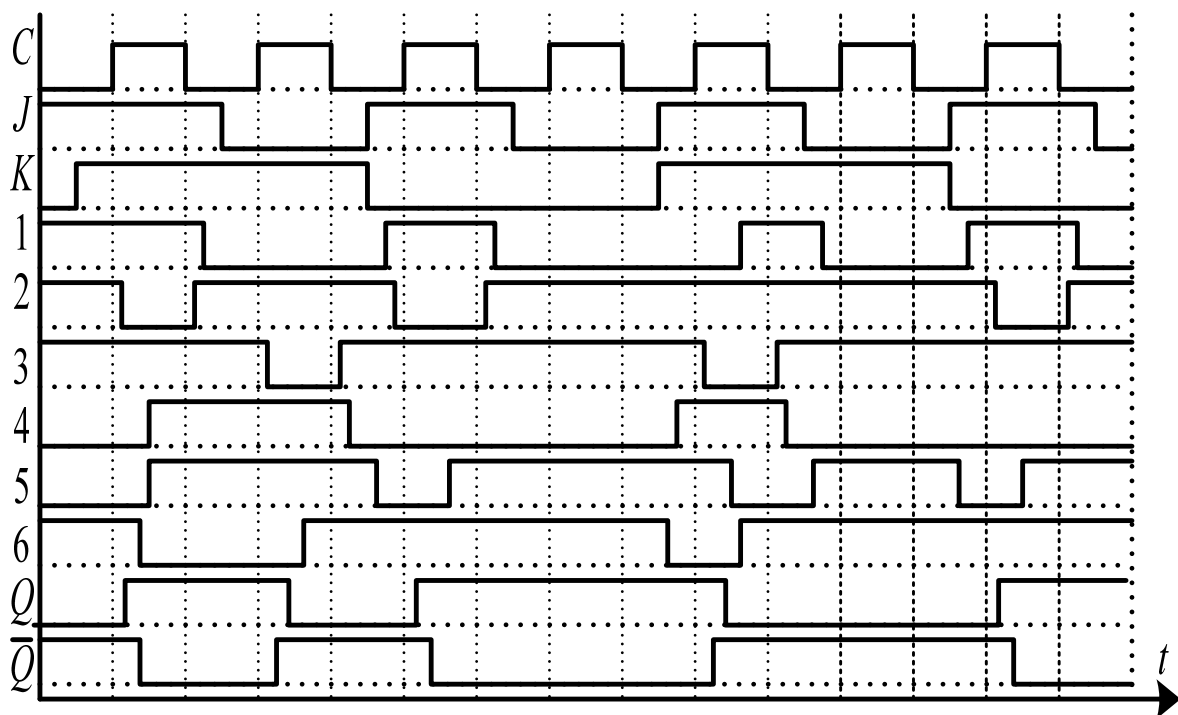


Рисунок 5.30 – Часова діаграма роботи JK -тригера з урахуванням затримок ЛЕ

5.5 Синхронні двоступеневі тригери за структурою MS

Для прийому та фіксації інформації в тригерах використовують, як мінімум, два фрагменти сигналу. Такими фрагментами сигналів можуть бути рівень і фронт зростання, рівень і зріз, фронт і зріз.

Синхронний двоступеневий тригер (*MS*-тригер) складається з двох частин – тригерів, одночасний прийом інформації в які заборонено. *Master*-частина – перший ступінь, *Slave*-частина – другий ступінь синхронного двоступеневого тригера. Для побудови першого та другого ступенів використовують синхронні тригери зі статичним управлінням записом. Інформація на другий ступінь може бути передана тільки після прийому її на перший ступінь і після закінчення синхронізувального сигналу, що дозволяє запис вхідної інформації на перший ступінь. Така послідовність у прийомі інформації звичайно досягається вмиканням інвертора в коло синхронізувальних сигналів для другого ступеня. Основна схема синхронного двоступеневого тригера наведена на рис. 5.31.

Якщо на синхронізувальний вхід *C* подати сигнал «1», то вхідна інформація приймається на перший ступінь (основний тригер). В цей час другий ступінь (допоміжний тригер) залишається у попередньому стані. Інформація на перший ступінь приймається під час дії синхронізувального сигналу. Якщо стан синхронізувального входу дорівнює «0» (синхронізувальний сигнал відсутній), то прийом інформації на перший ступінь заборонений (в цей час дозволяється зміна сигналів на логічних входах тригера), а другий ступінь приймає (копіює) інформацію, яка зберігається на першому ступені.

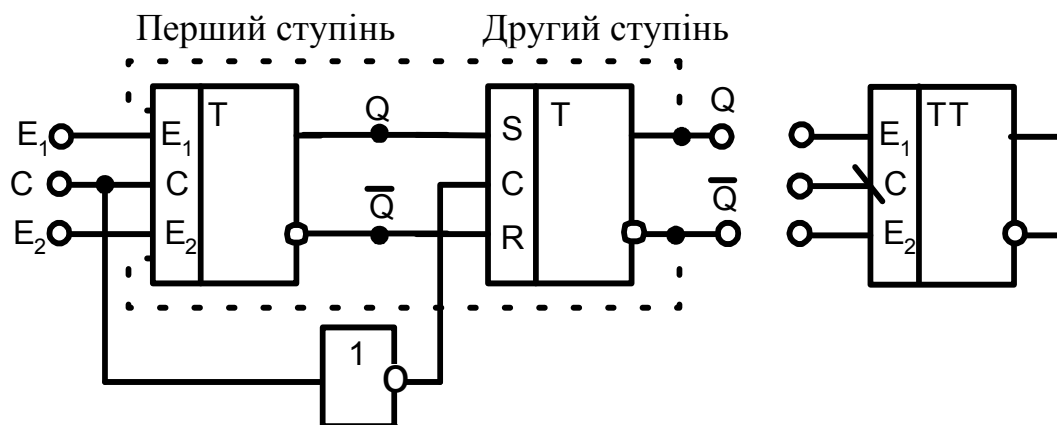


Рисунок 5.31 – Структура синхронного двоступеневого тригера за схемою *MS* з блокувальним інвертором та його УГП

З практики проектування тригерних пристроїв відомо більше схематичних варіантів щодо організації вузла блокування. Це, звичайно, потребує введення додаткової класифікації тригерів. В подальшому назви тригерів будемо уточнювати, відображаючи в назві схемні рішення вузла блокування. Наприклад, якщо вузол блокування виконується на одному інверторі, то тригер буде називатися *MS*-тригером з блокувальним інвертором.

Крім *MS*-тригерів з блокувальним інвертором застосовуються такі різновиди тригерів:

MS-тригери з двома блокувальними інверторами;

MS-тригери з внутрішніми заборонними зв'язками;
MS-тригери з різнополярним управлінням;
MS-тригери з комутаційними транзисторами.

Можливі й інші схемотехнічні рішення вузла блокування. Але всі вони мають одну і ту ж саму мету – дозволити передачу інформації з тригера *M* в тригер *S* під час відсутності *CC* та блокувати передачу стану тригера *M* в тригер *S* при дії *TI*. При цьому для правильної роботи пристрою необхідно, щоб процес запису та блокування передачі інформації між тригерами проходив в такій послідовності (рис. 5.32):

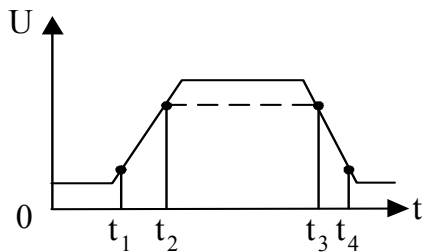


Рисунок 5.32 – Діаграма роботи *MS*-тригера

а) в момент t_1 заборонено передачу інформації з тригера *M* на тригер *S*;

б) в момент t_2 дозволено запис інформації на тригер *M*;

в) в момент t_3 заборонено прийом інформації на тригер *M*;

г) в момент t_4 дозволено передачу стану тригера *M* на тригер *S*.

Іншими словами, для правильної роботи тригера потрібно забезпечити випередження блокування порівняно з процесом запису інформації. Кожен з наведених *MS*-тригерів має певні переваги перед іншими, і тому далі вони будуть розглянуті більш детально.

5.5.1 Тригер типу *M-S* з інвертором

Цей тригер складається з двох тригерів (рис. 5.33), де між основним та допоміжним тригерами вмикають інвертор *D1*, який реалізує блокування перезапису інформації до допоміжного тригера під час запису інформації до основного тригера (при $C = 1$ на виході ЛЕ *D1* присутній «0», який закриває елементи *D6* та *D7*). Як видно, в схемі тригера (рис. 5.33) можливі змагання (гонки), що є одним з недоліків такої організації блокування. Наявність змагань є наслідком невиконання умов надійної роботи *MS*-тригера, сформульованих вище.

Щоб впевнитися у функціональній надійності тригера, потрібно виявити, за яких умов змагання стають критичними (небезпечними для правильного функціонування). Оскільки перемикання тригера *M* зі стану 0 ($Q = 0$) в стан 1 ($Q = 1$) та навпаки завжди здійснюється через неперехідний стан $Q = \bar{Q} = 0$, то тригер, схема якого наведена на рисунку 5.34, буде функціонувати надійно, якщо виконується умова $\tau_{01\max} \leq 2\tau_{cp.\min}$, де $\tau_{cp.\min} = (\tau_{01\min} + \tau_{10\min})/2$ – середня мінімальна затримка перемикання елементів І-НІ тригера *M*. Інакше кажучи, рівень «0» на виході блокувального інвертора повинен сформуватися раніше, ніж тригер *M* перемкнеться в інверсний стан.

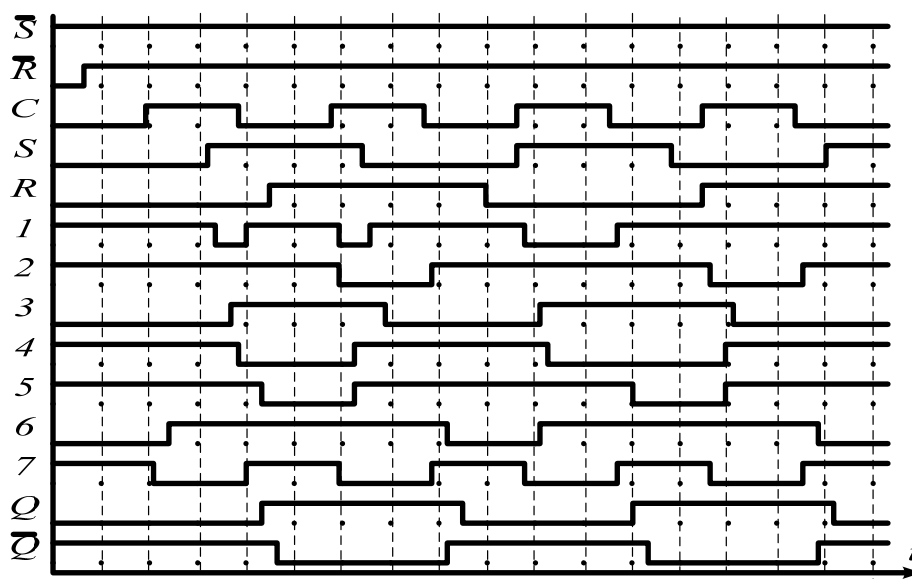
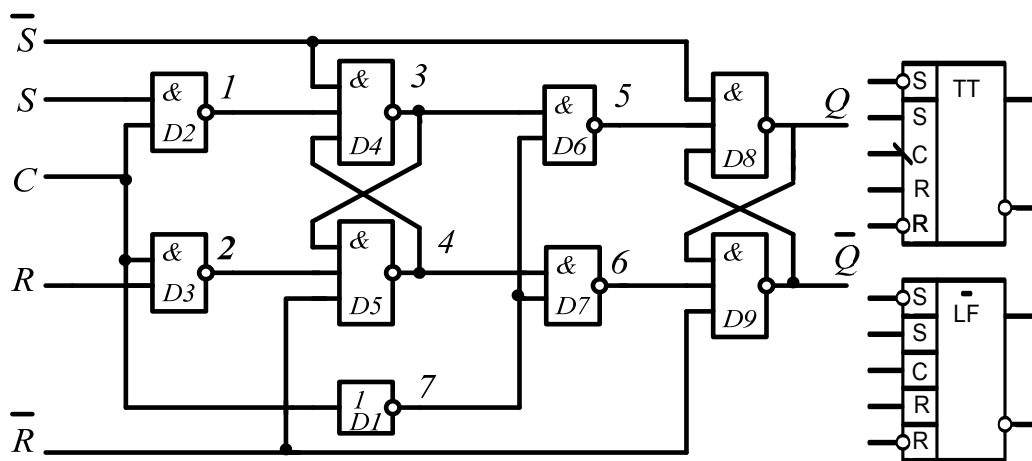


Рисунок 5.33 – Синхронний двоступеневий RS -тригер з асинхронними входами $\bar{S}$ і $\bar{R}$, його УГП та часові діаграми

В іншому випадку під час дії CC може виникнути потреба змінити стан допоміжного тригера S . При наявності у складі ЦІМС групи елементів, які мають підвищену швидкодію і для яких відомі значення $\tau_{\min}$ та $\tau_{\max}$, виконання даної умови не буде проблематичним. Якщо ж значення затримок елементів невідомі, схему тригера на рис. 5.33 застосовувати не слід. Важливою особливістю CRS -тригерів, які виконані за способом MS , є те, що вони можуть бути легко перетворені в інші типи тригерів (T -, CJK -типів). На рис. 5.33 наведені часові діаграми RS -тригера.

5.5.2 Тригер з двома інверторами. У таких тригерах вузол блокування виконується на двох інверторах, ввімкнених послідовно у тактове коло, як показано на рис. 5.34 на прикладі CRS -тригера.

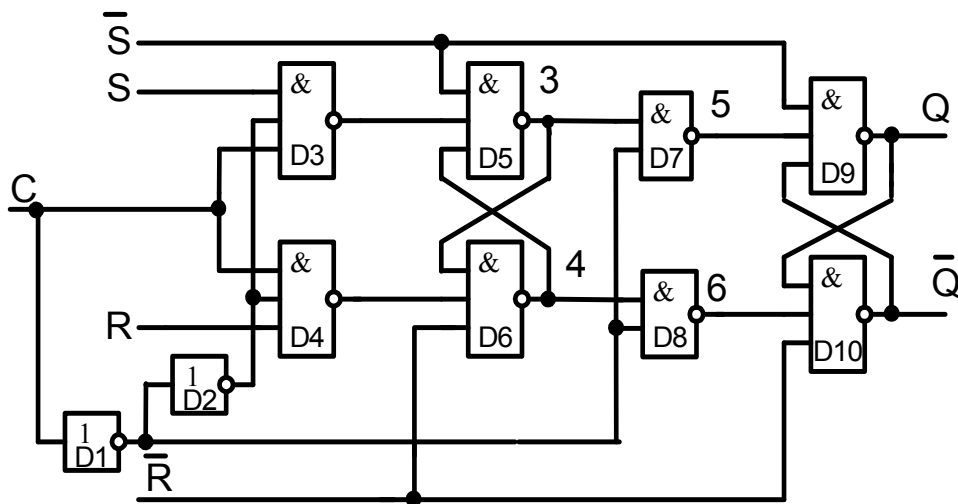


Рисунок 5.34 – Схема RS -тригера типу MS з двома інверторами

Завдяки такій побудові ідеально виконуються перераховані вище умови надійної роботи. При надходженні CC ($C = 1$) спочатку формується рівень 0 на виході ЛЕ $D1$ і забезпечується випередження блокування перед процесом запису інформації в тригер M . Потім формується рівень 1 на виході елемента $D2$ і тільки після цього дозволяється запис інформації на тригер M . Після закінчення CC ($C = 0$), навпаки, спочатку буде заборонений прийом інформації до тригера M , а після появи рівня 1 на виході ЛЕ $D1$ блокування знімається і тим самим дозволяється передача стану з тригера M на тригер S .

5.5.3 Тригери з заборонними зв'язками

Блокування передачі стану з тригера M на S під час дії CC в таких тригерах здійснюється за рахунок допоміжних блокувальних зв'язків з виходів ЛЕ тригера M . Більш детально принцип роботи таких тригерів розглянемо на прикладі рис. 5.35. За відсутності CC ($C = 0$) виходи ЛЕ $D1$ та $D2$ дорівнюють 1 і дозволена передача стану тригера M на S . При надходженні CC ($C = 1$) на виходах ЛЕ $D1$ (при $S = 1, R = 0$) або $D2$ (при $S = 0, R = 1$) формується рівень 0, який забороняє передачу стану M на S та встановлює тригер M в стан 1 або 0. Після закінчення CC ($C = 0$), навпаки, спочатку забороняється прийом інформації на тригер M (бо $C = 0$) і тільки потім, після формування рівня 1 на виходах ЛЕ $D1$ та $D2$, дозволяється передача стану тригера M на S .

Таким чином, виконуються умови надійної роботи і гарантується функціональна надійність пристрою. Цей тригер має важливу особливість, яку необхідно враховувати при його застосуванні: якщо під час дії CC інформація на одному із входів з одиничного значення змінюється на нульове, то на виходах ЛЕ $D1$ та $D2$ сформується дозвільний сигнал, і інформація з тригера M перепишеться на тригер S . Зміна інформації на входах тригера приводить

до порушення алгоритму роботи, бо інформація на входах тригера з'являється не після закінчення CC , а під час його дії, тобто не блокуються виходи під час дії CC . Розглянута схема є схемою тригера з неповним блокуванням виходів, тобто є тригером CRS -типу. На базі цієї схеми можлива організація тригерів T -, CJK -типів, асинхронних тригерів RS - і JK -типів, що здійснюється подачею на тактовий вхід C рівня 1. Потрібно сказати, що таких якостей не має жоден з відомих тригерів, розроблених за способом MS .

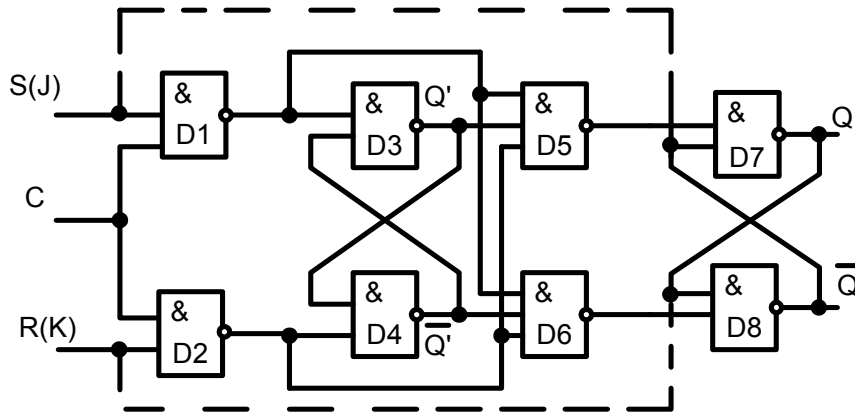


Рисунок 5.35 – Схема RS -тригерів із заборонними зв'язками

Особливість цієї схеми в тому, що блокування перезапису досягається шляхом тактування основного та допоміжного тригерів сигналами різної полярності. Цим забезпечується надійне блокування передачі інформації в допоміжний тригер у момент її запису в основний, чим і гарантується висока функціональна надійність.

Наприклад, якщо основний тригер побудувати на ЛЕ типу І-НІ, а допоміжний – на ЛЕ типу АБО-НІ, то, відповідно, перший буде керуватися сигналом «1», а другий – «0». Цим забезпечується надійне блокування передачі інформації на допоміжний тригер в момент її запису в основний, чим і забезпечується висока функціональна надійність.

5.5.4 Тригери з різнополярним управлінням

На рис. 5.36 зображена схема RS -тригера з різнополярним управлінням, в якій основний і допоміжний тригери реалізовані на елементах І-АБО-НІ. Порядок роботи тригерів, виконаних за даною схемою такий. При відсутності CC ($C = 0$) закриті ЛЕ I_1 , I_2 елемента $D3$ або $D4$ і стан тригера M постійно переписується в тригер S . Так, якщо тригер M знаходиться в стані $Q' = 0, \bar{Q}' = 1$, то будуть закриті ЛЕ I_1 і I_2 елемента $D4$, тобто на його виході рівень 1 і, як наслідок, тригер S у цьому випадку знаходиться у стані $Q = 0, \bar{Q} = 1$.

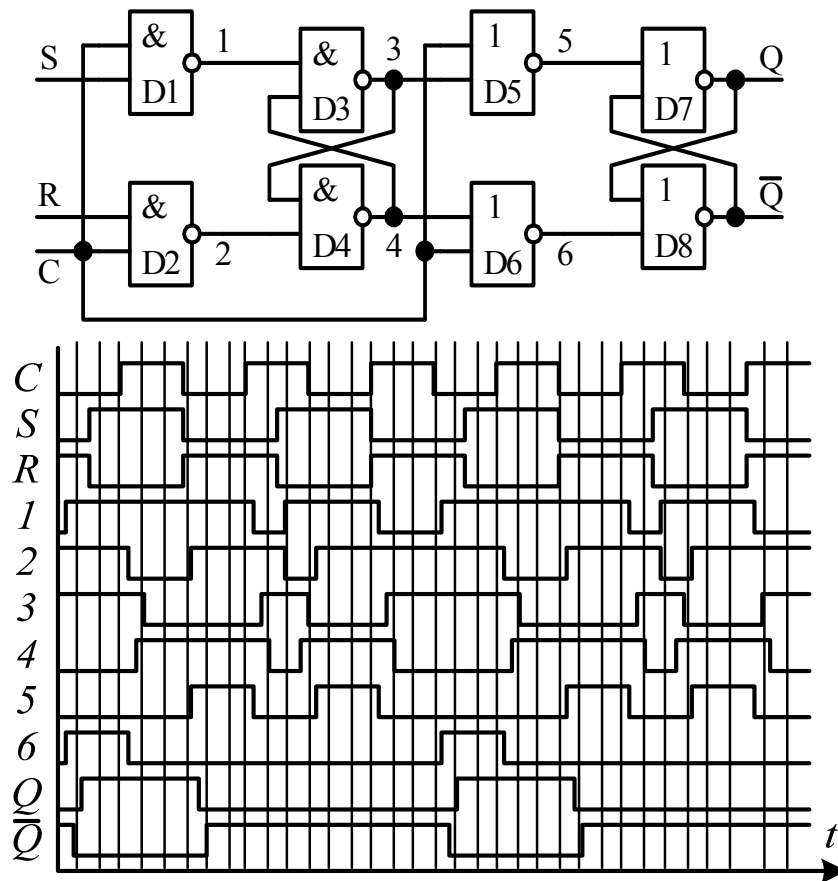


Рисунок 5.36 – Схема RS -тригера з різнополярним управлінням і часова діаграма його роботи

При надходженні CC ($C = 1$) відкривається ЛЕ I_1 елемента $D3$, тобто тригер S запам'ятає стан тригера M . Одночасно з процесом зберігання інформації допоміжним тригером відбувається запис інформації в тригер M . Проте інформація, яка записується в тригер M під час дії CC , не сприймається тригером S , тому що рівень 0 на виході Q утримує в закритому стані виходи I_1 і I_2 елемента $D4$, тобто діє заборона прийому інформації тригером S . Після закінчення CC заборона знімається ($C = 0$) і інформація буде прийнята тригером S , який під час переходу проходить через стан $Q = \bar{Q} = 1$. Цей тригер, як і всі розглянуті вище, може бути легко перетворений у тригер T - і JK -типів.

Порівнюючи цей тригер з іншими, розглянутими вище, важливо відзначити, що він має мінімальне число елементів і максимальну швидкодію. Крім того, має місце мінімальне значення потрібної навантажувальної здатності тригера і максимальне значення еквівалента навантаження за тактовим входом, що до певної міри обмежує можливості тригера при проектуванні більш складних цифрових вузлів.

Розглянутий тригер також не може бути перетворений в асинхронний, що звужує його функціональні можливості. Проте, завдяки високій швидкодії і малій кількості елементів, він знаходить досить широке застосування.

5.5.5 Схема тригера типу *MS* з комутувальними транзисторами

В такому тригері перемикання допоміжного тригера після закінчення дії *CC* забезпечується за допомогою ключових транзисторів, які вмикаються між основним і допоміжним тригерами. Схемна реалізація одного із варіантів такого тригера показана рис. 5.37. Слід зауважити, що, крім розглянутих, можливі й інші схемотехнічні прийоми організації блокування в *MS*-тригерах.

Проте наведені тут схемні варіанти одержали найбільше поширення завдяки їх високим схемотехнічним і функціональним якостям. Порівнюючи розглянуті схеми тригерів, виконаних за принципом *MS*, неважко бачити, що кожний з них має локальні переваги однак саме вони можуть виявитися вирішальними при виборі необхідного варіанта. Щодо розробника, то він повинен знати переваги та недоліки кожного тригера і вміло їх використовувати в процесі проектування.

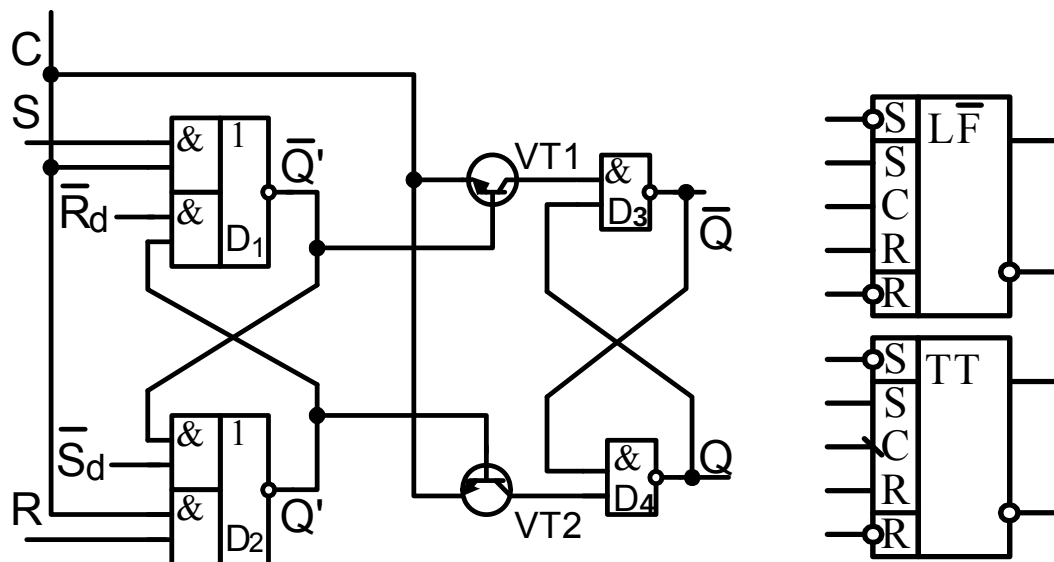


Рисунок 5.37 – Схема *RS*-тригера з комутувальними транзисторами і його УГП

Нижче перелічені основні переваги деяких із розглянутих тригерів, виконаних на типових ЦІМС.

MS-тригери з двома блокувальними інверторами – мають найбільшу функціональну надійність;

MS-тригери із заборонними зв'язками – мають найбільшу функціональну гнучкість.

MS-тригери з різнополярним тактуванням – мають максимальну швидкодію та мінімальне число елементів.

5.5.6 D-тригери типу MS

В цифрових пристроях *CD*-тригери мають таке ж функціональне призначення, як і тригери *CJK*-, *CRS*-типів, та застосовуються, в основному, при проектуванні регістрів зсуву та лічильників. Принципова можливість приймати інформацію за одним входом вигідно відрізняє їх від тригерів *CJK*-, *CRS*-типів за кількістю міжкаскадних зв'язків.

Проектування тригерів *CD*-типів здійснюється за принципом *MS*-тригера, де як *M*-тригер необхідно використовувати тригер *CD*-типу. В даному випадку можуть використовуватися ті ж схемотехнічні прийоми організації блокування, що і для *RS*-тригерів, і тому тут немає особливої необхідності розглядати їх знову. Обмежимося тільки схемами та часовими діаграмами їх роботи на рис. 5.38 та 5.39.

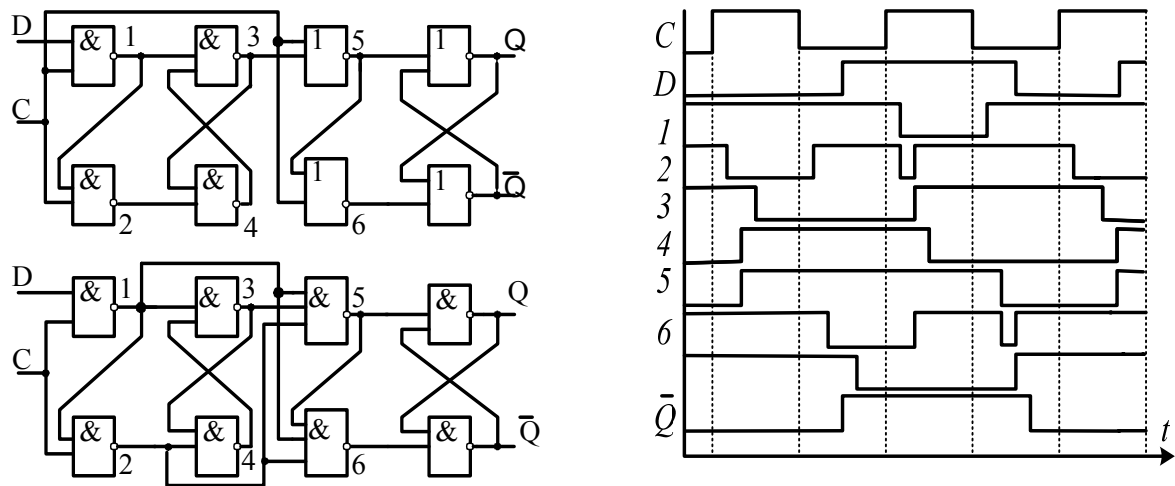


Рисунок 5.38 – Схеми *D*-тригерів та часова діаграма їх роботи

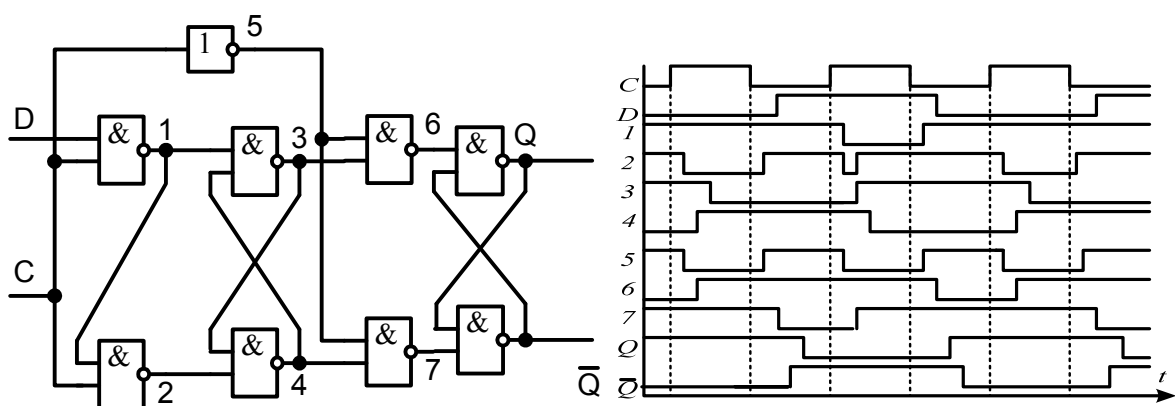


Рисунок 5.39 – Схема *D*-тригера з інвертором та часова діаграма його роботи

5.5.7 JK -тригер типу MS із заборонними зв'язками

Тригером JK -типу називається пристрій з двома стійкими станами та двома входами J та K , який за умови $J = K = 1$ здійснює інверсію попереднього стану тригера, але в інших випадках функціонує відповідно до таблиці істинності RS -тригера, при цьому вхід J еквівалентний входу S , а вхід K – входу R . Схема його наведена на рис. 5.40.

На підставі переходів JK -тригера запишемо СДНФ для функції Q_{t+1}

$$Q_{t+1} = J_t \bar{K}_t \bar{Q}_t + J_t K_t \bar{Q}_t + \bar{J}_t \bar{K}_t Q_t + J_t \bar{K}_t Q_t.$$

Характеристичне рівняння для цього JK -тригера

$$Q_{t+1} = J_t \bar{Q}_t + \bar{K}_t Q_t.$$

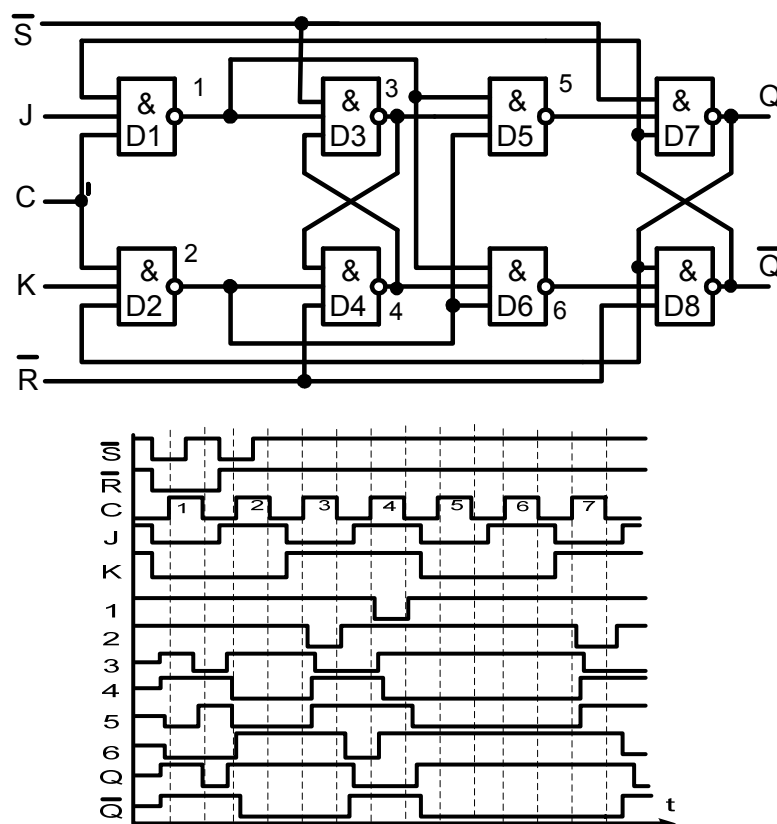


Рисунок 5.40 – Схема JK -тригера і його часова діаграма

Отже, з опису роботи JK -тригера видно, що інформація на виході з'являється в момент переходу CC зі стану «1» у стан «0», тобто маємо синхронний JK -тригер з динамічним записом, в якого є інверсний динамічний вхід.

Швидкодія JK -тригера ($t_{3,cr}$ середня затримка перемикання одного ЛЕ) визначається через:

а) тривалість τ_i вхідного синхронізувального імпульсу для стійкого запису інформації в основний тригер: $\tau_i \geq 3t_{3,cr}$;

б) затримка перемикання τ_t допоміжного тригера: $\tau_t \geq 4t_{3,sp}$ (час спрацювання RS -тригера – $3t_{3,sp}$ та час перемикання одного з ЛЕ $D1$ та $D2$ – $1t_{3,sp}$);

в) максимальна частота $f_{\max}$ перемикання JK -тригера

$$f_{\max} \leq \frac{1}{\tau_i + \tau_t} = \frac{1}{7t_{3,sp}}.$$

Асинхронні входи JK -тригера. Для початкового встановлення тригера існують асинхронні входи S (встановлення $Q = 1$) і $\bar{R}$ (встановлення тригера в $Q = 0$). Для прискорення початкового встановлення сигнали подаються на вихідні каскади допоміжного і головного тригерів. Швидкодія за асинхронними входами JK -тригера дорівнює $2t_{3,sp}$.

Універсальність JK -тригера

Із JK -тригера за допомогою незначних змін можна отримати інші тригери: RS -, D -, T -.

T -тригер на основі JK -тригера

Характеристичне рівняння T -тригера

$$Q_{t+1} = T_t \bar{Q}_t + \bar{T}_t Q_t.$$

Порівнюючи характеристичне рівняння T -тригера з рівнянням JK -тригера, помічаємо, що при виконанні умови $J = K = T$, рівняння JK -тригера та T -тригера тотожні. Функціональна схема D -тригера наведена на рисунку 5.41, а).

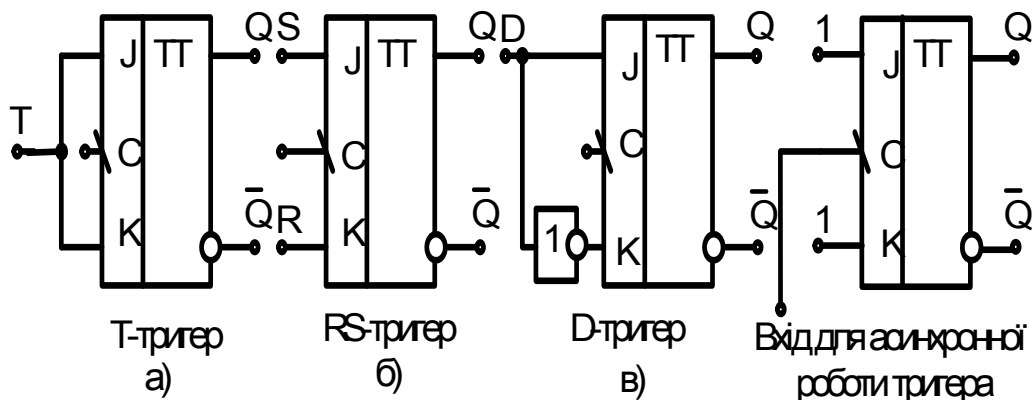


Рисунок 5.41 – Функції, виконувані JK -тригером

RS -тригер на базі JK -тригера

Порівнюючи характеристичне рівняння RS -тригера

$$Q_{t+1} = S_t + \bar{R}_t Q_t$$

з рівнянням JK -тригера, отримуємо їх тотожність за умови $J = S$, $K = R$ за

винятком комбінації $JK = 1$. Функціональна схема RS -тригера наведена на рисунку 5.41, б).

D-тригер на основі JK-тригера

При виконанні умови $J = \bar{K} = D$ в характеристичному рівнянні JK -тригера маємо

$$Q_{t+1} = D_t \bar{Q}_t + D_t Q_t = D_t,$$

що відповідає характеристичному рівнянню D -тригера. Функціональна схема D -тригера наведена на рисунку 5.41, в).

5.6 Асинхронні входи синхронних тригерів

Як правило, тригер, яким управляють за синхронізувальним входом, крім логічних елементів І-НІ, має асинхронні входи попереднього встановлення тригера у стани «0» та «1». Сигнали, які надходять на ці входи, мають пріоритет, тобто, незалежно від стану інших тригерів, ці сигнали відразу (за переднім фронтом) встановлюють тригер в певний стан. Літерою S позначають асинхронний вхід для встановлення тригера у стан «1», а літерою R – асинхронний вхід для встановлення тригера у стан «0». Синхронний тригер за встановлюваними входами реалізує таблицю переходів асинхронного RS -тригера. Наявність у синхронного тригера асинхронних входів вказують в його позначенні таким чином: після скороченого позначення логічних входів пишуть позначення асинхронних входів, наприклад: DRS -, $RSRS$ -тригер.

Асинхронні входи синхронних тригерів зі статичним записом отримують збільшенням числа входів у КП. Розглянемо організацію входів попереднього встановлення синхронного двоступеневого тригера. З аналізу роботи будь-якого двоступеневого тригера відомо, що при $C = 0$ сигнали на логічних входах не впливають на стан тригера, а КП другого рівня копіює стан тригера першого рівня, оскільки відкриті елементи зв'язку, які розділяють обидва тригера. Якщо тепер спеціальним сигналом змінити стан КП першого рівня, то, очевидно, змінить свій стан і тригер другого рівня (скопіює новий стан тригера першого рівня). Час, через який на виходах двоступеневого тригера встановиться новий стан, буде визначатися затримкою, яку вносять послідовно ввімкнені чотири логічних елементи. Для збільшення швидкодії сигнал попереднього встановлення тригера подається одночасно на КП першого та другого рівнів. При цьому новий стан тригера встановиться через час, який дорівнює затримці двох елементів.

5.7 Контрольні запитання і завдання

1. Що таке тригери? Комп'ютерні приклади їх використання.
2. Узагальнена структура тригера?
3. Класифікація тригерів за визначальними ознаками.

4. У чому полягає різниця між асинхронним і синхронним тригерами?
5. Властивості синхронних тригерів.
6. Різниця між статичним і динамічним записами інформації на тригер.
7. Чому не можна змінювати інформацію на D -вході в тригері зі статичним управлінням записом під час дії тактувального сигналу?
8. Охарактеризуйте динамічні входи D -тригера за схемою «трьох тригерів».
9. Яку роль виконують допоміжні тригери в схемі «трьох тригерів»?
10. Призначення асинхронних входів тригерів?
11. Яка інформація записується в D -тригер за схемою «трьох тригерів» у момент приходу переднього фронту тактувального сигналу?
12. Чи змінюється вихідний стан D -тригера за схемою «трьох тригерів» при зміні інформації тільки на D -вході; на D і на C з «1» у «0» і чому?
13. Сформулюйте означення RS -тригера.
14. Способи опису функціонування тригера.
15. Як проводиться запис інформації в тригері зі статичним записом?
16. Чи можна подавати послідовності сигналів з різною частотою на входи R і S (синхронізація входу C існує)?
17. Чи можна унеможливити заборонену комбінацію (для входів R і S) для RS -тригера?
18. Які функціональні типи тригерів існують на основі MS -принципу? Їх особливості.
19. Означення JK -тригера.
20. Характеристичне рівняння JK -тригера.
21. Який фронт синхросигналу впливає на JK -тригер?
22. Яку роль виконує допоміжний тригер і яку – основний?
23. Чи зміниться і чому вихідний стан JK -тригера при одночасній зміні інформації тільки на J , K і C входах; на входах з «0» в «1»; на входах J , K і C з «1» у «0»?
24. Як одержати інші типи тригерів на JK -тригерах?
25. Як впливають асинхронні входи S і R на роботу тригера?
26. У чому відмінність динамічного управління записом від статичного?
27. Якими параметрами та характеристиками визначається робота тригера у статичному та динамічному режимах?
28. Що таке елементарна комірка пам'яті?
29. Чим визначаються активні сигнали на входах елементарних комірок пам'яті та тригера?
30. Як можна описати закон функціонування тригера?
31. Що таке характеристична таблиця і для чого вона використовується?
32. Що таке лічильний тригер? Що таке лічильний режим роботи універсального тригера?
33. Сформулюйте означення універсального тригера.
34. Опишіть роботу універсального JK -тригера в асинхронному та синхронному режимах.

6 РЕГІСТРИ

Перелік скорочень і умовних позначень до розділу 6

СС – синхронізувальні сигнали; ОЗП – оперативний запам'ятовувальний пристрій.

Регістри – це пристрої, призначені для прийому, зберігання та передачі операндів протягом часу їх обробки. Крім перелічених мікрооперацій ре-гістри дозволяють виконувати зсуви операндів на певну кількість розрядів, а також перетворювати паралельне їх подання на послідовне і навпаки, а також виконувати деякі інші мікрооперації над словами. До складу регістрів входять тригери і ряд ЛЕ, призначених для управління прийомом і ви-дачею слів. Залежно від способу введення і виведення інформації розріз-няють паралельні, послідовні і паралельно-послідовні реєстри.

6.1 Паралельні реєстри

У реєстрах паралельної дії (реєстри пам'яті) всі розряди слова вводять і виводять одночасно (рис. 6.1). Для побудови n -розрядного паралельного реєстра необхідно використати n тригерів. На базі паралельних тригерних реєстрів організовують структури пристроїв оперативної пам'яті.

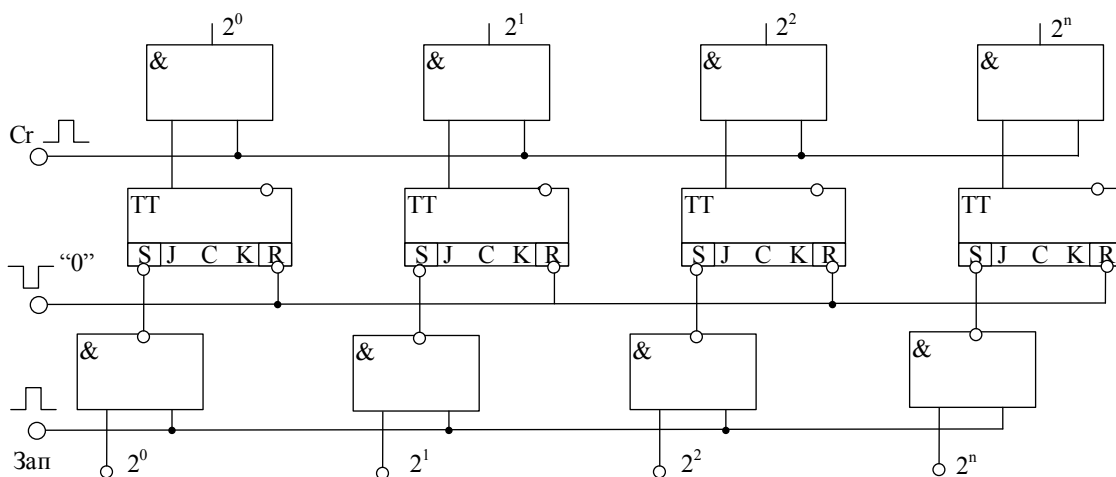


Рисунок 6.1 – Паралельний реєстр

6.2 Послідовні реєстри (реєстри зсуву)

Послідовний реєстр (зсувний реєстр, реєстр зсуву) забезпечує послі-довний запис і послідовне читання операндів. Для запису на інформацій-ний вхід реєстра послідовно подають значення двійкових розрядів опера-нда. Сигнали зсуву, що надходять на тактові входи, передають записану

інформацію від розряду до розряду вправо або вліво. У однотоктних послідовних регістрах надходження одного тактового сигналу забезпечує зсув одночасно всього числа на один розряд вправо або вліво. Залежно від кількості каналів, по яких надходить інформація на входи розрядів регістра, розрізняють регістри парафазні (інформація на кожен розряд надходить по двох каналах Q і $\bar{Q}$) і однофазні (інформація, надходить по одному каналу Q або $\bar{Q}$). При побудові регістрів застосовуються тригери RS -, D - і JK -типів? Найпростіше регістр зсуву реалізується на D -тригерах (рис. 6.2, а). Вихід Q попереднього розряду з'єднується з входом D наступного. Завдяки цьому кожен тактовий сигнал встановлює наступний тригер в стан, в якому до цього знаходився попередній, здійснюючи тим самим зсув інформації на розряд вправо, або вліво. (Як наслідок загальноприйнятого в науці і техніці позиційного способу запису цифрової інформації тут поняття «вправо» та «вліво» мають умовний відносний характер: «вправо» звичайно означає «в бік молодших розрядів», а «вліво» – «в бік старших розрядів»).

Вхід a_i першого розряду служить для прийому в регістр інформації у вигляді слова, яке передають послідовно (розряд за розрядом). З кожним синхронізуючим сигналом на цей вхід повинен подаватися новий розряд вхідної інформації. З виходу Q_3 останнього тригера знімають послідовний код вихідного слова з затримкою в часі щодо вхідного послідовного коду на число періодів CC , яке дорівнює кількості розрядів регістра. На рис. 6.2, б показана схема зсувного регістра на JK -тригерах. У цій схемі входи J і K наступних тригерів з'єднані з виходами відповідно Q і $\bar{Q}$ попередніх.

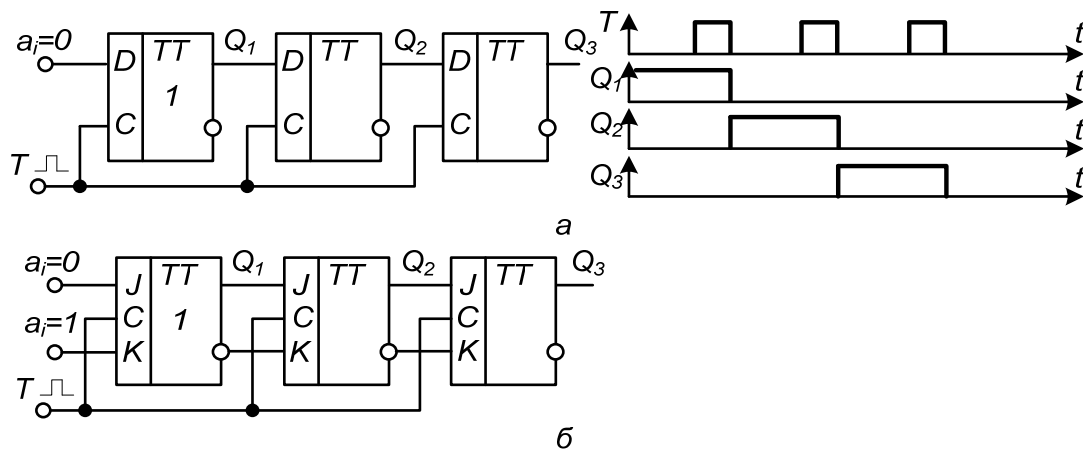


Рисунок 6.2 – Регістри зсуву

При проектуванні регістрів зсуву необхідно обов'язково застосовувати тригери, що синхронізуються фронтом, оскільки під час дії CC змінюються стани виходів тригерів, які під'єднані до входів наступних тригерів. Таким чином змінюється стан входів наступних тригерів, і якщо синхросигнал ще не скінчився, то тригери, що синхронізуються рівнем, перемикаються в но-

вий стан, тобто за час дії одного синхросигналу (один такт) інформація в регістрі просувається більше ніж на один розряд, отже, нормальне виконання зсуву на 1 розряд за 1 такт порушується.

Регістри зсуву можуть бути реверсивними, тобто можуть виконувати зсув і вправо, і вліво. Реверсивний регістр можна отримати, якщо в схемі паралельно-послідовного регістра (рис. 6.3, а) замість зовнішнього сигналу $\bar{A}_x$ під'єднати вихід подальшого розряду $\bar{Q}_{x+1}$ (показано пунктиром). Зсув вправо виконується при значенні сигналу $M = 1$, зсув вліво – при $M = 0$. Зсув числа вліво або вправо на один розряд відповідає його множенню або діленню на два.

Регістри зсуву застосовують також для перетворення паралельного подання операнда в послідовне і навпаки. Для цього кожен розряд регістра повинен мати додатковий вхід для паралельного прийому слова. Перетворення послідовного подання слова в паралельне здійснюється одночасним опитуванням станів всіх тригерів регістра (рис. 6.3, б).

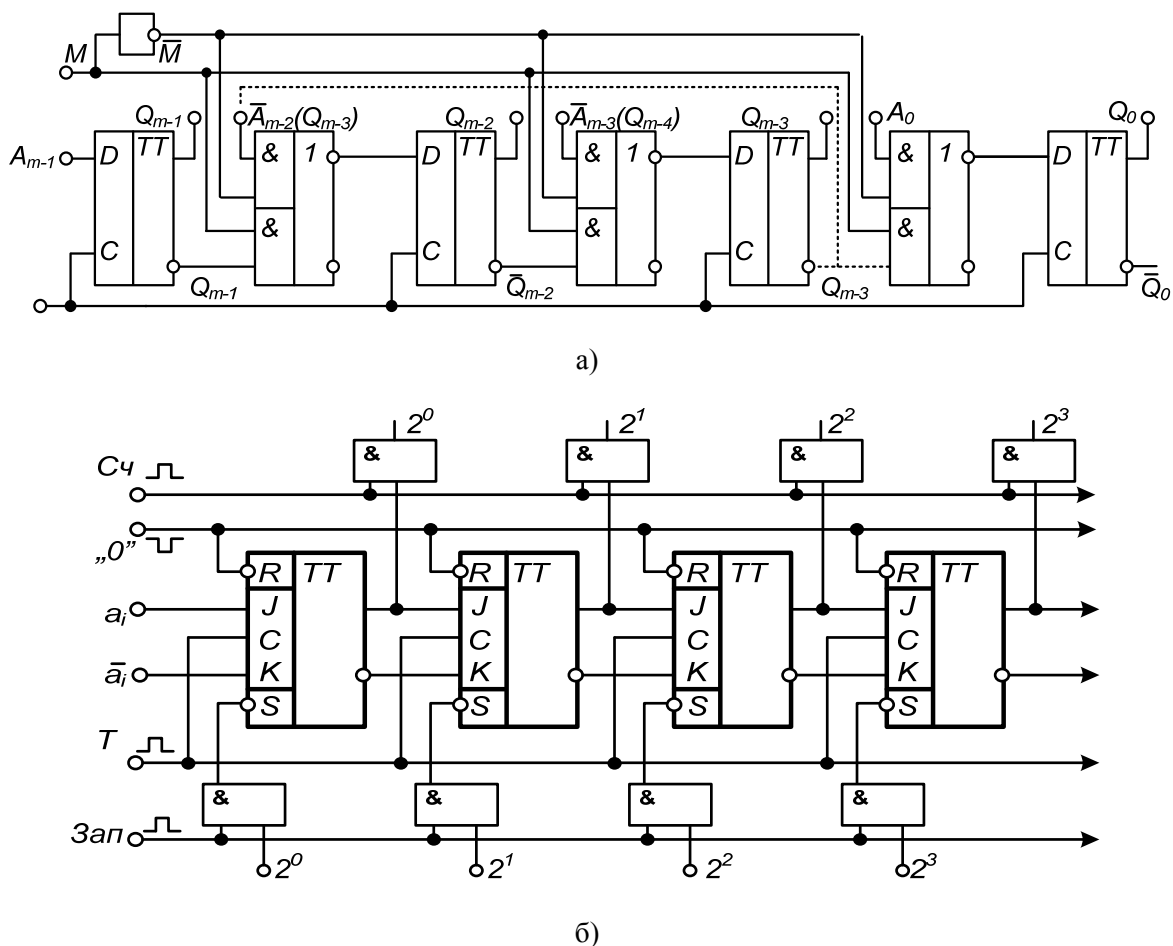


Рисунок 6.3 – Регістри зсуву:

- а – паралельно-послідовний регістр на D -тригерах;
- б – перетворювач паралельного подання слова в послідовний і навпаки на базі регістра зсуву

6.3 Послідовні регістри зі зворотними зв'язками (кільцеві лічильники)

Особливістю таких регістрів є те, що при введенні зворотних зв'язків в послідовний регістр останній перетворюється в замкнуте кільце, в якому під впливом СС циркулює введена в регістр інформація. Такі регістри називають кільцевими лічильниками. Схеми кільцевого лічильника, побудованого на D - і JK -тригерах зображені на рис. 6.4. У цих лічильниках спеціальна кодова комбінація (так званий «маркер»), введена в перший тригер, циркулює протягом усього часу існування послідовності СС, що подаються на входи C всіх тригерів лічильника (на рис. 6.4 маркером служить одиниця). У процесі «просування» маркерної одиниці вздовж регістра кожен тригер переходить з нульового стану в одиничний і знаходиться в ньому протягом періоду проходження СС. Ця властивість кільцевого лічильника використовується при побудові розподільників сигналів. При цьому маркерна одиниця з'являється на виході з частотою, що дорівнює відношенню частоти СС до кількості n тригерів в регістрі. Це відношення називають також коефіцієнтом перерахунку лічильника $K_{лч}$. Таким чином, для розглянутого лічильника $K_{лч} = n$.

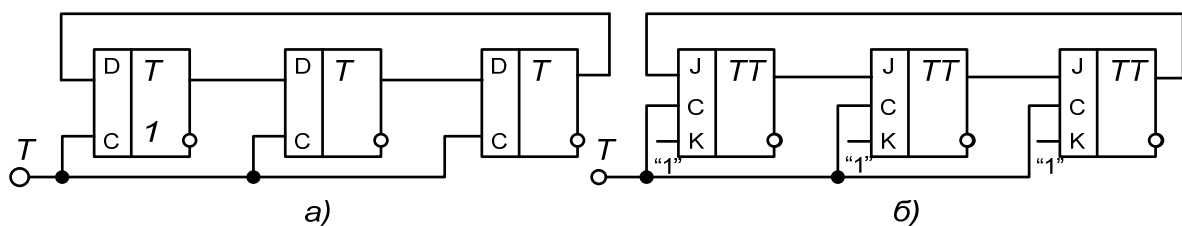


Рисунок 6.4 – Кільцеві лічильники на D -тригерах а) і JK -тригерах б)

Недоліком кільцевих лічильників такого типу є те, що потрібна велика кількість тригерів для забезпечення великих коефіцієнтів перерахунку. Цей недолік мінімізують в лічильниках на регістрі з перехресним зворотним зв'язком (так званий «лічильник Джонсона»). Лічильник Джонсона (рис. 6.5) має коефіцієнт перерахунку вдвічі більший числа складових його тригерів. Особливість роботи лічильника Джонсона полягає в тому, що як результат відліку в регістрі спочатку від першого тригера до останнього поширюється «хвиля 1», а потім «хвиля 0». Якщо потрібно забезпечити непарний коефіцієнт перевідліку $2n-1$, то вхід J першого тригера з'єднується з виходом $\overline{Q}_n$ останнього тригера, а вхід K першого тригера – з виходом $\overline{Q}_{n-1}$ передостаннього тригера. Як і в кільцевих лічильниках з маркерною одиницею, так і в лічильниках Джонсона можливі збої у вигляді появи зайвих одиниць («хвиль одиниць» або «хвиль нулів») або зникнення потрібних кодових комбінацій. Для їх усунення вводять в кільцевий лічильник функціональний логічний пристрій, який забезпечує роздільний перепис

«1» з останнього тригера в перший тільки за умови, що всі інші тригери перебувають у стані «0».

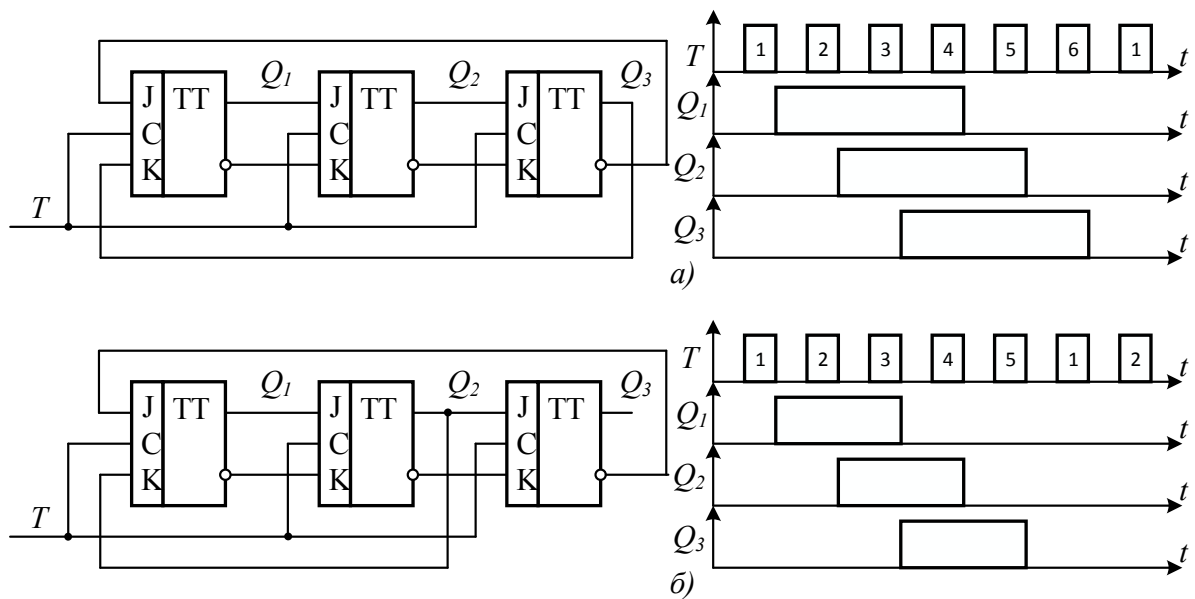


Рисунок 6.5 – Лічильники з $K_{dc} = 6$, а) і $K_{dc} = 5$, б)
на регістрі з перехресним зворотним зв'язком

6.4 Контрольні запитання і завдання

1. Для чого використовуються регістри?
2. Що таке довжина регістра?
3. Які регістри називають реверсивними?
4. Від чого залежить складність комбінаційної схеми регістра?
5. У яких випадках для побудови регістра необхідно використовувати тригери з внутрішньою затримкою?

7 ЛІЧИЛЬНИКИ

Лічильником називають пристрій для виконання мікрооперацій інкрементації та декрементації і формування певного числа, що виражає результат цих мікрооперацій, тобто для підрахунку (лічби) кількості вхідних сигналів, які свідчать про реалізацію деяких подій.

7.1 Двійкові лічильники з послідовним переносом

Це послідовні лічильники з коефіцієнтом перерахунку (див. попередній розділ) $K_{\text{ліч}} = 2^n$, які реалізують послідовним з'єднанням n лічильних триггерів. Структурна схема та часові діаграми інкрементного (який функціонує на додавання вхідних сигналів) лічильника з $K_{\text{ліч}} = 8$ показані на рис. 7.1. а.

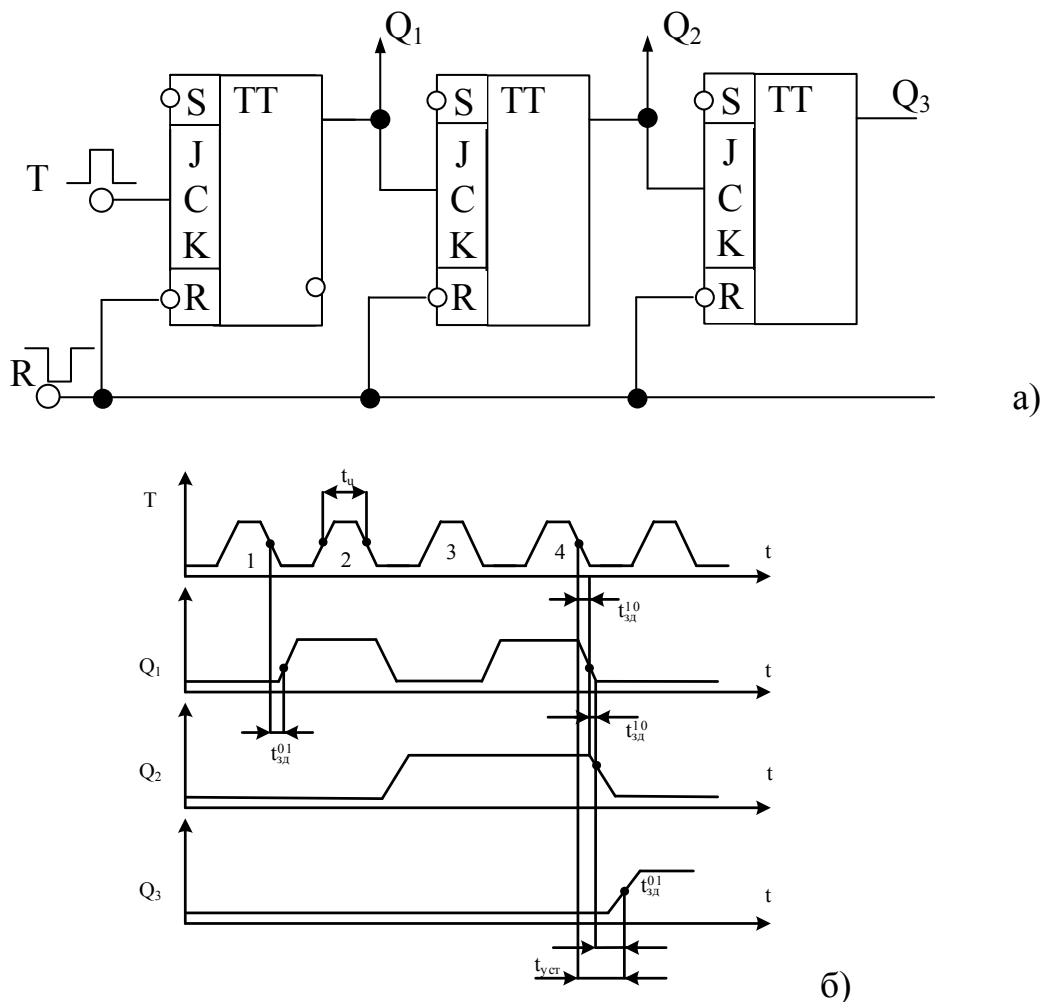


Рисунок 7.1 – Двійковий лічильник з безпосередніми зв'язками

Послідовні лічильники характерні тим, що сигналами управління для перемикання старших розрядів є сигнали, що знімаються з інформаційних

виходів молодших розрядів, а вхідні сигнали надходять на вхід першого розряду (рис. 7.1, б). Для таких двійкових лічильників максимальна частота надходження вхідних імпульсів визначається з виразу $f_{лнч} = (1/t_u + t_{ycm})$, де t_u – тривалість вхідних імпульсів; $t_{ycm} = n * t_{з.ср}$; n – число тригерів; $t_{з.ср} = 0,5(t_{з0}^{01} + t_{з0}^{10})$ – середня затримка перемикавання тригера. Перевага – технічна простота, а недолік – невисока швидкодія.

7.2 Двійкові лічильники з наскрізним і паралельним переносом

Синхронні двійкові лічильники – це паралельні лічильники (рис. 7.2) де вхідні сигнали подають на лічильні входи всіх розрядів одночасно, а зміна стану i -го розряду відбувається тільки при певному стані всіх попередніх розрядів.

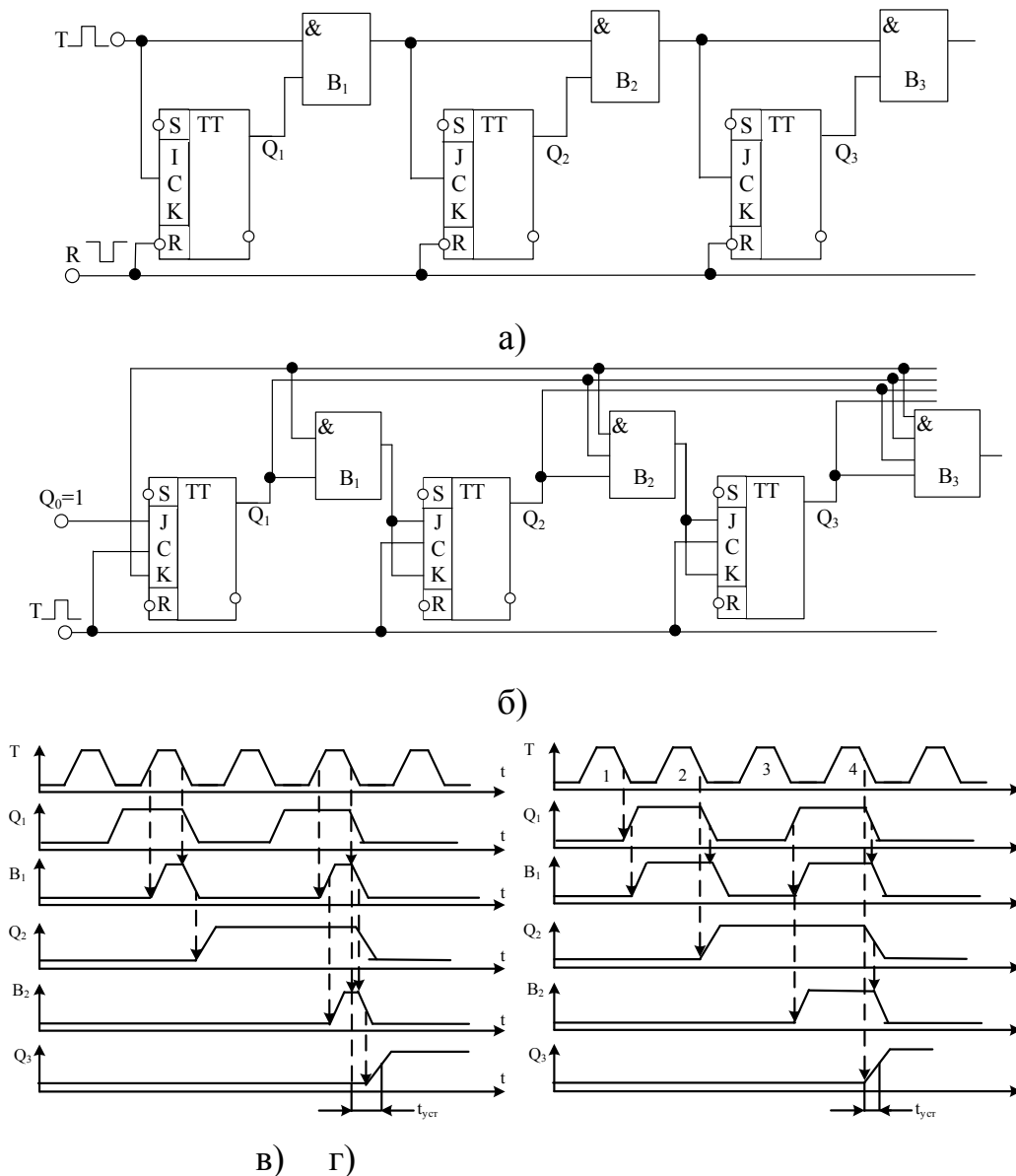


Рисунок 7.2 – Двійковий лічильник: а) з послідовним (асинхронним) б) з паралельним (синхронним) переносом і їх часові діаграми в, г)

Для цього типу лічильників максимальна частота проходження $f_{ліч} = 1/(t_u + t_{уст})$, де $t_{уст} = n * t_{з.ср}$. Перевага – висока швидкодія, оскільки $t_{уст}$ не залежить від числа розрядів. Недолік – потрібні додаткові функціональні пристрої для фіксації стану молодших розрядів, число входів яких дорівнює n (де n – порядковий номер розряду лічильника), і нерівномірне навантаження на входи тригерів.

7.3 Декрементні та реверсивні лічильники

Декрементні лічильники призначені для отримання різниці між числом N , записаним в лічильнику, і числом n сигналів, що надходять на його вхід (рис. 7.3). При цьому, на відміну від інкрементних лічильників, у декрементних лічильниках сигналами управління для старших розрядів є сигнали, що знімаються з інверсних інформаційних виходів молодших розрядів.

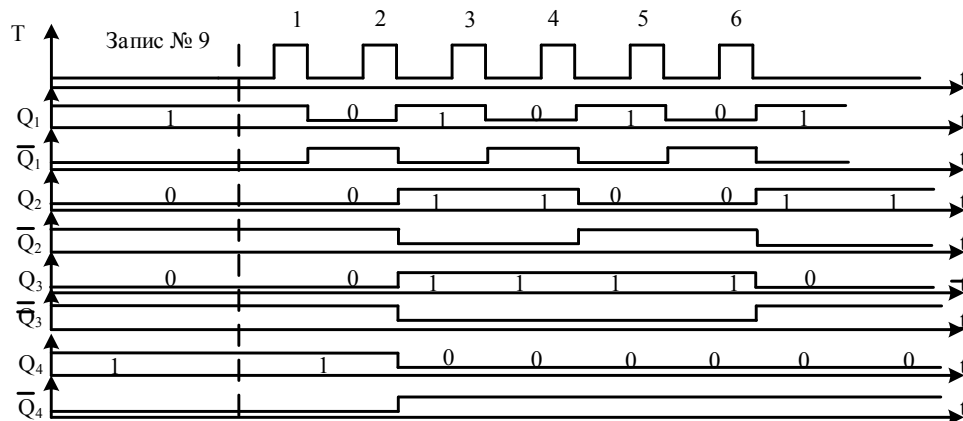
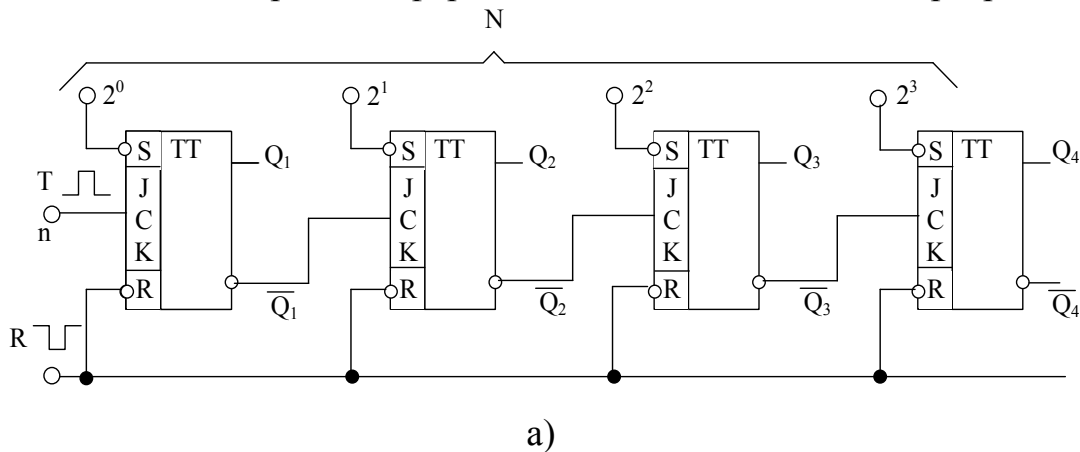


Рисунок 7.3 – Декрементні двійковий лічильник а) і часова діаграма б)

Реверсивні лічильники (рис.7.4) дозволяють виконувати операції додавання і віднімання входних сигналів. Залежно від зовнішнього сигналу управління $\pm V$ лічильні входи наступних тригерів з'єднуються або з пря-

мими, або з інверсними інформаційними виходами попередніх. У режимі додавання ($+V = 1, -V = 0$) відкриваються вентиля B_1, B_4 , і з надходженням кожного лічильного сигналу результат роботи лічильника буде збільшуватися на одиницю. У режимі віднімання ($+V = 0, -V = 1$) відкриваються вентиля B_2, B_5 , і під час надходження кожного лічильного сигналу число в лічильнику буде зменшуватися на одиницю.

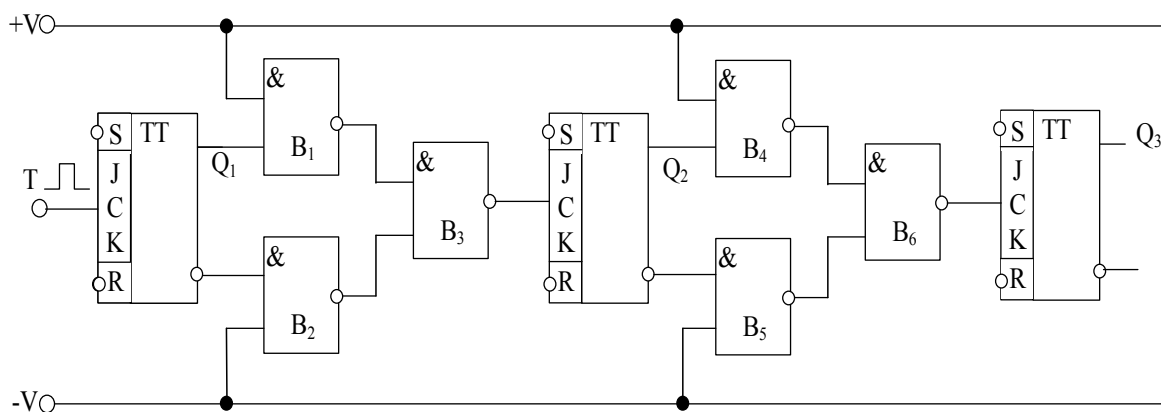


Рисунок 7.4 – Реверсивний двійковий лічильник

7.4 Лічильники з довільним коефіцієнтом лічби

Лічильники з коефіцієнтом лічби $K_{ліч} = 1 \dots 2^n$ будують на основі двійкових лічильників. Принцип роботи таких лічильників полягає у вилученні «зайвих» стійких M станів у двійкового лічильника з коефіцієнтом перерахунку 2^n , при цьому число заборонених станів $M = 2^n - K_{ліч}$. Лічильники з довільним коефіцієнтом лічби за способом побудови ділять на лічильники з природним і довільним порядком лічби. У лічильниках з природним порядком лічби порядок відліку починають з 0 і закінчують числом $K_{ліч} - 1$, а потрібний коефіцієнт лічби забезпечується пристроєм блокування переносу. На рис. 7.5, 7.6 та 7.7 показані схеми лічильників з природним порядком рахунку. На рис. 7.5 наведено схему з регульованим коефіцієнтом лічби, на рис. 7.6 – десятковий паралельний лічильник на JK-тригерах, а на рис. 7.7 – десятковий послідовний лічильник.

У лічильниках з довільним порядком лічби в процесі підрахунку лічильник приймає стани, що не відповідають їх еквівалентному поданню в двійковому численні. За способом побудови ці лічильники ділять на лічильники з примусовою лічбою і початковим устанавленням M .

У лічильниках з примусовою лічбою унеможливлення заборонених станів M досягається за рахунок зовнішнього примусового встановлення окремих його розрядів в стан «1». У результаті цього в процесі рахунку результати лічильника змінюються в природній формі, починаючи від 0 і закінчуючи деяким числом $X \leq K_{ліч} - 2$. З приходом чергового сигналу лічильник замість стану $X + 1$ приймає стан $X + 1 + M$, який в двійковому запису дорівнює $2^n - 1$, тобто всі розряди лічильника приймають стан «+».

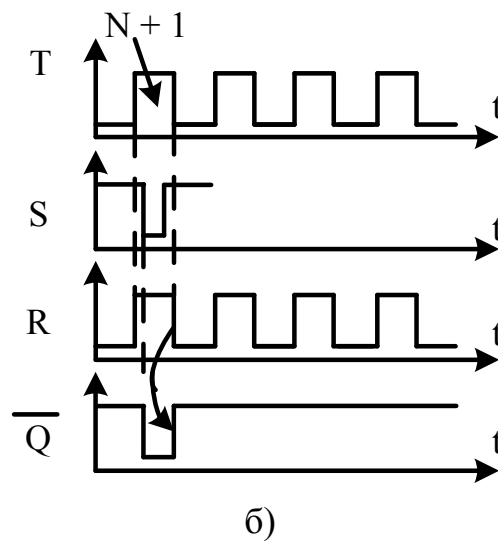
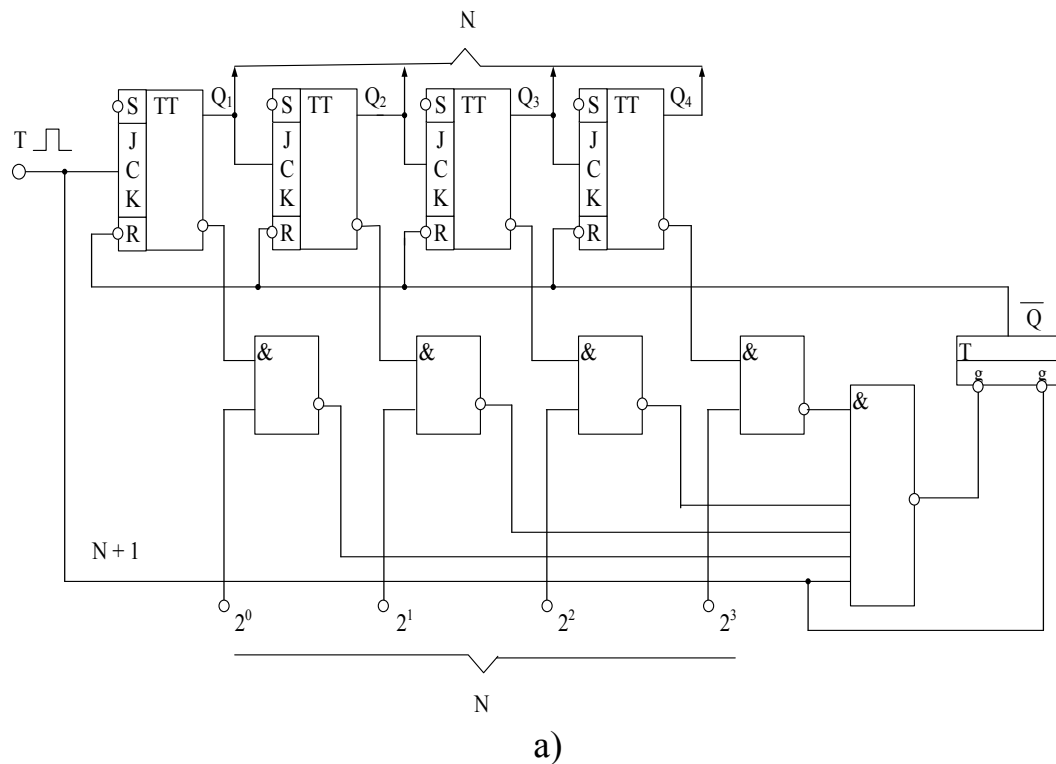


Рисунок 7.5 – Лічильник із регульованим коефіцієнтом лічби:
а) схема; б) часові діаграми

При цьому примусова лічба здійснюється за рахунок введення зворотніх зв'язків зі старших розрядів на молодші, під дією яких відповідні розряди лічильника, що знаходяться в стані «0», позачергово перемикаються в стан «1». Зворотні зв'язки зі старших розрядів на молодші підключаються або безпосередньо на лічильний вхід молодших розрядів, або на вхід установлення «1».

Для лічильника на рис.7.6 коефіцієнт лічби $K_{\text{ліч}} = N+1$ задають двійковим числом N на входах розрядів з вагою $2^0, 2^1, 2^2, 2^3$.

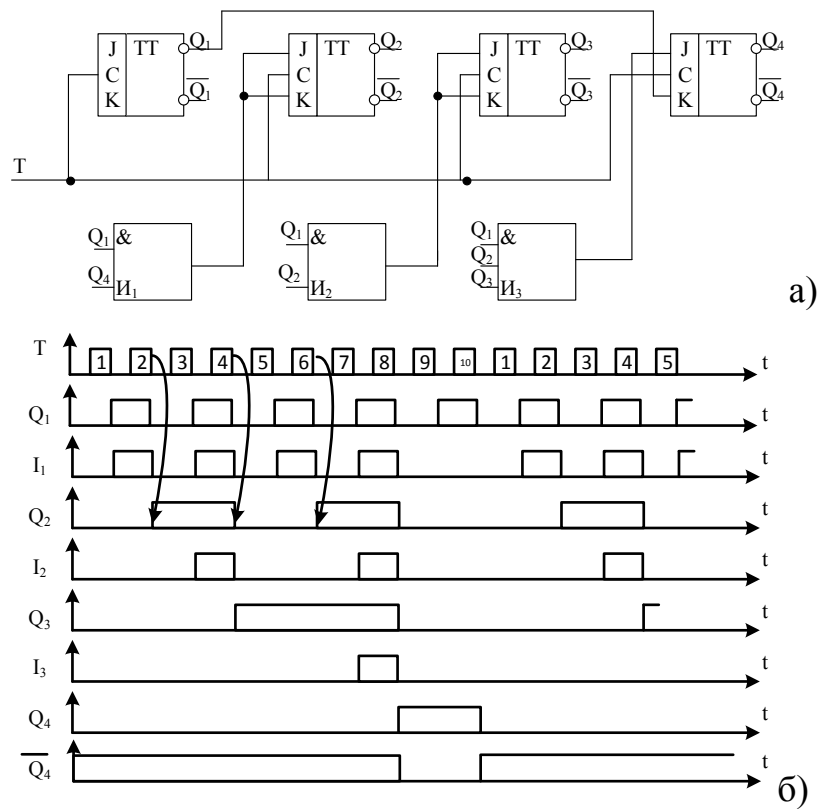


Рисунок 7.6 – Схема десяткового паралельного лічильника на *JK*-тригерах а) та його часові діаграми б)

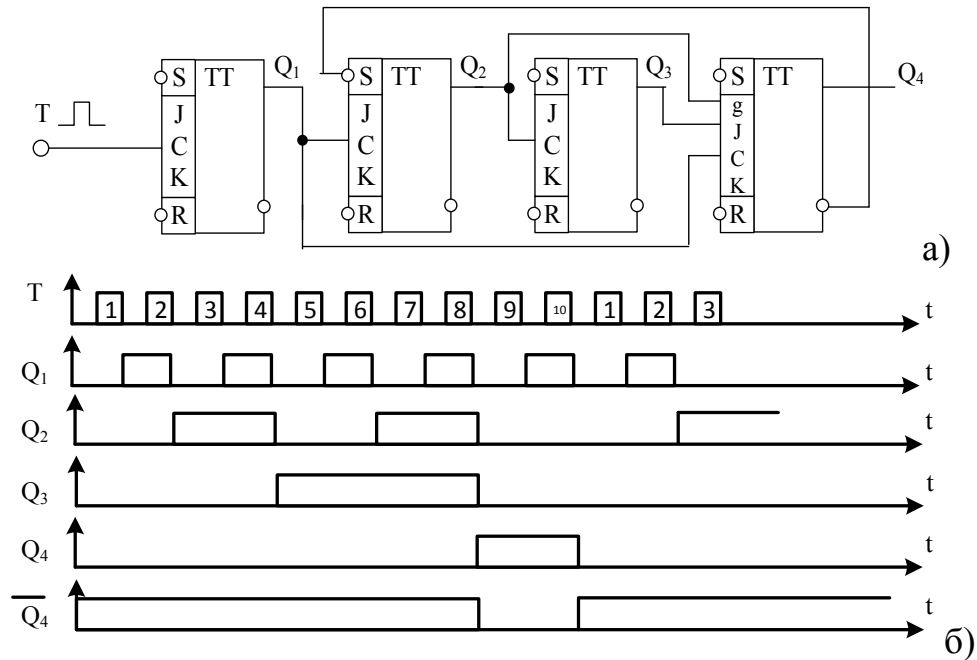
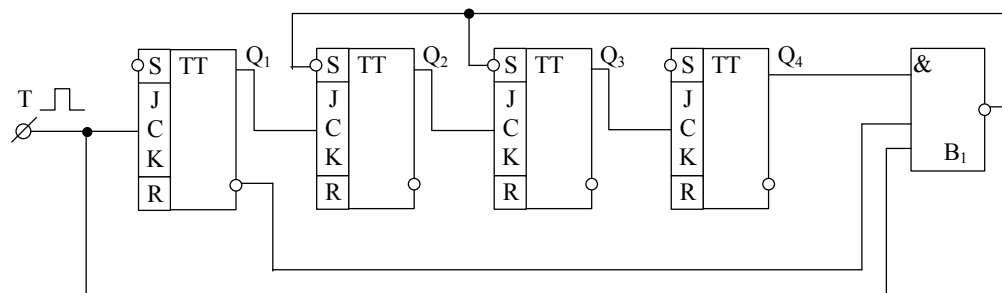


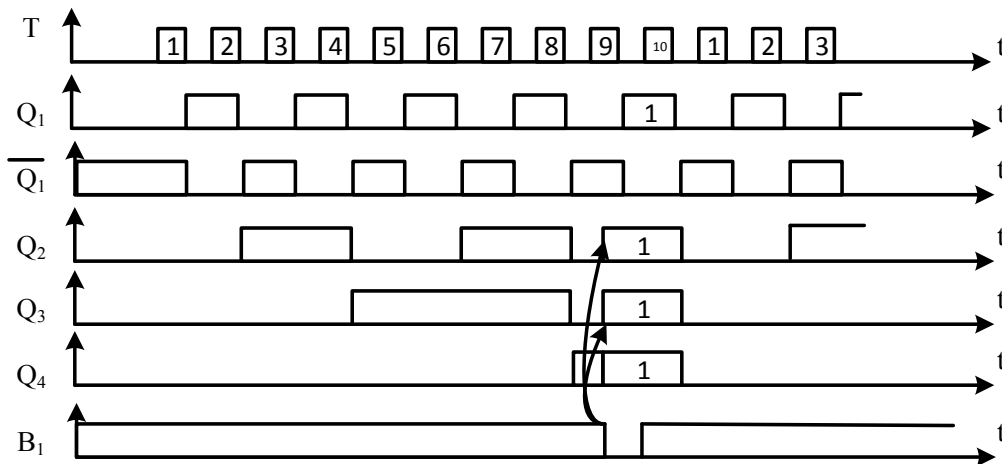
Рисунок 7.7 – Десятковий послідовний лічильник на JK-тригерах та його часові діаграми

7.5 Десяткові лічильники

Електронна промисловість серійно випускає велику кількість лічильників: декадні з фазо імпульсним поданням інформації, двійково-десяткові чотирирозрядні, лічильники-дільники на 10, 12 та інші. Принцип роботи десяткових лічильників розглянемо на деяких прикладах їх побудови. На рис. 7.8, а) зображена схема лічильника з $K_{\text{ліч}} = 10$, побудованого на JK -тригерах за схемою послідовного лічильника. Для вилучення шести надлишкових станів введено зворотний зв'язок з прямого виходу 4-го розряду та інверсного виходу i -го розряду через ЛЕ І-НЕ (B_1) на входи S третього і другого розрядів.



а)



б)

Рисунок 7.8 – Лічильник десятковий з примусовою лічбою

Під час перших восьми лічильних сигналів цей пристрій працює як інкрементний лічильник, послідовно приймаючи стани 0,1–7,8 (рис. 7.8, б). З приходом 9-го лічильного сигналу на виході вентилу B_1 формується сигнал логічного «0», що надходить на входи третього і другого розрядів відповідно, тобто після закінчення 9-го імпульсу в стані «1» виявиться не тільки перший розряд, але також другий і третій розряди. Таким чином, лічильник нараховує додаткові 6 одиниць, тобто зі стану 1000 переходить не в стан 1001, а в стан 1111. З приходом наступного, 10-го сигналу лічильник

знову повернуться в початковий стан 0000. Недоліком такої структури є те, що десятковій цифрі 9 відповідає двійкове число 1111, тобто десяткове 15, крім того неможливо побудувати лічильники з коефіцієнтами лічби 9, 11, 13, 15.

У лічильниках з початковим установленням M лічба здійснюється не в процесі роботи лічильника, а за допомогою його зовнішнього встановлення в початковий стан, що відповідає числу M заборонених станів на початку рахунку. У загальному випадку для побудови лічильника з початковим встановленням M і заданим коефіцієнтом перерахунку необхідно визначити різницю $n \geq \frac{\lg K_{\text{ліч}}}{\lg 2}$ двійкового лічильника, на основі якого повинен виконуватися лічильник, тобто $K_{\text{ліч}} \leq 2^n$; за виразом $M = 2^n - K_{\text{ліч}}$ знайти число зайвих станів і записати у вигляді n -розрядного двійкового числа; подати сигнали заборонного зворотного зв'язку зі старшого розряду на лічильні входи тих тригерів, які повинні знаходитися в стані «1», якщо в нього задалегідь записати число M (рис. 7.9).

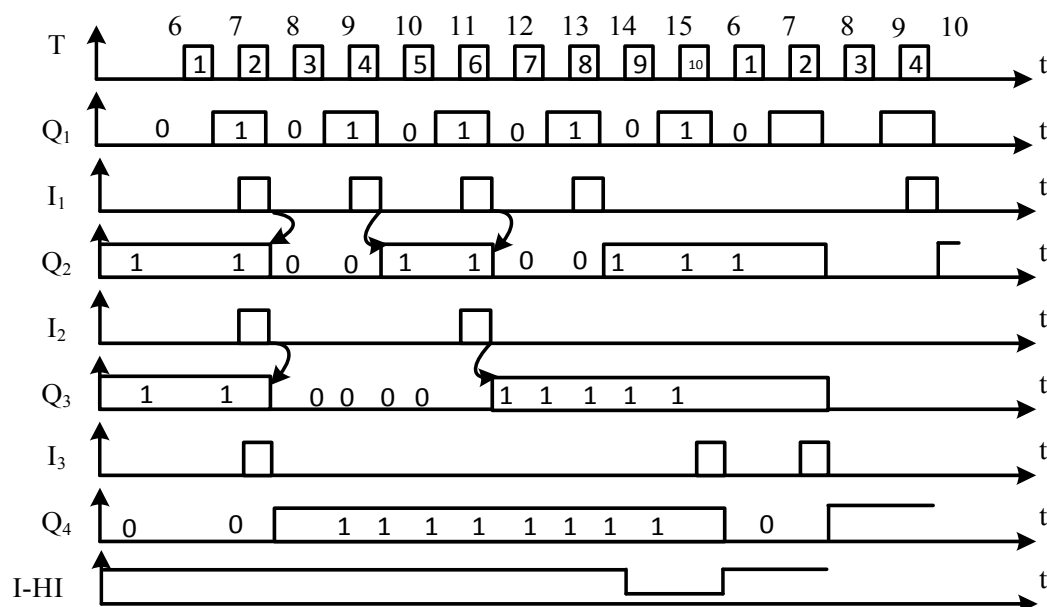


Рисунок 7.9 – Лічильник десятковий з початковим установленням $M=0110$

7.6 Лічильники в коді Фібоначчі

З усього переліку мікрооперацій, що застосовуються у цифровій техніці, лічба є найповільнішою операцією зміни числового значення деякого операнда. Тому підвищення швидкодії лічильників залишається актуальною задачею для комп'ютерних схемотехніків.

Відомо способи підвищення швидкодії лічильників за рахунок організації послідовного, наскрізного чи паралельного переносу та їх комбінацій. Проте всі вони мають недоліки. Паралельний перенос дозволяє отримати

найвищий показник швидкодії, але потребує значних апаратних витрат, що експоненціально зростають при нарощуванні розрядності. Тому цей спосіб практично не використовується при побудові багаторозрядних лічильників. Найменші апаратні витрати і найнижчу швидкодію мають лічильники з послідовним переносом. Наскрізний перенос дозволяє при порівняно невеликих апаратних витратах досягти значного підвищення швидкості лічби. Якщо швидкодія лічильників з наскрізним переносом є недостатньою, то використовуються різноманітні комбінації паралельного і наскрізного переносу, що дозволяють підвищити швидкість, але призводять при цьому до збільшення апаратних витрат. Тому відомі методи організації переносу є малоефективними при побудові багаторозрядних швидкодіючих лічильників. В той же час відомі спроби побудови швидкодіючого лічильника, стани якого закодовані двійковими символами згідно з ваговими рядами, які визначаються послідовностями Фібоначчі. Такий підхід при невеликих апаратних витратах дозволяє досягти високої швидкодії за рахунок того, що під час лічби на кожному кроці лише невелика кількість розрядів змінює свій стан. Суть запропонованого підходу полягає в тому, що на кожному кроці одночасно з додаванням одиниці у молодший розряд стану лічильника виконуються всі елементарні згортки в інші розряди, починаючи з третього. Елементарна згортка замінює дві сусідні одиниці Фібоначчієвого подання двійкового числа на одинцю більш старшого розряду (011→100). Збільшення будь-якого розряду стану лічильника, починаючи з другого, відбувається лише за рахунок переносу з молодших розрядів. При цьому вага переносу завжди більша ваги молодшого розряду. Тому не відбувається переповнення лічильника. Більш того, кількість сусідніх одиниць стану лічильника, через які можливий перенос на кожному такті, становить не більше двох. Дослідниками Фібоначчієвого кодування показано, що коли на кожному кроці роботи Фібоначчієвого лічильника додавати одиницю до молодшого розряду його стану та виконувати всі можливі елементарні згортки, то в двійковому запису його стану не може бути більше двох сусідніх одиниць, через які відбувається перенос. Звідси витікає висновок про можливість побудови лічильників, в яких перенос на кожному кроці розповсюджується не далі, ніж на два розряди. На рис. 7.10 показана структурна організація швидкодіючого шестирозрядного фібоначчієвого лічильника, який складається з розрядних блоків 1; 2; 3.1; 3.2; 3.3; 3.4.

На рис. 7.11, а зображено структуру блока першого розряду, а на рис. 7.11, б – структуру блока другого розряду лічильника. Оскільки при використанні фібоначчієвого кодування вагу першого і другого розрядів встановлюють інакше порівняно з вагою інших розрядів, то розрядні блоки 1 і 2 також відрізняються від інших розрядних блоків. Решта блоків 3.1, 3.2, 3.3 і 3.4 мають однакову структуру (рис. 7.11, в). З наведених рисунків видно, що у середньому кожен розряд лічильника містить крім лічильного тригера ще два логічних елементи І-НІ.

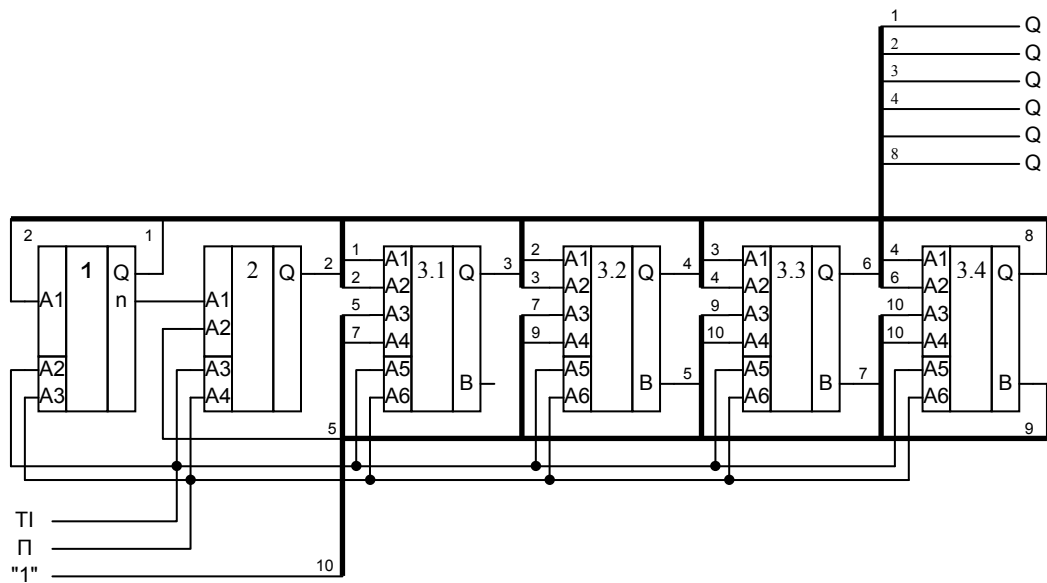


Рисунок 7.10 – Структурна організація швидкодіючого Фібоначчівського лічильника

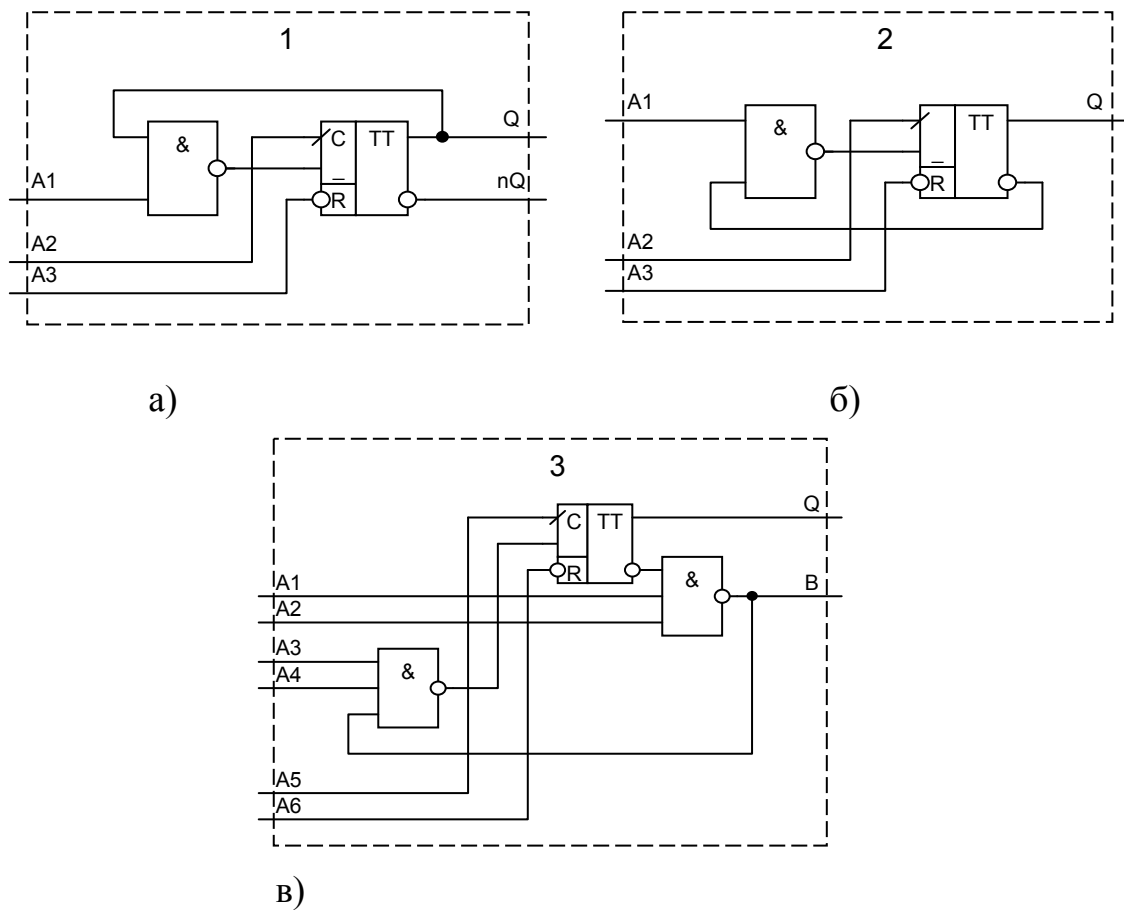


Рис. 7.11. – Структурна організація розрядних блоків лічильника:
а) структура блока першого розряду; б) структура блока другого розряду;
в) структура розрядних блоків, починаючи з третього

Отже, апаратні витрати на такий лічильник приблизно дорівнюють апаратним витратам на відомий лічильник з наскрізним переносом. З іншого боку, максимальна затримка лічильника на кожному кроці визначається часом виконання елементарної згортки і дорівнює часу перемикання лічильного тригера плюс затримки двох елементів І-НІ. Тобто, швидкодія даного лічильника відповідає швидкодії відомого лічильника з паралельним переносом.

На рис. 7.12 наведені часові діаграми, що пояснюють роботу лічильника і підтверджують його працездатність. На часових діаграмах не показано затримки тригерів і логічних елементів.

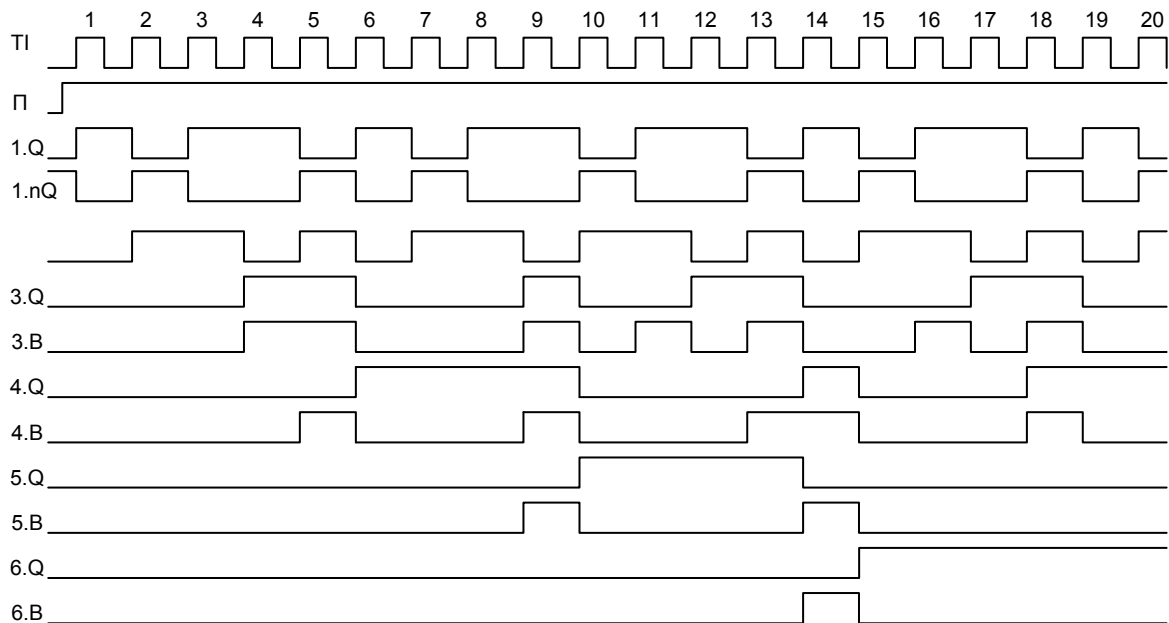


Рисунок 7.12 – Часові діаграми роботи лічильника

7.7 Контрольні запитання і завдання

1. Узагальнена структура синхронного лічильника.
2. Як класифікують лічильники з природним порядком лічби за способом організації переносу між розрядами?
3. Навести приклад лічильника з послідовним переносом.
4. Записати функції збудження i -го розряду інкрементного, декрементного та реверсивного лічильників з паралельним переносом.
5. Записати функції збудження i -го розряду інкрементного, декрементного та реверсивного лічильників з наскрізним переносом.
6. Порівняти особливості і характеристики паралельного та наскрізного переносу у лічильниках.
7. Назвати основні часові характеристики лічильників.
8. Сформулювати методику синтезу синхронних лічильників з не природним порядком лічби.

8 СУМАТОРИ

Перелік скорочень і умовних позначень до розділу 8

АЛП – арифметико-логічний пристрій; ПС – півсуматор; ПОС – повний однорозрядний суматор; ОС – однорозрядний суматор.

Суматори виконують мікрооперацію арифметичного (без врахування знаків) додавання операндів за правилами певної системи числення. Арифметичне додавання як мікрооперація є лише складовою частиною операції додавання, яка входить до системи команд комп'ютера та є більш повною і складною, оскільки крім елементарного арифметичного додавання потрібно ще виконувати ряд інших операцій і процедур, як-от: врахування знаків операндів і заданої операції, вирівнювання порядків доданків, нормалізації результату, округлення суми тощо. Зазначені дії виконуються в арифметико-логічних пристроях (АЛП), ядром яких є суматори. Класифікація суматорів показана на рис. 8.1.

За кількістю входів розрізняють півсуматори, однорозрядні суматори (ОС) і багаторозрядні суматори. Багаторозрядні суматори поділяють на послідовні, в яких обробку даних ведуть по черзі, розряд за розрядом, на одному й тому ж обладнанні, та паралельні, в яких операнди додають одночасно по всіх розрядах і для кожного розряду є відповідне обладнання. За способом організації міжрозрядних переносів паралельні суматори поділяють на суматори з послідовними, паралельними переносами і з груповою структурою. В останніх розрядна сітка розділена на поля, які оброблюють групи розрядних пристроїв. У групах і між ними можуть застосовуватися різні способи реалізації переносів, причому в найменуваннях суматорів спочатку вказують вид переносів всередині групи. Наприклад, термін «паралельний суматор з паралельно-паралельним переносом» означає, що суматор має групову структуру, в якій в групах і між ними реалізовано паралельний перенос. Комбінаційні суматори є, по суті, функціональними перемикальними пристроями в загальноприйнятому значенні цих слів. Накопичувальні суматори мають пам'ять, в якій акумулюють результати додавання так, що черговий операнд додають до попереднього результату, який вже знаходиться в спеціальному регістрі-акумуляторі.

За способом тактування розрізняють синхронні і асинхронні суматори. Для синхронних суматорів характерні постійність і максимальність часу, що відводиться для додавання, незалежно від числових значень доданків, в асинхронних виробляється ознака завершення операції, при цьому середній час додавання істотно менше максимального. Залежно від системи числення розрізняють двійкові, двійково-десяткові та інші види суматорів.

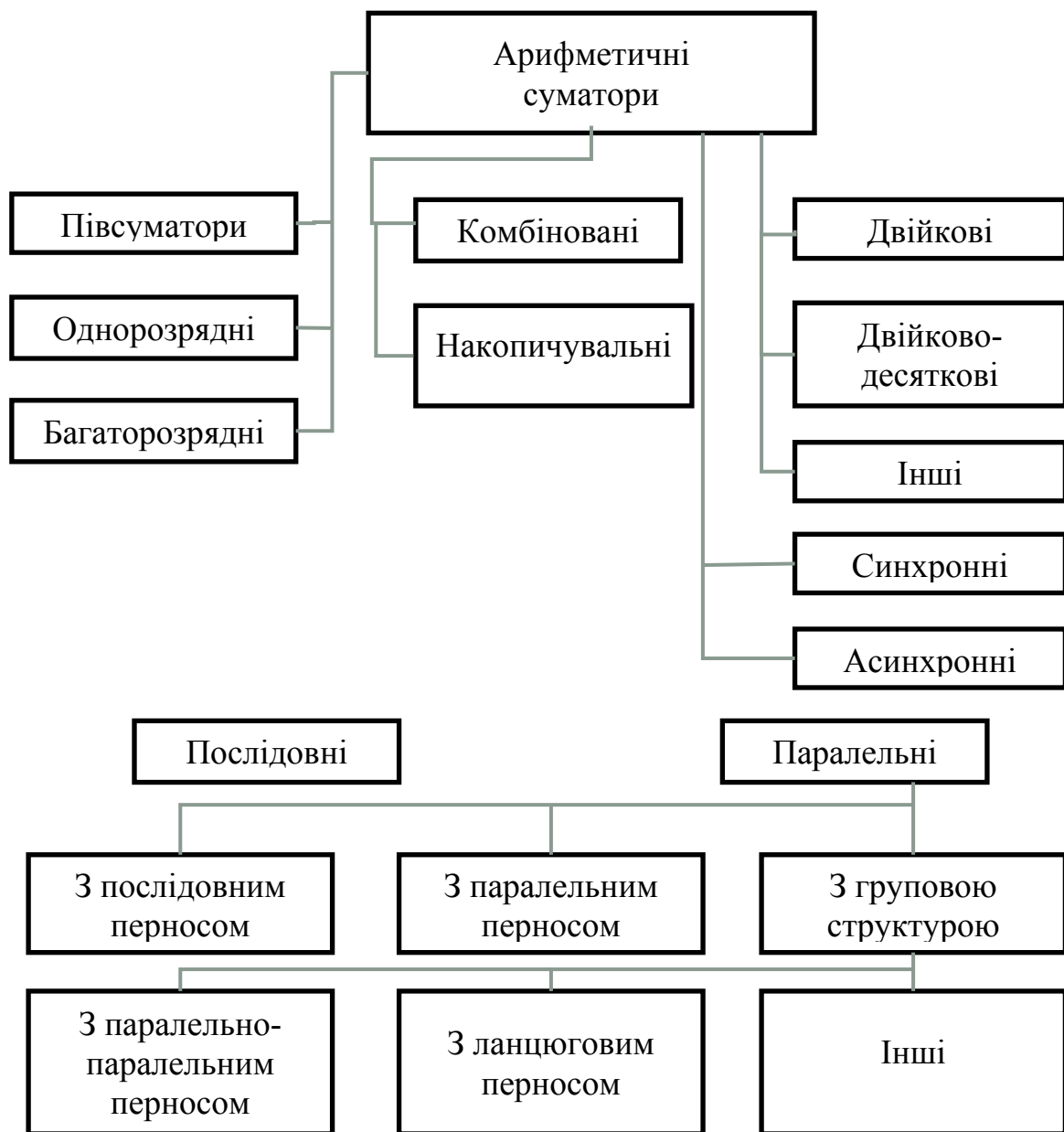


Рисунок 8.1 – Класифікація арифметичних суматорів

8.1 Однорозрядні півсуматори і суматори

В комп'ютерній схемотехніці застосовують однорозрядні суматори на два і на три входи. Суматор на два входи називають півсуматором (ПС), а на три входи – повним однорозрядним сумматором (ПОС). Півсуматор має два входи і два виходи, які відповідають сигналам суми і переносу відповідно до формул:

$$S_i = \bar{a}_i b_i + a_i \bar{b}_i = \overline{a_i \oplus b_i}, p_i = a_i b_i.$$

ПС реалізує лише частину операції додавання, оскільки не враховує переносу з сусіднього молодшого розряду. Враховуючи цю обставину

ПОС звичайно називають однорозрядними суматорами, що мають три входи і забезпечують додавання цифр доданків a_i і b_i з переносами з попереднього розряду p_{i-1} (табл. 8.1).

У цій таблиці в полі p_i об'єднані рядки, що відрізняються інверсіями аргументів. При цьому інвертуються і значення функцій суми та переносу. Функції, що мають таку властивість, називають самодвоїстими. Позначаючи X – вектор аргументів, можна записати:

$$s(\bar{X}) = \bar{s}(X), p(\bar{X}) = \bar{p}(X).$$

Таблиця 8.1 – Функція утворення переносу

a_i	b_i	p_{i-1}	s_i	p_i	
0	0	0	0	0	
0	0	1	1	0	
0	1	0	1	0	
0	1	1	0	1	
1	0	0	1	0	
1	0	1	0	1	
1	1	0	0	1	
1	1	1	1	1	

Функції, реалізовані ПС і ПОС, відповідають виразам:

$$\begin{aligned}
 s_i' &= \bar{a}_i b_i = a_i \bar{b}_i, \\
 s_i &= \bar{a}_i \bar{b}_i p_{i-1} + a_i \bar{b}_i \bar{p}_{i-1} + \bar{a}_i b_i \bar{p}_{i-1} + a_i b_i p_{i-1}, \\
 s_i &= p_{i-1} \oplus (a_i \oplus b_i), \\
 p_i' &= ab, \\
 p_i &= a_i b_i \bar{p}_{i-1} + \bar{a}_i b_i p_{i-1} + a_i \bar{b}_i p_{i-1} + a_i b_i p_{i-1} = a_i b_i + a_i p_{i-1} + b_i p_{i-1}, \\
 p_i &= a_i b_i (\bar{p}_{i-1} + p_{i-1}) + p_{i-1} (\bar{a}_i b_i + a_i b_i) = a_i b_i + p_{i-1} (a_i \oplus b_i).
 \end{aligned}$$

Таблиця 8.2 – Таблиця істинності ПС

a_i	b_i	s_i'	p_i'
0	0	0	0
1	0	1	0
0	1	1	0
1	1	0	1

Тут a_i, b_i, p_{i-1} – цифри доданків; s' і s – суми, реалізовані ПС і ПОС, p' і p – переноси ПС і ПОС. На рис. 8.2 показана реалізація ПОС на елементах І-АБО-НІ. Користуючись тотожними перетвореннями (див. розділ 1) перемикальних функцій, можна по-різному подавати функції ПС і ПОС і отримати багато варіантів реалізації функцій суми і переносу:

$$\begin{aligned}
s_i &= \overline{\overline{a_i}b_i\overline{p_{i-1}} + p_i(\overline{a_i} + \overline{b_i} + \overline{c_i})}, \\
s_i &= a_i b_i p_{i-1} + \overline{p_i}(a_i + b_i + p_{i-1}), \\
s_i &= (a_i \overline{b_i} + \overline{a_i} b_i) \overline{p_{i-1}} + (a_i b_i + \overline{a_i} \overline{b_i}) p_{i-1}, \\
p_i &= \overline{a_i} \overline{b_i} + \overline{a_i} p_{i-1} + \overline{b_i} p_{i-1}, \\
p_i &= a_i b_i + a_i p_{i-1} + b_i p_{i-1}, \\
p_i &= a_i b_i + \overline{(a_i b_i + \overline{a_i} \overline{b_i})} p_{i-1}.
\end{aligned}
\tag{8.1}$$

Залежно від виду базової функції, виконуваної ЦІМС, І-НІ, АБО-НІ, І-АБО-НІ і т. д., для реалізації суматора можуть бути взяті за основу різні вирази. Так, використання останнього виразу в ПОС на елементах І-НІ дозволяє реалізувати суму як

$$s_i = \overline{p_i p_{i-1}} + F_i p_{i-1} p_i = a_i b_i + \overline{F_i} p_{i-1},$$

де $F_i = \overline{a_i} \overline{b_i} + a_i b_i$.

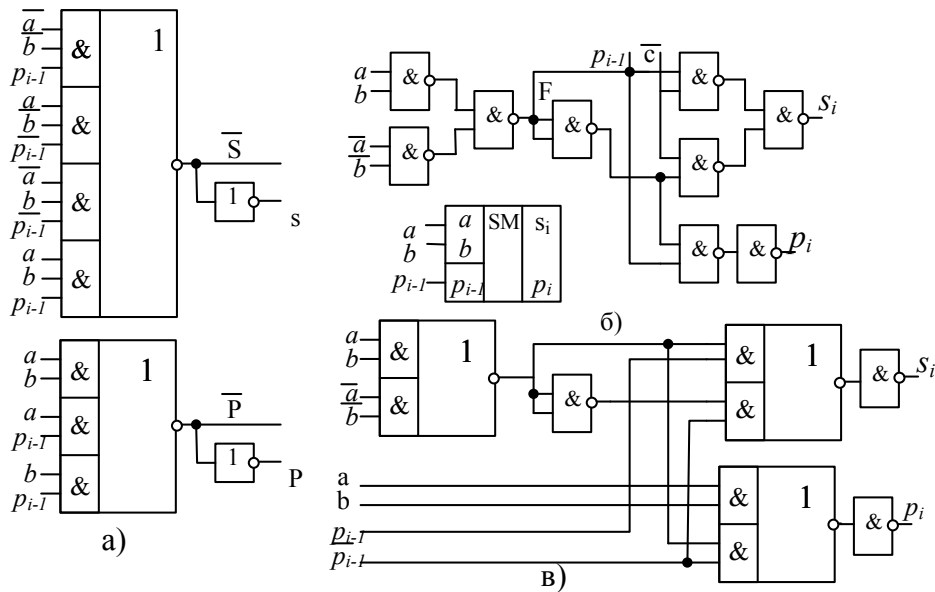


Рисунок 8.2 – Однорозрядний суматор з парафазними входами:

- а) – на елементах І-АБО-НІ; б) – на елементах І-НІ;
 в) – на елементах І-АБО-НІ та І-НІ

Швидкодія ПОС визначається часом встановлення вихідних сигналів суми S і переносу P після появи сигналів на вході. На рис. 8.2, б наведена схема ПОС на елементах І-НІ. Затримка формування суми S і переносу становить $5t_{зд.ср}$. Застосовуючи елементи І-АБО-НІ, можна побудувати ПОС, схема якого показана на рис. 8.2, в. Швидкодія цього ПОС визнача-

ється часом затримки формування суми, що дорівнює $4t_{зд.ср}$, а переносу – $3t_{зд.ср}$.

Зважаючи на багатоваріантність реалізацій функцій суми і переносу (8.1) на практиці варто враховувати особливості конкретної серії ЦІМС. Наприклад, на елементах ТТЛ при побудові ПОС перенос можна отримати всього на одному ЛЕ (а саме: К155ЛР3) типу АБО-НІ. В цьому разі слід реалізувати вираз для інверсного значення переносу:

$$\bar{p}_i = \overline{a_i b_i + a_i p_{i-1} + b_i p_{i-1}}.$$

Вираз для S_i подібним чином реалізувати не можна, якщо в серії елементів немає мікросхеми типу 3-3-3-3І-4АБО-НІ. Тому потрібно спробувати зменшити кількість літер в мінтермах, що входять у формулу для S_i . Ідея спрощення полягає в тому, щоб використовувати отримане значення переносу як допоміжний аргумент при обчисленні S_i . Складемо таблицю для S_i як функції чотирьох аргументів – a_i , b_i , p_{i-1} і p_i (табл. 8.3).

Таблиця 8.3 – Таблиця для суми S_i

a_i	b_i	p_{i-1}	p_i	s_i
0	0	0	0	0
0	0	0	1	X
0	0	1	0	1
0	0	1	1	X
0	1	0	0	1
0	1	0	1	X
0	1	1	0	X
0	1	1	1	0
1	0	0	0	1

a_i	b_i	p_{i-1}	p_i	s_i
1	0	0	1	X
1	0	1	0	X
1	0	1	1	0
1	1	0	0	X
1	1	0	1	0
1	1	1	0	X
1	1	1	1	1

У частині таблиці справа на наборах аргументів, які є нереальними (наприклад, одиничне значення переносу p_i при нульових значеннях всіх вхідних змінних), проставлені невизначені значення функції X, які можна трактувати довільно. Отже, з табл. 8.1 випливає:

$$s_i = \bar{a}_i \bar{b}_i p_{i-1} + \bar{a}_i b_i \bar{p}_{i-1} + a_i \bar{b}_i \bar{p}_{i-1} + a_i b_i p_{i-1},$$

$$p_i = a_i b_i + a_i p_{i-1} + b_i p_{i-1} = a_i b_i + p_{i-1}(a_i + b_i). \quad (8.2)$$

З використанням переносу як допоміжного аргумента і отриманого виразу для S_i після побудови карти Карно (рис. 8.3), отримаємо:

$$S_i = a_i \bar{p}_i + b_i \bar{p}_i + p_{i-1} \bar{p}_i + a_i b_i p_{i-1} = (a_i + b_i + p_{i-1}) \bar{p}_i + a_i b_i p_{i-1} \quad (8.3)$$

		$a_i b_i$			
		00	01	11	10
p_{i-1}	p_i				
00	0	1	x	1	
01	x	x	0	x	
11	x	0	1	0	
10	1	x	x	x	

Рисунок 8.3 – Карта Карно для суми як функції чотирьох аргументів

Схема ПОС, побудованого за формулами (8.2) і (8.3), показана на рис. 8.4, а), а УГП ПОС – на рис. 8.4, б).

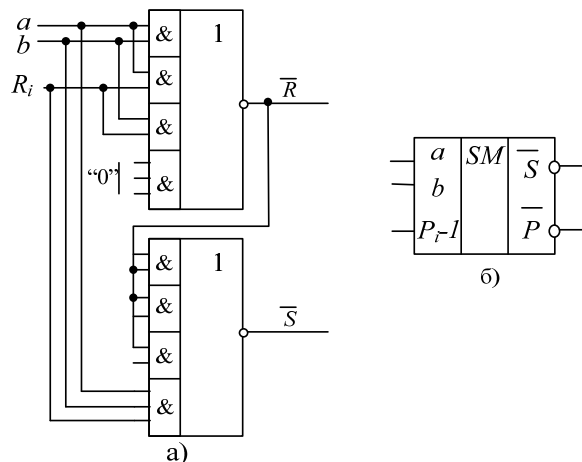


Рисунок 8.4 – Схема однорозрядного суматора а) і його УГП б)

Швидкодію ПОС оцінюють затримками поширення сигналів за чотирма напрямками: «доданок–сума», «доданок–перенос», «перенос–сума» та «перенос–перенос». Для розглянутого ПОС має місце:

$$t_{a\bar{P}} = t_{p\bar{P}} = t_{з.лр.},$$

$$t_{a\bar{S}} = t_{p\bar{S}} = 2t_{з.лр.}$$

Швидкодійні ПОС можна побудувати на елементах типу ЕЗЛ. Оскільки функції суми і переносу залежать від трьох аргументів, то доцільно їх реалізувати на основі багатоярусного перемикача струму. Для безпосередньої реалізації функції на такому перемикачі їх потрібно подати у вигляді інверсії логічної суми наборів аргументів, які відповідають нульовим значенням функції. В цьому випадку:

$$S_i = \overline{\bar{a}_i \bar{b}_i \bar{p}_i + \bar{a}_i \bar{p}_{i-1} + \bar{b}_i p_{i-1}},$$

$$P_i = \overline{\overline{a_i} \overline{b_i} + \overline{a_i} \overline{p_{i-1}} + \overline{b_i} p_{i-1}}.$$

Зауважимо, що реалізація отриманих виразів на двох триярусних перемикачах струму дозволяє отримати досить швидкодійну структуру ПОС. Крім того, можна застосувати більш економічну, за числом транзисторів, структуру на основі ЕЗЛ.

8.2 Накопичувальний послідовний суматор на базі лічильного тригера

Розглянуті вище структури ПОС відносять до пристроїв комбінаційного типу, які не в змозі забезпечити збереження результату додавання після зняття сигналів з входів a , b і c . Для запам'ятовування результатів додавання необхідно використати тригери (RS - або D -типу). Спільно з тригером комбінаційний ПОС виконує функції накопичувального суматора (НС). Роль накопичувального суматора може виконувати лічильний тригер (рис. 8.5) з пристроєм формування переносу, на лічильний вхід якого всі доданки мають подаватися послідовно в часі. При цьому додавання трьох доданків здійснюють за три такти. Тоді при $a = 1$; $b = 0$; $c = 0$ лічильний тригер після першого сигналу $b = 1$ прийме стан «1» і зберігатиме його після трьох тактів додавання (рис. 8.5, б). При значеннях доданків $a = 1$; $b = 1$; $c = 0$ лічильний тригер після другого сигналу $b = 1$ повернеться в стан «0» і сформує сигнал переносу на виході P . Якщо на лічильний вхід тригера надійдуть всі три сигнали $a = 1$, $b = 1$, $c = 1$, то після другого доданка $b = 1$ на виході P формується перенос, а після третього доданка $c = 1$ тригер прийме стан «1». Прямий інформаційний вихід лічильного тригера є виходом суми S однорозрядного накопичувального суматора. Однак швидкодія такого накопичувального суматора може бути в декілька разів вищою, ніж у комбінаційного суматора, оскільки додавання всіх трьох доданків в ньому по всіх розрядах може відбуватися за один такт.

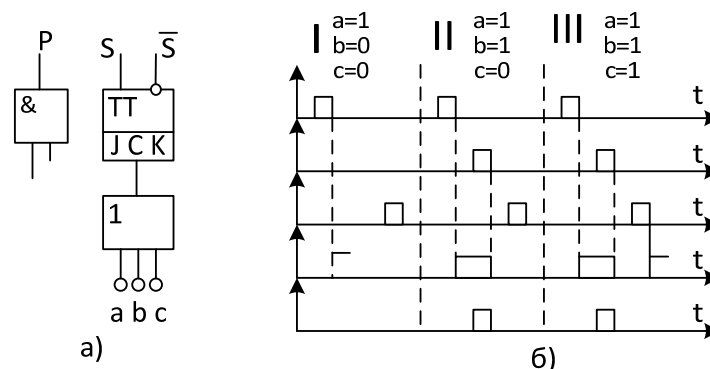


Рисунок 8.5 – Однорозрядний накопичувальний суматор на лічильному тригері а); часова діаграма його роботи б)

В сучасних технічних комп'ютерних засобах НС реалізують найчастіше за допомогою комбінаційного суматора і регістра (рис. 8.6). При цьому комбінаційні суматори функціонують згідно з формулою $S := A + B$, а накопичуючі реалізують формулу $S := S + A$, тобто до вмісту суматора, що має пам'ять, додають черговий доданок і результат залишають в пам'яті, заміщаючи попереднє значення суми. Перед початком додавання регістр встановлюють в нульовий стан за входом R . На вхід A надходить доданок, який потрібно додати до вмісту регістра. При нульовому стані регістра сума дорівнює A . Сигнал C є виконавчим для регістра, під його дією суму S приймають в регістр (на першому такті приймають слово A). Надалі з виходів регістра знімають один доданок (слово B), а за входами A приймають інший. Результат приймають на регістр за сигналом C і надалі цей результат буде другим доданком для наступного циклу додавання.

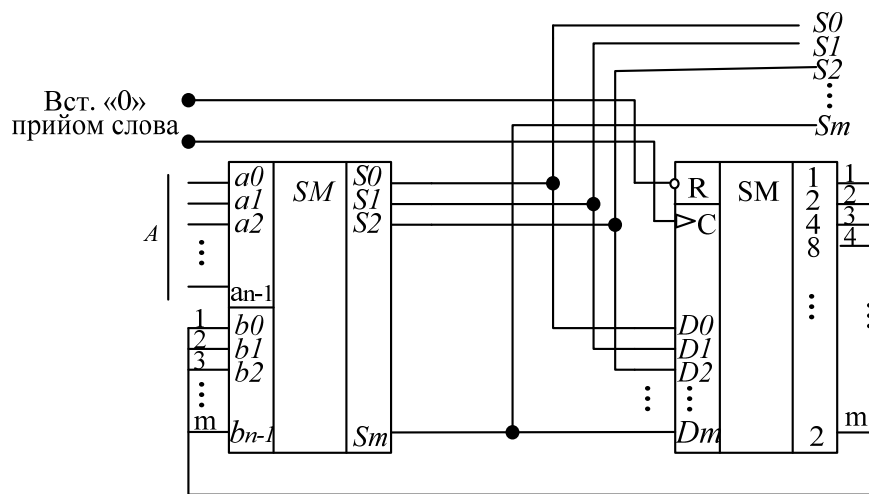


Рисунок 8.6 – Схема накопичувального суматора

Довжина циклу в цьому випадку складає

$$t_s = t_{sm} + t_{pr},$$

де t_{pr} – час встановлення вмісту регістра. Відповідно до особливостей функціонування накопичувальні суматори називають також акумуляторами. У таких суматорах з послідовним переносом (рис. 8.7) доданки мають подаватися на паралельні входи розрядів суматора послідовно в часі, забезпечуючи часовий інтервал між надходженням двох чисел, достатній для формування сигналу переносу в розрядах суматора і для його передачі через лінії затримки (ЛЗ), під'єднані між кожними двома сусідніми розрядами в напрямку від молодших розрядів до старших. Для початкового встановлення всіх розрядів НС в стан 0 використовують окреме коло «Скидання». НС з послідовним переносом мають невисоку швидкодію:

$$t_{заг.} = (t_{лз.} + t_{зд.ср.})(n - 1) + t_n,$$

де $t_{зд.ср.}$ – середня затримка сигналу в ЛЕ формування переносу; $t_{ЛЗ}$, $t_i + t_n$ – час затримки ЛЗ; t_n – затримка перемикавання тригера; t_i – тривалість розрядного сигналу, що надходить на вхід суматора.

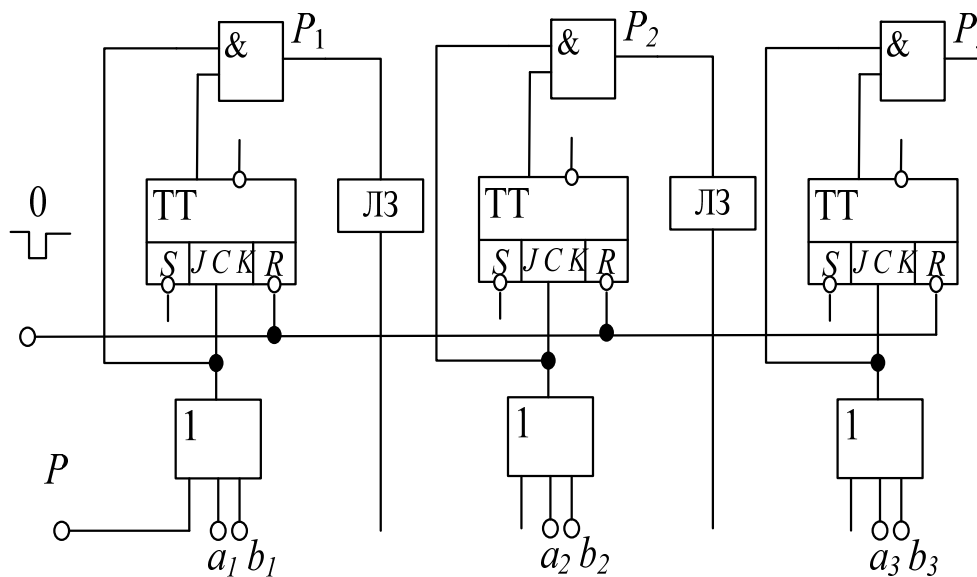


Рисунок 8.7 – Накопичувальний суматор з послідовним переносом

За числа розрядів $n > 4...8$ час додавання виявляється порівняно великим, тому для підвищення швидкодії в паралельних суматорах звичайно використовуються прискорені способи формування сигналів переносу. При цьому часто використовують спосіб з одночасним формуванням переносу для декількох розрядів. Враховуючи, що перенос, сформований n -розрядом, $P_{n+i} = a_n b_n + (a_n \oplus b_n) c_n$, можна отримати значення переносу на вході другого розряду $P_2 = a_1 b_1 + (a_1 \oplus b_1) c_1$. Для третього розряду маємо $P_3 = a_2 b_2 + (a_2 \oplus b_2) c_2$, оскільки $c_2 = p_2$, отримаємо

$$P_3 = a_2 b_2 + (a_2 \oplus b_2) [(a_1 \oplus b_1) c_1 + a_1 b_1].$$

Аналогічно $P_4 = a_3 b_3 + (a_3 \oplus b_3) c_3$. З цих формул видно, що значення переносу для кожного розряду можна визначити за допомогою логічного пристрою, що містить ЛЕ І, АБО і виключне АБО, знаючи значення доданків, і наявність переносу c_1 на вході першого розряду.

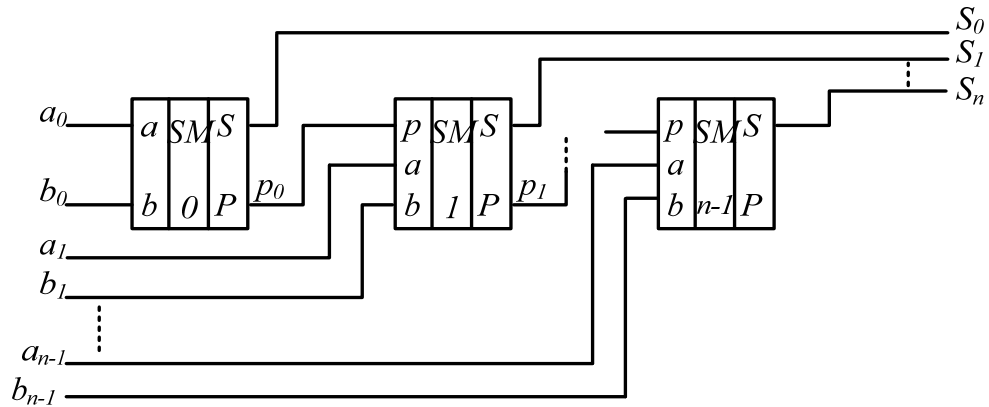
8.3 Паралельний багаторозрядний суматор із послідовним переносом

Паралельні суматори з послідовним переносом (рис. 8.8, а) приймають доданки паралельно. Після цього утворюють попередні розрядні суми, а після появи і розповсюдження переносів суми набувають остаточних значень. Час додавання в цьому випадку

$$t_s \leq t_{ap} + (n-2)t_{pp} + t_{ps},$$

де t_{ap} – час формування переносу в молодшому розряді; t_{pp} – час поширення переносу через розряд; t_{ps} – час отримання суми в старшому розряді після надходження на його вхід переносу від передостаннього розряду.

а)



б)

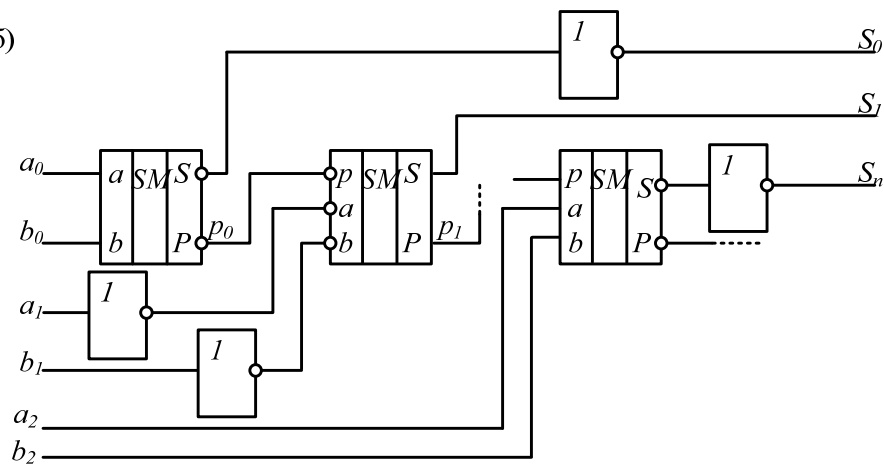


Рисунок 8.8 – Схеми паралельних суматорів з послідовним переносом

Основну частину часу додавання займає проходження переносу через розряди суматора, тому забезпечення мінімальності t_{pp} має велике значення.

8.4 Паралельний багаторозрядний суматор із паралельним переносом

Суматори з паралельним переносом мають максимальну швидкодію. У них відсутні процеси поширення переносів від розряду до розряду. У кожному розряді одночасно отримуються вихідні величини.

Можливість побудови суматора з вказаними властивостями базується на відтворенні функцій суми і переносу, що залежать тільки від значень доданків незалежно від місця розташування розряду в розрядній сітці.

У першому (молодшому) розряді реалізують функції, що залежать лише від доданків тільки цього розряду

$$s_0 = f_0(a_0, b_0); p_0 = \phi_0(a_0, b_0).$$

У другому розряді реалізують функції

$$s_1 = f_1(a_1, b_1, p_0); p_1 = \phi_1(a_1, b_1, p_0),$$

які залежать як від доданків даного розряду, так і від переносів з попереднього розряду. Однак цей перенос є функцією доданків a_0 і b_0 , тому величини s_1 і p_1 можна подати у вигляді

$$s_2 = f_2(a_1, b_1, a_0, b_0); p_2 = \phi_2(a_1, b_1, a_0, b_0).$$

Аналогічно в $(i - 1)$ -розряді маємо

$$s_i = f(a_i, b_i, a_{i-1}, b_{i-1}, \dots, a_0, b_0),$$

$$p_i = \phi(a_i, b_i, a_{i-1}, b_{i-1}, \dots, a_0, b_0);$$

в останньому (старшому) розряді

$$s_{n-1} = f(a_{n-1}, b_{n-1}, a_{n-2}, b_{n-2}, \dots, a_0, b_0).$$

При $a_i b_i = 1$ перенос даного розряду породжується незалежно від значення переносу з попередніх розрядів. Тоді

$$p_0 = g_0 - ph_0;$$

$$p_1 = g_1 - p_0 h_1 = g_1 + (g_0 + ph_0)h_1 = g_1 + g_0 h_1 + ph_0 h_1;$$

$$p_i = g_i - g_{i-1} h_i + g_{i-2} h_{i-1} + \dots + ph_0 h_1 \dots h_i. \quad (8.4)$$

де $g_i = a_i b_i$ – функція генерування (породження) переносу, $h_i = a_i + b_i = \overline{a_i b_i}$ – функція передачі цього переносу.

Перенос для i -го розряду (8.4) виразимо через операцію І-НІ

$$\begin{aligned} P_i &= \overline{\overline{g_i \cdot g_{i-1} \cdot h_{i-1} \cdot g_{i-2} \cdot h_{i-1} \cdot h_i \dots} + ph_0 h_1 \dots h_i} = \\ &= \overline{a_i \cdot b_i \cdot a_{i-1} \cdot b_{i-1} \cdot h_i \cdot a_{i-2} \cdot b_{i-2} \cdot h_{i-1} \cdot h_i \dots ph_0 h_1 \dots h_i}. \end{aligned} \quad (8.5)$$

Виразам (8.4) і (8.5) відповідає схема паралельного суматора з паралельним переносом на рис. 8.9, що є сукупністю не пов'язаних між собою колами передачі переносу розрядних пристроїв.

Тривалість додавання для паралельного переносу не залежить від розрядності доданків (якщо знехтувати затримкою елементів від навантаження, яка збільшується зі зростанням розрядності доданків). Для суматора на рис. 8.9 тривалість додавання складається з затримок формування функцій передачі переносу ($t_{з.ла}$), часу формування функцій передачі переносу ($2t_{з.ла}$) і затримки однорозрядних суматорів t_{ps} тобто

$$t_s = 3t_{з.ла} + t_{ps}.$$

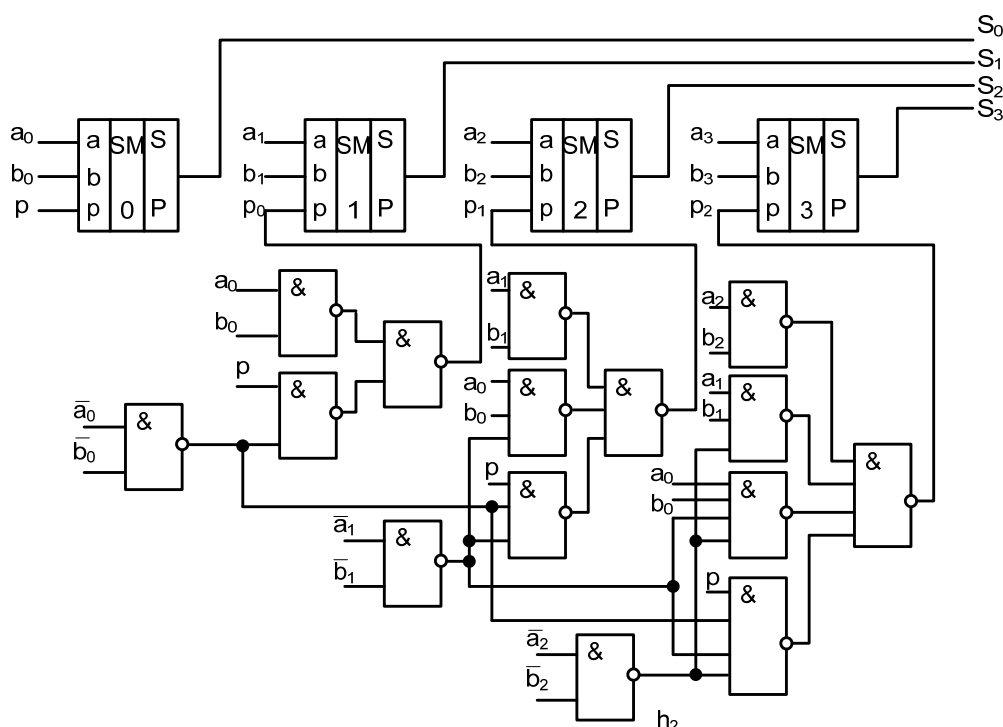


Рисунок 8.9 – Схема паралельного суматора з паралельним переносом

Зауважимо, що для суматорів з паралельним переносом можна використовувати спрощені однорозрядні суматори, що мають лише один вихід суми.

Залежно від типу однорозрядного суматора час додавання при паралельному переносі становить (5...6) t_3 .

Зі зростанням числа розрядів реалізація паралельного переносу утруднюється тим, що виникає потреба в ЛЕ з великими числом входів і навантажувальною здатністю. Для формування переносів у старшому розряді суматора потрібні елементи з числом входів, що дорівнює розрядності суматора. Найбільшу навантажувальну здатність мають мати пристрої формування функцій h_i . Отримана в даному розряді функція h_i використову-

Вже при побудові 8-розрядного суматора потрібні елементи з числом входів 8 і коефіцієнтом розгалуження 16, що може перевищувати можливості базових ЛЕ. Тому суматори з паралельним переносом звичайно реалізують лише для невеликого числа розрядів як складові частини суматорів з груповою структурою.

Суматори з груповою структурою, що мають n розрядів, складаються з груп по l розрядів. У групах і між ними можливі різні типи переносів. Пристрої, що отримують сигнали переносу (розглядалися в підрозділі 8.2), називають формувачами наскрізного переносу і використовують в паралельних суматорах з наскрізним переносом (рис. 8.10) і в суматорах з ланцюговим переносом (рис. 8.11).



168

носу використовують блоки переносу (БП), які аналізують доданки в межах групи. Зрозуміло, що при цьому розмірність схем, які генерують перенесення, зменшується, але при цьому з'являється послідовна передача переносів між групами.

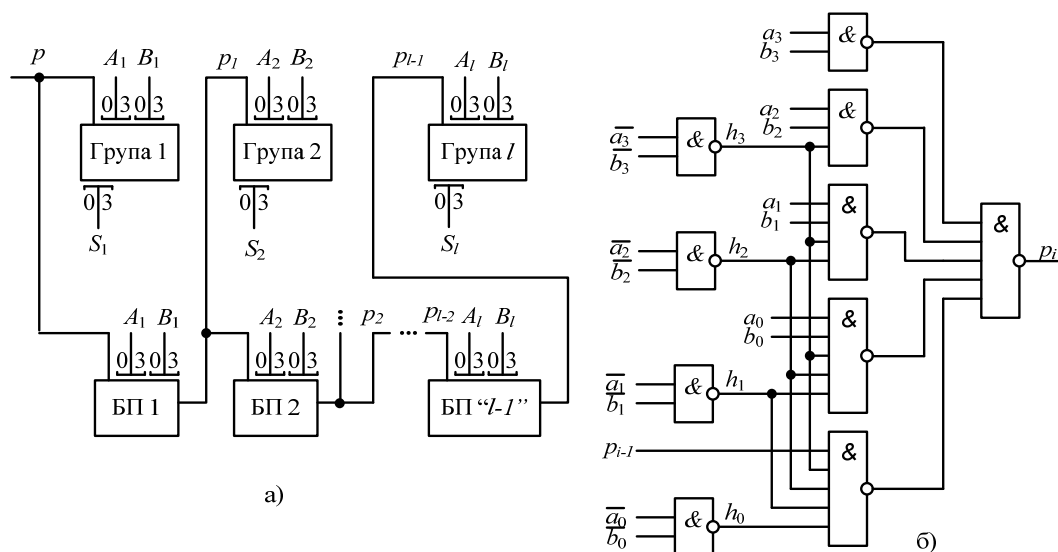


Рисунок 8.11 – Схема суматора з ланцюговим переносом а);
схема ланцюгового переносу для 4-х розрядів б)

Перенос з групи визначається через функції генерації і передачі переносів розрядів таким чином

$$p = g_{m-1} + g_{m-2}h_{m-1} + \dots + g_0h_1h_2\dots h_{m-1} + ph_0h_1\dots h_{m-1},$$

$$g_i = a_ib_i \quad (i = 0, 1, \dots, m-1),$$

де p – вхідний перенос групи.

Блок переносу для $m = 4$ показано на рис. 8.11, б.

Тривалість додавання в цьому випадку оцінюється як

$$t_s = t_h + (l-1) t_{\text{он}} + t_{sm},$$

де t_h , $t_{\text{он}}$ і t_{sm} – затримки генерування функцій передачі переносу h у групі, в блоці переносу і додавання у групі відповідно.

Для 32-розрядного суматора з 8-ма групами і з паралельним переносом у групах час додавання оцінюється приблизно в 20–25 затримок ЛЕ. В них міжгрупові переноси здійснюються паралельно.

Побудова суматорів з груповою структурою забезпечується функціональним складом сучасних серій ІМС. Багато з серій ІМС містять пристрої скороченого перенесення осу, у яких реалізовані функції генерування і пе-

редачі переносу та отримуються сигнали паралельних переносів, що дозволяють застосувати скорочені переноси на різних рівнях суматора (у групах та між ними).

8.6 Суматор з паралельно-паралельним переносом

Суматор з паралельно-паралельним переносом має паралельний перенос як в групах, так і між ними, що забезпечує мінімальний час додавання. У такому суматорі двічі (у групах і між ними) повторюється принципово однакова організація переносів, причому для суматора в цілому групи відіграють ту ж роль, що і однорозрядні суматори для групи. Для груп виробляються функції генерування і передачі переносів, причому функція генерування

$$G = g_{m-1} + g_{m-2}h_{m-1} + g_{m-3}h_{m-2}h_{m-1} + \dots + g_0h_1\dots h_{m-1}$$

приймає одиничне значення, якщо з даної групи з'являється перенос незалежно від наявності або відсутності вхідного переносу.

Функція передачі переносів

$$H = h_0h_1\dots h_{m-1}$$

приймає одиничне значення, якщо вихідний перенос з групи з'являється тільки при наявності вхідного. Зауважимо, що функції передачі переносу h і H називають також функціями прозорості або транзиту. У даному випадку термін «функція прозорості» добре пояснює співвідношення для H : група прозора при прозорості всіх її розрядів.

Суматор з 4-ма чотирирозрядними групами (рис. 8.12) виробляє за допомогою груп значення суми в межах групи шляхом додавання слів $A_1 = a_3a_2a_1a_0$, $B_1 = b_3b_2b_1b_0$, $A_2 = a_7a_6a_5a_4$, $B_2 = b_7b_6b_5b_4$ і т. д. Сигнали переносів для груп формуються за допомогою блоків G , H і P , що реалізують записані вище вирази.

Тривалість додавання в структурі суматора з паралельно-паралельним переносом можна оцінити таким чином:

- 1) після подачі доданків у групах формують функції прозорості розрядів, отже, стають відомими всі аргументи для отримання функцій G і H ;
- 2) за відомими значеннями G , H та зовнішнього переносу p отримують переноси в групи, після чого в групах формують вихідні змінні $S_m, S_{m-1}, \dots, S_0$ (для цього, як зазначалося при розгляді суматорів з паралельним переносом, потрібно 5...6 елементарних затримок).

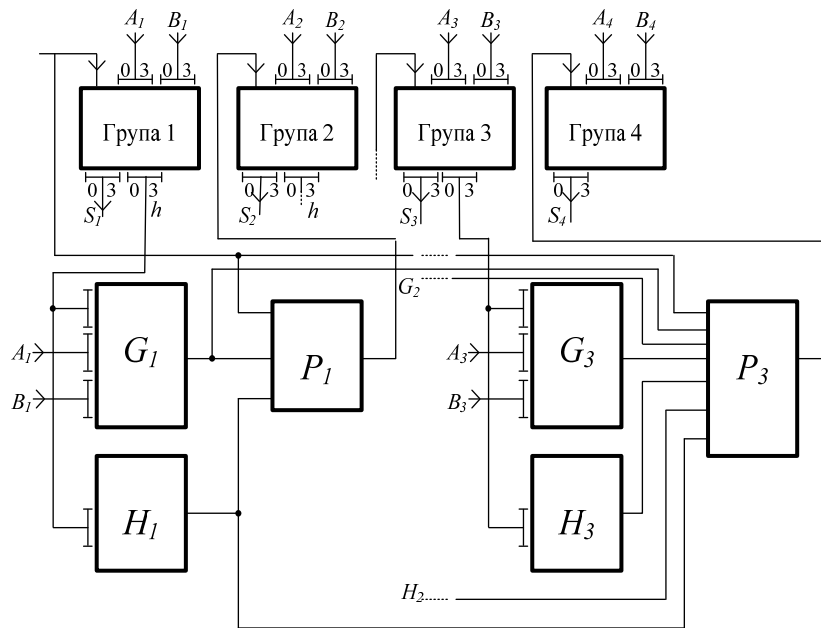


Рисунок 8.12 – Схема суматора з паралельно-паралельним переносом

Сигнал переносу для групи визначають відповідно до виразу

$$P_j = G_j + G_{j-1}H_j + G_{j-2}H_{j-1}H_j + \dots + G_1H_2\dots H_j + pH_1\dots H_j, \\ j = 1, l-1.$$

Зважаючи на тривалості вказаних етапів процесу додавання, можна оцінити час додавання в суматорах з паралельно-паралельним переносом значенням $10\dots 11 t_3$. Це час досяжний вже й для багаторозрядних суматорів, тому є практичною оцінкою для суматорів найбільшої швидкодії. Якщо число розрядів дуже велике, то труднощі побудови (необхідність елементів з великим числом входів і коефіцієнтом розгалуження) з'являться і в суматорі групової структури з паралельно-паралельним переносом.

8.7 Послідовний n -розрядний суматор

Послідовні суматори перетворюють послідовні коди операндів в послідовний код суми цих доданків. Додавання починається з молодшого розряду і виконується порозрядно послідовно за стільки тактів, скільки розрядів міститься в операндах. До складу багаторозрядного суматора послідовної дії, крім комбінаційного одnorозрядного суматора, входять три регістри зсуву для двох операндів A і B і для результату S , тригер переносу TrP і пристрої управління введенням і виведенням чисел (рис. 8.13).

Два числа, що їх додають, завантажують в регістри A і B у послідовних або паралельних входах. Тригер переносу спочатку встановлений в 0, отже $C = 0$, а сигнал суми записується в регістр зсуву S . Кожен сигнал переносу,

що виробляється суматором, з'являється на виході P і запам'ятовується тригером переносу. Тактовий сигнал вводить біт суми в регістр S і одночасно зсуває на один розряд регістри A і B . Крім того, він вносить значення переносу P в TrP , в результаті чого на вході C завжди діє значення розряду переносу, що мало місце при додаванні двох попередніх розрядів. Так додають всі розряди двійкових чисел, і результат записують в регістр S . Цей результат можна зчитувати на паралельних або послідовних виходах. Основна перевага такого суматора – невелика кількість обладнання; до недоліків віднесемо невисоку швидкодію, тому час додавання двох n -розрядних чисел $t_{\Sigma} = nT$, де T – період слідування тактових сигналів.

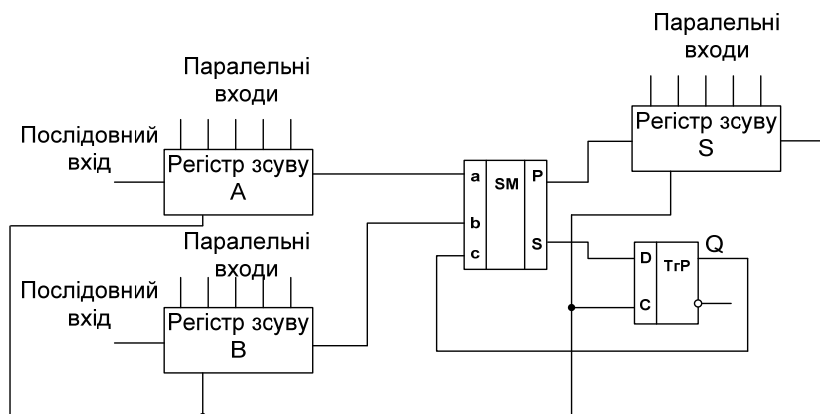


Рисунок 8.13 – Послідовний n -розрядний суматор

8.8 Десяткові та двійково-десяткові суматори

Багаторозрядні десяткові суматори будують з використанням однорозрядних десяткових суматорів. Останні виконують операцію додавання десяткових цифр $a_i b_i$ і переносів P_{i+1} , формують на виходах десяткову цифру суми S_i і переносу P_{i+1} для передачі в наступний десятковий розряд.

При використанні десяткової системи числення цифри десяткових розрядів звичайно подають у двійково-десятковому коді 8421 (рис. 8.14). Двійкові подання десяткових цифр додають за правилами додавання двійкових чисел. Якщо отримана сума містить десять або більше одиниць, то формують одиницю переносу для передачі в наступний десятковий розряд, а з суми віднімають десять одиниць. Наявність в отриманій сумі десяти або більше одиниць виявляється за такими ознаками: поява переносу з розряду P_8 при додаванні цифр, наявність 1 в розрядах S_8 і S_4 або S_8 і S_2 в отриманій сумі. Необхідну при цьому корекцію суми з одночасним відніманням з неї десяти одиниць зручно робити додаванням шести одиниць (0110) і відніманням 16 одиниць шляхом відкидання переносу, що виникає у розряді P_8 .

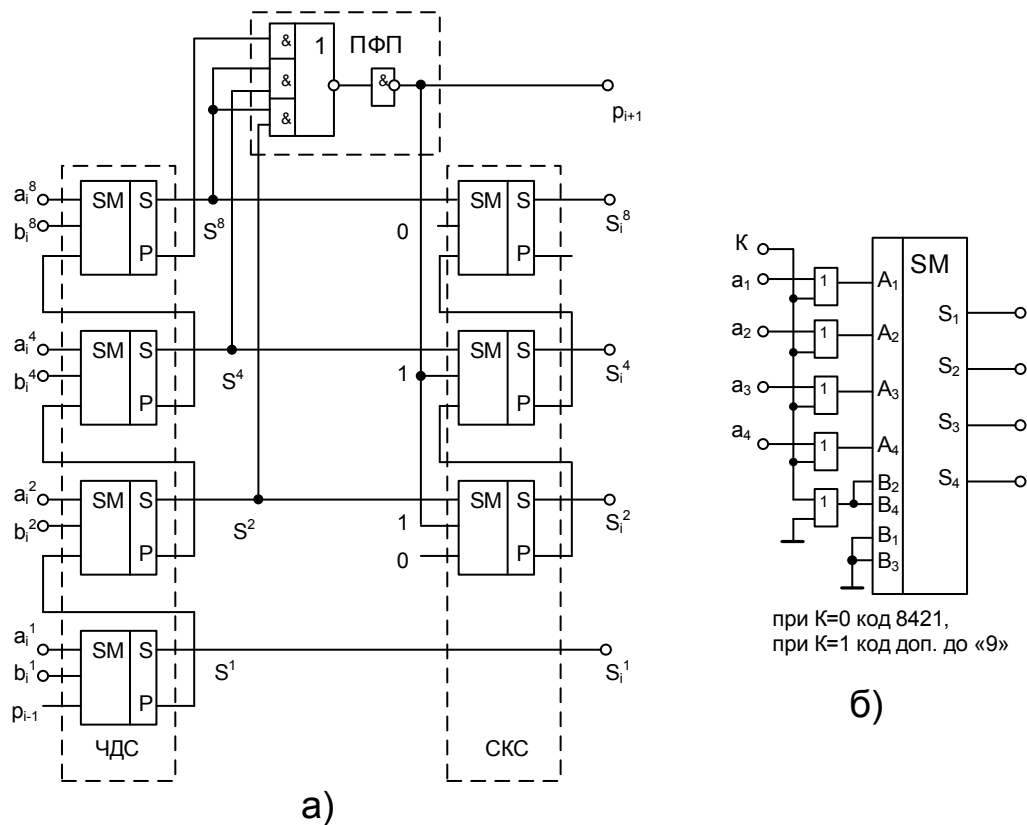


Рисунок 8.14 – Схеми однорозрядного десяткового суматора а); та перетворювача коду попередньої суми на доповнення до 9 б)

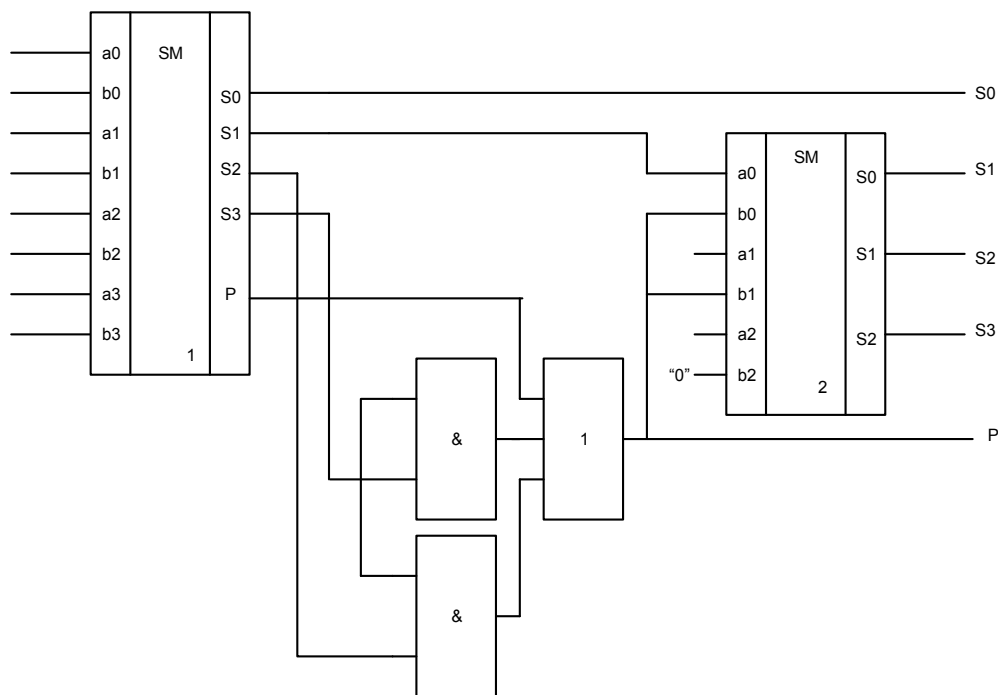
Відповідно до правил додавання в десятковій системі числення, на рис. 8.14, а) наведена схема однорозрядного десяткового суматора, що містить в собі чотирирозрядний двійковий суматор (ЧДС), пристрій формування переносу (ПФП) P_{i+1} в наступний розряд і пристрій корекції суми (ПКС). Останній являє собою трирозрядний суматор, в якому, при $P_{i+1} = 1$, виробляється сигнал додавання одиниці в розряди нескорегованої суми з ваговими коефіцієнтами 2 і 4.

Двійково-десяткові суматори виконують дії над десятковими числами, розряди яких закодовані двійковими тетрадами. Звичайний спосіб побудови двійково-десятькового суматора передбачає первинне додавання тетрад звичайним двійковим суматором та наступну корекцію результату. Корекція необхідна, оскільки при додаванні тетрад результат може перевищувати 9, тоді як тетрада має містити двійкові числа від 0 до 9.

Якщо результат додавання тетрад менше 9, то корекція не потрібна.

Якщо при додаванні тетрад та переносу з сусідньої тетради результат лежить в межах 10...15, то розрядна сітка не переповнюється, переносу не виникає, однак потрібен перенос у сусідню тетраду з одночасним зменшенням отриманого числа на 10. Віднімання 10 можна замінити додаванням з доповняльним кодом 10, який має двійкове подання 0110. Таким чином, у даному випадку до результату потрібно додати коректувальну поправку 6 та забезпечити сигнал переносу в старшу тетраду.

Тетрада двійково-десятькового суматора (рис. 8.15) містить суматори 1 і 2 та послідовність ЛЕ для формування коректувальної поправки при переповненні тетради або отриманні в ній чисел $10\dots 15$. У останньому випадку формується і сигнал переносу. Перший етап додавання виконується суматором 1 , на який надходять тетради доданків. Корекція результатів першого етапу відбувається в суматорі 2 , на одні входи якого подаються початкові значення сум, а на інші – (за необхідності) подають корегувальну поправку 0110 (в молодший розряд завжди надходить при цьому нуль, тому він передається на вихід безпосередньо). Перенос з першого суматора проходить через елемент АБО та встановлює на входах b_0 , b_1 другого суматора одиниці, що і відповідає двійковому зображенню числа 6 , яке дорівнює 0110 .



Якщо початкове значення суми лежить в межах 10...15, то необхідно штучно створити на виході тетради перенос. На наявність чисел у вказаному діапазоні вказує одиничне значення логічної функції

$$F = s_1 s_3 + s_2 s_4.$$

8.9 Суматори-віднімачі

Віднімання в комп'ютерних арифметичних пристроях найчастіше виконують шляхом додавання двійкових чисел в оберненому або доповняльному кодах (нагадаємо, доповняльний код від'ємного числа утворюють з оберненого коду шляхом додавання 1 до молодшого розряду). Віднімання ж двійково-десяткових чисел виконують в доповняльному (до 9 або до 10) коді.

Віднімання двійкових чисел виконують так. Від'ємник, також його знаковий розряд, подають в оберненому коді, а зменшуване – в прямому двійковому коді. Тоді різницю можна отримати арифметичним додаванням зменшуваного та від'ємника разом з їх знаковими розрядами. Якщо в знаковому розряді при цьому утворюється перенос, то цю 1, яку ще називають циклічним переносом, додають до молодшого розряду суми. Знак результату визначається отриманим значенням знакового розряду ZS . Якщо результат операції від'ємний ($ZS = 1$), то він буде поданий в оберненому коді, якщо результат додатний ($ZS = 0$), то його можна правильно читати в прямому двійковому коді.

Недоліком використання оберненого коду є утворення циклічного переносу, що призводить до можливого повторення операції, яке значно збільшує час виконання віднімання. Тому краще використовувати доповняльний код одного з операндів. При цьому відпадає необхідність врахування циклічного переносу, оскільки перенос зі знакового розряду числа не враховується. Таким чином віднімання замінюють додаванням з переведенням від'ємника у доповняльний код. Якщо знаковий розряд результату $ZS = 1$, то отримане число від'ємне та подано в доповняльному коді. Якщо $ZS = 0$, то результат додатний та поданий в прямому коді.

Додавання та віднімання двійкових чисел з застосуванням доповняльного коду виконується простіше та швидше, хоча перетворення чисел в доповняльний код дещо складніше, порівняно з перетворенням в обернений код.

На рис. 8.16 показана структурна схема суматора-віднімача з використанням доповняльного коду.

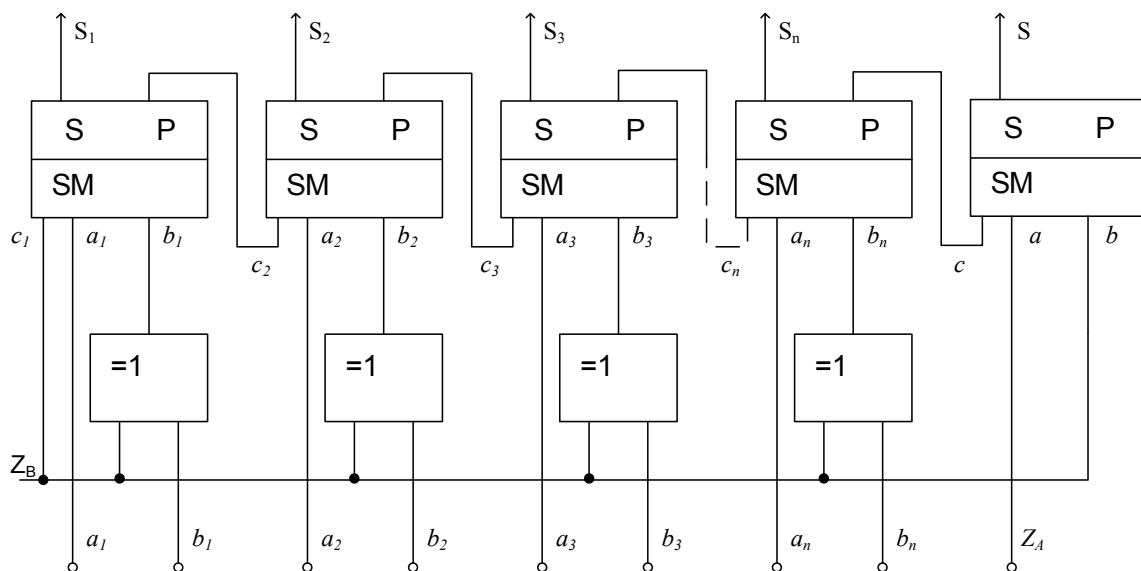


Рисунок 8.16 – Структурна схема суматора-віднімача

8.10 Контрольні запитання і завдання

1. Призначення суматорів. Приклад додавання однорозрядних двійкових чисел.
2. Поясніть різницю в застосуванні півсуматора та повного однорозрядного суматора.
3. Побудуйте схему додавання двох трирозрядних двійкових чисел.
4. Поясніть, як виконується віднімання двійкових чисел.
5. В чому особливість функціонування двійково-десяткових суматорів?
6. В якому випадку відпадає необхідність у врахуванні циклічного переносу для суматорів-віднімачів ?

9 СХЕМОТЕХНІЧНІ КОМП'ЮТЕРНІ ПРИСТРОЇ І ВУЗЛИ ВИСОКОЇ ВИРОБНИЧОЇ СЕРІЙНОСТІ

Перелік скорочень і умовних позначень до розділу 9

АЗП – асоціативний запам'ятовувальний пристрій; АЛП – арифметико-логічний пристрій; АШ (ША) – адресна шина (шина адрес); БУП – блок управління пам'яттю; ЗМ – запам'ятовувальний масив; КП – комірка пам'яті; ЛПС – лічильник показчика стека; МБ – блок множення; МДБ – блок множення і додавання; РШ0(1) – розрядна шина нулів (одиниць); ТП – технічна пам'ять.

Сучасний рівень розвитку інтегральної технології дозволяє виготовляти вироби, складність яких оцінюється сотнями тисяч і мільйонами ЛЕ. Це свідчить про принципову можливість виготовлення будь-якого функціонального комп'ютерного пристрою чи вузла у вигляді ЦІМС. Однак зрозуміло, що виробництво та, особливо, підготовка до нього, унікальних за своїми функціями виробів без сподівань на їх хоча б невелику серійність – надзвичайно дорога справа. Очевидно, що доцільніше спершу забезпечити виробництво тих схемотехнічних компонент, на які може бути гарантований високий попит. З числа виробів комп'ютерної схемотехніки досить високий та стабільний в часі попит існує на пристрої для обробки і зберігання інформації (інакше, на арифметико-логічні пристрої і пристрої технічної пам'яті). Тому далі розглянемо схемотехнічні аспекти побудови структур деяких пристроїв як інформаційно-обробного, так і інформаційно-зберігаючого призначення.

9.1 Матричні помножувачі

Множення двійкових чисел в комп'ютерних системах досить часто реалізують як послідовності мікрооперацій зсуву та додавання, які чергуються в певному порядку. Однак час виконання такого множення досить великий (особливо порівняно з часом виконання інших типових комп'ютерних операцій). Незважаючи на те, що в комп'ютерній арифметиці відомий величезний доробок певних алгоритмічно-логічних способів прискорення множення, час множення у підсумку не вдається зробити хоча б сумірним з часом виконання інших операцій або незалежним від довжини операндів. Тільки застосуванням апаратних способів комп'ютерного прискорення можна досягти результату з такими якісними характеристиками. Суть апаратного прискорення множення полягає в одночасному формуванні всіх n^2 (або $n^2/2$, $n^2/4$, $n^2/8, \dots$) розрядних добутків двох операндів. Це можна зробити за допомогою квадратної або прямокутної матриці з ЛЕ типу І, оскільки таблиця істинності логічного І однорозрядних двійкових цифр

повністю відповідає таблиці їх множення за правилами двійкової арифметики. Після формування в такий спосіб всіх розрядних добутків потрібно виконати апаратним способом додавання розрядних добутків з однаковою вагою і врахувати ефект розповсюдження переносів у напрямку від молодших розрядів до старших. Такі матричні пристрої множення мають високу швидкодію і виготовляються у вигляді ЦІМС високого рівня інтеграції, тобто належать до типових функціональних пристроїв широкого застосування.

Структура матричних помножувачів в тій їх частині, що виконує додавання рівноважних розрядних добутків, зумовлюється структурою математичних виразів (суперпозицій), які покроково описують повну операцію множення. Оскільки і множення, і додавання є комутативними операціями, то взагалі можна використовувати довільний порядок як формування часткових добутків (тобто, зі старших чи молодших розрядів), так і їх додавання (тобто, з якого часткового добутку починати). При цьому принципово важливим є врахування переносів із i -го розряду суми розрядних добутків в сусідній старший $(i+1)$ -й розряд.

Нехай задано два цілих двійкових операнди без знаків $A = a_{m-1}a_{m-2} \dots a_0$ та $B = b_{n-1}b_{n-2} \dots b_0$. Тоді множення цих чисел з використанням матричного формування розрядних добутків за вищевикладеним алгоритмом можна пояснити діаграмою на рис. 9.1.

Операнд A (множник)	$a_3 \ a_2 \ a_1 \ a_0$							
Операнд B (множене)	$b_3 \ b_2 \ b_1 \ b_0$							
Частковий добуток a_0B				a_0b_3	a_0b_2	a_0b_1	a_0b_0	
Частковий добуток a_1B			a_1b_3	a_1b_2	a_1b_1	a_1b_0		
Частковий добуток a_2B			a_2b_3	a_2b_2	a_2b_1	a_2b_0		
Частковий добуток a_3B	a_3b_3	a_3b_2	a_3b_1	a_3b_0				
Суми часткових добутків	p_7	p_6	p_5	p_4	p_3	p_2	p_1	p_0

Рисунок 9.1 – Діаграма матричного множення двох двійкових чисел, де $m=n=4$

Тут цифра суми часткових добутків p_7 утворюється як результат додавання переносів, самі ж розрядні добутки подані парами двійкових цифр, що формують саме ці добутки.

Щодо конструкції: блоки матричних пристроїв множення відомі як просто помножувальні (МБ) або помножувально-додавальні (МДБ). Блок для перемноження чотирирозрядних операндів (рис. 9.2, а) містить 16 ЛЕ І та 12 однорозрядних суматорів. Час додавання всіх розрядних добутків, з

урахуванням переносів, і буде визначати час множення. Для МБ з однаковою довжиною операндів кількість ЛЕ та однорозрядних суматорів визначають відповідно як n^2 та $n(n-1)$ (n – розрядність операндів). МДБ реалізує операцію $P = A * B + C$. Для операндів $C = c_{m-1}c_{m-2} \dots c_0$ та $D = d_{n-1}d_{n-2} \dots d_0$ в структурі МДБ передбачені додаткові входи (рис. 9.2, б). Умовне позначення МДБ показано на рис. 9.2, в (для прикладу тут $m = 4$ та $n = 2$). В МБ та МДБ (рис. 9.2) максимальний час множення визначається найдовшим шляхом передачі сигналу (права діагональ плюс нижній ряд матриці) однорозрядних суматорів.

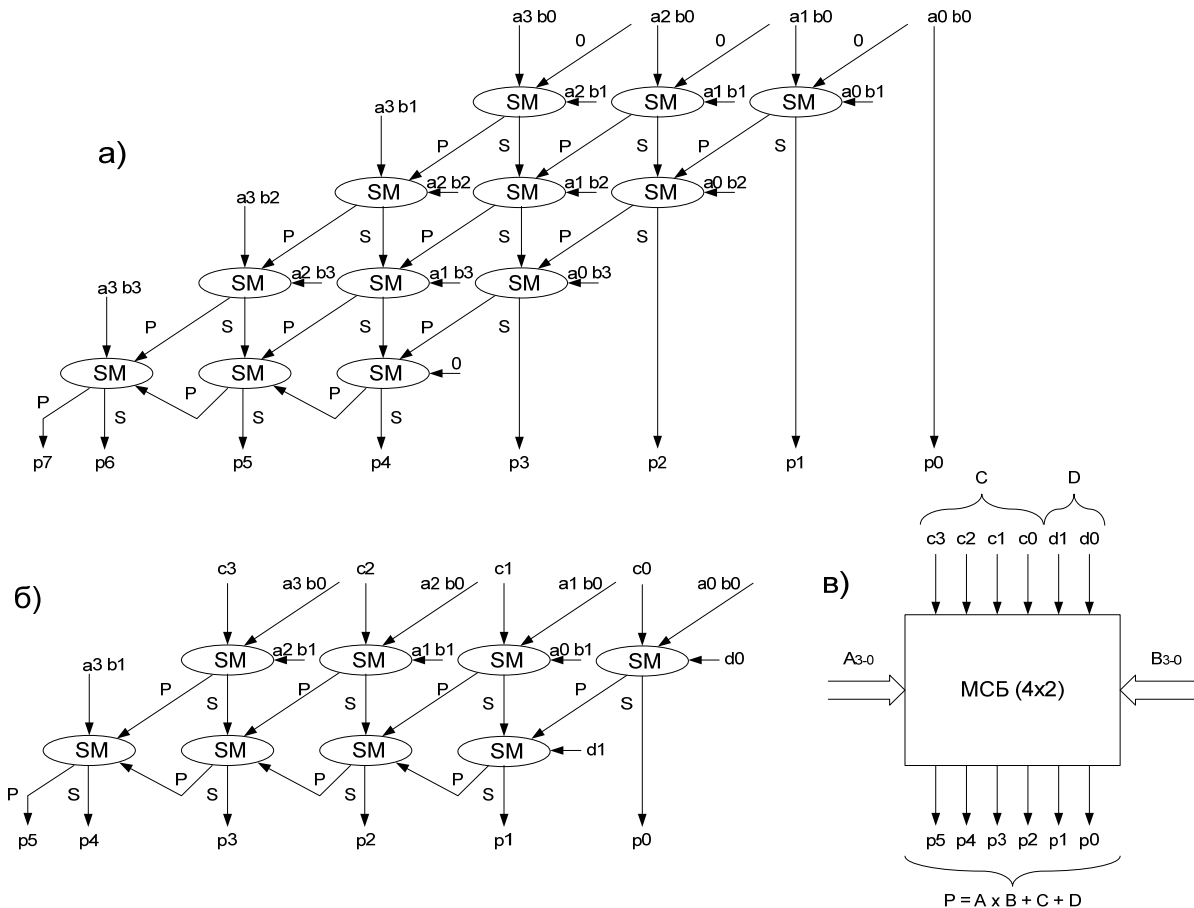


Рисунок 9.2 – Структурні схеми блока помножувального матричного пристрою а); блока помножувально-додавального пристрою б); його умовне позначення в)

Для множення двох n -розрядних чисел це складає

$$t_M = t_K + 2(n - 1)t_{sm},$$

де t_K – затримка ЛЕ типу I; t_{sm} – час додавання в однорозрядному суматорі.

Нижній ряд однорозрядних суматорів в матриці є, насправді, паралельним суматором з послідовним переносом. Якщо замінити його суматором з паралельним переносом, то час виконання операції зменшиться до

$$t_M = t_K + (n - 1)t_{sm} + t_{cn},$$

де t_{cn} – час додавання в суматорі з паралельним переносом. Матричні помножувачі більшої розмірності будують із модулів меншої розмірності, для яких можна використовувати як МБ, так і МДБ. У першому випадку необхідні додаткові модулі спеціальних суматорів порозрядної структури, які звичайно наявні в серіях ЦІМС. У другому випадку додаткові модулі не потрібні.

9.2 Арифметико-логічні пристрої

Більшість серій мікросхем мають у своєму складі ЦІМС, які названі арифметико-логічними пристроями (АЛП – англ. термін arithmetic logic unit – ALU). Однак такі АЛП не використовують як основу центральні комп'ютерні процесори. Основна відмінність процесорів від АЛП полягає в тому, що їх структура та інші ресурси націлені на забезпечення найкращої, в певному сенсі, реалізації концепції програмного управління процесора. Ця концепція звичайно зафіксована як система досить складних команд універсального засобу обробки інформації, яким насправді і є процесор. АЛП ж, про які йде мова тут, зорієнтовані на найкраще виконання ряду не дуже складних логічних та арифметичних операцій (які за змістом ближчі до мікрооперацій) над словами. Прикладом такого АЛП можуть бути мікросхеми К155ІПЗ або 564ІПЗ. Мікросхема К155ІПЗ – швидкодійний АЛП, який виконує 16 логічних і 16 арифметичних операцій (рис. 9.3, а). $\overline{A_0} \dots \overline{A_3}$, $\overline{B_0} \dots \overline{B_3}$ – інформаційні входи чотирирозрядних операндів A та B , над якими виконуються логічні й арифметичні операції.

Вибір операції (одну з 16 арифметичних або логічних) визначається комбінацією сигналів на входах $S0 \dots S3$. При $M = 1$ мікросхеми виконують логічні операції, а при $M = 0$ – арифметичні. Мікросхема може працювати як в позитивній (на входи A , B та C_n подають істинні значення розрядів операндів та переносу, а з виходів 4 знімають також істинні значення результатів операцій), так і в негативній логіці (на входи A , B та $C_n + 4$ подають інверсні значення інформації, а з виходів $\overline{F_0} \dots \overline{F_3}$, $\overline{P}$, $\overline{G}$, $C_n + 4$ знімають також інверсні значення результатів операції). Вхід C_n використовують тоді, коли виконують операції з операндами, розрядність яких більша 4. У випадку прискореного переносу по групі АЛП використовуються входи $\overline{G}$, – утворення переносу, $\overline{P}$ – поширення переносу, $C_n + 4$ – формування переносу при об'єднанні декількох АЛП за допомогою мікросхеми прискореного переносу К155ІП4 (рис. 9.3, в). АЛП можна використовувати також для переведення чисел із доповняльного (оберненого) коду у прямий та навпаки.

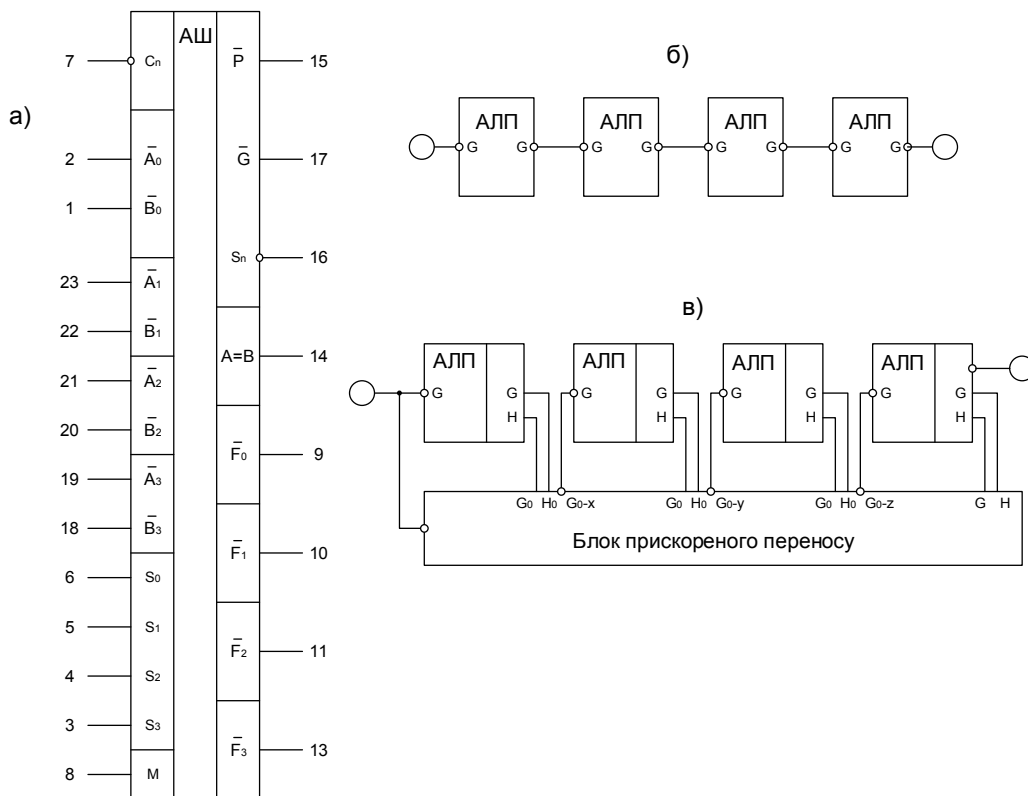


Рисунок 9.3 – Умовне позначення АЛП К155ІПЗ а),
схема з'єднання АЛП з наскрізним переносом б),
з'єднання АЛП з прискореним переносом в)

На виходах F_i формується вихідне слово, вихід $\bar{C}_n$ дає вихідний перенос, який можна використовувати як вхідний для наступного АЛП, виходи G та H формують функції генерації та передачі переносу, які використовуються для утворення переносу в швидкодійних суматорах.

Вихід K – це вихід компаратора. Вихід компаратора виконується за схемою з відкритим колектором, і тому допускає реалізацію монтажною логіки шляхом паралельного з'єднання аналогічних виходів декількох АЛП. Вихід компаратора формує функцію $F_{A-B} = \bar{F}_0 \bar{F}_1 \bar{F}_2 \bar{F}_3$.

При цьому АЛП працює в режимі віднімання $A-B$ (яке виконується як додавання A з інверсним словом $\bar{B}$). Якщо $A = B$, то в усіх розрядах вихідного слова будуть нулі, а всі інверсні значення $\bar{F}_i$ – одиничні, що забезпечує одиничне значення вихідної функції. В усіх інших випадках серед значень F_i знайдеться хоча б одна одиниця, тобто серед функцій буде хоча б один нуль і буде отримано $F_{A-B} = 0$. Паралельне об'єднання виходів K декількох АЛП відповідає реалізації логічної операції І монтажним способом, що дозволяє проводити порівняння на рівність багаторозрядних слів, які обробляються декількома АЛП (рис. 9.4).

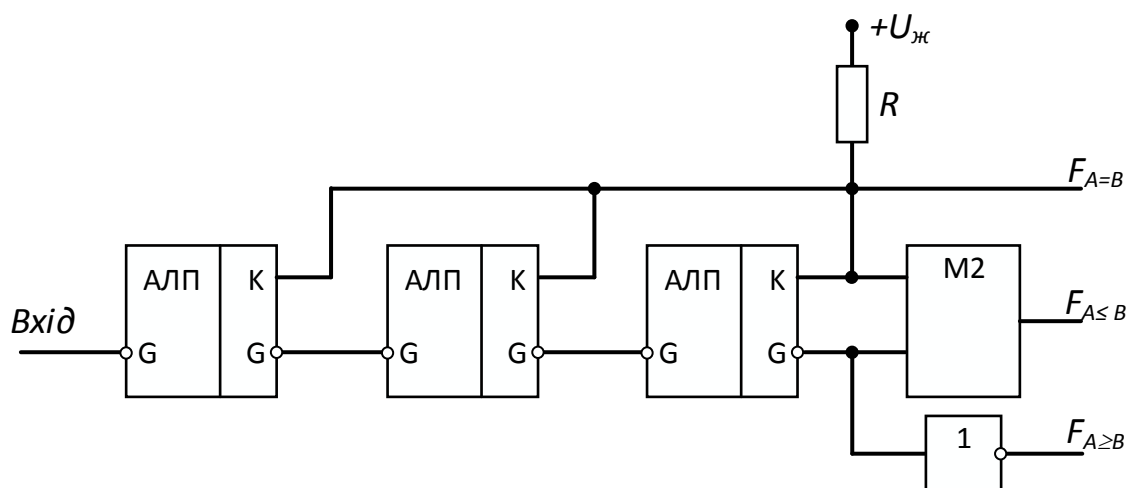


Рисунок 9.4 – Реалізація операції порівняння з використанням АЛП

Комбінуючи сигнал F_{A-B} з сигналом переносу на виході, можна одночасно з ознаками рівності отримати і ознаки нерівності $F_{A \geq B}$ та $F_{A \leq B}$ (рис. 9.4). Якщо $A < B$, то $F_{A-B} = 0$ і при виконанні віднімання виникає перенос із старшого розряду, тому $F_{A \leq B} = 1$. Якщо $A > B$, то $F_{A-B} = 0$ і перенос відсутній, виходячи з цього, $F_{A \leq B} = 0$. Одночасно з цим, як видно з рис. 9.4, $F_{A \geq B} = 1$.

Логічні операції виконуються незалежно в кожному розряді. Арифметичні операції виконуються з урахуванням переносів і позичок. Обидва типи операцій можуть зустрічатися одночасно. Наприклад, запис $(A + \bar{B}) + AB$ означає, що на початку порозрядно виконуються операції інверсії ($\bar{B}$), логічного додавання ($A + \bar{B}$) та логічного множення (AB), а потім отримані вказаним чином два чотирирозрядних числа додаються арифметично (з урахуванням переносів). Повний перелік операцій, які виконує АЛП К155ІПЗ, наведено в табл. 9.1.

Для обробки операндів більшої розрядності АЛП об'єднують послідовно. У цьому випадку основний вплив на швидкодію пристрою має час поширення переносу, який може проходити від молодшого розряду по усій розрядній сітці. В пристроях на основі АЛП намагаються забезпечити мінімальний час затримки переносу (в одну–дві елементарні затримки), але при додаванні слів з великою розрядністю час додавання може стати неприйнятним. Тоді разом з АЛП застосовують спеціальну мікросхему, яка називають блоком прискореного переносу, в якому перенос утворюють за допомогою функцій генерації G та передачі переносу H , які було розглянуто при описуванні паралельних суматорів.

Один блок прискореного переносу створює прискорені переноси для декількох АЛП (найчастіше чотирьох), а також функції G та H більш високого рівня, які дозволяють, за необхідності, організувати і паралельний перенос на другому рівні. Схеми спільної роботи АЛП при їх послідовному

ввімкненні та застосуванні блоків прискореного переносу показані на рис. 9.3, б, в.

Таблиця 9.1 – Перелік операцій, які виконує АЛП К155ИПЗ

Сигнали вибору функції				Логічні функції ($M = 1$)	Арифметико-логічні функції при $C_0 = 0$ ($M = 0$)
S_3	S_2	S_1	S_0		
0	0	0	0	$\bar{A}$	A
0	0	0	1	$\overline{A+B}$	$A+B$
0	0	1	0	$\overline{AB}$	$A+\bar{B}$
0	0	1	1	0	$-I$
0	1	0	0	$\overline{AB}$	$A+\bar{A}\bar{B}$
0	1	0	1	$\bar{B}$	$(A+B)+\bar{A}\bar{B}$
0	1	1	0	$A\oplus B$	$A-B-I$
0	1	1	1	$A\bar{B}$	$\bar{A}\bar{B}-1$
1	0	0	0	$\overline{A+B}$	$A+AB$
1	0	0	1	$A\oplus B$	$A+B$
1	0	1	0	B	$(A+\bar{B})+AB$
1	0	1	1	AB	$AB-I$
1	1	0	0	I	$A+A$
1	1	0	1	$A+\bar{B}$	$(A+B)+A$
1	1	1	0	$A+B$	$(A+\bar{B})+A$
1	1	1	1	A	$A-I$

Слід зауважити, що окрім вищерозглянутих функціональних пристроїв комбінованого типу широке застосування в цифрових комп'ютерних структурах знаходять інші високосерійні ЦІМС, наприклад, вказівники старшої одиниці (старших розрядів), вузли контролю тощо, які розглядаються в спеціалізованій літературі з проектування комп'ютерних засобів.

9.3 Пристрої технічної пам'яті

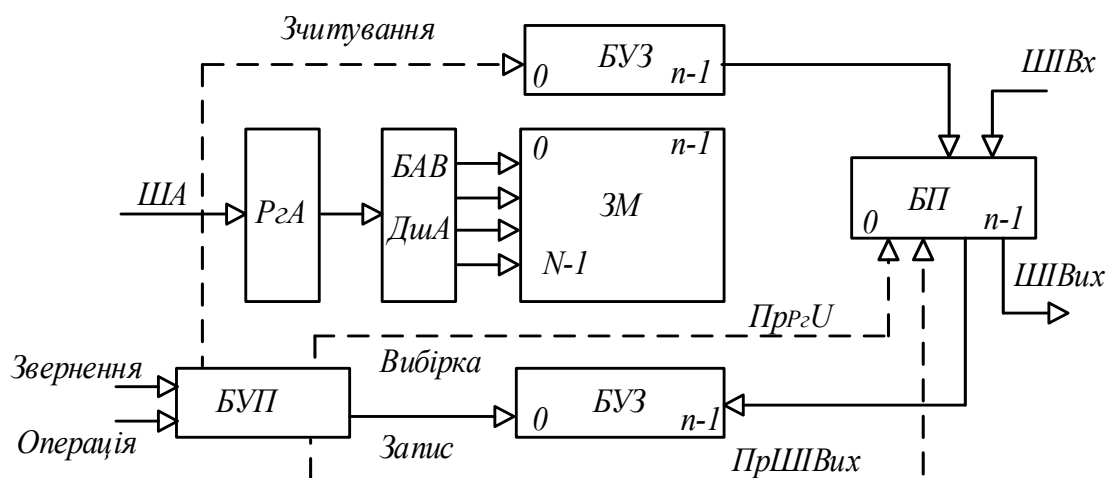
Пристрої технічної пам'яті (запам'ятовувальні пристрої) – комплекси технічних засобів, що реалізують функцію пам'яті. Взагалі функції пам'яті полягають у виконанні декількох операцій, а саме: розміщення (запис) в технічній пам'яті (ТП) інформації, поданої на вхід ТП, збереження цієї інформації протягом певного часу, відтворення (читання, зчитування) раніше записаної інформації на виходах ТП, пошук заданої певними ознаками ін-

формації в ТП, знищення збереженої інформації. Елемент пам'яті (ЕП) – частина технічної пам'яті (ТП), використовувана для зберігання найменшої одиниці даних – 1 біта. Основні параметри ТП – інформаційна ємність, розрядність, швидкодія, споживана потужність та ін.

Множина ЕП в ТП утворює запам'ятовувальний масив (ЗМ). Масив розділений на окремі комірки (КП), кожна з них призначена для зберігання двійкового слова, кількість розрядів в якому визначається шириною вибірки пам'яті. За способом організації пам'яті (розміщення та пошук інформації) в ЗМ розрізняють адресну, асоціативну і безадресну (нульадресну, стекову або магазинну) ТП.

У пам'яті з адресною організацією розміщення і пошук інформації в ЗМ реалізують шляхом використання адреси (тобто координат в просторі або в часі) слова (числа), яким служить номер КП, в якій це слово розміщується. Факт виконання операцій запису або читання (зчитування) називають також звертанням (зверненням) до ТП. Для запису (зчитування) слова в ЗМ ініціюють цю операцію, команда звертання до ТП вказує адресу, за якою здійснюється звертання.

Структура адресної ТП частин (рис. 9.5 а) містить ЗМ з N n -розрядних комірок і його електронне обрамлення, що містить регістр адреси PzA , що має K ($K \geq \log_2 N$) розрядів, блок пам'яті БП, блок вибірки БАВ з дешифратором адреси ДшА, блок підсилювачів зчитування БПЗч, блок розрядних підсилювачів сигналів запису БПЗп і блок управління пам'яттю БУП. За кодом адреси PzA БАВ формує сигнали, які дозволяють зробити в відповідній комірці зчитування або запис слова. Вважають, що будь-який ЗМ в ТП складається з окремих комірок, кожна з яких здатна зберігати одне слово певної довжини.



а)

Рисунок 9.5 – Структура адресної пам'яті з довільним звертанням

При запису після виконання вказаної загальної частини циклу звернення проводиться прийом слова з вхідної інформаційної шини *ШВх* на *P_{ГІ}*. Запис складається з двох фаз: очищення комірки (встановлення в «0») і власне запис. Для цього *Б АВ* спочатку проводить вибірку та очищення комірки, заданої адресою в *P_{2А}*. Очищення виконується сигналами зчитування слова в комірку, але при цьому блокуються підсилювачі зчитування і з *БПЗз* в *P_{ГІ}* інформація не надходить. Потім в обрану *Б АВ* комірку записують слово з *БП*.

Блок управління БУП генерує необхідні послідовності сигналів управління, що ініціюють роботу окремих вузлів пам'яті. Кола передачі сигналів управління показані пунктирними лініями на рис. 9.5, а).

Інформаційна ємність ТП визначається найбільшою кількістю інформації, яка може бути зафіксована в ТП. Вимірюють ємність ТП кількістю *N* збережених *n*-розрядних слів. При цьому одиницею виміру слугує біт або байт (1 байт дорівнює 2^3 бітів). Тоді $N = 1 \text{ Кбіт} = 2^{10} \text{ бітів} = 1024 \text{ бітів}$; $\text{Кбайт} = 2^{10} \text{ байтів} = 1024 \text{ байтів}$; $\text{Мбіт} = 2^{20} (1024^2) \text{ бітів}$; $\text{Мбайт} = 2^{20} \text{ байтів} = (1024)^2 \text{ байтів}$. Розрядність ТП – довжина в двійкових розрядах слів, з якими в ТП виконують основні операції розміщення (запису) інформації, її зберігання та її отримання (читання). Звичайно розрядність слів в ТП кратна байту.

Швидкодія пам'яті визначається тривалістю операцій звернення, тобто тривалістю процесів запису або читання інформації. Тривалість звернення до пам'яті при читанні $t_{зв.лч}$ – час від моменту подачі адреси на регістр адреси ТП до моменту появи шуканого слова на регістрі числа ТП. Тривалість звернення до пам'яті при запису $t_{зв.лч}$ – час від моменту подачі адреси на регістр адреси ТП до закінчення процесу запису в ЗМ ТП.

Найбільш повною характеристикою швидкодії ТП є час циклу $T_{ц}$ – мінімальний час між двома послідовними зверненнями до ТП.

$$T_{ц} = t_{зв.лч} + t_{зв.зан.}$$

У загальному випадку комп'ютер може мати в своєму складі зовнішні і внутрішні пристрої ТП. Зовнішні слугують для зберігання великих обсягів інформації: сховищ даних і програмного забезпечення комп'ютерних систем. Внутрішні пристрої ТП за функціональністю поділяють на оперативні та постійні. Пристрої оперативної ТП (ОТП) виконують запис, зберігання і читання довільної двійкової інформації. Пристрої постійної ТП (ПТП) здійснюють зберігання і видачу постійно записаної інформації, зміст якої не змінюється в ході роботи комп'ютерної системи. Це використовувані в процесі роботи стандартні підпрограми і мікропрограми, табличні дані для перетворювачів кодів, а також таблиці різних інших функцій, константи та ін. Коли необхідно періодично змінювати інформацію в ПТП, то використовують перепрограмовні ПТП (ППТП).

Щодо фізичних принципів створення ОТП, то можна стверджувати (і аналіз багаточисельних публікацій про це свідчить), що протягом 20-го та 21-го століть практично всі відомі фізичні ефекти, явища і прилади пройшли через спроби створення ЕП на їх основі. Як результат такого потужного відбору в сучасній теорії і практиці ТП утвердилася думка, що найефективнішими (і в осяжному майбутньому перспективними) для схемотехніки ТП є напівпровідникові інтегральні структури. ТП на основі інших фізичних принципів побудови ЕП мають зараз скоріше епізодичні застосування, ніж регулярні.

Переваги напівпровідникових ЕП, їх використання при створенні ТП дають можливість значно підвищити швидкодію, зменшити масу, габарити і споживану потужність, підвищити надійність роботи ТП.

Основою напівпровідникових пристроїв ТП є тригер з колами управління для запису і зчитування інформації. За фізико-технологічними ознаками ЕП, як і ЛЕ, розрізняють на біполярних і на МОН-транзисторах. Пристрої ТП на біполярних транзисторах мають більш високу швидкодію $T_{\text{ц}} = 10 \dots 20$ нс (для ЕЗЛ) і $T_{\text{ц}} = 40 \dots 100$ нс (для ТТЛШ), але меншу щільність розміщення ЕП на кристалі (4 Кбіти), відносно малу споживану потужність, але й невисоку швидкодію.

Конфігурація ОЗП (рис. 9.5, б) типова для МОН ВІС з організацією 1024×1 . 10 – розрядний адресний код $A_0 \dots A_9$ надходить на дешифратори вибірки рядка ДШВС і колонки ДШВК, які спільно вибирають потрібний ЗЕ.

Одна з можливих організацій ОЗП з ЗЕ статичного типу показана на рис. 9.5, в). ОЗП містить 8-розрядний адресний вхід $A_0 \dots A_7$. Для читання інформації з ОЗП на вхід подається сигнал, дозвіл видачі при записі забороняється видача інформації з матриці ЗЕ на шину даних; дані, що підлягають запису, подаються на вхід ОЗП, і надходить сигнал запису інформації в обох режимах (запису і читання) адресні сигнали подаються до надходження сигналів.

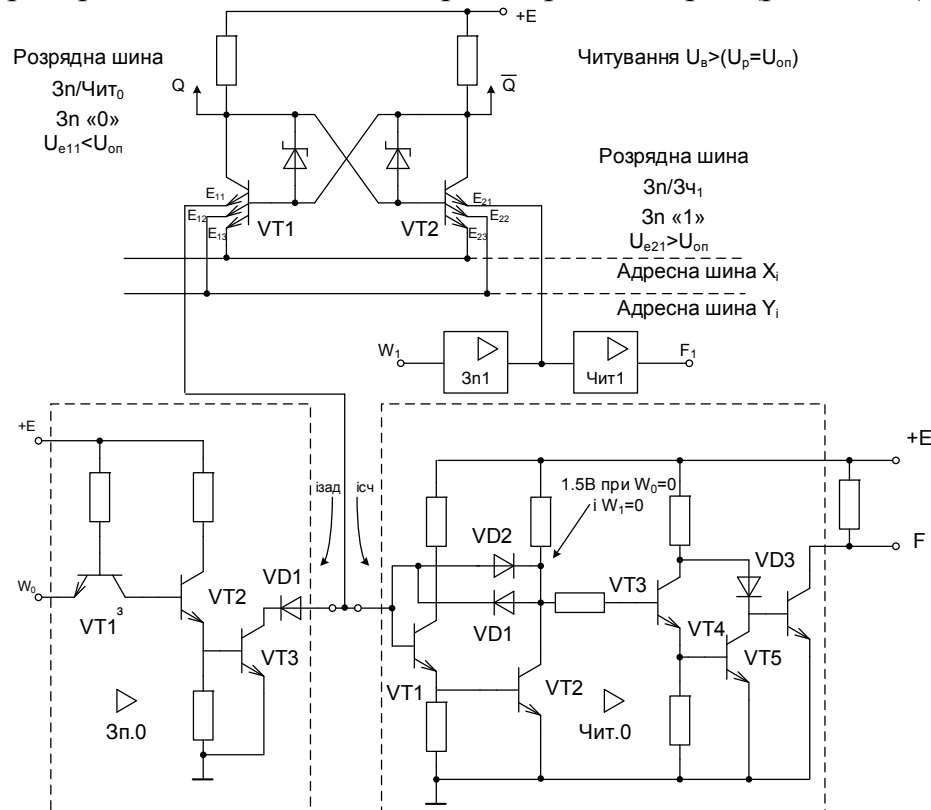
При побудові систем пам'яті найбільшого поширення набули ОЗП з конфігурацією $n \times 1$ ($n = 256, 512, 1024, 2048, 4096$). Наприклад, пам'ять багатьох мікроЕОМ має розрядність, яка дорівнює або кратна розрядності мікропроцесора. Тому для 8-розрядного мікропроцесора необхідна пам'ять з довжиною слова 8 бітів. Необхідна довжина слова пам'яті досягається паралельним вмиканням ВІС-пам'яті, де m – довжина слова в бітах ($m = 8, 12, 16, 24, 32, 64$). Сукупність 8 однорозрядних ліній даних всіх 8 ВІС утворює 8-розрядну шину даних системи пам'яті. Лінії адреси та читання-запису всіх ВІС вмикаються паралельно для одночасного звернення до всіх ВІС П.

Організація ОЗП ємністю 128×8 на основі двох ВІС ОЗУ ємністю 256×4 бітів показана на рис. 9.5, з, на рис. 9.5, д – ОЗП ємністю 2048 байтів на основі ВІС ЗП з організацією 1024×1 , на ньому вказані тільки адресні

кола ЗП. Сигнали з виходу тригера Тг подаються на входи РВ дозволу вибирання ВІС ЗП. Адресні лінії А0... А9 (рис. 9.5, д) служать для адресації комірок пам'яті всередині вибраних ВІС ЗП.

9.4 Елементи пам'яті на біполярних транзисторах

Як основу біполярних інтегральних ЕП найчастіше використовують статичні тригери на двох багатоемітерних транзисторах (рис. 9.6, а).



а)



б)

Рисунок 9.6 – Схема біполярного ЕП напівпровідникової ТП а) і часова діаграма б) її роботи

Для запису інформації в ЕП спочатку здійснюють вибірку потрібного ЕП подачею на адресні шини $АШХ_i$ і $АШУ_i$ позитивних потенціалів ($\geq 2,4$ В), що відповідають логічній «1», при цьому адресні емітери E_{13} , E_{23} і E_{12} , E_{22} виявляються замкнутими (див. рис. 9.6, б). Потім при запису «0» («1») в ЕП на вхід $W_0(W_1)$ підсилювача запису $Зп_0(Зп_1)$ подають логічну «1», а на вхід $W_0(W_1)$ підсилювача запису $Зп_1(Зп_0)$ подають логічний «0». При цьому з підсилювача запису $Зп_0(Зп_1)$ на розрядну шину РШЗп/Чит₀ (РШЗп/Чит₁) передається потенціал ($\leq 0,4$ В) логічного «0», а на іншу РШЗп/Чит₁ (РШЗп/Чит₀) з незбудженого підсилювача $Зп_1(Зп_0)$ передається потенціал, що дорівнює $\sim 1,5$ В. Якщо до запису тригер знаходиться в стані «0» (VT1 відкритий, VT2 закритий), то подача низького потенціалу на емітер E_{11} не змінює стан тригера. Якщо до запису тригер знаходився у стані «1» (VT1 закритий, VT2 відкритий), то при подачі низького потенціалу на емітер E_{11} відкривається транзистор VT1, а VT2 закривається, і тригер, таким чином, встановлюється в стан «0».

У режимі зберігання (ЕП не вибраний) на адресних шинах $АШХ_i$ і $АШУ_i$ та на виходах W_0 , W_1 – логічний «0». При цьому інформаційні емітери і E_{11} і E_{21} замкнені, оскільки на них подано потенціал 1... 1,5 В з колектора транзистора VT2 через діод, а емітерний струм відкритого транзистора VT1 замикається на землю через адресні шини $АШХ_i$ і $АШУ_i$.

У режимі зчитування (ЕП вибраний) на адресні шини $АШХ_i$ і $АШУ_i$ подається потенціал ($\geq 2,4$ В) логічної «1», а на входи W_0 і W_1 – потенціал логічного «0». Тому адресні емітери E_{12} і E_{13} , E_{23} і E_{22} виявляються замкнутими, а колекторний струм відкритого транзистора VT1 протікає через інформаційний емітер E_{11} в базове коло вхідного транзистора підсилювача зчитування Чит₀. При цьому вхідний транзистор відкривається і на виході підсилювача зчитування Чит₀ з'являється логічний «0» (рис. 9.6, б). Зчитування відбувається без руйнування інформації в ЕП. Збережена в ЕП інформація доступна для зчитування весь час, поки ЕП знаходиться в вибраному стані і в нього не проводиться запис. Для підвищення швидкодії ЕП в тригері застосовують діоди Шоттки в колах нелінійного негативного зворотного зв'язку, запобігаючи тим самим переходу транзисторів VT1 і VT2 в режим насичення.

ЦІМС біполярного ЗП являє собою кристал кремнію, в якому утворені ЗЕ (тригерів) з усіма з'єднаннями, а також адресні дешифратори, підсилювачі-формуваці запису, зчитування й інші функціональні вузли для управління адресним вибиранням, записом і зчитуванням. На рис. 9.7, б показана схема модуля ЦІМС KI55PYI (16 слів $\times$ 1 розряд) напівпровідникової біполярної ТП. Модуль має адресні входи $X_1... X_4$ і $Y_1... Y_4$, інформаційні входи W_0 , W_1 і виходи F_0 , F_1 .

Подібні ЦІМС часто мають вхід для сигналів вибору модуля. На рис. 9.7, а зображена структурна схема побудови інтегральних напівпровідникових біполярних багаторозрядних ЗП з модулів (корпусів) однорозрядних слів. На рис. 9.7, в показана структура ЗП для багаторозрядних слів.

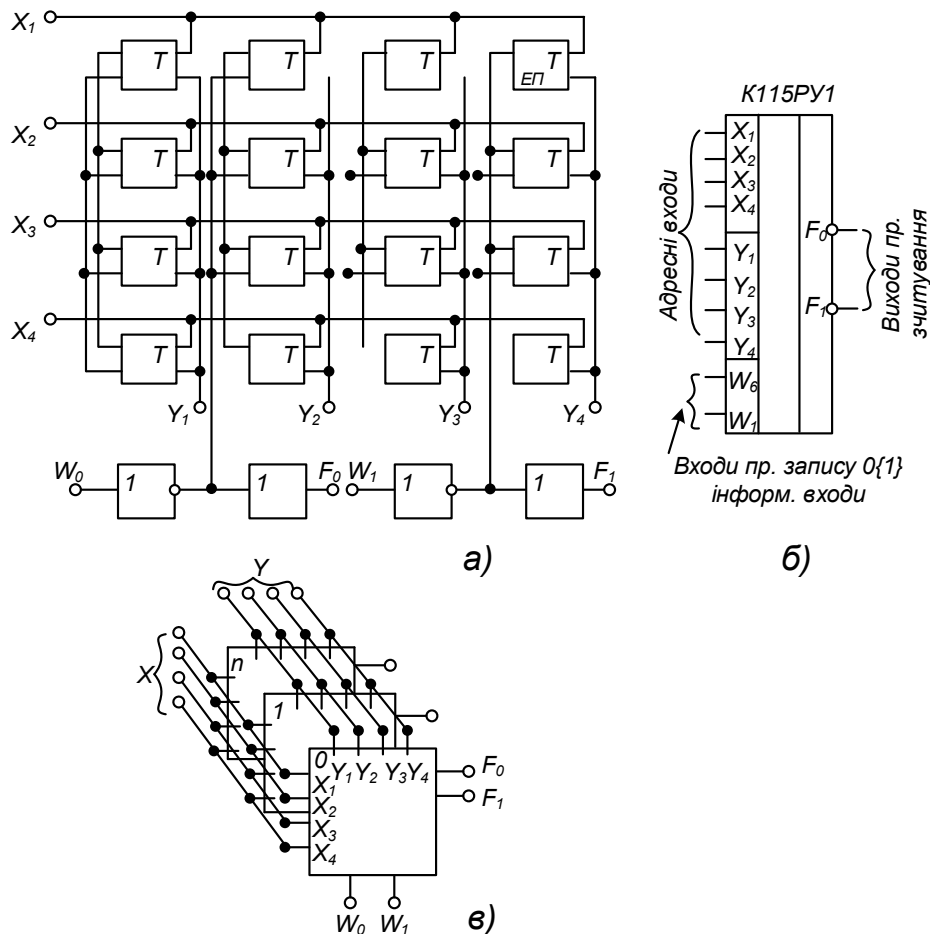


Рисунок 9.7 – Запам'ятовувальні пристрої:

- а) – організація елементів пам'яті; б) – позначення мікросхеми K115PY1;
 в) – тривимірна організація елементів пам'яті

9.5 Елементи технічної пам'яті на МОН-транзисторах

Пристрої ТП на МОН-транзисторах можуть бути виконані як на статичних, так і на динамічних елементах. Статичні ЕП базуються на тригері, що складається з двох МОН-транзисторів VT1, VT2 (рис. 9.8, а). Тут транзистори VT3 і VT4 працюють у режимі динамічного навантаження. Транзистори VT5, VT6 використовують як вентиля для звертання до тригера по адресній шині X_i , а VT7, VT8 – по адресній шині Y_i . У режимі зберігання на адресні шини X_i і Y_i подається потенціал ~ 0 , а на розрядні шини РШ0 і РШ1 – додатний потенціал, близький до E , при цьому транзистори VT5, VT6, VT7, VT8 замкнені, ізолюючи тим самим тригер від розрядних шин. У режимі запису на адресні шини X_i, Y_i подають позитивний потенціал і потенціал ~ 0 у відповідну розрядну шину РШ0 або РШ1, при цьому на виходах тригера встановлюються необхідні рівні. При зчитуванні в результаті подачі сигналу $>E$ на адресні шини транзистори VT6 і VT8 відкриваються й вимірюється струм, що протікає через відкриту частину тригера (рис. 9.8, б).

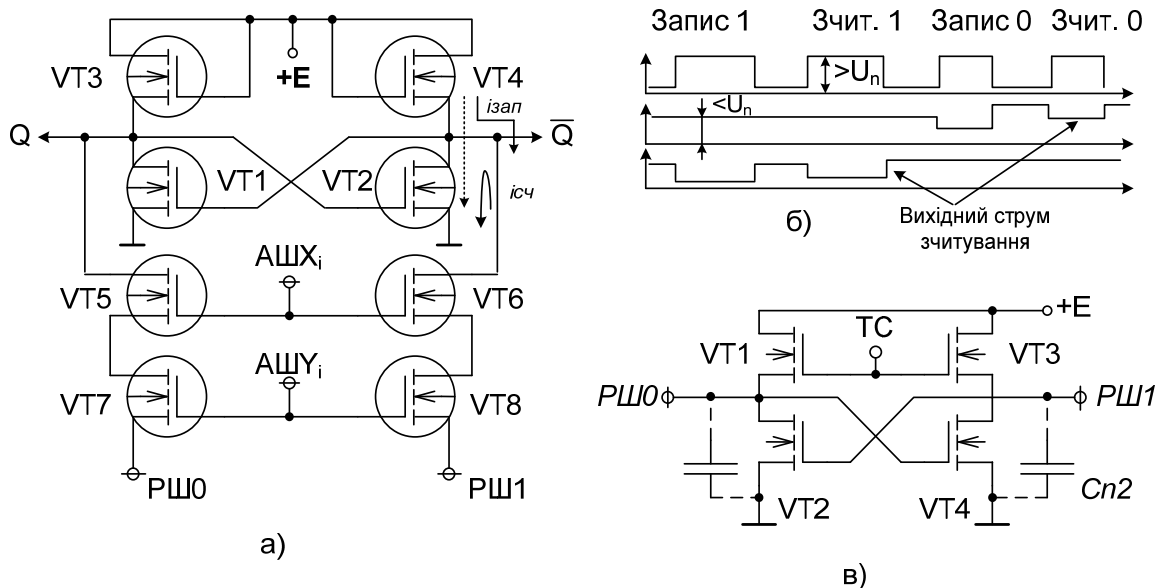


Рисунок 9.8 – Статичний ЗЕ на МОН-транзисторах: а) – схема; б) – діаграма його роботи; в) – критичний підсилювач зчитування

Як підсилювач зчитування звичайно використовують диференціальний каскад тригерного типу (рис. 9.8, в), який має високу чутливість. Підсилювач зчитування пов'язаний безпосередньо з ЕП. У режимі зберігання (тактовий сигнал $TC = 0$) транзистори $VT1$, $VT3$ закриті, а потенціали шин $PШ0$, $PШ1$ встановлені в нуль додатковими вузлами, що не показані на рис. 9.8, в. У режимі зчитування на обидві розрядні $PШ0$, $PШ1$ подають сигнал зчитування від обраного ЕП. Сигнал $TC = 1$ відкриває $VT1$, $VT3$ і паразитні ємності $CП1$, $CП2$ заряджаються від джерела живлення E через $VT1$, $VT3$ і транзистори ЕП. Залежно від збереженої в ЕП інформації на розрядних шинах $PШ0$, $PШ1$ встановлюється дисбаланс напруг, і після досягнення порогу спрацьовування тригера на $PШ1$, $PШ0$ встановлюються логічні рівні згідно з інформацією одиницею, що зберігається.

Істотного зниження споживаної потужності і збільшення швидкодії статичних ЕП можна досягти за рахунок застосування КМОН-транзисторів (рис. 9.9, а). Тут один з транзисторів в послідовній парі завжди відкритий, а другий – завжди закритий, тому споживана потужність мала і визначається в статичному режимі лише величиною струмів витоку. Швидкодія ЕП на КМОН-транзисторах значно вища, ніж на одноканальних МОН-транзисторах. Це обумовлено тим, що перезарядження навантажувальних ємностей відбувається через малий опір відкритого транзистора протилежного типу провідності на відміну від великого нелінійного опору навантажувального МОН-транзистора того ж типу провідності, що і активний прилад.

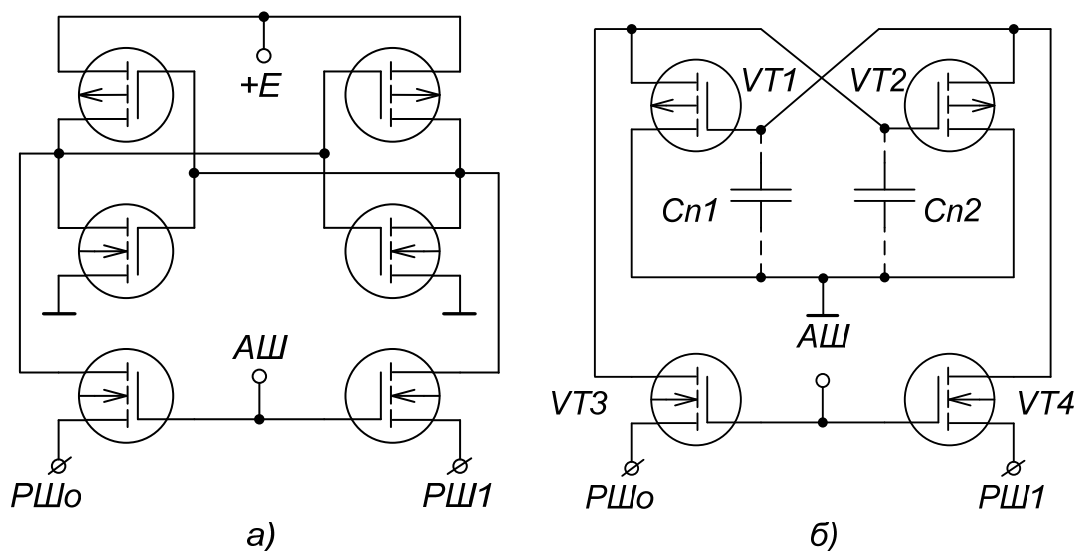


Рисунок 9.9 – Схеми ЕП на КМОН-транзисторах: **а)** статичного типу; **б)** динамічного типу

В основу роботи динамічних ЕП покладено принцип накопичення та відновлення заряду на затворі МОН-транзистора. Основа динамічного ЕП – конденсатор, що накопичує заряд (надалі вважаємо, що відсутність заряду на конденсаторі означає стан «0», а наявність – стан «1»). Оскільки конденсаторам властивий витік, то для збереження інформації накопичений заряд повинен періодично відновлюватися. Періодичне, а не постійне надходження енергії в ЕП дозволяє значно зменшити потужність споживання і спростити ЕП та підвищити щільність електронних компонент на кристалі.

На рис. 9.9, **б** показана схема динамічного ЕП. Цей елемент являє собою тригер без навантажувальних резисторів. Транзистори VT3 і VT4 суміщають функції збудження числових шин і навантаження. Запис інформації в ЕП проводиться подачею позитивного сигналу $+E$ на адресну шину (АШ) і негативного сигналу на відповідну розрядну шину. При цьому VT3 і VT4 відкриваються, інформація з розрядних шин PШ0 і PШ1 заноситься на тригер. У режимі зберігання інформація в ЕП зберігається у вигляді зарядів паразитних ємностей $C_{п1}$ і $C_{п2}$. Заряд, накопичений в режимі запису на паразитних ємностях $C_{п1}$ або $C_{п2}$, зменшується через струми витоку. Тому для відновлення інформації ЕП періодично збуджується АШ і одночасно на розрядні шини подається однакова за знаком напруга (негативна для p -канальних транзисторів). При зчитуванні інформації збуджується АШ, і струм зчитування через транзистор VT3 і VT4 надходить в розрядну шину, приєднану до відкритого тригера.

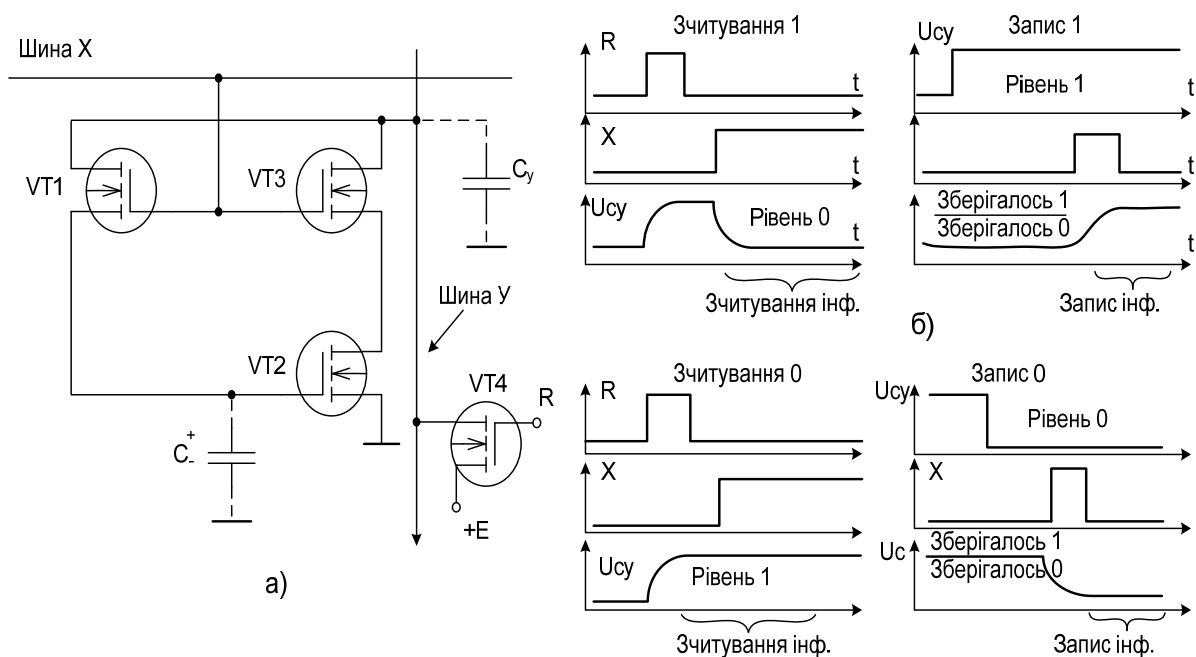


Рисунок 9.10 – Схема а) та часові діаграми б) роботи динамічного ЗЕ на трьох МОН-транзисторах

На сьогодні широке застосування в ТП знаходять тритранзисторні динамічні ЕП. Схема і часові діаграми роботи такого ЕП зображені на рис. 9.10. Запам'ятовувальний ефект забезпечує паразитна ємність C заслону транзистора VT2. Шина Y використовується для передавання в ЕП біта інформації при запису і отримання його при зчитуванні. Оскільки джерело живлення використовується тільки при зчитуванні, то ним може служити паразитна ємність C Y-шини. Попередньо перед зчитуванням подається сигнал R , що відкриває VT4, і конденсатор C_y Y заряджається від джерела $+E$. Потім на шину X подають сигнал синхронізації запису/читання Зп/Чит, який відкриває VT3, але не може відкрити VT1. Якщо ЗЕ зберігає «1», то конденсатор заряджений і відкритий VT2. У цьому випадку через відкриті VT3 і VT2 конденсатор C_y CY розряджається, і низький рівень «0» сигналу U_{cy} на шині Y вказує, що ЕП зберігав інверсну величину, тобто «1». Якщо ЕП зберігає «0», то конденсатор C розряджений, VT2 закритий, і сигнал на шині X не може викликати розряд конденсатора C_y CY. Високий рівень сигналу U_{cx} (рівень «1») вказує, що ЗЕ зберігав «0». Таким чином, підсилювач зчитування отримує з шини Y інверсне значення двійкового числа, яке зберігається в ЗЕ. Для запису на шину Y подають сигнал U_{cx} («1» або «0»). Потім на шину X подають високий рівень сигналу Зп/Чит, що відкриває VT1, під'єднує до шини Y конденсатор C . В результаті, незалежно від свого попереднього стану, конденсатор виявляється зарядженим, якщо записують «1», і розрядженим, якщо записують «0». У ЕП періодично проводиться регенерування інформації. При регенерації в ЗЕ записують

інверсне значення числа, що в ньому зберігалось. Після кожної парної регенерації в ЗЕ виявляється вихідне слово, а після кожної непарної – його інверсія. У ТП є пристрій, сигнал з якого вказує в даний момент ЕП пряме чи інверсне значення слова. У ОТП динамічного типу зберігання інформації без спотворення гарантується тільки протягом певного проміжку часу (часу зберігання чи періоду регенерації $T_{рег}$), після закінчення якого необхідно провести відновлення (регенерування). Залежно від способу організації регенерації інформації в динамічній у ОТП розрізняють пристрої з планарною регенерацією, з накачуванням заряду та з малою регенерацією. При планарній регенерації відновлення інформації здійснюють в спеціальному циклі (циклі регенерації), повторюваному з частотою ~ 10 кГц незалежно від режиму роботи мікросхеми. За іншого способу організації планарної регенерації кожен цикл звернення дещо збільшують, і в ньому відводять спеціальний інтервал часу, протягом якого проводять повну регенерацію інформації одночасно в усіх комірках пам'яті. Однак якщо протягом цього інтервалу до ОТП не було звернення, то для запобігання втрат інформації повинен бути сформований службовий цикл звернення. Особливість регенерації з накачуванням заряду полягає в одночасному відновленні інформації в усіх ЕП за допомогою спеціального сигналу регенерації (накачування заряду), який звичайно подають асинхронно на додатковий вхід ТП з періодом регенерації $T_{рег} = 0,01 \dots 1$ мс. При рядковій регенерації відновлення інформації здійснюється в спеціальному циклі, який являє собою або цикл зчитування за адресою регенерації. У цьому циклі регенеруються одночасно всі ЕП одного з рядків матриці ЕП за обраною адресою.

9.6 Стекова і магазинна пам'ять

Ці види ТП – безадресні. У стековій пам'яті комірки утворюють одновимірний ЗМ, в якому сусідні комірки пов'язані одна з одною колами передачі слів (рис. 9.11, а). Стек заповнюється з одного боку, при цьому слова записуються у вільні комірки з послідовними номерами, починаючи з комірки 0, а зчитування з видаленням слова з пам'яті проводиться з іншого боку стека і тільки з нижньої комірки (комірки 0). При цьому інші слова в стеку зсуваються вниз в сусідні комірки з меншими номерами. Таким чином, порядок зчитування відповідає правилу: першим надійшов – першим обслуговується. При запису слова в стек вміст лічильника показчика стека (ЛПС) збільшується на 1, при зчитуванні зменшується на 1. При ЛПС = 0 виробляється сигнал КС = 0, що позначає, що стек порожній. При ЛПС = N-1 формується сигнал КС = N-1, який вказує, що стек заповнений.

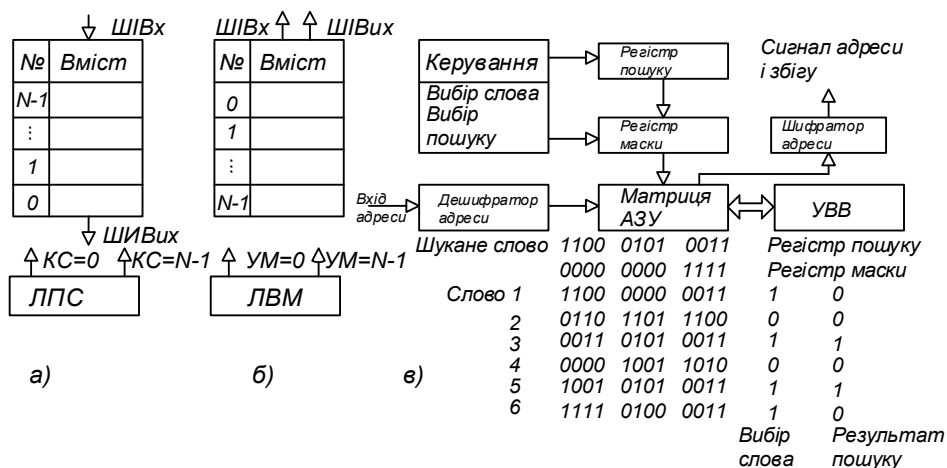


Рисунок 9.11 – Структура ТП: а) – стекової; б) – магазинної; в) – асоціативної

У магазинній пам'яті (рис. 9.11, б) запис нового слова проводиться у верхню комірку (комірку 0). При цьому раніше записані слова, також і слово, що знаходилося в комірці (0), зсуваються вниз, в сусідні комірки з більшими на 1 номерами. Зчитування можливе тільки з верхньої (нульової) комірки магазинної пам'яті. Якщо проводиться зчитування з видаленням, то всі інші слова в пам'яті зсуваються вгору, в сусідні комірки з більшими номерами. У цій пам'яті порядок зчитування слів відповідає правилу: останнім надійшов – першим обслуговується, в ряді пристроїв передбачається також операція простого зчитування слова з нульової комірки (без його видалення і без зсуву слів у пам'яті). Магазинна пам'ять забезпечується лічильником-показчиком магазину ЛВМ, принцип роботи якого аналогічний ЛПС.

9.6.1 Асоціативна ТП. У цій пам'яті пошук потрібної інформації проводиться не за адресою, а за її змістом (тобто, за асоціативною ознакою, яка властива збереженій інформації, незалежно від фізичних координат КП). При цьому пошук слова відбувається одночасно, по всіх розрядах паралельно. У асоціативних ЗП (АЗП) передбачається можливість замаскувати деякі розряди слова і вести пошук лише незамаскованих бітів.

На рис. 9.11, в показана структура АЗП. Нехай шукане слово є 1100 0101 0011. Якщо нас цікавлять лише останні чотири розряди, то перші 8 можна закрити маскою за допомогою операції «І» зі словом 0000 0000 1111. В такому разі працюють тільки останні чотири розряди в регістрі пошуку і кожне слово в ЗУ. Всією операцією управляє система, яка містить регістр вибору слова. Це дає можливість вести пошук лише певних слів. Так, у наведеному прикладі 1-е і 6-е слова не можна відшукати, навіть якщо їх останні чотири розряди збігаються зі словом в регістрі пошуку; тому на виході буде формуватися неприйнятний результат пошуку. Вміст регістра пошуку збігається лише з 3-м і 6-м словами, і тому з пам'яті виводиться повний зміст цих слів. АЗП в даний час використовуються не так широко, як адресні, стекові і магазинні. Одна з причин цього полягає в відсутності

загальноприйнятих вимог щодо способів формування асоціативних ознак, уніфікації їх структури і, як наслідок, їх складність у виготовленні і відлагодженні.

9.6.2 Постійні та перепрограмовні засоби ТП. Постійною ТП (частіше – постійними запам'ятовувальними пристроями – ПЗП) називають ТП, в якій існує постійна залежність між вхідною (адресою) і вихідною (числом, словом) інформацією. При виклику однієї і тієї ж адреси в ПЗП завжди читають одну певну інформацію, відповідну цій адресі. ПЗП звичайно проектується як адресні ЗП (рис. 9.12). Функціонування ПЗП можна розглядати як виконання однозначного перетворення k -розрядного слова адреси КП в n -розрядне слово, тобто ПЗП можна вважати перетворювачем кодів (комбінаційною схемою) з k -входами і n -виходами. ЗМ утворюється системою перпендикулярних шин, в перерізі яких встановлюються ЗЕ, які пов'язують (стан «1») між собою відповідні горизонтальні і вертикальні шини. Дешифратор ДС за адресою в РгА вибирає одну з горизонтальних шин, в яку подається сигнал вибірки. Вихідний сигнал (сигнал «1») з'являється на тих вертикальних розрядних шинах, які мають зв'язок зі збудженою адресною шиною.

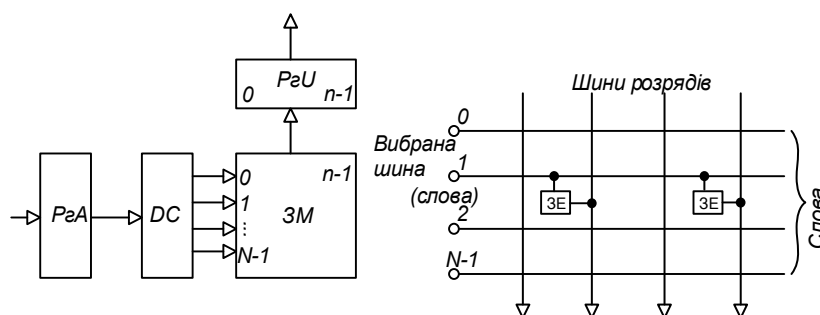


Рисунок 9. 12 – Структура ПЗП

За типом ЗЕ встановлюють або розривають зв'язок між горизонтальними і вертикальними шинами, розрізняють біполярні і МОН-структури ПЗП. Біполярні ПЗП мають високу швидкодію (час вибирання 30–50 нс), та невелику ємність (256 бітів, 1 К на один кристал). ПЗП на МОН-схемах мають велику ємність (від 2 К до 16 К на один кристал), але досить низьку швидкодію (час вибирання 0,5–2 мкс).

За способом запису інформації розрізняють три типи інтегральних напівпровідникових ПЗП: 1) з програмним в процесі виготовлення нанесенням контактних перетинок за допомогою фотошаблонів в потрібних користувачу точках; 2) з програмним випалюванням перетинок або пробоїв p - n переходів (одноразове програмування); 3) з електричним програмуванням, при якому інформація заноситься в ЗМ електричним шляхом, а стирання інформації виконується впливом на ЗМ ультрафіолетового випромінювання або електричним шляхом (ПЗП-ППЗП з можливістю перепрограмування).

На рис. 9.13, а показаний біполярний транзисторний ЗЕ з випалюваною перемичкою, що з'єднує горизонтальну і вертикальну шини. При «програмуванні» ПЗП перетинка випалюється в потрібних місцях імпульсами струму амплітудою 20... 30 мА. При виборі адресним дешифратором горизонтальної шини X на базу транзистора ЗЕ надходить сигнал, що його відкриває, і у разі наявності перемички (стан «1») на вертикальній шині з'являється потенціал колектора транзистора +5 В. На рис. 9.13, б зображений ЗЕ, програмований пошкодженням р-n переходу. У початковому стані з'єднані зустрічно діоди ізолюють шини X і Y (стан «0»). При подачі підвищеної напруги діод *VD1* пробивається і закорочується (стан «1»).

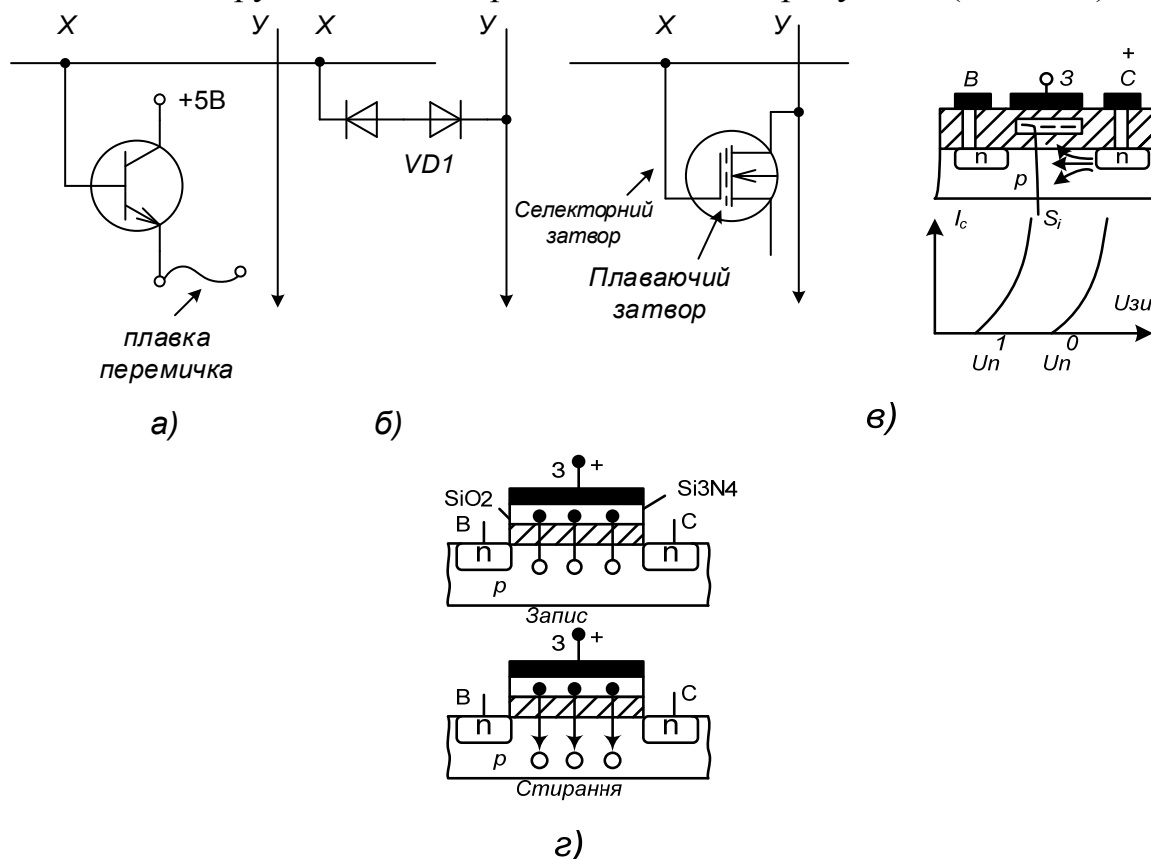


Рисунок 9.13 – Реалізація ЗЕ в ПЗП: а) – з плавкою перемичкою; б) – з р-n переходом, що пробивається; в) – на МОН-транзисторі з плаваючим затвором; г) – на МНОН-транзисторі в режимі запису і стирання

На рис. 9.13, в зображений ЗЕ, що використовується в ПЗП на лавинно-інжекційному МОН-транзисторі з плаваючим і селекторним затворами. Плаваючий (кремнієвий) затвор не має електричного підведення і призначений для зберігання заряду. Селекторний затвор приєднаний до одного з виходів дешифратора рядків (шини X), а стік – до шини Y. У початковому стані відсутній заряд на плаваючому затворі (стан «1», низького опору), транзистор має невелику порогову напругу. Вибір елемента здійснюється подачею на селекторний затвор вихідної напруги адресного дешифратора,

при цьому вмикається транзистор і через ланцюг стік-витік протікає значний струм. Програмування проводиться електричним шляхом – з подачею на стік імпульсу. При цьому відбувається інжекція електронів, що мають високу енергію, через окисел при лавинному пробіі стокового р-н переходу на ізолюваний затвор, який одержує негативний заряд. В результаті збільшується порогова напруга (стан «0»). Тепер подача на селекторний затвор вихідної напруги дешифратора не вмикає цей транзистор. Встановлений елементу стан зберігається як завгодно довго.

Стирання інформації проводиться ультрафіолетовим опроміненням структури ПЗП, результаті чого накопичений на плаваючому затворі заряд стікає на землю.

У ППЗП з електричним записом, стиранням і збереженням інформації при ввімкненому живленні як ЗЕ використовуються напівпровідникові структури «метал-нітрид-окисел-напівпровідник МНОН». У МНОН-структурах застосовують композиційний діелектрик, що складається з тонкого (~5 нм) шару SiO_2 і товстого (80–100 нм) шару нітриду Si_3N_4 (див. рис. 9.13, а), в якому є «пастки» електронів.

У нормальному стані транзистор ввімкнений (низька порогова напруга U_{nn}). У процесі запису на затвор МНОН-транзистора подають позитивну напругу (+30 В), при цьому електрони з підкладки проходять через тонкий шар SiO_2 і захоплюються «пастками». Накопичений ними негативний заряд в шарі нітриду підвищує порогову напругу U_{nv} транзистора і переводить його в закритий стан. Збереження заряду в «пастках» досить тривале (10 000 годин). Зчитування з ЗЕ на МНОН-транзисторах здійснюється при подачі на затвор напруги зчитування, що задовольняє умову $U_{nn} < U < U_{nv}$. Щоб зняти накопичений заряд, на затвор подається негативна напруга такої ж амплітуди. Ця напруга створює ефект, обернений процесу запису. МОН-транзистори з плаваючим затвором і МНОН-структури мають один загальний недолік – великий час запису (~10–3 с) і необхідністю управління високими напругами (~30 В) для запису.

Як ЗЕ в ППЗП можуть використовуватися аморфні напівпровідники (АН) наприклад, V_2O_5 – п'ятиокисел ванадію. Ці матеріали здатні зберігати один з двох стійких станів. У першому стані АП має аморфну структуру і високий питомий опір $\sim 5 \times 10^4$ Ом•см, а в другому – полікристалічну структуру з малим питомим опором 0,3 Ом•см (рис. 9.14, а). Значення опорів в цих двох станах порядку $\sim 10^3 \dots 10^5$. ЗЕ, для запобігання паразитних зв'язків між ними, ізолюють один від одного діодами або транзисторами (рис. 9.14, б, в).

Щоб перевести ЗЕ з аморфного стану в полікристалічний, необхідно спочатку подати імпульс високої напруги (+30 В), а потім подати імпульс струму ~7 мА і тривалістю ~10 мс. Після цього ЗЕ буде залишатися в низькоомному стані практично необмежений час. Щоб повернути ЗЕ в високоомний аморфний стан («0»), необхідно подати імпульс струму амплітудою

150 мА і тривалістю ~ 5 мкс. Зчитування здійснюється подачею струму ~ 1 мА і вимірюванням спаду напруги на ЗЕ.

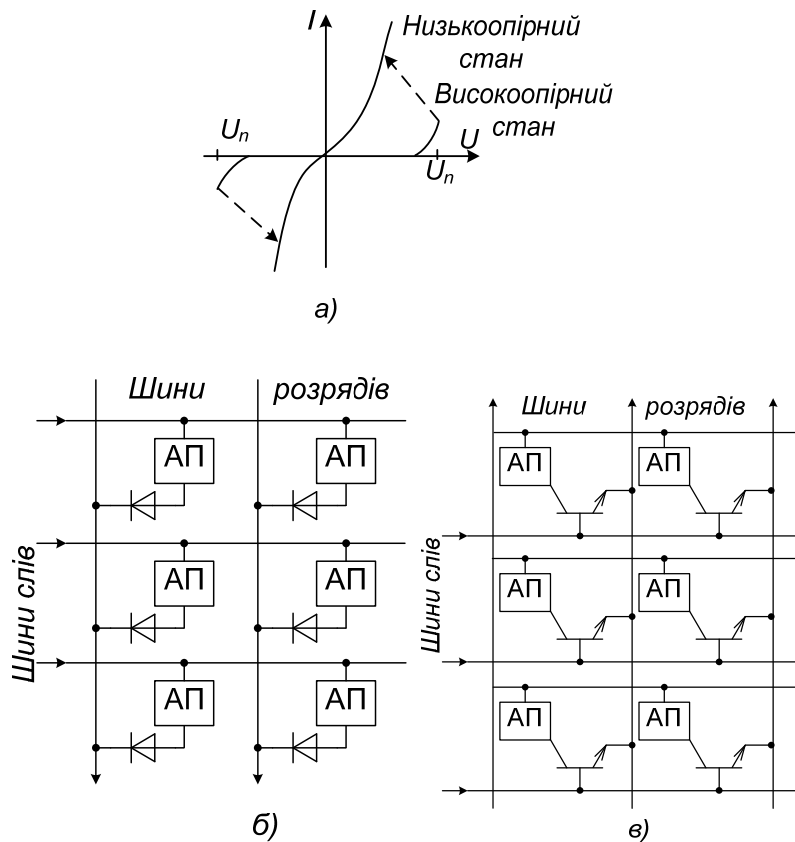


Рисунок 9.14 – ЗЕ на аморфних напівпровідниках: а) – характеристика перемикання АН; матриці ЗЕ на АН з діодними (б) і транзисторними (в) розв'язками

Структура перепрограмовного ПЗП ємністю 8 Кб (1 Кб $\times$ 8) на одному кристалі (в одному корпусі) зображена на рис 9.15. ЗМ містить 64 горизонтальних X і 128 вертикальних Y шин, на перетині яких розташовані МОН-транзистори з плаваючим і селекторним заслонами. Вертикальні шини розбиті на вісім груп по 16 у кожній. Кількість груп відповідає кількості розрядів, збережених в мікросхемі (корпусі) слів.

Шини Y однієї групи за допомогою звичайних МОН-транзисторів приєднані до відповідного розрядного підсилювача зчитування-запису.

Вибірка 8-розрядної комірки в ЗМ проводиться за принципом збігу збуджених горизонтальних і вертикальних шин. Старші розрядні адреси A_0, A_1, A_2, A_3 через буфери подають на дешифратор шин Y , в результаті чого збуджується одна з 16 вихідних ліній цього дешифратора, а приєднані до цієї лінії своїми заслонами вісім МОН-транзисторів під'єднують до розрядних підсилювачів зчитування-запису вісім вертикальних шин (по одній з кожної групи). До кожної з обраних таким чином вертикальних шин під-

ключені своїми стоками 64 ЗЕ, заслони яких з'єднані з відповідними горизонтальними шинами.

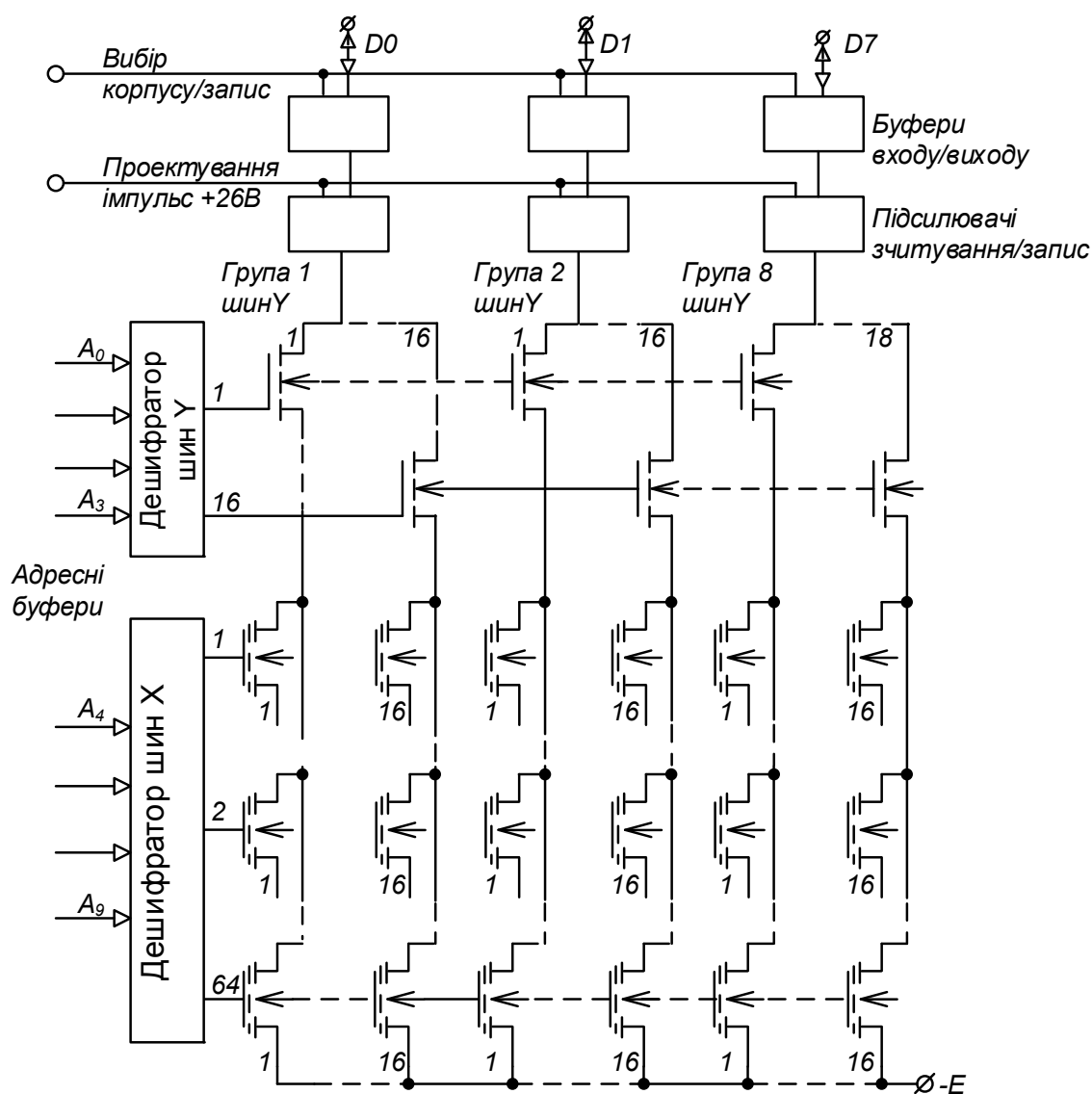


Рисунок 9.15 – Перепрограмовний МОНПЗП ємністю 8К (1К×8) бітів

При подачі молодших розрядів адреси A₄, ..., A₉ на дешифратор шин Х вибирається одна з 64 горизонтальних шин, в результаті чого управляльний потенціал надходить на приєднані до цієї шини селекторні заслони 128 ЗЕ. У кожній групі шин вибирається один ЗЕ, що опинився на перетині обраних вертикальних і горизонтальних шин. Якщо обраний для цього ЗЕ зберігає «1» (відсутній заряд на плаваючому заслоні), то подача на селекторний заслон вихідної напруги адресного дешифратора вмикає транзистор ЗЕ, і через його стік протікає струм на вхід розрядного підсилювача. Якщо ж ЗЕ зберігає «0» (присутній заряд на плаваючому заслоні), то сигнал з виходу дешифратора шин Х не може ввімкнути цей транзистор, і струм не надходить на вхід зарядного підсилювача. Розрядні підсилювачі

формують і через буфери входу/виходу видають на входи D0,... D1 мікросхеми сигнали логічних «0» та «1», що відповідають інформації у вибраній КП. При зчитуванні інформація в комірці зберігається.

Розглянутий ПЗП може працювати в двох режимах: 1) зчитування інформації, що зберігається; 2) запис (програмування ПЗП). Режим зчитування встановлюється подачею сигналу вибору, який налаштовує схеми буферів входу/виходу і розрядних підсилювачів на формування та передачу на входи D0, ..., D7 сигналів. Перед записом проводиться стирання інформації ультрафіолетовим опроміненням напівпровідникового кристала ПЗП. Під впливом світла заряди стікають з заслонів. Після стирання всі ЗЕ перебувають у стані «1» (провідному).

Режим запису здійснюється при одночасній подачі сигналу «Запис» і імпульсу «Програмування». На входи A0, ..., A9 надходить адреса комірки, в яку проводиться запис, а на виходи D0, ..., D1 (в режимі запису стали входами) надходить записуване 8-розрядне слово. Сигнал «Запис» налаштовує буфера входу/виходу і розрядні підсилювачі на передачу інформаційних сигналів з входів D0, ..., D7 до стоків ЗЕ. При цьому відкриваються підсилювачі запису тих розрядів, в яких в записуваному слові знаходиться 0. Ці підсилювачі пропускають у вибрані дешифратори шин Y вертикальні шини імпульс «Програмування» +26 В, який забезпечує лавинний пробій і занесення негативного заряду в плаваючі затвори (запис 0) тільки тих ЗЕ, селекторні затвори яких приєднані до горизонтальної шини, збудженої вихідним сигналом дешифратора шин X. Розглядається пристрій, що має час вибирання 0,4–1 мкс, час програмування 30–100 с.

ПЗП, що перепрограмовуються, широко використовуються при розробці мікропроцесорних системів і мікро-ЕОМ.

9.7 Контрольні запитання і завдання

1. Призначення запам'ятовувальних пристроїв (ЗП) та їхня класифікація.
2. Перерахуйте основні параметри ТП.
3. Наведіть приклади різних структур ЗП.
4. Наведіть приклад структури ЗП із однорозрядною організацією.
5. Пам'ять ЕОМ на базі статичних ЕП.
6. Пам'ять ЕОМ на базі динамічних ЕП.
7. Де застосовуються статичні і динамічні ЕП?
8. Пам'ять ЕОМ на базі постійних ЗП (ПЗП). Як можна класифікувати ПЗП?
9. Наведіть фрагмент схеми маскового ПЗП і поясніть принцип запису.
10. Наведіть фрагмент схеми діодної матриці перепрограмовного ПЗП і поясніть принцип запису.

10 СТРУКТУРНА ОРГАНІЗАЦІЯ ПРОГРАМОВНИХ ЛОГІЧНИХ ІНТЕГРАЛЬНИХ СХЕМ ТА ЇХ ВИКОРИСТАННЯ В КОМП'ЮТЕРНІЙ СХЕМОТЕХНІЦІ

Перелік скорочень і умовних позначень до розділу 10

БМК – базові матричні кристали; КЛБ – конфігуровні логічні блоки; ЛТ – логічна таблиця; ОЗП – оперативний запам'ятовувальний пристрій; ПЛІС – програмовні логічні інтегральні схеми; ПЛМ – програмовні логічні матриці; ПМЛ – програмовні матриці логіки; ППЗП – програмовні постійні запам'ятовувальні пристрої; САПР – системи автоматизації проектування.

10.1 Загальні відомості щодо програмовних логічних інтегральних схем

Програмовна логічна інтегральна схема (ПЛІС, англ. Programmable Logic Device, PLD) – електронний компонент високого рівня інтеграції, який використовується для створення сучасних обчислювальних засобів. Сама по собі така цифрова інтегральна схема з програмовною структурою є напівфабрикатом. ***Щоб створити певний функціонально закінчений обчислювальний пристрій на основі ПЛІС, необхідно виконати конфігурування ПЛІС, тобто завантажити до неї програму (так звану «прошивку»), яка визначатиме логіку роботи пристрою.*** Тоді, враховуючи цю обставину, ПЛІС вже може розглядатися як програмовний логічний пристрій. Створити прошивку (тобто, запрограмувати ПЛІС) можна кількома способами, використовуючи спеціалізовані системи автоматизації проектування (САПР), а саме: відтворити логіку роботи пристрою схематично або графічно; описати її за допомогою спеціалізованих алгоритмічних мов опису апаратури (як, наприклад, VHDL, Verilog, System C, SystemVerilog тощо). Слід пам'ятати, ***що ПЛІС без спеціалізованих САПР не дають розробнику обчислювальних засобів жодних переваг. Тільки спільно ПЛІС і САПР стають могутнім знаряддям створення високоефективних засобів обчислювальної техніки.***

Розвиток ПЛІС-технологій стартував на початку 70-х років 20-го століття. Поштовхом до цього була поява програмовних постійних запам'ятовувальних пристроїв (ППЗП, англ. Programmable Read Only Memory – PROM), які, крім основного їх призначення – зберігання даних, стали застосовувати для реалізації логічних функцій. Однак необхідність приведення логічних функцій до досконалої диз'юнктивної нормальної форми (ДДНФ) не дозволяла ефективно використовувати PROM для реалізації функцій від багатьох змінних. Перші програмовні логічні пристрої з ПЛІС-реалізацією систем булевих функцій багатьох змінних були створені на основі програмовних логічних матриць (ПЛМ, англ. Programmable Logic Arrays, PLA), які, врешті, набули широкого використання як універсальна

елементна база цифрових пристроїв. Наступною сходинкою розвитку ПЛІС стали програмовні матриці логіки (ПМЛ, Programmable Arrays Logic, PAL) які визначили архітектуру (тобто, організацію структур і системне програмне забезпечення) сучасних ПЛІС типу CPLD (Complex Programmable Logic Devices). Також варто відзначити роль базових матричних кристалів (БМК, англ. Gate Array, GA) та їхній вплив на формування таких архітектур сучасних ПЛІС, як FPGA (Field Programmable Gate Array).

Далі в цьому розділі для загального позначення всіх структурних утворень, що мають бути класифіковані, будемо використовувати одну аббревіатуру – ПЛІС. Крім того, у конкретних випадках будемо застосовувати і англійські варіанти відповідних аббревіатур, оскільки вони вже міцно закріпилися на практиці і, зважаючи на особливості розвитку національної електронної промисловості, однозначних загальновизнаних україномовних аналогів може і не існувати.

ПЛІС можуть бути класифіковані за:

- рівнем інтеграції і, отже, пов'язаною з цим складністю (кількість елементів та міжз'єднань, наявність перехресних зв'язків тощо);
- архітектурними властивостями (типи функціональних блоків, характер системи міжз'єднань);
- кількістю допустимих циклів програмування (репрограмування);
- типом пам'яті конфігурування («тіньової» пам'яті);
- ступенем залежності затримок поширення сигналів від довжини ліній зв'язку;
- системними властивостями;
- схемотехнологічною реалізацією (МОН, ТТЛ, ТТЛШ, ЕЗЛ та ін.);
- однорідністю або гібридністю (за наявністю або відсутністю в мікросхемах зон із різними (за методами проектування) пристроями).

Очевидно, що наведений перелік не вичерпує всіх можливих класифікаційних ознак ПЛІС. Усі вони мають важливе значення і стосуються різних якостей об'єктів класифікації. Однак розглядати ПЛІС з метою їх вивчення за всіма цими ознаками було б досить громіздкою справою. Тому для навчання доцільно укрупнити і виділити основні класифікаційні ознаки. Отже, далі будемо притримуватися класифікацій за:

- архітектурними властивостями;
- рівнем інтеграції і однорідності/гібридності;
- кратністю програмування, що визначає також тип пам'яті конфігурування.

10.2 Класифікація ПЛІС за архітектурними властивостями

Класифікація за архітектурними властивостями (рис.10.1) дає найповніше уявлення про класи задач, які вирішуються за допомогою ПЛІС того чи іншого типу. За цією класифікацією ПЛІС поділяють на чотири класи.

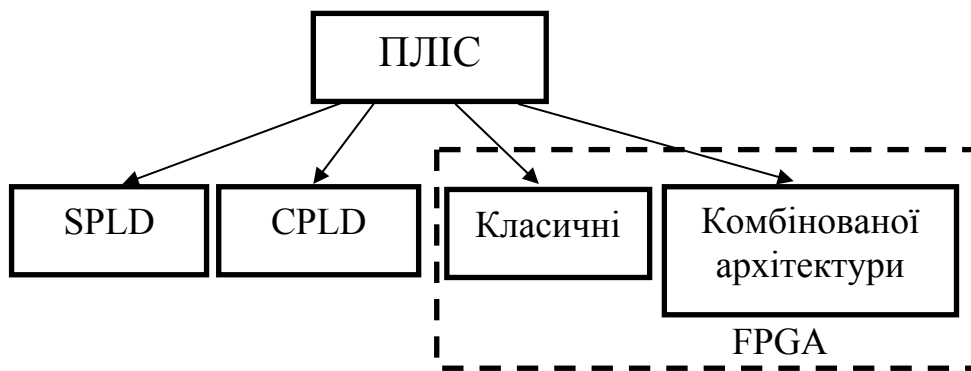


Рисунок 10.1 – Класифікація ПЛІС за архітектурними властивостями

Першим з них розглянемо клас простих програмовних логічних пристроїв (англ. Simple Programmable Logic Devices, SPLD). За архітектурними особливостями SPLD, в свою чергу, ділять на підкласи: програмовні логічні матриці (ПЛМ) та програмовні матриці логіки (ПМЛ).

Ці підкласи ПЛІС реалізують диз'юнктивні нормальні форми перемикальних функцій, а їх основними структурними блоками є дві матриці: матриця елементів «І» (кон'юнктивна матриця) і матриця елементів «АБО» (диз'юнктивна матриця), що з'єднані послідовно. Узагальнена структура ПЛМ показана на рис. 10.2.

В такій структурі ПЛМ обидві матриці логічних елементів «І» та «АБО» є програмовними. Програмування (прошивка) здійснюється шляхом керованої (в спеціальному електричному режимі) комутації входів елементів «І» та «АБО» до провідників, по яких передаються прямі та інверсні значення входних і проміжних логічних змінних. Число логічних елементів «І» в кон'юнктивній матриці має бути достатнім для того, щоб можна було реалізувати будь-який мінтерм ДДНФ, а набір логічних елементів «АБО» в диз'юнктивній матриці має забезпечувати можливість реалізації будь-якої логічної суми термів, які потрібні відповідно для подання заданих функцій у вигляді ДДНФ. В ПЛМ, яка наведена на рис.10. 2, ранг терма обмежений кількістю входів і дорівнює чотирьом, кількість термів теж дорівнює чотирьом. У мікросхемах ПЛМ, які серійно випускалися промисловістю, максимальна кількість входів дорівнювала шістнадцяти (максимальний ранг мін термів – 16), кількість термів дорівнювала 32 і кількість виходів мікросхеми – 8. Процес програмування таких ПЛМ на фізичному рівні здійснюється так. Там, де потрібно зберегти плавку перемичку, на її вхід і вихід подається висока напруга, там, де з'єднання не потрібно, на вхід подається потенціал нуля, а на вихід – напруга живлення. Таким чином, перемичка з полікристалічного кремнію під впливом високої температури струму короткого замикання випаровується.

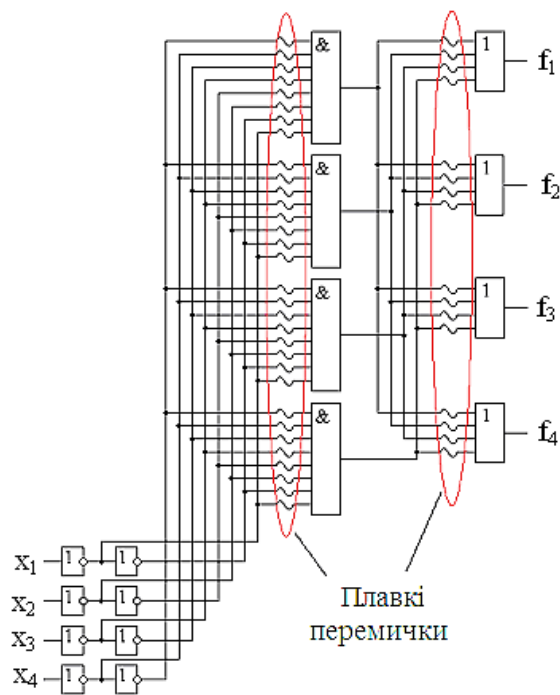


Рисунок 10.2 – Узагальнена структура програмовних логічних матриць

Оскільки повна принципова схема ПЛМ є досить складною для візуального сприйняття, то в літературі (особливо англомовній) досить часто застосовується шинне подання провідників (рис. 10.3), де логічний елемент «І», який реалізує мінтерм ДДНФ, зображується як окремий горизонтальний рядок з умовно-графічним позначенням елемента «І».

До входів цього елемента підводиться набір провідників, а на виході під'єднано окремий провідник. Якщо вхідний провідник під'єднується до входу логічного елемента «І» (перемичка збережена), то це місце позначається хрестиком «х», а якщо з'єднання відсутнє (перемичка спалена), то хрестик не проставляється. Аналогічні позначення застосовуються і для багатовходових елементів «АБО».

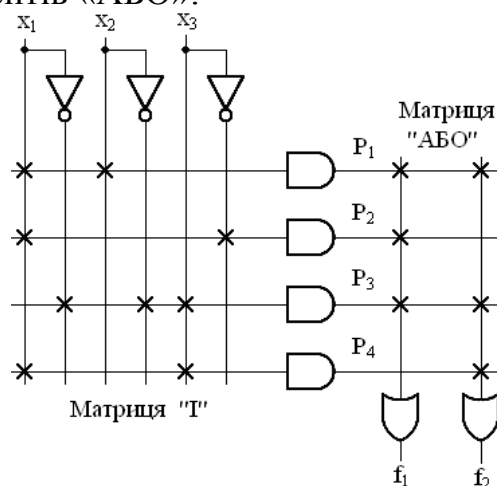


Рисунок 10.3 – Спрощене подання внутрішньої структури ПЛМ

Визначення реалізованих логічних функцій відповідно до такого схематичного позначення не викликає жодних труднощів. Наприклад, ПЛМ на рис. 10.3 реалізує логічні функції f_1 і f_2

$$f_1 = x_1x_2 + x_1\overline{x_3} + \overline{x_1}\overline{x_2}x_3,$$

$$f_2 = x_1x_2 + \overline{x_1}\overline{x_2}x_3 + x_1x_3.$$

Головним недоліком ПЛС такої архітектури є неефективне використання ресурсів програмовної матриці логічних елементів «АБО» (в практиці проектування обчислювальних засобів звичайно треба реалізувати суттєво менше ДДНФ, ніж це дозволяють можливості прошивки диз'юнктивної матриці). Ця обставина привела до появи ще одного виду ПЛС – програмовних матриць логіки або PAL, в яких матриця логічних елементів «І» є програмовною, а матриця логічних елементів «АБО» - фіксованою. Як приклад розглянемо ПМЛ з трьома входами та двома виходами, що містять чотири логічних елементи «4-АБО», які реалізують чотири мінтерми шостого рангу (рис. 10.4).

Для правильного читання і розуміння схеми пристрою, яка показана на рис. 10.4, треба зважати на те, що кількість входів у логічних елементів «І» дорівнює кількості провідників на вході пристрою (в даному випадку шість), а з'єднання, які відмічені позначкою «х», під'єднані до входів логічного елемента; інші ж входи цього логічного елемента під'єднані через резистор до джерела живлення і тому не впливають на результат.

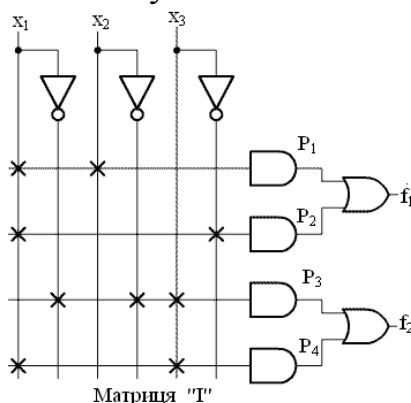


Рисунок 10.4 – Спрощене подання внутрішньої структури ПМЛ

Як видно, ПМЛ на рис.10.4 реалізує логічні функції:

$$f_1 = x_1x_2 + x_1\overline{x_3},$$

$$f_2 = x_1x_2x_3 + x_1x_3.$$

В ПМЛ на рис. 10.4 кон'юнктивну матрицю утворюють логічні елементи «4-І», які реалізують терми P_1 і P_2 , що з'єднані з першим елементом «АБО» диз'юнктивної матриці та логічні елементи «4-І», які реалізують терми P_3 і P_4 , що з'єднані з другим елементом «АБО». Однак ці з'єднання

між виходами кон'юнктивної матриці і входами диз'юнктивної матриці не можуть програмуватися. Тому, порівняно з ПЛМ, програмовні матриці логіки взагалі мають менше можливостей для програмування. В конкретній же ПМЛ на рис. 10.4 дозволяється використовувати чотири терми як входи для елементів АБО, але елементи АБО в ПМЛ мають тільки два фіксованих входи. Для компенсації цього недоліку виробники програмовних напівфабрикатів звичайно випускають декілька серій мікросхем ПМЛ, які відрізняються обсягом, числом входів і виходів та числом входів логічних елементів в диз'юнктивних матрицях.

Відомий також підклас ПЛІС класу SPLD, які подібні до ПЛМ і ПМЛ, але, на відміну від них, містять єдину програмовну матрицю «АБО-НІ» або «І-НІ». Саме така матриця за рахунок багатьох інверсних зворотних зв'язків дозволяє формувати досить складні логічні функції.

В цілому програмовні логічні матриці як апаратна основа проектування високоінтегрованих засобів обчислювальної техніки на сьогоднішній день вже вважаються морально застарілими і застосовуються для побудови відносно простих пристроїв, для яких не існує серійних мікросхем середнього рівня інтеграції. Для реалізації ж складних цифрових пристроїв найчастіше застосовуються ПЛІС типу CPLD (англ. Complex Programmable Logic Devices). Тут (рис. 10.5) кілька блоків, подібних до ПМЛ, поєднують засобами програмовної комутаційної матриці. Впливаючи на програмовні з'єднання комутаційної матриці і ПМЛ, що входять до складу CPLD, можна реалізувати необхідну структуру.



Рисунок 10.5 – Внутрішня організація CPLD

Спрощена внутрішня організація CPLD показана на рис. 10.5. CPLD складається з деякої кількості макрокомірок, розташованих на одному кристалі, кожна з яких з'єднана з блоками введення–виведення, що здійснюють формування необхідного типу входів або виходів для роботи з зовнішніми пристроями за допомогою програмовної комутаційної матриці.

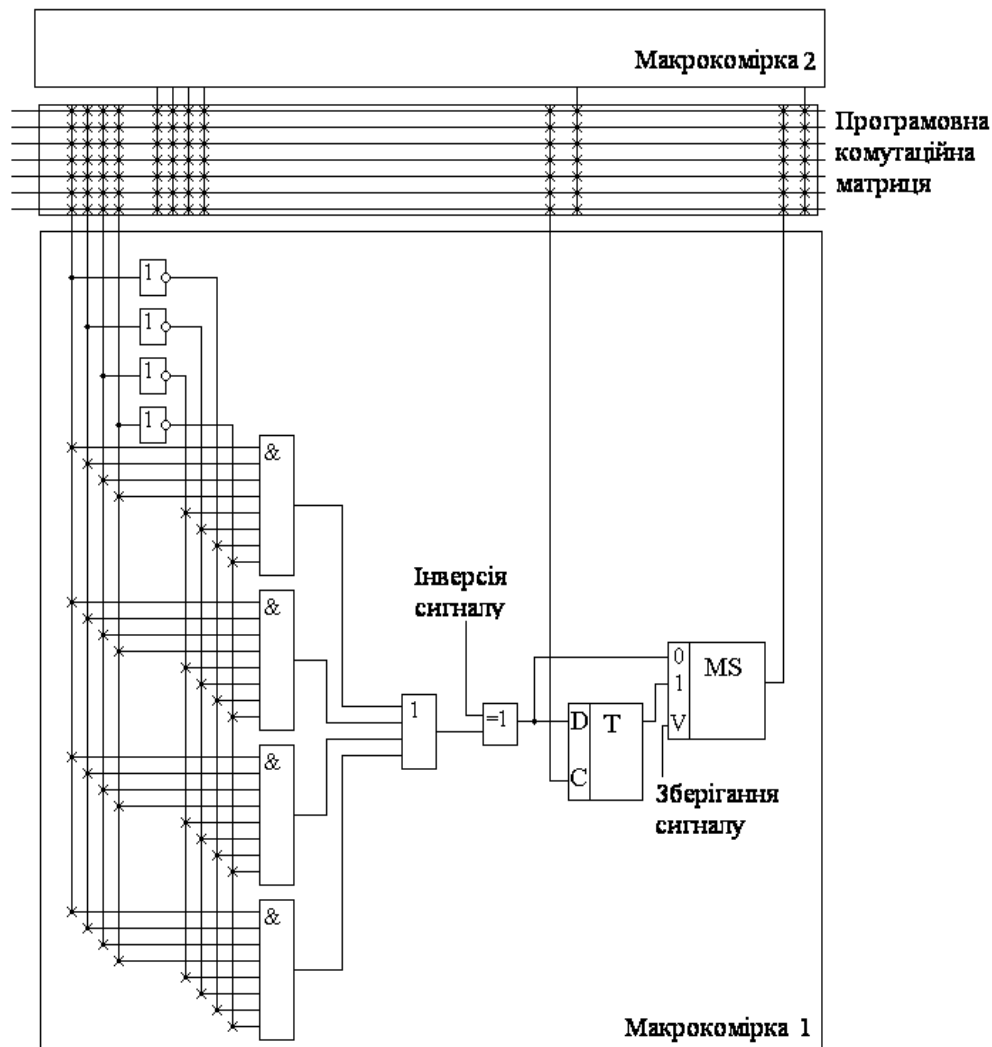


Рисунок 10.6 – Внутрішня структура макрокомірки CPLD

До макрокомірки (рис. 10.6) входять чотири шестивходових логічних елементи «6-І» і один 4-входовий логічний елемент «4-АБО». Його вихід з'єднаний зі входом логічного елемента «ВИКЛЮЧНЕ АБО». Цей логічний елемент призначений для реалізації інверсії логічної функції, яка реалізована ПЛМ-подібною структурою макрокомірки. Для цього на другий вхід елемента може подаватися логічний нуль або логічна одиниця. Якщо цей вхід запрограмований на подачу логічної одиниці, то елемент «ВИКЛЮЧНЕ АБО» буде інвертувати значення, яке надходить з виходу логічного елемента «4-АБО». Якщо ж на вхід надходить логічний нуль, то елемент «ВИКЛЮЧНЕ АБО» передаватиме на вихід сигнал без зміни. На ви-

ході макрокомірки знаходиться двовходовий мультиплексор, який дозволяє передавати на вихід поточне значення сигналу з виходу ПЛМ-подібної структури або вміст D-тригера.

Блок введення–виведення складається з буферних регістрів, які дозволяють налаштовувати виводи мікросхеми на введення або на виведення сигналів. Маршрут проходження сигналу по мікросхемі може бути запрограмований МОН-ключами, які розміщені на перетинах вертикальних і горизонтальних ліній провідників комутаційної матриці (на рис. 10.6 ці з'єднання позначено символом «x»). Необхідно відзначити, що коли вивід мікросхеми запрограмований на прийом інформації (тобто як вхід), то він не може бути використаний як вихід мікросхеми. Як правило, в супровідній до мікросхеми документації фірми-виробники вказують максимально можливу кількість вхідних та вихідних ліній.

Програмування CPLD відбувається у тій же системі, де застосовується ця ПЛІС. Для цього не потрібен програматор, оскільки процес програмування здійснюється через спеціальні виводи (JTAG-порт) за допомогою контролера програмування, що є в структурі CPLD (англ. In-system programming controller).

Розробка цифрових пристроїв на ПЛІС типу CPLD практично не має суттєвих відмінностей від проектування цифрових пристроїв на мікросхемах невисокого рівня інтеграції. Для цього не потрібно вивчати особливості внутрішньої структури мікросхеми або проектувати комутаційну матрицю. Розробники ПЛІС надають проектувальнику обчислювальних засобів пакет САПР, за допомогою якого можна вести розробку цифрового пристрою у схемному редакторі, а потім транслювати схему в файл комутаційної матриці міжз'єднань CPLD. Цей файл завантажується в ПЗП мікросхеми CPLD і, тим самим, мікросхема перетворюється у розроблений цифровий пристрій.

Перш, ніж перейти до ПЛІС типу FPGA, розглянемо клас інтегральних мікросхем із програмовною структурою, який отримав назву базових матричних кристалів (БМК). БМК називають також вентильними матрицями (англ. Gate Array, GA або Uncommitted Logic Array, ULA). Термін БМК використовується, в основному, в україно- та російськомовній літературі, хоча назва «вентильна матриця», очевидно, більш точно відповідає суті.

Для програмування БМК необхідне досить складне промислове обладнання. БМК — кристал, на прямокутній поверхні якого розміщені внутрішня та периферійна ділянки. У внутрішній ділянці по рядках і стовпцях (у вигляді матриці) розташовані базові комірки — групи некомутованих (не з'єднаних) схемних елементів (транзисторів, резисторів). Елементний склад базової комірки при різних варіантах з'єднань елементів допускає реалізацію деякої кількості пристроїв певного класу, кожен із яких відпо-

відає певній функціональній комірці. Для стандартних БМК створюється бібліотека функціональних блоків – набір фотошаблонів з'єднань, що реалізують певний пристрій. Бібліотеки функціональних комірок БМК нараховують звичайно десятки або сотні типових вузлів, реалізованих на одній або декількох базових комірках.

Розрізняють каналні, безканалні та блокові архітектури БМК. У БМК каналної структури (рис. 10.7) між рядками і стовпцями базових комірок або їхніх компактних груп залишаються горизонтальні й вертикальні вільні зони (канали) для міжз'єднань. В периферійній ділянці кристала розміщуються зони, що реалізують введення/виведення сигналів через контактні площадки, які розташовані по краях кристала.

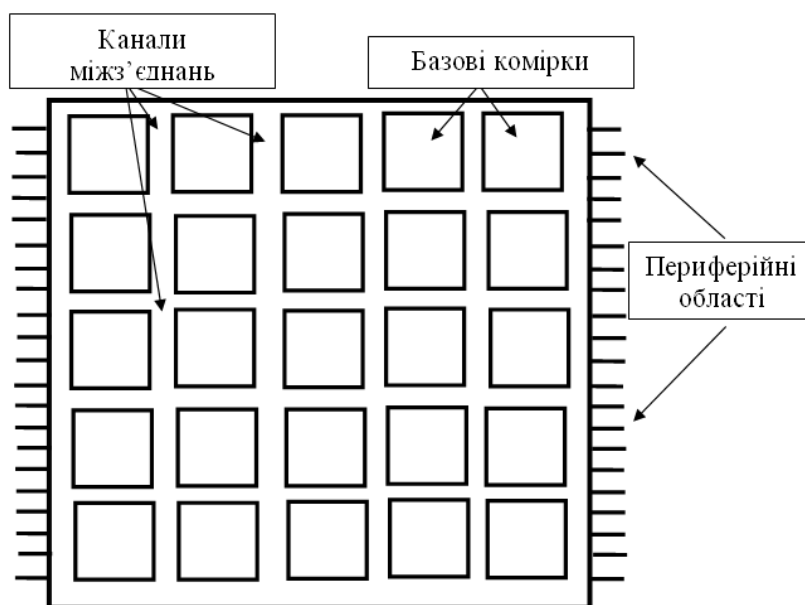


Рисунок 10.7 – Внутрішня структура каналних БМК

Для зменшення витрат площини кристала на трасувальні канали існують безканалні структури БМК типу «море вентилів» (містить масив завершених логічних елементів) або «море транзисторів» (містить масив транзисторів), в яких не існує вільних зон кристала для міжз'єднань. В таких БМК вся внутрішня ділянка щільно запаковується базовими комірками, а міжз'єднання проводяться тільки там, де це необхідно, причому базові комірки, що перебувають під ними, в конкретних випадках можуть бути невикористаними.

В блокових БМК (рис. 10.8) на кристалі створюють спеціальні блоки (області, підматриці) для реалізації заздалегідь відомих функцій (логічної обробки даних, пам'яті й ін.). Між підматрицями розміщуються спеціальні трасувальні канали. На периферії підматриць розташовуються внутрішні буферні каскади для передачі більш потужних сигналів по міжблоковим з'єднанням, що мають відносно більшу довжину.

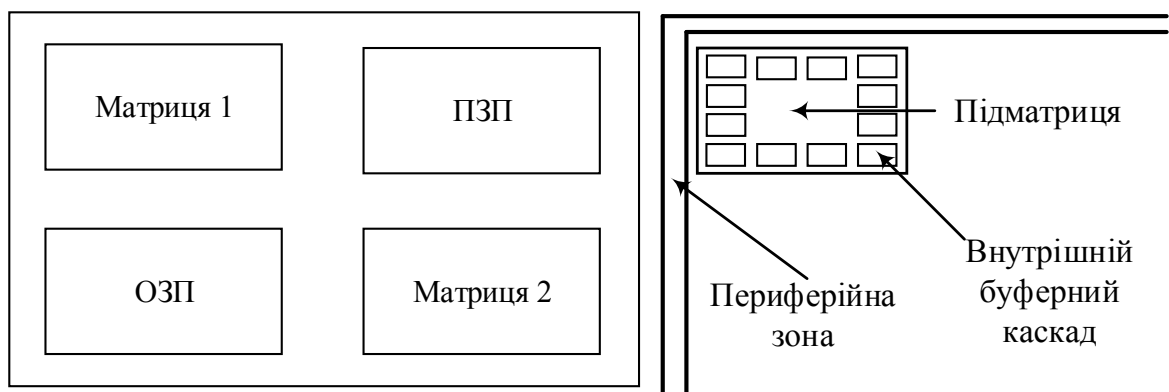


Рисунок 10.8 – Внутрішня структура блокових БМК

З точки зору розробника обчислювальних засобів основні переваги БМК є такими:

- БМК мають фіксоване геометричне розташування комірок, що спрощує автоматичне розміщення та трасування елементів;
- формування великої інтегральної схеми пристрою на основі БМК виконується за допомогою невеликої кількості фотошаблонів, що спрощує виробництво;
- наявність розвиненої бібліотеки логічних елементів і типових схемотехнічних рішень значно спрощує процес розробки;
- в складі одного БМК можуть бути реалізовані як цифрові, так і цифроаналогові елементи;
- пристрій, що розроблено на основі БМК, не потребує проведення кваліфікаційних випробувань, оскільки параметри пристрою на основі БМК є стандартизованими.

Для більш чіткого розуміння місця БМК в ієрархії інтегральних схем корисно доповнити абревіатуру GA (Gate Array) ще однією англійською абревіатурою – MPGA (англ. Mask Programmable Gate Array, програмовна за допомогою фотошаблону вентильна матриця). Також відомі БМК із лазерним програмуванням міжз'єднань – LPGA (англ. Laser Programmable Gate Array). При виготовленні цих БМК спочатку за допомогою металізованих доріжок виконують усі можливі міжз'єднання елементів, а потім при програмуванні частина з'єднань під дією точно сфокусованих керованих лазерних променів розривається. Такі БМК подібні до звичайних тому, що для їх програмування користувачу потрібно звертатися до виробника на підприємство електронної промисловості. Однак терміни і вартість виконання замовлення, порівняно зі звичайними БМК (MPGA), виявляються меншими.

ПЛІС типу CPLD призначені для застосування в цифрових пристроях з великою кількістю логічних елементів. Але в CPLD важко забезпечити ефективне застосування всіх макрокомірок, оскільки завжди деяка їх частина залишається невикористаною, займає площу кристала та споживає струм. Щоб уникнути цього при реалізації складних цифрових структур

застосовують ПЛІС, які мають більшу логічну ємність та вищий ступінь використання кристала. Розглянемо клас інтегральних схем із програмовною логікою, які об'єднали в собі властивості замовних інтегральних схем (англ. Application Specific Integral Circuit, ASIC) щодо можливостей реалізації складних проектів та простоту CPLD з точки зору програмування. До подібних мікросхем відносять ПЛІС типу FPGA. Принцип роботи FPGA суттєво відрізняється від принципу роботи CPLD. Основною особливістю FPGA є наявність матриці логічних елементів, що мають від двох до шести входів, тригерів, відрізків ліній зв'язку, що з'єднуються перемичками з польових транзисторів. Field Programmable Gate Array програмуються зміною напруженості електричного поля (field) в затворах цих транзисторів. Затвори всіх таких програмовних польових транзисторів під'єднані до виходів тригерів багаторозрядного регістра зсуву, в який записуються відповідні двійкові значення для програмування ПЛІС. Деякі з ділянок цього регістра можуть також виконувати роль оперативних (ОЗП) або постійних запам'ятовувальних пристроїв (ПЗП). Програма (прошивка) звичайно зберігається в ПЗП, що використовується разом із ПЛІС. Після подачі живлення або за сигналом скидання вона автоматично переписується в програмувальний регістр зсуву ПЛІС. Таким чином відбувається конфігурування ПЛІС.

Розглянемо типову архітектуру FPGA на прикладі ПЛІС фірми Xilinx. Слід зауважити, що в основі лінійки продукції мікросхем програмовної логіки фірми Xilinx знаходяться ПЛІС з архітектурою FPGA, що здатні повністю задовольнити переліки сучасних вимог щодо створення цифрових систем: від невисокої вартості, малих габаритів, високої ефективності використання в серійному виробництві і до надвисокої швидкості передачі даних, великого обсягу логічних ресурсів, можливостей цифрової обробки сигналів. До таких ПЛІС належать FPGA серій Virtex та Spartan різних поколінь. Принципи функціонування структурних елементів ПЛІС цієї серії можна відслідковувати і в архітектурах ПЛІС таких фірм-виробників, як Altera, Microsemi (Actel), Lattice, QuickLogic.

Поверхню кристала FPGA (рис. 10.9) займають основні елементи ПЛІС: матриця конфігурованих логічних блоків (КЛБ, англ. Configurable Logic Block, CLB); матриця програмовних електронних ключів міжз'єднань (польових транзисторів із плаваючим затвором, що розташовані на перетині вертикальних та горизонтальних ліній) для утворення ліній електричного зв'язку між внутрішніми блоками FPGA; блоки ОЗП (англ. BlockRAM), які розміщені уздовж кристала; блоки DSP-48 (англ. Digital Signal Processing) з помножувачем 25×18 , 48-бітовим акумулятором і передсуматором для високошвидкісної фільтрації; блоки введення/виведення (англ. Input Output Block, IOB).

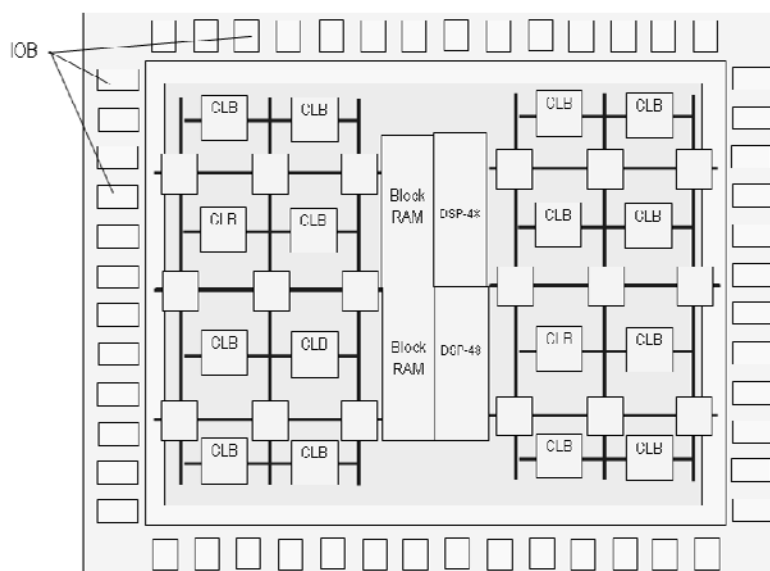


Рисунок 10.9 – Розташування елементів ПЛІС Virtex-4

Основним логічним елементом в ПЛІС є логічна таблиця (ЛТ, англ. look-up table, LUT), що являє собою однобітовий ПЗП на 16 комірок (рис. 10.10). У LUT за адресою G3, G2, G1, G0 записана одиниця, якщо код адреси являє собою мінтерм заданої чотиривходової логічної функції. Наприклад, якщо за адресою 1,1,1,1 записана одиниця, а за рештою адрес – нулі, то LUT реалізує чотиривходову функцію «І». На рис.10.10 показано приклад кодування функції «ВИКЛЮЧНЕ АБО» за допомогою LUT на чотири входи.

Тригери LUT входять до складу програмувального регістру зсуву, і їх початковий стан заповнюється під час конфігурування ПЛІС.

В сучасних ПЛІС використовуються LUT підвищеної ємності, які реалізують логічні функції від 6 аргументів. Крім того, інколи окремі частини пам'яті блокового ОЗП (BlockRAM) конфігуруються як 10..12-аргументна логічна функція.

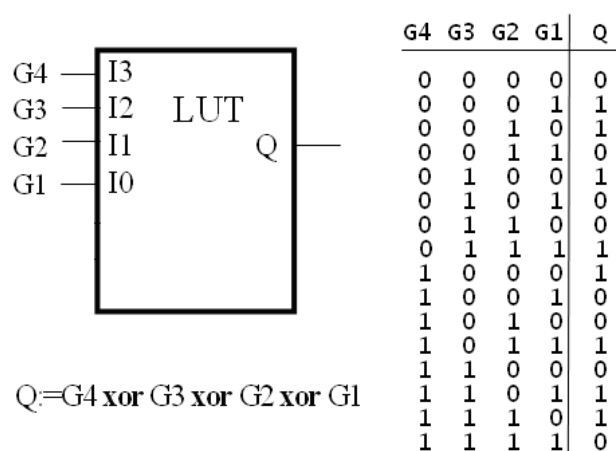


Рисунок 10.10 – Кодування функції «ВИКЛЮЧНЕ АБО» від чотирьох аргументів

У ПЛІС також використовують програмовні синхронні D-тригери (рис. 10.11). Завдяки їх програмному конфігуруванню можна задати такі режими роботи: тригер з початковим скиданням (R) або з початковим установленням (S); тригер із записом за фронтом або спадом синхросигналу; тригер із сигналом дозволу запису (E) або без нього.

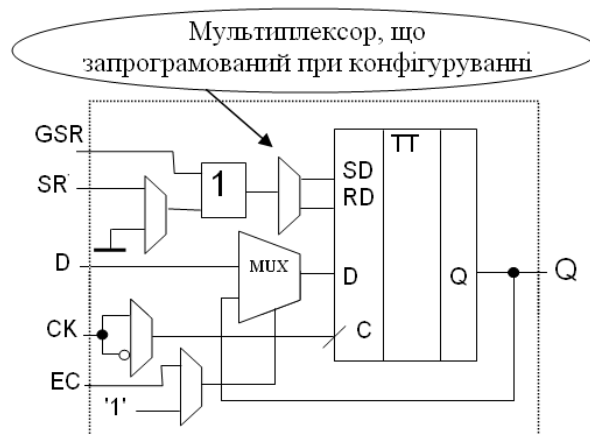


Рисунок 10.11 – Схема підключення тригера в ПЛІС

Після завершення конфігурування ПЛІС формує сигнал загального скидання (англ. global set-reset, GSR), який встановлює всі тригери в 0 або 1. Інші типи таких тригерів, як J-K-, R-S-, T-тригери (звичайно, якщо вони потрібні для проектуваного пристрою), реалізують на основі D-тригерів, до входів яких ввімкнено LUT, яка відповідає необхідній функції збудження.

Фірми, які створюють САПР для ПЛІС, звичайно не рекомендують конфігурувати тригери, що працюють у асинхронному режимі, внаслідок складності налаштування пристроїв з такими тригерами. Вони також забороняють реалізовувати на ПЛІС будь-які тригери, використовуючи комбінаційні схеми. Подібні комбінації логічних елементів можуть сприйматися системами автоматизованого проектування як фатальні помилки.

Сигнал СК подається на всі тригери ПЛІС від загальної мережі синхросигналів. Фронт синхросигналу, поширюючись по такій мережі, досягає усіх тригерів майже одночасно. Завдяки цьому спрощується проектування пристроїв і досягається максимально можлива тактова частота синхросигналу.

Для збереження такої якості у проектованих пристроїв в цілому небажано управляти тригером в ПЛІС шляхом під'єднання до нього синхросигналів через які-небудь логічні структури. Замість цього рекомендують використовувати вхід дозволу запису тригера, що на рис. 10.11 позначений як EC.

Для реалізації швидкодійних суматорів LUT комбінують з елементами «ВИКЛЮЧНЕ АБО» та двовходовими мультиплексорами, які служать для прискорення перенесення. Таке рішення характерне для ПЛІС фірм Xilinx та Lattice. Кілька LUT і тригерів (від 2 до 8) разом з лініями прискореного

перенесення групують в так званий конфігуровний логічний блок (КЛБ, англ. CLB). CLB розміщують у вузлах решітки з вертикальних і горизонтальних провідників.

Завдяки можливості з'єднання провідників за допомогою електронних ключів (польових транзисторів) один з одним, можна створювати потрібні з'єднання між логічними блоками. З'єднання, що здійснюються електронними ключами, можна програмувати, замикаючи і розмикаючи ці ключі. Двовимірний масив двійкових значень, які задають ці з'єднання, зберігається в конфігураційному ПЗП і може бути змінений при програмуванні ПЛІС. Кількість CLB, залежно від моделі ПЛІС, може досягати декількох мільйонів.

CLB із двох тригерів та двох LUT називають сегментом CLB (англ. CLB slice). На рис. 10.12 показана структура сегмента CLB ПЛІС фірми Xilinx. При збільшенні кількості входів LUT можливості CLB значно розширюються. Так, із 6-входовими LUT на CLB-slice можна реалізувати три-входовий суматор або суматор із мультиплексором на вході.

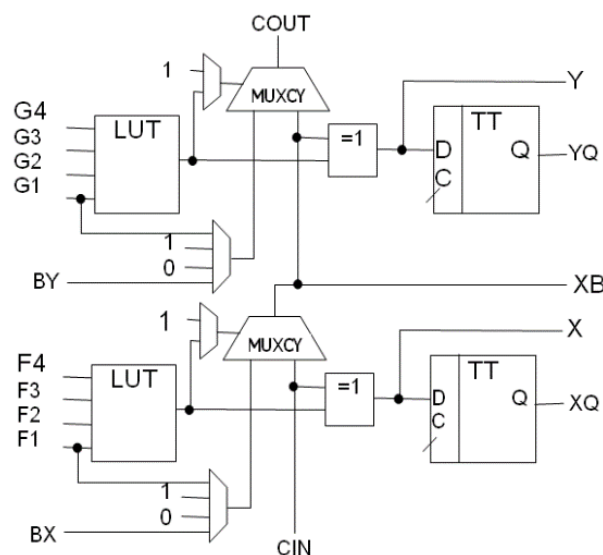


Рисунок 10.12 – Структура сегмента CLB (CLB slice)

У CLB-slice на рис. 10.12 кожний із двох LUT та двох тригерів з'єднаний через вузол, що реалізує прискорений перенос і два однобітових півсуматори. Така структура дозволяє ефективно реалізувати багаторозрядні паралельні суматори, а на їх основі – лічильники та арифметико-логічні пристрої.

У внутрішній структурі сучасних замовних великих інтегральних мікросхем рідко використовують шини з трьома станами через технологічну складність їх надійної реалізації, а також внаслідок зростання затримки сигналів під час перемикавання таких шин. Крім того, не всі технології ПЛІС дозволяють застосовувати шину з трьома станами. Тому в ПЛІС всі двонаправлені шини з багатьма джерелами і приймачами даних проекту-

ють на основі мультиплексорів. Оскільки на одній 4-входовій LUT можна реалізувати лише двовходовий мультиплексор, а на 6-входовій LUT – чотиривходовий, то для ефективної реалізації 4-х і більше входів мультиплексорів в CLB є мультиплексори F5 і F6, які виконують роль верхніх рівнів деревоподібного мультиплексора (рис. 10.13). Ці мультиплексори також використовуються для реалізації логічних функцій з п'ятьма (F5) та шістьма (F6) аргументами.

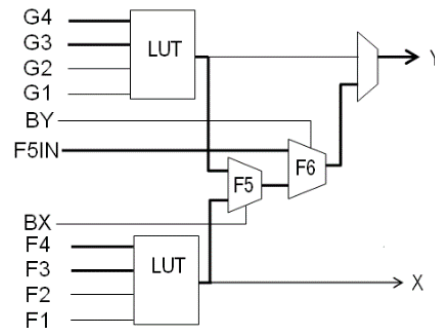


Рисунок 10.13 – Реалізація 5-входового мультиплексора на LUT

В ПЛІС останніх поколінь з'явилися ресурси для реалізації багатовходових мультиплексорів. Так, 6-входова LUT реалізує 4-входовий мультиплексор замість 2-входового мультиплексора, що було можливим у 4-входовій LUT. Додаткові мультиплексори у складі конфігурованих логічних блоків забезпечують побудову деревоподібних мультиплексорів із 8, 16 і більше входами. Такі мультиплексори необхідні для пересилання операндів в арифметико-логічний пристрій від різних джерел, а також для реалізації внутрішніх загальних шин обчислювальних засобів.

Для створення швидкодіяного однорозрядного ОЗП використовують ресурси LUT ПЛІС. Пари сусідніх LUT можна об'єднати у двопортовий ОЗП. При цьому запис виконується за однією адресою в обидві LUT, а читання – за двома різними адресами (рис. 10.14).

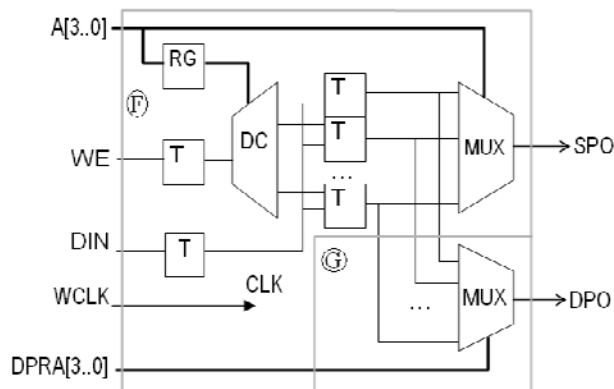


Рисунок 10.14 – Реалізація 2-портового 16-бітового ОЗП на двох сусідніх LUT (F і G)

При цьому для реалізації синхронного режиму запису вхідний біт даних, сигнал запису та адресу запам'ятовують на тригерах-засувках (Т), а для читання за іншою адресою з блока другого LUT (сегмент G на рис. 10.14) використовують тільки мультиплексор читання.

У сучасних ПЛІС ємність такого ОЗП може досягати 64 бітів. Для на-рощування ємності пам'яті виходи декількох CLB з модулями ОЗП об'єднують через мультиплексори. При цьому необхідні додаткові апаратні витрати для побудови пристрою дешифрування адреси, яка передає сигнали LUT на входи мультиплексора. Таким чином, оперативна пам'ять виявляється розподіленою по площині кристала і тому має назву розподілений ОЗП (англ. Distributed RAM).

Для зберігання масивів даних та для їх буферизації в ПЛІС використовують від десятків до тисяч окремих блоків ОЗП (англ. Block RAM). Кожний такий блок в ПЛІС фірми Xilinx насправді є двопортовою пам'яттю ємністю 36 Кбітів. Розрядність даних може бути встановлена від 1 до 36 розрядів. Наприклад, в ПЛІС від Xilinx серії Virtex блоковий ОЗП може бути налаштованим як ОЗП ємністю 256 16-розрядних слів або 512 8-розрядних слів і т. д. Ця пам'ять може бути запрограмована як однопортова або як повністю двопортова пам'ять. Початковий стан цієї пам'яті може задаватися при її конфігуруванні, тому вона може виконувати і роль ПЗП.

На базі LUT також можна реалізувати 16-розрядний регістр зсуву з од-норозрядним входом і програмовним номером вихідного розряду, що від-повідає структурі пам'яті FIFO змінної довжини.

Обчислювальні засоби на ПЛІС знаходять широке застосування завдя-ки тому, що їх можна легко інтегрувати до більшості проектів цифрових засобів та, крім того, ними досить просто можна замінювати пристрої, які ре-алізовані на застарілій елементній базі. Таке інтегрування реалізується за допомогою великої кількості блоків введення/виведення (IOB), які налаш-товують під різні стандарти електричного з'єднання входів мікросхем. На рис. 10.15 показано структуру одного IOB.

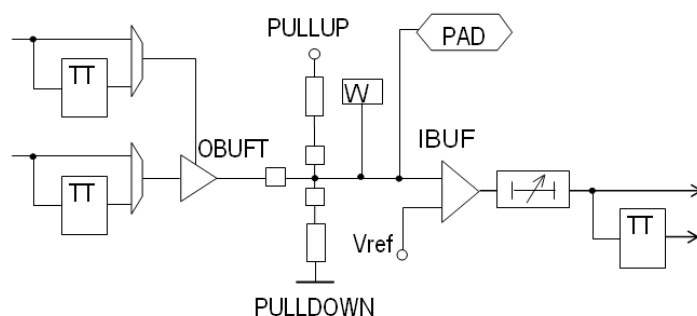


Рисунок 10.15 – Структура IOB

Сигнальний вивід ПЛІС отримав назву PAD. До нього можна під'єднувати внутрішній навантажувальний резистор PULLUP або резис-тор PULLDOWN, які з'єднані з шиною живлення або шиною землі, відпо-

відно. Ці резистори забезпечують режим виходу з відкритим колектором (стоком) для систем з різними рівнями сигналів. Вхідний сигнал з PAD надходить на компаратор IBUF, поріг спрацьовування якого можна налаштувати на рівні TTL, КМОН, шини PCI та багатьох інших, а також можна змінювати шляхом регулювання зовнішньої напруги.

Вихідний сигнал в IOB формується в трирівневому буфері OBUFT, причому рівень струму в навантаженні програмується ступінчасто і може досягати 20 мА. Для забезпечення високої швидкодії введення/виведення біти даних та сигнал управління трирівневим буфером запам'ятовують на тригерах.

Для діагностики, налагодження та, в одному з режимів, конфігурування, ПЛІС може бути перемкнута переключена в так званий режим граничного сканування (Boundary Scan). У цьому режимі всі IOB з'єднуються в ланцюжок одного довгого регістра зсуву. Керуючи цим регістром зсуву через інтерфейс JTAG можна зчитувати стан виводів, подавати тестові сигнали, конфігурувати ПЛІС.

Для програмування та конфігурування ПЛІС можуть використовуватись програматори, зовнішні ПЗП та завантажувальні кабелі. Завантажувальні кабелі можуть використовуватись при програмуванні мікросхеми, яка вже встановлена на друковану плату. Така технологія носить назву внутрішньосхемного програмування (англ. In-System programming – ISP).

При використанні зовнішніх ПЗП конфігураційна інформація зберігається в спеціалізованих мікросхемах і при вмиканні живлення завантажуються до ПЛІС. Слід відмітити, що конфігураційні ПЗП відрізняються від звичайних ПЗП тим, що, крім зберігання інформації, вони ще мають працювати з мікросхемою ПЛІС за стандартним протоколом конфігурування. Протокол залежить від режиму конфігурування. Для мікросхем ПЛІС можливі такі режими програмування:

- пасивний послідовний (англ. Passive Serial);
- асинхронний пасивний послідовний (англ. Passive Serial Asynchronous);
- пасивний паралельний (англ. Passive Parallel);
- швидкий пасивний паралельний (англ. Fast Passive Parallel);
- програмування через JTAG-інтерфейс;
- активний послідовний (англ. Active Serial).

Наприклад, для конфігурування ПЛІС серії Spartan передбачено спеціальні три входи задання одного з режимів, для виведення сигналу синхросерії програмування CCLK, вхід послідовності конфігурації PROGRAM, вихід прапорця закінчення конфігурування DONE і виводи порту JTAG. Залежно від встановленого режиму можна завантажувати прошивку ПЛІС одним із трьох способів: через однорозрядні PROGRAM, порт JTAG або 8-розрядну шину D з використанням для управління виводів WRITE та BUSY.

Конфігурування через однорозрядний вхід триває не більше 10 секунд. Це стандартний спосіб конфігурування і для нього не потрібно додаткового обладнання, крім ПЗП-прошивання з однорозрядним виходом.

Для реалізації конфігурування через шину D необхідно додаткове зовнішнє обладнання (автомат або мікроконтролер), що керує записом і подає послідовність адрес для читання на входи 8-розрядного ПЗП. Цей режим забезпечує швидкість конфігурування, що сумірна з максимальним часом читання даних із ПЗП.

Блоки множення з накопиченням практично стали обов'язковою частиною більшості ПЛІС. Наприклад, DSP48 виконують множення 18×18 або 18×25 з акумулюванням результату в 48-розрядному швидкодійному акумуляторі. Крім множення з накопиченням на цих блоках рекомендується реалізувати блок зсуву, швидкодійні суматори. Для роботи з високою тактовою частотою, що досягає 400–700 МГц, рекомендується ці блоки використовувати у конвеєрному режимі обробки інформації.

10.3 Сучасні тенденції розвитку пристроїв із програмовною логікою

Однією з сучасних тенденцій використання ПЛІС є створення таких класів цифрових пристроїв, як, наприклад, система на кристалі – SoC (System-on-a-Chip). SoC – це спеціалізований цифровий виріб, що конструктивно виконаний на кремнієвій підкладці і який характеризується низьким енергоспоживанням, малими розмірами, високою швидкодією. Система на кристалі має такі основні компоненти: процесор, пам'ять, логіку, інтерфейс введення/виведення, перетворювачі інформації, шини обміну даними та інфраструктура сервісного обслуговування. Перевагами SoC є прозорість щодо наявності вбудованих компонентів, висока швидкодія при низькому енергоспоживанні, мініатюрні розміри, ефективне використання різних типів пам'яті, висока надійність, низька вартість готового виробу та високий коефіцієнт використання його функціональностей. При проектуванні сучасних систем на кристалі широко використовують готові модулі, які називають IP-блоками, IP-core (intellectual property core) або просто IP, що реалізують досить складні функції перетворення і обробки даних. IP-блок – це складний, вже протестований, верифікований і оптимізований функціональний модуль, який може бути багато разів використаний як компонент при проектуванні більш складних цифрових виробів з метою скорочення часу їх розробки та мінімізації ризиків проектування. Це зменшує час верифікації (перевірки відповідності заданим властивостям) проекту всієї системи. Основний недолік використання IP полягає у складності інтеграції цих блоків із різних джерел.

Використання передових технологій в процесі виробництва ПЛІС привело до значного розширення їх функціональних можливостей. У першу чергу ця тенденція стосується ПЛІС з архітектурою FPGA.

Внаслідок цього, в процесі проектування цифрових пристроїв ПЛІС може розглядатися як деяке мікропроцесорне ядро, що оточене вільними логічними ресурсами. За способом реалізації процесорні ядра, що використовуються при проектуванні сучасних обчислювальних засобів на основі ПЛІС, поділяють на програмні (soft-процесори, soft-ядра) та апаратні (hard processor).

Програмні ядра або ядра, що конфігуруються, формуються на основі стандартних логічних ресурсів кристалів. Перевагами програмних процесорних ядер, порівняно з апаратними, є висока гнучкість, низька собівартість, відносно невеликий обсяг використовуваних ресурсів кристалів, можливість застосування в проектах, що реалізуються на базі найпоширеніших і доступних сімей ПЛІС.

Апаратні процесори – мікропроцесорні ядра, які виконані у вигляді відповідних інтегрованих апаратних блоків ПЛІС. Головна перевага апаратних мікропроцесорних ядер полягає в можливості функціонування з високими тактовими частотами. Наслідком цього є більш висока швидкодія порівняно з мікропроцесорними ядрами, що конфігуруються. Недоліками апаратних ядер є обмежена кількість моделей кристалів ПЛІС, в яких вони застосовуються, та їх висока вартість.

Кількість початкових проектів на ПЛІС, що використовують процесорні ядра, збільшується з кожним роком за рахунок використання багатопроцесорної організації (розпаралелювання обчислювальних процесів), розширення сфер застосування ПЛІС, поступового витіснення середовищами з програмовною структурою традиційних замовних мікросхем тощо.

Класифікація ПЛІС за рівнем інтеграції відбиває, головним чином, сучасні тенденції, а саме: зростання рівня інтеграції ПЛІС і виділення з них класу «системи на кристалі».

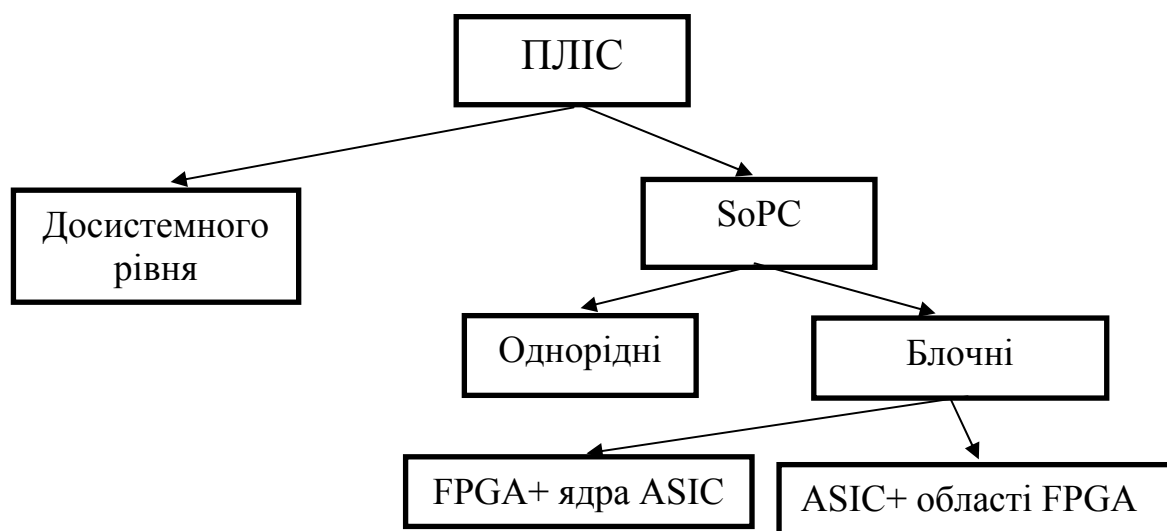


Рисунок 10.16 – Класифікація ПЛІС за рівнем інтеграції

До ПЛІС «досистемного» рівня відносять ті, для яких не розглядається питання створення на них завершених SoC.

Напівфабрикати класу «система на програмовному кристалі» (англ. System on Programmable Chip, SoPC) належать до ПЛІС найвищого рівня інтеграції. На таких ПЛІС можлива побудова повної високопродуктивної обчислювальної системи. В літературі поряд із терміном SoPC часто зустрічаються терміни SoC та PSoC (англ. Programmable System on Chip). Взагалі прийнято використовувати загальне поняття «цілісна вбудована система, що може бути розміщена на одному кристалі», і для цього поняття використовують одну з абревіатур SoC, SoPC або PsoC.

Клас SoPC ділиться на підкласи однорідних та блокових систем на кристалі. В однорідних SoPC різні блоки системи реалізуються завдяки програмуванню складових своєї ж системи. При розробці таких систем використовуються IP. Блокові SoPC можуть містити апаратні ядра, які спроектовані за методологією ASIC. Ці ядра насправді є спеціалізованими ділянками кристала, які виділені для певних функцій. З іншого боку, існують блокові SoPC, що являють собою спеціалізовану платформу, тобто ASIC, із ділянками програмовної логіки. Відповідно, в кожному з цих видів ПЛІС може міститися або бути відсутнім процесорне ядро.

10.4 Обчислювальні системи на ПЛІС із soft-процесором

Далі розглядаються архітектурні особливості ядра RISC-мікропроцесора на ПЛІС. Цей центральний процесорний елемент (ЦПЕ) є 16-розрядним софт-ядром, що призначене для конфігурування ПЛІС фірм Xilinx, Altera або інших. На даний час найбільшого поширення набули софт-ядра мікропроцесорів з розрядністю 8 і 32. Ядра з розрядністю 8 застосовуються, в основному, як скінченні автомати управління з невеликою кількістю станів (наприклад, ядро Picoblaze фірми Xilinx). Ядра з розрядністю 32 (наприклад, софт-процесорні ядра Microblaze тієї ж фірми), застосовуються, в основному, як керувальні ЦПЕ, що функціонують під управлінням операційної системи реального часу (наприклад Linux). До сфери застосувань такого софт-ядра можна віднести обслуговування файлової системи, реалізацію стека протоколів Ethernet, обробку складних запитів користувача, виведення інформації на монітор, реалізацію складних алгоритмів. Деякі софт-процесорні ядра надаються користувачам фірмами-виробниками ПЛІС як IP-блоки.

До особливостей архітектури RISC-процесорів відносять: використання спрощеного формату команд, конвеєризацію (концепцію, що розбиває процес обробки машинної команди на декілька практично незалежних етапів, які потім можна суміщати в часі для декількох команд у відповідній апаратурі (конвеєрі команди), виконання інших команд під час виконання команди переходу, використання регістрової пам'яті. Для роботи в реальному часі процесор має високу швидкодію не тільки за рахунок одноктакто-

вого виконання команд, але і завдяки апаратно реалізованому стеку протоколів та системі переривань. Розрядності 16 звичайно достатньо для вирішення задач, що базуються на застосуванні SoC.

Характерною особливістю такого софт-процесора є те, що за його допомогою вирішують задачі, для яких не потрібна реалізація великої кількості логічних переходів та складних маніпуляцій з даними. Такими задачами можуть бути:

- початкове встановлення параметрів і динамічне управління віртуальними модулями, встановленими в ПЛІС;
- запуск завдань у віртуальних модулях, сконфігурованих в ПЛІС і реакція на закінчення їх виконання;
- адаптація затримок вхідних сигналів для стійкого прийому швидкісних сигналів від зовнішніх пристроїв;
- розрахунок математичних функцій, які використовуються як параметри для віртуальних модулів, що встановлені в ПЛІС;
- прийом і передача структурованих даних через такі послідовні порти введення/виведення, як RS232, CAN, I2C, SPI, USB;
- організація передачі даних через інтерфейси SATA, PCIe, DVI, HDMI;
- ущільнення та розпаковування даних за алгоритмами GZIP, LZW, GIF, кодів Хаффмана та ін.;
- забезпечення пересилання даних між блоками пам'яті;
- обробка переривань від зовнішніх джерел, більшість яких є віртуальними модулями;
- реалізація алгоритмів, що вимагають непрямої адресації, наприклад, обробка деревоподібних структур даних, ведення таблиць, словників;
- нескладні операції з файлами, що зберігаються у зовнішній пам'яті;
- реалізація простих універсальних і спеціалізованих протоколів Ethernet;
- реалізація багатопроцесорної обробки в обчислювальній системі.

Розглядуваний ЦПЕ має такі основні характеристики:

- розрядність даних та шини адреси команд – 16;
- розрядність команд – 18;
- розрядність шини адреси даних – 24;
- розрядність шини адреси регістрової пам'яті – 4.

Крім того, структура ЦПЕ на ПЛІС має задовольняти такі вимоги:

- розрядність шини адреси даних може бути збільшена до 32 додаванням регістра сторінок;
- дані адресуються байтами, а команди адресуються словами;
- пам'ять даних розташовується частково на ПЛІС, а частково – в зовнішній, щодо ПЛІС, пам'яті;
- під час затримки при доступі до пам'яті процесорне ядро зупиняється, доки не прийде сигнал готовності від зовнішнього джерела;

- дані адресуються побайтово;
- пам'ять даних під'єднується через стандартний інтерфейс шини (до цього інтерфейсу також під'єднуються інші повільні периферійні пристрої та процесори);
- пам'ять програм завжди знаходиться в кристалі ПЛІС; доступ до пам'яті програм реалізується послідовним каналом; команди адресуються послівно;
- адресний простір регістрів периферійних пристроїв складає 256 адрес; регістри периферійних пристроїв адресуються послівно;
- стек, який зберігає адреси повернення і прапорці, має обсяг від 32 до 128 і реалізований апаратно;
- до регістрової пам'яті обсягом 32 байти можливий доступ в режимі читання за трьома адресами, а доступ в режимі запису – тільки за однією;
- одночасно адресуються і використовуються лише 16 регістрів: R0,..., R7, R8, ..., R15.

ЦПЕ функціонує у трьох режимах: нормальний режим роботи, режим з альтернативними регістрами і режим переривання. Для цих трьох режимів загальними є регістри R0, ..., R7. А в режимі переривання замість регістрів R8, ..., R15 підставляються регістри R16, ..., R23. У режимі з альтернативними регістрами підставляються R24, ..., R31. Для всіх режимів старші регістри адресуються як R8, ..., R15. Перемикання груп регістрів спрощує передачу параметрів і організацію тимчасового зберігання даних.

Умовні переходи виконуються в межах адрес ± 127 від поточної команди залежно від стану прапорців регістру стану (англ. Status Register, SR).

Система переривань передбачає обробку переривань від 15 джерел, прапорці яких запам'ятовуються в регістрі IR (англ. Interrupt Register, IR). Периферійний регістр маски переривань IMR (англ. Interrupt Mask Register, IMR) зберігає маску переривань. Пріоритети переривань фіксовані. Таблиця векторів переривань знаходиться за адресами 2, ..., 31. В ній містяться адреси переходу до підпрограм обслуговування переривання або дві команди, з яких остання – команда повернення з підпрограми переривання зі скиданням прапорця переривання. Доступ до регістрів SR, IR, IMR виконується як до регістрів периферійних пристроїв. Команди переходів та команди, які виконуються більше ніж за 1 такт (наприклад, MUL) не мають перериватися. Число стадій конвеєра – 3.

Система команд містить типову для процесора з RISC-архітектурою кількість команд. Це дозволяє спростити систему дешифрування команди та, відповідно, збільшити тактову частоту. Кожна команда, крім команд переходу, виконується за 1 такт. Команда переходу виконується за 2 такти.

Ще одна архітектурна особливість полягає в тому, що при виконанні переходів конвеєр очищується, тобто команда, що стоїть за командою переходу, не виконується, а її результати не зберігаються. Це необхідно для спрощення програмування на рівні асемблера.

При виконанні двох підряд команд звертання до пам'яті даних (запису або читання) конвеєр не зупиняється. Для унеможливлення колізій потрібно між ними вставляти інші команди, як наприклад, пусту команду (NOR), а також стежити за блокуваннями при перериванні. Щоб нівелювати такі недоліки необхідно, щоб кількість ступенів конвеєра була не більше 3.

Для реалізації зупинок процесора в точках зупинки та зчитування стану процесора через стандартну послідовну шину введення/виведення типу UART або I2C застосовуються засоби для відлагодження, що можуть бути під'єднані при синтезі системи.

Структура конвеєра процесора показана на рис. 10.17. На цій схемі позначено:

- PC – лічильник команд (program counter);
 - ROMI – ПЗП команд (instruction ROM);
 - RGI – регістр команди (instruction register), INSTR_R ;
 - OPC – код операції (operation code);
 - RAMD – ОЗП даних (data RAM);
 - RRAM – регістровий ОЗП (registered RAM);
 - DCI – дешифратор команди (instruction DC);
 - MI – мікрокоманда (microinstruction);
 - RGC – регістр константи (CONST register);
 - RGS – регістр стану (state register);
 - Cond – умова переходу (jump condition);
 - DIP – дані в периферійний регістр (input data for peripheral register);
 - DOP – дані з периферійного регістра (output data from peripheral register);
 - AP – адреса периферійного регістра (address of peripheral register);
 - ADR_O, DAT_O, DAT_I – сигнали інтерфейсу;
 - RP – регістр сторінки (page register).
- Арифметико-логічний пристрій (АЛП, англ. ALU) містить регістри:
- RSH – регістр зберігання другої частини зсунутого даного (register for shift);
 - RH – регістр зберігання старшої частини добутку (product high part).
- Процес виконання команд у конвеєрі ЦПЕ можна описати так. Перший такт командного циклу передбачає нижчевказані дії:
- видача адреси з PC, RD або зі стека;
 - видача прапорців з RGS;
 - формування адреси нової команди і подача її на ROMI;
 - закінчення запису результату в RAMD.
- В другому такті командного циклу відбувається зчитування команди з ROMI та її запис в RGI.

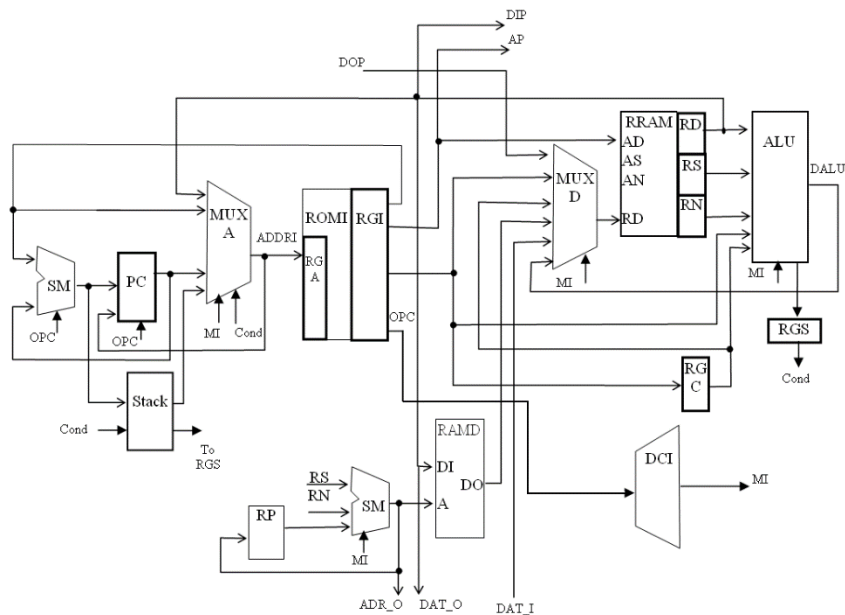


Рисунок 10.17 – Структура конвеєра ЦПЕ

Під час третього такту командного циклу відбувається:

- видача команди з RGI;
- дешифрування OPC;
- подача адрес на RRAM;
- визначення адреси переходу і запис її в RGNA, зберігання адреси в стеку (CALL);
- зчитування операндів з RRAM;
- зчитування з RAMD та периферійного регістра;
- видача адреси з RRAM на RAMD;
- операція ALU;
- запис результату в RRAM та в периферійний регістр;
- запис прапорців в RGS.

Четвертий та інші такти командного циклу:

- запис/читання з RAMD при ACK_I=0/ACK_O=0, відповідно;
- закінчення операцій, що виконуються в спеціалізованих співпроцесорах.

Із вищевикладеного видно, що перший і другий такти сусідніх команд перекриваються, крім випадку, коли поточна команда – команда переходу, або периферійні пристрої видають сигнали затримки операції ACK_I = 0 або ACK_O = 0.

Мікропроцесор складається з мікропроцесорного ядра, ОЗП даних, ОЗП програм і периферійних пристроїв. Мікропроцесорне ядро (рис. 10.18) складається з модулів лічильника команд (PC), регістрової пам'яті (RRAM), арифметико-логічного пристрою (ALU) і контролера перериваннями (INT_CNTRL). Опис цих структур виконується мовою VHDL, що дає змогу ввести проект в САПР ПЛІС та скомпілювати його у прошивку ПЛІС.

10.5 Контрольні запитання і завдання

1. Навести класифікації програмовних логічних інтегральних схем (ПЛІС).
2. В чому полягає відмінність програмовних логічних матриць (ПЛМ) від програмовних матриць логіки (ПМЛ)? Навести приклад структур ПЛМ та ПМЛ.
3. ПЛІС якого типу стала продовженням архітектур ПЛМ та ПМЛ? Навести приклади застосування цих мікросхем програмовної логіки?
4. Охарактеризувати такий пристрій з програмовною структурою як базовий матричний кристал.
5. Інтегральні мікросхеми з програмовною логікою типу FPGA. Структура, властивості та особливості.
6. Софт-процесорні ядра на ПЛІС. Основні характеристики, особливості проектування та застосування.

СПИСОК ЛІТЕРАТУРИ

1. Лехин С. М. Схемотехника ЭВМ / С. М. Лехин. – СПб. : БХВ-Петербург, 2010. – 672 с.
2. Бабич М. П. Комп'ютерна схемотехніка : навч. пос. / М. П. Бабич, І. А. Жуков. – К. : «МК-Прес», 2004. – 412 с.
3. Компьютерная схемотехника (краткий курс) / Процюк Р. О., Корнейчук В. И., Кузьменко П. В., Тарасенко В. П. – К. : «Корнійчук», 2006. – 433 с.
4. Гикавий В. А. Цифрова і аналогова схемотехніка : лабораторний практикум / В. А. Гикавий. – Вінниця : ВДТУ, 2001. – 99 с.
5. Бабич Н. П. Компьютерная схемотехніка. Методы построения и проектирования : учебное пособие / Н. П. Бабич, И. А. Жуков. – К. : МК-Пресс, 2004. – 576 с.
6. Угрюмов Е. П. Цифровая схемотехніка : учебное пособие / Угрюмов Е. П. – [2-е изд., перераб. и доп.]. – СПб. : БХВ-Петербург, 2007. – 800 с : ил.
7. Рябенський В. М. Цифрова схемотехніка : навчальний посібник / Рябенський В. М., Жуйков В. Я., Гулий В. Д. – Львів : Новий Світ, 2000, 2009. – 736 с.
8. Білінський Й. Й. Цифрова схемотехніка : навч. посібник. Ч. 1. Базові поняття цифрової схемотехніки / Білінський Й. Й., Гикавий В. А., Мельничук А. О. – Вінниця : ВНТУ, 2011. – 133 с.
9. Бойт К. Цифровая электроника / К. Бойт ; пер. с нем. М. М. Ташлицкого. – М. : Техносфера, 2007. – 472 с.
10. Цифровая схемотехника и архитектура компьютера / Д. М. Харрис, С. Л. Харрис. – Morgan Kaufman, 2013. – 1662 с.
11. Теоретичні основи комп'ютерних напівпровідникових електронних компонентів : навч. пос. / Азаров О. Д., Гарнага В. А., Сапсай Т. Г., Тарасенко В. П. – Вінниця : ВНТУ, 2015. – 134 с.
12. Азаров О. Д. Комп'ютерна електроніка. Елементи цифрових : навч. пос. / Азаров О. Д., Байко В. В., Обертюх М. Р. ; під загальною редакцією О. Д. Азарова. – Вінниця : УНІВЕРСУМ-Вінниця, 2003. – 170 с.
13. Азаров О. Д. Комп'ютерна електроніка : лабораторний практикум. / Азаров О. Д., Байко В. В., Крупельницький Л. В. – Вінниця : ВДТУ, 2001. – 111 с.
14. Азаров О. Д. Комп'ютерна електроніка. Частина 2. Елементи цифрових схем : навчальний посібник / Азаров О. Д., Байко В. В., Обертюх М. Р. ; під загальною редакцією доктора технічних наук, професора О. Д. Азарова. – Вінниця, 2002. – 170 с.
15. Азаров О. Д. Комп'ютерна електроніка. Основи теорії транзисторів та транзисторних схем : навч. посібник / Азаров О. Д., Суприган О. І., Байко В. В. – Вінниця : ВНТУ, 2004. – 128 с.

16. Методичні вказівки до виконання курсової роботи з курсу «Комп'ютерна електроніка» для студентів бакалаврського напрямку 6.0915 «Комп'ютерна інженерія» спеціальності 7.091501 «Комп'ютерні системи та мережі» ступеневої підготовки спеціалістів з вищою технічною освітою / Уклад. : Азаров О. Д., Байко В. В., Крупельницький Л. В. – Вінниця : ВДТУ, 2001. – 46 с.

17. Методичні вказівки до виконання курсового проекту з дисципліни «Комп'ютерна електроніка» для студентів напряму підготовки 0915 «Комп'ютерна електроніка» / Уклад. : Азаров О. Д., Байко В. В., Крупельницький Л. В. – [2-ге вид., доповнене] – Вінниця : ВНТУ, 2004. – С. 1–75.

Навчальне видання

**Азаров Олексій Дмитрович
Гарнага Володимир Анатолійович
Клятченко Ярослав Михайлович
Тарасенко Володимир Петрович**

Комп'ютерна схемотехніка

Підручник для студентів ВНЗ

Рукопис оформлено В. Гарнагою

Науковий редактор В. Тарасенко

Редактор В. Дружиніна

Оригінал-макет виготовлено О. Ткачуком

Підписано до друку 19.06.2018.

Формат 29,7×42¹/₄. Папір офсетний.

Гарнітура Times New Roman.

Друк різнографічний. Ум. друк. арк. 13,8.

Наклад 500 (1-й запуск 1–50) пр. Зам. 2018-124.

Видавець та виготовлювач
інформаційний редакційно-видавничий центр.

ВНТУ, ГНК, к. 114.

Хмельницьке шосе, 95,

м. Вінниця, 21021.

Тел. (0432) 65-18-06.

press.vntu.edu.ua;

E-mail: kivc.vntu@gmail.com.

Свідоцтво суб'єкта видавничої справи
серія ДК № 3516 від 01.07.2009 р.