



**МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”
Кафедра електричних мереж та систем**

МІКРОПРОЦЕСОРНА ТЕХНІКА

Навчальний посібник

для студентів усіх форм навчання та студентів-іноземців напряму
підготовки 6.050701 “Електротехніка та електротехнології”

Затверджено методичною радою НТУУ «КПІ»

**Київ
«Політехніка»
2014**

Мікропроцесорна техніка: Навчальний посібник з дисципліни для всіх форм навчання та студентів іноземців напряму підготовки 6.050701 “Електротехніка та електротехнології”/Уклад. В.В.Кирик.-К.: ІВЦ «Видавництво «Політехніка», 2014.-183с.

В посібнику представлена ідеологія побудови мікропроцесорних систем на основі сучасних досягнень мікроелектроніки та інформаційно-комунікаційних систем для використання в електроенергетиці. Детально розглянуто функціональні елементи цифрової техніки, структура і архітектура мікропроцесорних систем, використання мікроконтролерів в електротехнічних системах та принципи побудови інформаційно – діагностичних та інформаційно – управляючих систем в електроенергетиці.

Для студентів усіх форм навчання та студентів-іноземців напряму підготовки 6.050701 “Електротехніка та електротехнології”

В пособии представлена идеология построения микропроцессорных систем на основе современных достижений микроэлектроники и информационно-коммуникационных систем для использования в электроэнергетике. Детально рассмотрены функциональные элементы цифровой техники, структура и архитектура микропроцессорных систем, использование микроконтроллеров в электротехнических системах и принципы построения информационно–диагностических и информационно – управляющих систем в электроэнергетике.

Для студентов всех форм обучения и студентов-иностранцев направления подготовки 6.050701 “Электротехника и электротехнологии”

Укладач: *В.В. Кирик, д-р техн. наук, професор*

Відповідальний редактор *Т.Л.Кацадзе, канд.техн.наук, доц.*

Рецензенти
*О.Д.Подольцев, д-р т.н., ст наук.с.
В.В.Гавриленко, д-р ф-м.н., проф.
М.Ф.Сопель, канд.техн. наук, ст.наук.с.*

СИСТЕМА ОЦІНЮВАННЯ ЗНАНЬ

Одиниця – Студент тільки-но наблизився до знань, по своєму недоліку природних здібностей чи тому, що зовсім не старався, маючи тягу до чого не будь другого;

Двійка – Студент фрагментарно володіє знаннями з дисципліни, але і ті зафіксував своєю пам'яттю, не маючи конспекту. Не проник в основи і взаємозв'язок складових частин, які формують єдине ціле дисципліни;

Трійка – Студент знає дисципліну в тому вигляді, в якому вона була йому викладена на аудиторних заняттях, тобто обмежується конспектом або словами викладача. При цьому вивчене використовує він з трудом та великими потугами;

Чотири – Студент чітко володіє знаннями наданими на аудиторних заняттях, уміє пояснити всі частини дисципліни, розуміє їх взаємозв'язок і легко використовує надбані навички на практиці. Студент приймає за неможливе вивчити що не будь не розуміючи;

П'ять – Студент володіє знаннями з дисципліни, уміє легко пояснити всі частини дисципліни, взаємно пов'язує любі точки дисципліни, з глибоким проникненням розбирає нові або складні моменти, знає слабкі місця теорії і може формулювати певні свої бачення.

ЗМІСТ

Перелік умовних посилань та скорочень.....	6
Передмова.....	9
1. Загальні положення, ідеологія побудови мікроелектронних і мікропроцесорних систем.....	10
1.1 Технології створення та функціонування електронної техніки.....	10
1.2 Загальні відомості, функціонування та використання цифрових мікропроцесорних систем.....	14
1.3 Особливості мікропроцесорних систем.....	22
2. Функціональні елементи цифрової техніки.....	29
2.1 Дискретні електронні пристрої цифрової техніки.....	29
2.2 Основні операції над логічними змінними та їх реалізація.....	34
2.3 Спеціальні цифрові схеми – асинхронні тригери.....	40
2.4 Синхронні тригери.....	42
2.5 Генератори імпульсів в мікропроцесорних системах.....	46
3. Представлення та дії над двійковими числами в мікропроцесорних систем.....	50
3.1 Представлення чисел в цифрових системах.....	50
3.2 Основні дії над двійковими числами.....	54
3.3 Комбінаційні пристрої в мікропроцесорних системах.....	56
3.4 Послідовні пристрої в мікропроцесорних системах.....	61
3.5 Цифро-аналогові і аналого-цифрові перетворювачі.....	67
4. Функціональні структури і архітектура мікропроцесорних систем.....	72
4.1 Структурна схема мікропроцесора.....	72
4.2 Типова структура мікропроцесорної системи.....	78
4.3 Ідеологія побудови мікропроцесорних систем.....	83
4.4 Функціональні структури і архітектура комп'ютерних систем в електроенергетиці.....	91
5. Мікроконтролери в електротехнічних системах.....	95
5.1 Основні визначення та класифікація мікроконтролерів.....	95
5.2 Архітектура мікроконтролерів.....	98
5.3 Мікроконтролери сімейства MCS.....	105
5.4 Мікроконтролери сімейства AVR.....	108
5.5 Мікроконтролери сімейства PIC.....	113
5.6 Промислові контролери.....	115
6. Інформаційно - діагностичні та інформаційно -управляючі	

систем в електроенергетиці...../.....	122
6.1 Принципи організації МПС в електроенергетиці.....	122
6.2 Класифікація вхідної інформації при розробці інформаційного забезпечення МПС.....	128
6.3 Принципи побудови мікропроцесорних інформаційно-управляючих систем (ІУС) в електроенергетиці.....	135
6.4 Принципи побудови інформаційно-діагностичних комплексів.....	142
ДОДАТОК А	
Автоматизована мікропроцесорна система комерційного обліку електроенергії.....	150
ДОДАТОК Б	
Оперативно-інформаційний комплекс автоматизованої мікропроцесорної системи диспетчерського керування на базі scada-системи трейс моуд.....	160
Рекомендована література.....	182

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

АВР	– автоматичне введення резерву
АК	– аналоговий комутатор
АЛП	– арифметико-логічний пристрій
АРМ	– автоматизоване робоче місце
АРН	– автоматичне регулювання напруги
АРЧП	– автоматичне регулювання частоти і перетоків потужності
АСДК	– автоматизована система диспетчерського керування
АСКОЕ	– автоматизована система комерційного обліку електроенергії
АЦП	– аналогово-цифровий перетворювач
БД	– база даних
ВОК	– волоконно-оптичні канали
ВОЛЗ	– волоконно-оптична лінія зв'язку
ВП	– виконавчий пристрій
ГІ	– генератор імпульсів
ГТІ	– генератор тактових імпульсів
ДП	– диспетчерський пункт
ДТЛ	– діодно-транзисторна логіка
ЕЕО	– електроенергетичний об'єкт
ЕЕС	– електроенергетична система
ЕЗЛ	– емітерно-зв'язана логіка
ЕОМ	– електронна обчислювальна машина
ЕС	– енергосистема
ЗОІ	– система збору і обробки інформації
ЗП	– запам'ятовуючий пристрій
ІМС	– інтегральні мікросхеми
ІУС	– інформаційно-управляючі системи
К	– компаратор
КОП	– код операції
КП	– комірка пам'яті
КР	– керуючий регістр
КТЗ	– комплекс технічних засобів
МДМ	– модем (модулятор-демодулятор)
МК	– мікроконтролери
МП	– мікропроцесор
МПС	– мікропроцесорна система

МРЧ	– монітор реального часу
НВІС	– надвелика інтегральна схема
НД	– нормативні документи
ОЗП	– оперативний запам'ятовуючий пристрій
ОІК	– оперативно-інформаційний комплекс
ОК	– оптичні кабелі
ОРЕ	– оптовий ринок електроенергії
П	– електронний підсилювач
ПД	– пам'ять для даних
ПЕОМ	– персональна електронна обчислювальна машина
ПЗП	– постійний запам'ятовуючий пристрій
ПК	– персональний комп'ютер
ПрК	– пристрій керування
ПЛ	– програмний лічильник
ПЛУ	– програмно-логічне управління
ПрП	– пристрої пам'яті
ПП	– погоджуючий пристрій
ППЗП	– перепрограмований постійний запам'ятовуючий пристрій
ПС	– пристрій сполуки
ПТМ	– пристрій телемеханіки
РЗП	– реєстри загального призначення
РКІ	– рідинно-кристалевий індикатор
РОО	– рівень об'єктів обліку електроенергії
РПД	– резидентна пам'ять даних
РСП	– реєстри спеціального призначення
РТЛ	– резистивно-транзисторна логіка
РПП	– резидентна пам'ять програм
РТО	– рівень точок обліку електроенергії
РЦП	– рівень центрального пункту
ТВ	– телевимірювання
ТН	– трансформатор напруги
ТО	– технологічний об'єкт
ТСг	– телесигналізація
ТС	– трансформатор струму
ТТЛ	– транзисторно-транзисторна логіка
ЦАП	– цифро-аналоговий перетворювач
ЦАУ	– система цифрового автоматичного управління
ЦДС	– центральна диспетчерська система

ША	– шина адреси
ШД	– шина даних
ШУ	– шина управління
ШИМ	– широтно-імпульсний модулятор
ARM	– Advanced RISC – machine
ASIC	– application-specific integrated circuit, інтегральна схема для специфічного використання
CAN	– Controller Area Network
CISC	– Complex Instruction Set Commands
COTS	– Commercial Off-The-Shelf
DMI	– Direct Media Interface
DWDM	– Dense Wavelength Division Multiplexing
EEPROM	– Electrically Erasable Programmable Read-Only Memory
FSB	– Front side bus
GPIO	– General Purpose Input/Output
HT	– Hyper-Threading
IC	– Integrated Circuit
PIC	– Peripheral Interface Controller
QPI	– Quick Path Interconnect
RISC	– Reduce Instruction Set Commands
Soc	– System-on-Crystal, система на кристалі
UART	– Universal Asynchronous Receiver/Transmitter
WDM	– Wavelength Division Multiplexing

ПЕРЕДМОВА

Електроенергетика є базовою галуззю народного господарства з широким використанням засобів автоматизації на основі сучасної мікропроцесорної техніки. В електроенергетиці намітилися тенденції до створення так званої «розумної мережі» (Smart Grid) з поступовим формуванням на ряду з силовою системою інформаційно-комунікаційної мережі, яка призвана надати галузі новий стан для забезпечення значно вищих показників якості електроенергії та надійності електропостачання.

На сьогодні мікропроцесорні системи докорінно змінили як засоби автоматизації, так і технології створення електронної техніки, підняли суттєво технологічну культуру та рівень знань розробників нової техніки, користувачів і ремонтного персоналу.

Уміння раціонально використовувати мікропроцесорну техніку стало сьогодні самим важливим показником рівня кваліфікації сучасного спеціаліста.

В зв'язку з цим спеціалісти в галузі електротехніки, електричних мереж та систем повинні знати не лише можливості універсальних мікропроцесорних пристроїв, комп'ютерів, але і володіти знаннями архітектури, принципів роботи та мати навички використання програмного забезпечення інформаційних систем керування.

Дисципліна «Мікропроцесорна техніка» націлена на надання студенту знань, вмінь та навичок саме в галузі використання нових засобів мікропроцесорної та обчислювальної техніки при передачі, перетворенні, регулюванні, розподілі та споживанні електричної енергії. При цьому неминучим є системний підхід до вивчення функціонуванні об'єкту та місця в мережі мікропроцесорних систем, як однієї з підсистем енергетики.

Посібник налічує 6 розділів в яких враховано досвід викладання навчального курсу «Мікропроцесорна техніка» на кафедрах «Електричні мережі та системи» і «Електричні станції» Національного технічного університету «Київський політехнічний інститут».

Автор щиро вдячний рецензентам – головному науковому співробітнику ІЕД НАНУ О.Д.Подольцеву, професору НТУ В.В.Гавриленку та заступнику директора фірми «Анігер» М.Ф.Сопелю за цінні поради та пропозиції щодо змісту навчального посібника.

Розділ перший

ЗАГАЛЬНІ ПОЛОЖЕННЯ, ІДЕОЛОГІЯ ПОБУДОВИ МІКРОЕЛЕКТРОННИХ І МІКРОПРОЦЕСОРНИХ СИСТЕМ

1.1 Технології створення та функціонування електронної техніки

Електронна техніка стрімко увійшла в повсякденне життя та діяльність людини в вигляді офісних, промислових комп'ютерів, потужних електронно-обчислювальних машин та в вигляді контролерів і мікроконтролерів, які вбудовані на сьогодні практично у всі побутові прилади та промислові установки і виконують функції управління, контролю, захисту та діагностики.

Мікропроцесорні системи докорінно змінили технології створення електронної техніки, підняли суттєво технологічну культуру та рівень знань розробників нової техніки, користувачів і ремонтного персоналу.

Така ситуація позначилась і на самих мікропроцесорних системах, які сьогодні базуються на напівпровідникових мікросхемах, що створюються на основі нанотехнологій. На сьогодні електронна промисловість освоїла 25÷35 нанометрові процеси, при яких товщина ізоляційного прошарку в транзисторах знаходиться на рівні 1 нм, що складає по товщині близько п'яти атомів водню.

Вихід на рівень нанотехнологій призвів до докорінної зміни фізики процесів, при цьому класичні закони, як виявилось, не працюють на цьому рівні.

Сучасні електронні пристрої потребують елементної бази, яка відповідає таким вимогам як: висока швидкість обробки даних,

компактність і низьке енергоспоживання. Об'єднати ці вимоги дозволяє методологія Soc (System-on-Crystal, система на кристалі).

З комерційної точки зору, ринок Soc-мікросхем можна розділити на два сегменти. Один утворюють ті програмовані мікросхеми, що практично замінили класичні ASIC (application-specific integrated circuit, «інтегральна схема для специфічного використання») з жорстко детермінованою логікою у великих OEM- проектах. Другий утворюють комерційні (COTS, Commercial Off-The-Shelf) Soc-процесори, що поєднують у своїй конструкції ядра на основі відкритих процесорних архітектур або архітектур, що стали стандартами де-факто, а також набір апаратних прискорювачів або спеціалізованих співпроцесорів і схемотехніку стандартних базових інтерфейсів для обміну даними і підключення стандартних периферійних пристроїв.

Такі процесори дозволяють здійснювати розробку як вбудованих систем для промислових застосувань, так і електроніки в інтересах масового споживчого ринку. При цьому важливою перевагою процесорів Cots-класу в промислових розробках є забезпечення сучасних характеристик у частині масогабаритних параметрів і енергоспоживання, а при розробках електроніки споживчого класу — можливість прискорення виводу кінцевих виробів на ринок за рахунок економії часу на створення апаратної платформи. Величезну допомогу розробникам сучасних систем автоматизації створює також «масована» програмна підтримка виробниками своєї мікропроцесорної бази.

Концепція Soc використана корпорацією Intel для розширення експансії своїх x86-процесорів на ринок вбудованих систем. В 2008 р. Intel вивела на ринок Soc-Процесор EP80579 Integrated Processor (Intel EP80579) на основі ядра Pentium M.

Периферія «промислового класу» (три MAC-Інтерфейси Gigabit Ethernet, два інтерфейси CAN (Controller Area Network, диференційної пари) безліч портів GPIO (General Purpose Input/Output) і два контролери UART, Universal Asynchronous Receiver/Transmitter) і наявність у сімействі Intel EP80579 процесорів з розширеним температурним діапазоном - 40...85 °С відкриває широкі перспективи для створення на основі цих процесорів компактних материнських плат промислового рівня для систем автоматизації електроенергетичних процесів..

На відміну від ринку процесорів для персональних комп'ютерів і серверів, де представлені небагато процесорних архітектур, на ринку вбудованих (embedded) систем використовується близько півсотні архітектур мікроконтролерів. По даним аналітиків компанії Frost & Sullivan, в 2007 г, дві компанії, що займали вищі позиції в рейтингу ринку, поставляли на нього виробу на основі десяти різних архітектур.

Компанія Atmel, безумовний технологічний лідер на ринку мікроконтролерів, в 2010 р. представила на виставці embedded world 2010 близько двох десятків мікроконтролерів «загального» призначення на основі ядра Cortex-M3 сімейства SAM3S. При тактовій робочій частоті до 64 МГц ці мікросхеми мають питому потужність 1,45 мВт/МГц.

На сьогодні використання 32-розрядної платформи в безлічі додатків не є оптимальним навіть для досягнення високої продуктивності. Хоча сучасні 32-розрядні МК часом мають ціну, яку раніше мали 8- і 16-розрядні мікросхеми, сучасні 8- і 16-розрядні мікроконтролери також дешевшають. Якщо 32-розрядні МК можуть швидко перемножувати числа із плаваючою комою, те це ще не говорить, що подібна функціональність буде затребувана. Більш важливими можуть виявитися такі особливості 8-розрядного мікроконтролера, як напруга живлення 5 В, робота в умовах електромагнітних перешкод, можливість прямо управляти ЖК-дисплеєм,

наявність надійної пам'яті EEPROM (Electrically Erasable Programmable Read-Only Memory), струм споживання в режимі очікування на рівні наноампер, — і все це в мініатюрному корпусі площею кілька квадратних міліметрів. Ці фактори можуть виявитися вирішальними для розробника системи, яка повинна працювати від одного комплекту батарей 5-10 років.

Дані останніх років підтверджують, що 8- і 16-розрядні МК, як і раніше, цікаві замовникам, і новинки в цій ніші ринку пропонуються не тільки новими розробниками, але і лідерами промисловості.

Зростаючим ринком додатків для 8-розрядних мікроконтролерів є підтримка технології Ethernet на рівні 10/100 Мбіт/с. Інтеграція цієї технології у вбудовані системи дозволяє здійснювати їхню дистанційну діагностику, відновлення програмного забезпечення та збір даних у системах промислової і побутової автоматики. Мікросхеми на основі 8 розрядного ядра для таких додатків випускаються компаніями Microchip, Maxim і Silicon Labs.

Фірма Renesas допомагає сучасним розробникам реалізувати їхні вимоги в нових поколіннях мікроконтролерів. Один з розробників замовляє високу продуктивність, другий — щільну інтеграцію, третій — кращі характеристики флеш-пам'яті, четвертий прагне підвищити щільність коду, а п'ятий зацікавлений у розширенні можливостей. Renesas втілила всі їх побажання в сімейство RX- мікроконтролерів.

На сьогодні електронні пристрої розробляються для використання в автоматизованих системах з непростим радіоелектронним оточенням, тобто при досить суттєвих електромагнітних перешкодах (електростанції, підстанції), де вимоги до безпеки і надійності зростають прямо пропорційно складності системи. Тому з боку розробників систем, так і певних державних стандартів, висувається вимога до таких властивостей апаратного забезпечення, як: захист мікропроцесорної системи від

випадкових програмних змін, аварійних збоїв коду, порушень пам'яті, помилок у програмі і т.п. Розробка складного програмного забезпечення, як правило, виконується декількома групами або навіть компаніями. Використовуючи апаратні засоби для поділу програмних модулів, можна блокувати проблемні частини коду, відокремивши їх від іншого програмного забезпечення.

Тому виробники сучасних мікропроцесорних систем використовують апаратний захист, наприклад, у мікроконтролери сімейства RX інтегрований модуль захисту пам'яті (MPU), який може розглядатися як апаратний брандмауер, що відокремлює, привілейований код операційної системи від програми користувача. Модуль MPU дозволяє легко виявити збій і забезпечує гнучкі засоби усунення помилок, що підвищує надійність програмного забезпечення та автоматизованої системи в цілому.

1.2 Загальні відомості, функціонування та використання цифрових мікропроцесорних систем

Розглянемо базові терміни мікропроцесорної техніки, принципи організації мікропроцесорних систем, структуру зв'язків, режими роботи та основні типи мікропроцесорних систем.

Електроенергетика належать до числа технічних систем, де автоматизація, а потім і комп'ютеризація розвивались досить успішно. Однак на теперішній час мова йде про перехід у якісно новий стан — в енергетику інформатизовану, що відрізняється суттєво більш високими показниками по відношенню якості енергії, продуктивність праці, економічності при дуже високій надійності.

Інформатизація технічних систем, у тому числі електроенергетики,— це створення єдиного інформаційного середовища з інфраструктурою, що дозволяє вирішувати весь комплекс завдань у цій області (наукові, технічні, технологічні, економічні й організаційні).

Важливим кроком у перетворенні енергетики в інформатизовану було створення ієрархічної системи протиаварійного управління, самою нижчою ланкою якого є кінцеві точки та вузли енергосистеми (ЕС): генератори, їх власні потреби, високовольтні пристрої, системні підстанції, споживчі підстанції й безпосередньо комутаційне електрообладнання. Це значне число вузлів і точок, де реалізуються процеси генерації, перетворення й розподілення енергії; у них чи навкруги їх відбуваються практично всі аварії й порушення режиму. Тому без вирішення проблеми комплексної автоматизації й побудови розвинутої системи управління «знизу до верху», узгодження математичних моделей різних рівнів управління і забезпечення необхідного обміну інформацією між ними на основі використання мікропроцесорної техніки (і, у тому числі, мікроконтролерів) і з впровадженням нових інформаційних технологій неможливо підняти рівень енергетики на якісно нову рівень.

Звернемося до історії. Робота англійського математика Алана Метісона Тьюринга (1912-1954) “On Computable Numbers, with an Application to the Entscheidungsproblem” у журналі “Journal of Symbolic Logic” в 1937 році, у якій вперше описана «машина Тьюринга», поклала початок комп'ютерній ері. Іменем Тьюринга названа найвища нагорода в області програмування.

Державний комітет Заради міністрів СРСР по впровадження передової техніки в народне господарство видав І.С.Бруку та Б.І.Рамееву Авторське Свідоцтво за №10475 на винаходження цифрової обчислювальної машини з пріоритетом від 4 грудня 1948 року. Це перший

офіційно зареєстрований документ, що стосується розвитку обчислювальної техніки в СРСР.

Перший транзистор був створений в 1947 році (а в 1956 році за його відкриття американським фізиком Бардіну, Браттейну і Шоклі вручили Нобелівську премію по фізиці). Перша мікросхема запрацювала 12 вересня 1958 долі в компанії Texas Instruments (за її винахід Нобелівську премію по фізиці присудили лише в 2000 році) «Першовідкривачами» мікросхеми (тобто батьками сучасної мікроелектроніки) по праву вважаються Джек Кілбі і один із засновників Intel Роберт Нойс. Особливого розвитку фізики як такої при створенні мікросхеми не було, але Кілбі і Нойс придумали технологію, яка здійснила повний переворот в електронній промисловості. До 1965 році в самій складній мікросхемі компанії Fairchild було всього лише 64 транзистора. В 1970 році (Intel Pentium)– 4004 (МК №25/2007.-с.22), 2000р. – 100 000 шт., 2005р.(P4 Core Duo) – 1 000 000 шт. з технологічного процесу 45 нанометрів, на сьогодні ядро Intel Core i7 – це 1170 млн. шт., при технології 32 нм.

Для представлення розмірів транзисторів в надрмініатюрних мікросхемах слід відмітити, що, наприклад, у віруса імуннодефіциту діаметр близький до 120 нм, червоного кров'яного тільця (еритроциту) — 6000...8000 нм, волосся на голові — приблизно 80 000 нм. А по технології 65 нм на кінці кулькової авторучки вдалося б розмістити більше 1 млн. транзисторів.

Еволюція розмірів мікропроцесорів відбувається в основному за рахунок зменшення довжини затворів польових транзисторів рис. 1.1, (транзистор у розрізі із збільшенням приблизно 500 000:1) із збереженням товщини ізоляційної оксидної прошарку 1,2 нм (приблизно п'ять атомів водню). У технології 90 нм довжина затвору дорівнює 45...50 нм, а по технології 65 нм — 30...35 нм. Завдяки цьому в кожного транзистора

маємо зменшення струму витоку, опору каналу, ємності затвору, що призводить до меншої постійної часу його перемикавання, підвищенню швидкодії, пониженню енергоспоживання. Це дозволяє формувати і більш високі тактові частоти. У січні 2006 р. INTEL оголосила про наступне покоління мобільної платформи INTEL Centrino, що базується на новому процесорі Yonah (означає "ведмідь" у перекладі з мови індіців північного заходу США). Yonah представляє собою перший комбінований мікропроцесор INTEL, що побудований по технології 65 нм і одночасно наслідуючий мікроархітектуру прототипів — мікросхем Dothan і Banias Pentium M, але відрізняється наявністю двох аналогічних по функціям ядер Core 1 і Core 2 і загальної кеш пам'яті. Він має більше число активних елементів — 151 000 000 на площі 90,3 мм² (проти 140 000 000 на 83,6 мм² Dothan і 77 000 000 на 82,8 мм² в Banias). Інтервал його тактових частот — від 1,5 до 2,33...2,55 ГГц.

Два незалежних ядра через загальну процесорну (FSB) шину взаємодіють один з одним і, завдяки "динамічному розподіленню" (Dynamic Cache Allocation), із загальною кеш пам'ятю обсягом 2 МБ.

Максимальне енерговикористання Yonah не перевищує 31 Вт. Використання динамічно змінюється (Dynamic Power Coordination) у залежності від завантаження обчислювальними програмами й числа використовуваних блоків кеш пам'яті. При зменшенні навантаження тактова частота знижується, як і сама напруга живлення. Зниження енергоспоживання дозволяє обчислювальному засобу більше години працювати в автономному режимі.

Історію й корені ділення обчислювальної техніки на великі ЕОМ, міні-ЕОМ, мікро-ЕОМ і мікроконтролери можна представити наступним чином.

Перша вітчизняна ЕОМ – МЕОМ створена в Інституті електродинаміки в 1951 р. (тоді - Інститут електротехніки) під керівництвом З.А.Лебедьєва. Перше покоління великих ЕОМ – «Урал-2»; друге – М-220, БЕСМ-4, М-222; третє – М4030 і ЕС ЕОМ. Міні-ЕОМ Промінь-М, НАІРІ, Саратов, Мир-1, Мир-2, СМ-3, СМ-4. Мікро-ЕОМ «Електроніка-60» (DEC), ЕС-1800, ЕС-1840, К1-10, К1-20 (Intel), Spektrum (Sinklera).

Подальший розвиток МПС був пов'язаний з появою – мікросхем серії К580 (8080 Intel), К588 (PDP-11/LSI-11 фірми DEC), 6800 - 6809 Motorola, Z8000 фірми Zilog, 9900 Texas Instruments, 8086 і MS-48, MS-51 фірми Intel.

Чіткі межі між типами МПС іноді провести доволі складно. Швидкодія всіх типів мікропроцесорів постійно росла, і неминучі ситуації, коли новий мікроконтроллер виявляється швидшим міні-ЕОМ, наприклад, застарілого персонального комп'ютера.

До кінця минулого століття процесори ПК, досягнувши тактових частот у сотні мегагерц, стали здібні практично непомітно для користувача виконувати обсяги робіт, що покривають більше 90% потреб у використанні ПК. По суті справи індустрії довелося вигадати нові задачі для ПК, щоб пояснити користувачу необхідність подальшого нарощування обчислювальної потужності. Свою роль тут зіграли й розробники програмного забезпечення: програми, написані в 80-х роках минулого століття, частіше працювали на комп'ютерах тих років швидше, а ніж сучасні аналогічні по призначенню на комп'ютерах у тисячі раз продуктивніших.

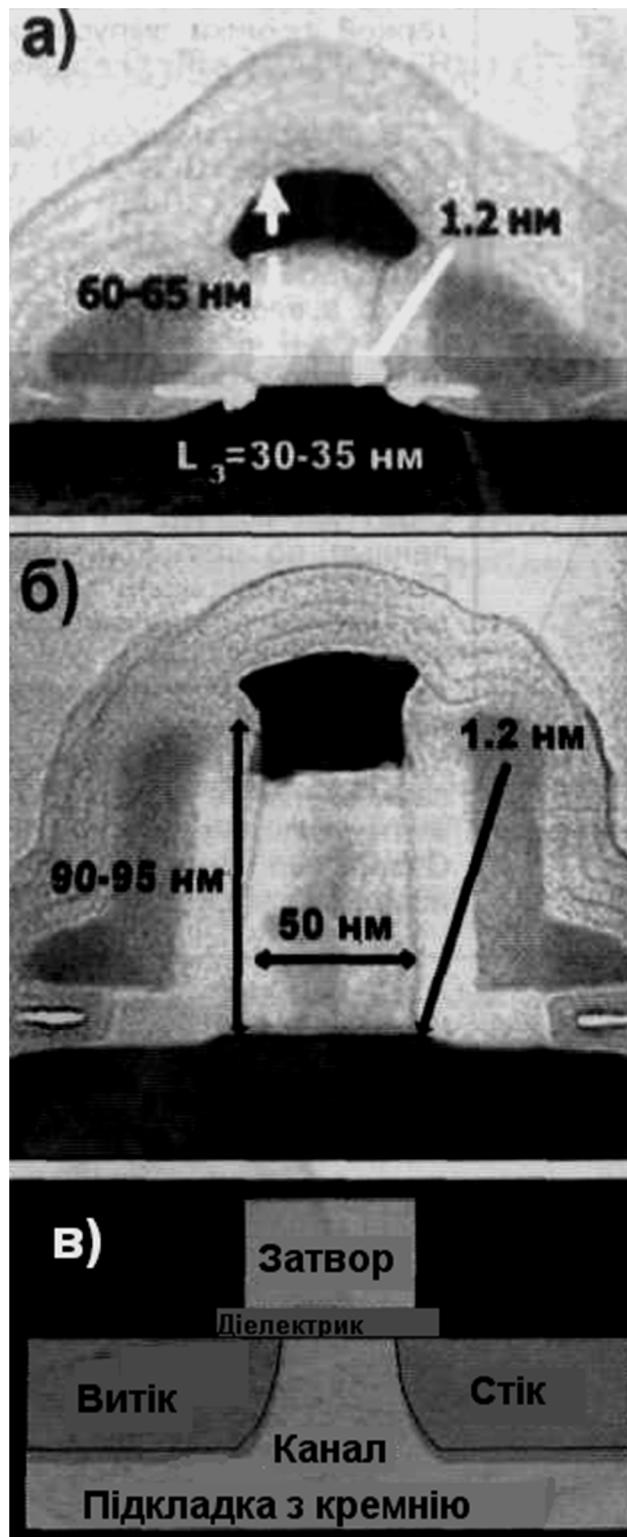


Рис. 1.1. Структура польових транзисторів мікропроцесора

Придумати нові задачі також виявилось неважко, з одного боку, цьому сприяла індустрія комп'ютерних ігор, здатна за рахунок залучення більш складних алгоритмів поглинути практично нескінчені обчислювальні ресурси, з іншого — залучення в сферу впливу ПК тих областей, які в силу своєї ресурсоємності раніше були недоступні комп'ютерам, наприклад відео.

Так чи інакше, але запити на збільшення процесорної потужності є. Той факт, що для вирішення переважної більшості практичних завдань цілком достатньо було потужності будь-якого навіть самого слабкого процесора, дозволив Intel в 2000 р. перейти на нову архітектуру Netburst, мало корисну для обчислень, але, це дозволило досягати суттєво більш високих тактових частот. Точніше сказати, архітектура була оптимізована для обробки потокових даних, але вкрай неефективна з погляду звичайних програм. Тобто відео на ній оброблялося добре, а зниження продуктивності на «звичайних» додатках не було помітно за рахунок її свідомо надлишкового запасу. Мікроархітектура Netburst значно випередила свій час. У той час канали передачі не були готові до передачі таких обсягів інформації, і можливості цієї мікроархітектури виявилися просто незатребуваними. Центральний процесор розсіював потужність як 100-ваттна електрична лампочка. Для ефективного завантаження конвеєра був придуманий блок пророкування розгалужень, який намагався вгадати роботу програми й заздалегідь підготувати дані в кеші.

Але й про реальну обчислювальну потужність теж не потрібно було забувати. Причина низької продуктивності полягала в наддовгому конвеєрі, через який частина виконавчих блоків процесора простоювала. Щоб завантажити їх, була придумана нова технологія — Hyper-Threading (HT), що дозволяє виконувати другий потік команд. При цьому якщо у звичайному одноядерному процесорі ресурсомісткий потік міг захопити

більш 99% ресурсів, то із НТ ця величина була обмежена приблизно 80% (уведення другого логічного ядра збільшує сумарну продуктивність приблизно на чверть у порівнянні з одиночним процесором; в однопотоковому режимі ця чверть залишається вільною), У випадку ресурсномісткого додатка це досить погано, ресурси кристала не можна використовувати повністю. Але у відношенні всіх інших процесів у системі — дуже добре: замість часток відсотка залишаються «вільними» відсотків двадцять, тобто цілком достатньо, щоб користувач, що запустив один «важкий» процес (наприклад, архівацію, антивірусну перевірку або відеостиск), не відчув від нього гальмування всієї іншої роботи (скажемо, роботи в текстовому процесорі або браузері).

Інакше кажучи, навіть така «віртуальна багатоядерність» сприяла істотному збільшенню комфорту при роботі із ПК, але величезне енергоспоживання не дозволило підняти частоту процесорного ядра понад 3,8 ГГц, а інших шляхів для підвищення продуктивності в цієї архітектури не було.

Трохи пізніше Intel представила флагманську архітектуру Intel Core, яка в якійсь мірі була кроком назад стосовно процесора Pentium 4, але зненацька успішно проникла в усі сегменти комп'ютерного ринку - мобільні рішення, настільні комп'ютери та сервера.

Багатоядерні версії процесорів повернули втрачені позиції Intel.

У цьому зв'язку цікаво відзначити, що, на відміну від платформи IBM PC/Intel/Windows, у комп'ютерах Apple уже давно використовувалися два процесори, навіть у моделях середнього рівня, — ще з тих часів, коли для цього доводилося ставити два окремі процесорні гнізда. Недарма комп'ютери саме цієї компанії завжди вважалися найбільш дружніми стосовно користувача.

На сьогодні промисловість перейшла на МПС з декількома фізичними ядрами при збереженні віртуального розгалуження.

1.3 Особливості мікропроцесорних систем

Розглянемо основні визначення.

Електронна система — це будь-який електронний вузол, блок, прилад або комплекс, що виконує обробку інформації.

Інтегральна мікросхема (Integrated Circuit (IC)) - активні (діоди, транзистори) і пасивні (резистори, конденсатори) елементи, а також провідники, що їх з'єднують відповідно до принципової електричної схеми пристрою, виготовлені як одне ціле у єдиному технологічному циклі на кристалі напівпровідникового матеріалу.

Напівпровідникові мікросхеми — це мікросхеми, у яких інтегральні елементи (транзистори, діоди, резистори, конденсатори) виготовляються у поверхневому шарі напівпровідникової пластини.

Завдання — це набір функцій, виконання яких потрібно від електронної системи.

Швидкодія — це показник швидкості виконання електронною системою її функцій.

Гнучкість — це здатність системи підлаштовуватися під різні завдання.

Надмірність — це показник ступеня відповідності можливостей системи розв'язуваною даною системою завдання.

Інтерфейс — угода про обмін інформацією, правила обміну інформацією, що включають електричну, логічну й конструктивну сумісність пристроїв, які приймають участь в обміні. Інша назва — сполучення.

Процесор – електронний пристрій, призначений для розшифрування програмного коду та виконання послідовності логічних і арифметичних операцій над бітами.

Мікропроцесор (МП) - функціонально закінчений процесор для побудови мікро-ЕОМ, реалізований у вигляді однієї або декількох великих інтегральних схем (ВІС) і призначений для обробки цифрової інформації по заданих програмах.

Мікропроцесорна система (МПС) – мікропроцесор, що програмно керує зовнішніми периферійними пристроями й елементами, призначений для цілей обчислення, контролю та керування різними пристроями (рішення спеціалізованих завдань).

МПС може реалізовуватися на:

- однокристальних мікропроцесорах ОМП (МК);
- секційних (багатокристальних) МП;
- складних матричних програмованих логічних схемах (ПЛІС, PLD, CPLD і ін.).

Драйвер — це програмний модуль між операційною системою та апаратним забезпеченням, який дозволяє оператору (програмісту) керувати різними пристроями будь-якої складності за допомогою певних команд, які зрозумілі людині: не на рівні електричних сигналів, що подаються на елементи плати, а на рівні логіки.

Операційна система - це програма, яка може інтегрувати в себе безліч драйверів у єдине ціле, тобто забезпечує безконфліктне функціонування пристроїв в єдиному програмному середовищі. Завдяки операційній системі кінцеві користувачі — звичайні люди, не фахівці — можуть упевнено працювати з логікою папок, файлів і окремих програм, не вникаючи ні в конфігурацію комп'ютера, ні вже тим більше в те, як мікросхеми передають один одному електричні сигнали.

Програми користувачів - це програмні модулі, які знаходяться над операційною системою, — ще далі від машинної логіки та ще ближче до

людської: до логіки фахівця. Наприклад, інженер-електрик глибоко не знаючи комп'ютер та операційну систему без проблем може завантажити програму моделювання і в ній працювати, користуючись знаннями електротехніки.

Платформа — це програма, яка дозволяє встановлювати на неї спеціалізовані програмні модулі. Операційна система — це теж платформа, тому що на неї встановлюються всі інші програми, але частіше мається на увазі платформа спеціалізована, на яку можна в різних комбінаціях встановлювати готові модулі для рішення вузьких завдань. У платформі є ядро, яке організовує взаємодію модулів — аналогічно материнській платі, куди підключаються різні пристрої. За рахунок готових модулів функціональність такого комплексу можна розширювати.

Код (програма) — це по суті теж саме (текст): з боку користувача — програма, з боку програміста представляє собою код. Якщо ми використовуємо її, як знаряддя, не заглиблюючись в суть, то частіше говоримо про програму, а якщо переписуємо або дописуємо, — то про код.

Код і алгоритм співвідносяться як текст і його зміст. Допустимо, потрібно порахувати швидкості руху електронів в провіднику. Формули нам дасть математик. Алгоритм — це описані людською мовою кроки, які людині потрібно зробити, щоб з вихідних даних, наприклад діаметра провідника і його довжини, одержати нові необхідні дані. Наприклад: «Взяти формулу номер один. Підставити в неї значення. Розрахувати... Взяти формулу номер два...» Маючи алгоритм, можна одержати бажаний результат, навіть не розуміючи, що та навіщо ми робимо. Код — це бінарний текст, який «пояснює» комп'ютеру, як йому виконувати алгоритм, виконаний на одній з мов програмування.

Розглянемо особливості мікропроцесорних систем.

МПС – це багаторівнева система. Найнижчим рівнем є апаратні або технічні засоби, тобто електронні схеми. Наступний рівень - це архітектура, тобто з'єднання елементів, а також структура й характеристики МПС, до яких має безпосереднє відношення користувач (його також можна віднести до апаратного рівня). Найвищий рівень – це програмний (програмне забезпечення). Представляє набір команд, які управляють роботою МПС і в основі яких лежить булева алгебра.

За способом програмного керування розрізняють МПС:

- Схемного керування з фіксованим набором команд, який не може змінюватися користувачем;
- Мікропрограмного керування, систему команд якого розробляють під час проектування конкретної МПС на основі набору найпростіших мікрокоманд і з урахуванням конкретного рішення завдань (Fuzzy,..).

Мікропроцесорна система може розглядатися як окремий випадок електронної системи, призначеної для обробки вхідних сигналів і видачі вихідних сигналів. У якості вхідних і вихідних сигналів при цьому можуть використовуватися аналогові сигнали, одиночні цифрові сигнали, цифрові коди, послідовності цифрових кодів. У середині системи може проводитися зберігання, нагромадження сигналів (або інформації), але суть від цього не міняється. Якщо система цифрова (а мікропроцесорні системи відносяться до розряду цифрових), то вхідні аналогові сигнали перетворюються в послідовності кодів вибірок за допомогою АЦП, а вихідні аналогові сигнали формуються з послідовності кодів вибірок за допомогою ЦАП. Обробка й зберігання інформації проводяться в цифровому виді.

Характерна риса традиційної цифрової системи полягає в тому, що алгоритми обробки й зберігання інформації в ній жорстко зв'язані зі схемотехнікою системи. Тобто зміна цих алгоритмів можливо тільки

шляхом зміни структури системи, заміни електронних вузлів, що входять у систему, і/або зв'язків між ними. Наприклад, якщо нам потрібна додаткова операція додавання, то необхідно включити в структуру системи зайвий суматор. Або якщо потрібна додаткова функція зберігання коду протягом одного такту, то ми повинні додати в структуру ще один регістр. Природно, це практично неможливо зробити в процесі експлуатації, обов'язково потрібний новий виробничий цикл проектування, виготовлення, налагодження всієї системи. Саме тому традиційна цифрова система часто називається системою на «жорсткій логіці».

Будь-яка система на «жорсткій логіці» обов'язково являє собою спеціалізовану систему, налаштовану винятково на одне завдання або (рідше) на кілька близьких, заздалегідь відомих завдань. Це має свої безперечні переваги.

По-перше, спеціалізована система (на відміну від універсальної) ніколи не має апаратної надмірності, тобто кожний її елемент обов'язково працює на повну силу (звичайно, якщо ця система грамотно спроектована).

По-друге, саме спеціалізована система може забезпечити максимально високу швидкодія, тому що швидкість виконання алгоритмів обробки інформації визначається в ній тільки швидкістю окремих логічних елементів і обраною схемою шляхів проходження інформації. А саме логічні елементи завжди мають максимальний на даний момент швидкодію.

Але в той же час більшим недоліком цифрової системи на «жорсткій логіці» є те, що для кожного нового завдання її треба проектувати й виготовляти заново. Це процес тривалий та дорогий і вимагає високої кваліфікації виконавців. А якщо розв'язуване завдання раптом змінюється, то вся апаратура повинна бути повністю замінена. У нашому швидко мінливому світі це досить марнотратно.

Шлях подолання цього недоліку досить очевидний: треба побудувати таку систему, яка могла б легко адаптуватися під будь-яке завдання, перебудовуватися з одного алгоритму роботи на іншій без зміни апаратури. І задавати той або інший алгоритм ми тоді могли б шляхом уведення в систему якоїсь додаткової керуючої інформації, програми роботи системи. Тоді система стане універсальною, або програмованою, не жорсткою, а гнучкою. Саме це й забезпечує мікропроцесорна система.

Але будь-яка універсальність обов'язково приводить до надмірності. Адже рішення максимально важкого завдання вимагає набагато більше засобів, а ніж рішення максимально простого завдання. Тому складність універсальної системи повинна бути такою, щоб забезпечувати рішення самого важкого завдання, а при рішенні простого завдання система буде працювати далеко не на повну силу, буде використовувати не всі свої ресурси. І чим простіше розв'язуване завдання, тим більша надмірність, і тем менш виправданою стає універсальність. Надмірність веде до збільшення вартості системи, зниженню її надійності, збільшенню споживаної потужності і т.д.

Крім того, універсальність, як правило, приводить до істотного зниження швидкодії. Оптимізувати універсальну систему так, щоб кожне нове завдання вирішувалося максимально швидко, попросту неможливо. Загальне правило таке: чим більше універсальність, гнучкість, тем менше швидкодія. Більше того, для універсальних систем не існує таких завдань (нехай навіть і найпростіших), які б вони вирішували з максимально можливою швидкістю. За все доводиться платити.

Таким чином, можна зробити наступний висновок. Системи на «жорсткій логіці» потрібні там, де розв'язувана завдання не міняється тривалий час, де потрібна найвища швидкодія, де алгоритми обробки інформації гранично прості. А універсальні, програмовані системи

доцільні там, де часто міняються розв'язувані завдання, де висока швидкодія не занадто важлива, де алгоритми обробки інформації складні. Тобто будь-яка система необхідна на своєму місці.

Однак за останні десятиліття швидкодія універсальних (мікропроцесорних) систем сильно виросла (на кілька порядків). До того ж великий обсяг випуску мікросхем для цих систем привів до різкого зниження їх вартості. У результаті область застосування систем на «твердій логіці» різко звужилася. Крім цього, актуальним стало питання захисту інтелектуальної власності.

Контрольні запитання

- 1. Електронна система та її визначення.*
- 2. Задача в електронній системі та її визначення.*
- 3. Швидкодія електронної системи та її визначення.*
- 4. Гнучкість електронної системи та її визначення.*
- 5. Надмірність електронної системи та її визначення.*
- 6. Інтерфейс електронної системи та його визначення.*
- 7. Дайте визначення мікропроцесорної системи?*
- 8. Як розрізняють способом програмного керування?*
- 9. Дайте тлумачення поняттю «жорстка логіка».*
- 10. Характеристика систем на «жорсткій логіці».*
- 11. В чому полягають переваги цифрового способу подання і опрацювання інформації у порівнянні з аналоговим?*

Розділ другий

ФУНКЦІОНАЛЬНІ ЕЛЕМЕНТИ ЦИФРОВОЇ ТЕХНІКИ

2.1 Дискретні електронні пристрої цифрової техніки

У цифрових пристроях інформація подається у вигляді напруги (або струму) імпульсної форми. При цьому визначений набір імпульсів відповідає числовому значенню представленої цим набором фізичної величини. Напруга сигналів в цифровому пристрої має лише два значення: близьке до *нуля* вольт — умовно названо *логічним нулем*, і напругою у декілька вольт (звичайно близько до 5 В) — умовно названо *логічною одиницею*. Такий розподіл рівнів сигналів дозволяє повністю уникнути впливу завад (рівень яких звичайно не перевищує десятих частин вольта). Окрім того, використання числових методів опрацювання інформації дозволяє досягти практично будь-якої (заданої) точності.

Другий недолік аналогових систем — неможливість змінення алгоритму функціонування без апаратних змін пристрою — усувається використанням програмованих цифрових пристроїв.

Розглянемо імпульсні напруги і струми.

Електричним імпульсом називають напругу або струм, що мають постійне значення лише протягом короткого відрізка часу, більшого або порівняного з тривалістю встановлення процесів у даному електричному колі. Розрізняють: **відеоімпульси** (рис.2.1а) і **радіоімпульси** (рис. 2.1 б).

Відеоімпульс є короткочасною появою напруги (або струму) у колі і має відмінну від нуля постійну складову (середнє значення).

Радіоімпульс — це короткочасний пакет високочастотних коливань напруги (або струму), обвідна якого має форму відеоімпульса. Радіоімпульси використовують в основному в системах радіозв'язку і практично не використовують в пристроях цифрового опрацювання інформації. Тому ми обмежимося розглядом відео імпульсів (в подальшому імпульси). Імпульси за формою є прямокутні, трапецієподібні, трикутні та ін.

Основні параметри імпульсів:

- амплітуда імпульсу U_m ;
- тривалість імпульсу t_i ;
- тривалість переднього фронту t_{ϕ} - час зростання напруги від $0,1U_m$ до $0,9U_m$;
- тривалість заднього фронту t_c -час спадання напруги від $0,9U_m$ до $0,1U_m$;
- період імпульсу T ;
- сквапність $Q = T/t_i$;
- коефіцієнт заповнення $\kappa_3 = 1/Q = t_i/T$, величина зворотна скважності.

Звичайно в цифрових електронних пристроях використовують імпульси напруги максимально наближені до прямокутної форми. Передача інформації за допомогою таких імпульсів дозволяє практично повністю виключити вплив електромагнітних перешкод на точність роботи пристроїв.

Розглянемо ключовий транзисторний підсилювач.

В основі більшості імпульсних пристроїв, у тому числі цифрових, лежить **ключовий транзисторний підсилювач**, схема якого подана на рис.2.2а.

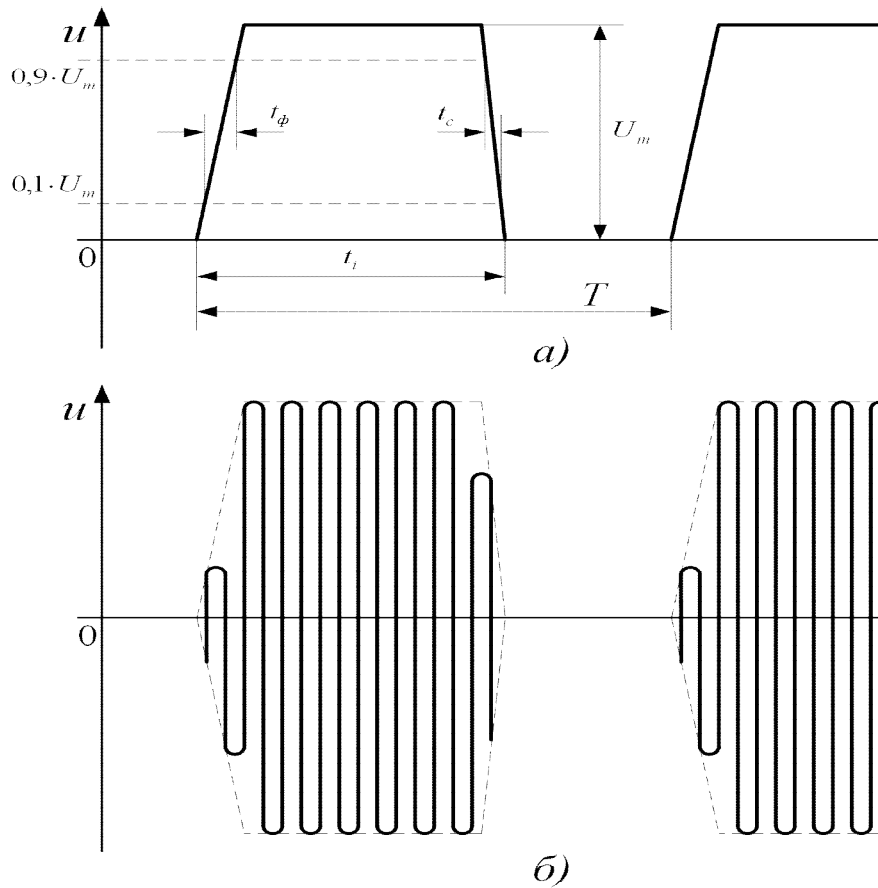


Рис. 2.1. Вигляд і параметри імпульсів

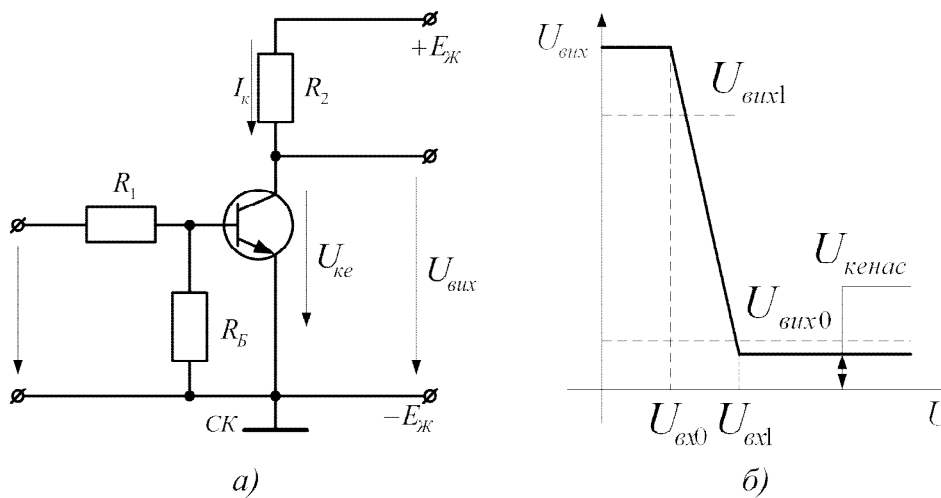


Рис. 2.2. Ключовий транзисторний підсилювач:
а – електрична схема; б – діаграма роботи

На відміну від схем лінійних підсилювачів, в цьому випадку зсув робочої точки транзистора відносно нуля (режиму відсікання) відсутній, так, що при $U_{ВХ} = 0$ транзистор закрито (у відсутність вхідного сигналу база через резистор R_B з'єднується з спільною точкою схеми). Для обмеження струму бази вхідна напруга подається через резистор R_1 . Струм колектора при цьому також дорівнює нулю, а вихідна напруга $U_{Вих} = U_{КЕ} = E_{ж}$. Такий стан названо режимом **відсікання**. Цьому режиму відповідає зона вхідних напруг від $U_{ВХ}=0$ до $U_{ВХ} \leq U_{ВХ0}$ на передавальній характеристиці підсилювача, що представлена на рис. 2.2 б (приблизно $U_{ВХ0} \approx 0,4 В$ — напруга переходу база-емітер).

Коли $U_{ВХ} > U_{ВХ0}$, транзистор відкривається, у подальшому струм колектора I_K і напруга $U_{КЕ}$ пов'язані відповідним співвідношенням: $U_{КЕ} = E_{ж} - I_K \times R_2$. Це відповідає лінійній ділянці передавальної характеристики. Нахил її визначається параметрами транзистора, а також величинами R_1 , R_2 і $E_{ж}$. Очевидно, що з досягненням струмом колектора максимального значення, подальший збільшення вхідної напруги за $U_{ВХ1}$ (рис. 2.2 б), не викликає змін I_K і $U_{КЕ}$, тому що транзистор знаходиться в режимі **насичення**. Вихідна напруга при цьому

(звичайно $U_{КЕНАС} \approx 0,2 В$). Таким чином, якщо обмежити зміну вхідної напруги значеннями $U_{ВХ} < U_{ВХ0}$ і $U_{ВХ} > U_{ВХ1}$, то пристрій працює в режимі ключа:

- з $U_{ВХ} < U_{ВХ0}$ $I_K = 0$, $U_{Вих} = E_{ж}$;
- з $U_{ВХ} > U_{ВХ1}$ $I_K \approx E_{ж}/R_K = I_{Кнас}$, $U_{Вих} < U_{ВХ0} \approx 0$.

Такий підсилювач часто використовують в якості підсилювача потужності сигналів керування виконавчими пристроями і контролю технологічних процесів. При цьому замість резистора R_2 вмикається обмотка електромагнітного реле, малопотужний електродвигун, сигнальна лампа та інше (рис. 2, в). Для керування вмиканням цих пристроїв

достатньо подавати на вхід підсилювача порівняно невелику напругу (порядку декількох вольт) при невеликому вхідному струмі (як правило, не більше 10 мА). Транзистор при цьому переходить в режим насичення («ключ замкнений») і підключає навантаження до джерела $E_{жс}$ (величина $E_{жс}$ складає десятки вольт). Струм навантаження при цьому (визначається величиною $E_{жс}$ і опором навантаження), в залежності від типу використаного транзистора, дорівнює 1...15 А. Якщо вхідну напругу вимкнути, транзистор повертається в режим відсікання («ключ вимкнуто») і навантаження вимикається від джерела $E_{жс}$. Таким чином, малопотужним сигналом від керуючого пристрою (наприклад, мікропроцесора) можна вмикати і вимикати порівняно потужні навантаження.

Ключовий транзисторний підсилювач є найпростішим цифровим пристроєм, на основі якого будуються усі інші цифрові схеми. Так, якщо прийняти напругу, що відповідає рівню логічної одиниці, близькою до величини напруги живлення схеми (обираючи при цьому відповідні параметри її елементів), то вихідні напруги схеми будуть відповідати необхідним для її роботи в режимі ключа вхідним напругам. Це дозволяє підключати вхід схеми до виходу такої ж схеми, поєднувати їх у різних сполученнях, створюючи більш складні дискретні пристрої. На практиці для надійної роботи таких схем значення напруги логічного нуля приймають трохи вище величини $U_{кенас}$ (рівень $U_{вих0}$, рис. 2.2 б), а напругу логічної одиниці — трохи нижче величини $E_{жс}$ (рівень $U_{вих1}$). При цьому зміна значень вихідної напруги схеми у незначних межах внаслідок впливу підключених до її виходу входів аналогових схем не викликає зміни логічних значень цих напруг. Параметри схеми підсилювача обирають таким чином, що з підключенням входів декількох (як правило, не менше 5) підсилювачів до виходу такого ж

підсилювача рівні напруг, що відповідають логічним 0 і 1, знаходяться у межах прийнятих значень. Допустиму кількість входів, що підключаються до виходу підсилювача, названо коефіцієнтом розгалуження. На основі ключового підсилювача створюються складні цифрові пристрої.

2.2 Основні операції над логічними змінними та їх реалізація

Під логічною змінною розуміємо величину, що отримує значення 0 або 1 в залежності від істинності події, що нею описується. Наприклад, змінна, що умовно названа «тепло», дорівнює 1, коли обігрівач ввімкнено, і дорівнює 0 при вимкненому обігрівачі. Слід підкреслити: значення, що приймають логічні змінні, не є числами, тобто не підлягають правилам арифметики.

Для операцій над значеннями логічних змінних використовується спеціальна алгебра логіки (булева алгебра). Розглянемо її положення (основні теореми алгебри логіки, розглянуті в [20]):

Правило повторювання:

$$X \wedge X = X \quad (X * X) = X, \text{ або } X \vee X = X, (X + X) = X, \quad (2.1)$$

де \wedge і \vee — символи логічного множення (\bullet) і додавання ($+$). Правило виконання операцій з інверсією:

$$X \wedge \bar{X} = 0 \quad (X * \bar{X}) = 0, \text{ або } X \vee \bar{X} = 1, \quad (X + \bar{X}) = 1, \quad (2.2)$$

де $(\bar{\quad})$ — символ інверсії.

Правило подвійної інверсії:

$$\overline{(\bar{X})} = X. \quad (2.3)$$

Теорема де Моргана:

$$\overline{X_1 \wedge X_2} = \bar{X}_1 \vee \bar{X}_2 \quad \text{або} \quad \overline{X_1 \vee X_2} = \bar{X}_1 \wedge \bar{X}_2. \quad (2.4)$$

Операції з 0 і 1:

$$X \wedge 1 = X$$

$$X \vee 0 = X$$

$$X \wedge 0 = 0$$

$$X \vee 1 = 1$$

$$\bar{0} = 1$$

$$\bar{1} = 0.$$

Розглянемо основні логічні операції.

Операція ТАК — логічне повторення. Вихідна змінна Y повторює сигнал на вході X : $Y=X$. Елемент, що реалізує цю операцію, називають буфер, або буферний повторювач (підсилювач). Використовують його для підсилення за потужністю у якості шинних формувачів у мікропроцесорних системах .

Звичайно має третій керуючий вхід — дозволу на видачу даних, що переводить вихід у особливий стан Z "відключений". Це дозволяє по чергово підключати до одної лінії декілька елементів. При цьому до лінії підключено лише один з елементів, на керуючий вхід котрого подано сигнал дозволу — інші відключені.

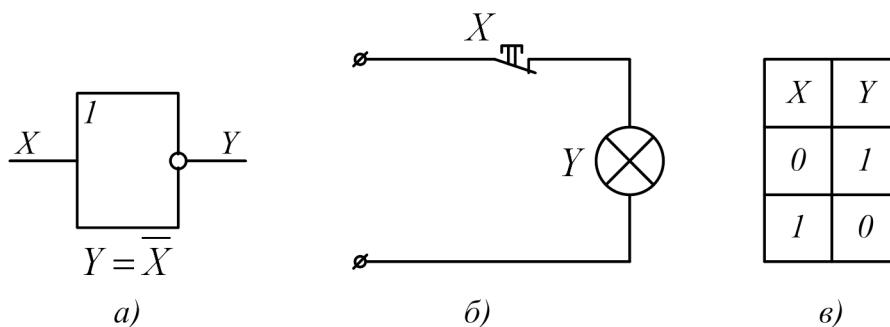


Рис.2.3. Операція НІ: а – зображення; б – інтерпретація, в – таблиця істинності

Операція НІ (NOT, \neg) — логічне заперечення, або **інверсія**: $Y = \bar{X}$ (рис. 2.3). Таку операцію виконує, наприклад, кнопка з нормально замкненими контактами (рис. 2.3 б): якщо вона не натиснута (вхідна величина $X=0$ — логічний нуль), то коло замкнене — лампа горить

(вихідна величина $Y = 1$ — логічна одиниця), і навпаки. Цю ж операцію виконує ключовий транзисторний підсилювач: якщо вхідна напруга відповідає рівню логічного нуля, то вихідна — логічній одиниці, і навпаки. Такий підсилювач названо **логічним елементом НІ (інвертором)** і позначається згідно з рис. 2.3 а (у позначенні виходу присутнє коло — символ інверсії). Замість вхідної і вихідної напруг вказують їхні логічні значення. Окрім того, не зображено кіл живлення елемента і спільний провід.

Логіку роботи елемента прийнято показувати у вигляді **таблиці істинності**, в якій всім можливим значенням однієї або декількох вхідних логічних змінних поставлено у відповідність значення вихідної змінної. Таблиця істинності елемента НІ наведена на рис. 2.3 в.

Операції АБО (OR, \vee) та АБО — НІ (NOR). Операція АБО (диз'юнкція) - **логічне додавання**. Зміст операції витікає з її назви, наприклад: «Двигун можна запустити вручну або за командою від ЕОМ».

Умовне позначення елемента АБО і функція, що ним реалізується, наведені на рис. 2.4. Як і елемент НІ, елемент АБО побудований на основі ключового транзисторного підсилювача, що має два вхідних кола.

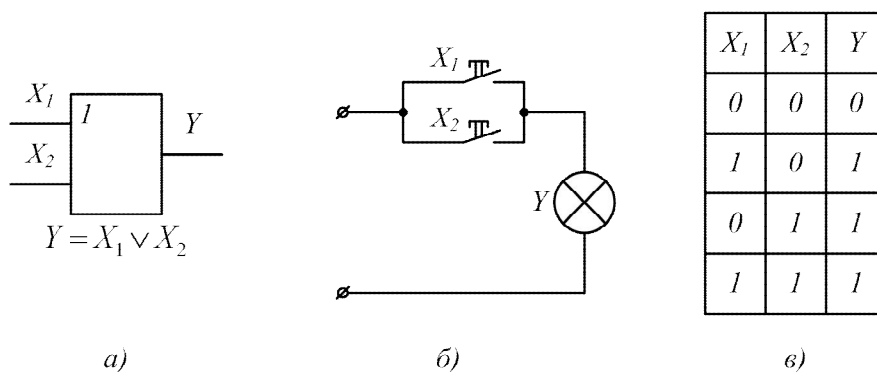


Рис.2.4. Операція АБО: а – зображення; б – інтерпретація. в – таблиця істинності

На рис. 2.4 б подано приклад реалізації функції ключів (кнопок). Щоб замкнути коло, потрібно натиснути одну з кнопок (або одну, або другу, або обидві разом).

Часто функція АБО використовується з подальшою інверсією вихідної змінної, тобто до виходу елемента АБО підключають інвертор. Отримана в результаті функція АБО-НІ звичайно реалізується одним спеціальним елементом АБО-НІ, позначення і реалізація якого наведені на рис.2.5.

З таблиці істинності елемента АБО (рис. 2.4 в) слідує, що вихідна змінна Y приймає значення логічної одиниці, якщо хоча б на одному з входів сигнал дорівнює 1. Сигнал на виході елемента АБО-НІ є інверсією відповідних значень елемента АБО (рис. 2.5 в).

Кількість входів елементів може бути і більше двох. У цьому випадку функцію, що реалізується, наприклад, елементом АБО, запишемо у вигляді:

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

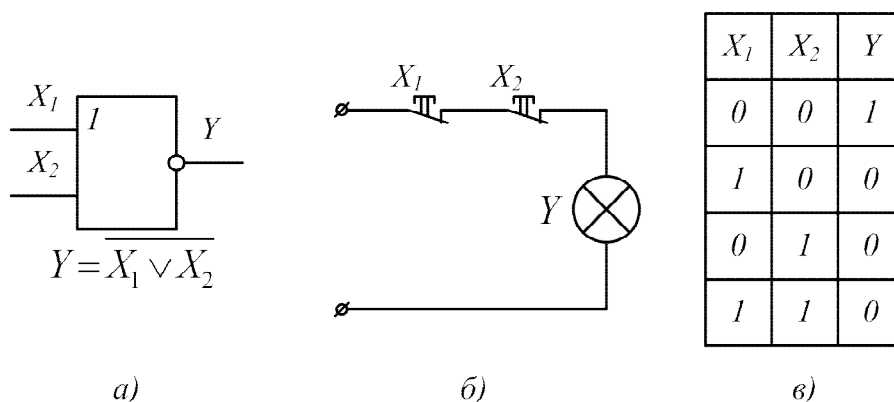


Рис.2.5. Операція АБО-НІ: а – зображення; б – інтерпретація, в – таблиця істинності

Кількість входів прийнято вказувати в позначенні елемента, наприклад: двохвходовий елемент 2АБО-НІ, чотиривходовий 4АБО-НІ.

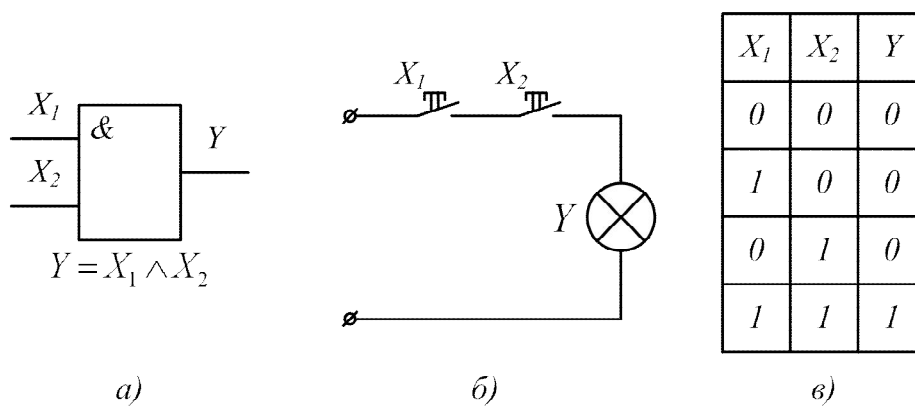


Рис.2.6. Операція I: а – зображення; б – інтерпретація, в – таблиця істинності

Операції I (AND, &) та I-НІ (NAND). Операція I (кон'юнкція) — **логічне множення**. Зміст її, як і операції АБО, співпадає з назвою, наприклад: «Станок увімкнеться — якщо ввімкнути вимикач двигуна і ввімкнути систему охолодження». Функцію I можна реалізувати послідовним з'єднанням нормально розімкнених кнопок (рис. 2.6 б): коло замкнено — лампа горить, якщо натиснуті і перша, і друга кнопки.

Таким чином, логічна одиниця на виході елемента I згідно з його таблицею істинності (рис. 2.6 в), виникає за умови одиничних сигналів на всіх входах (котрих, як і у елемента АБО, може бути більше двох). Функція I-НІ, що реалізується відповідним елементом, є інверсією функції I (рис. 2.7).

Операція виключне АБО (XOR, =1) – додавання по модулю два. Ця операція може бути визначена як нерівнозначність, тобто вихідний сигнал приймає значення 1 лише у випадках, коли сигнали на його входах відрізняються (рис. 2.8 б), чому відповідає вираз:

$$Y = (\overline{X_1} \wedge X_2) \vee (X_1 \wedge \overline{X_2}).$$

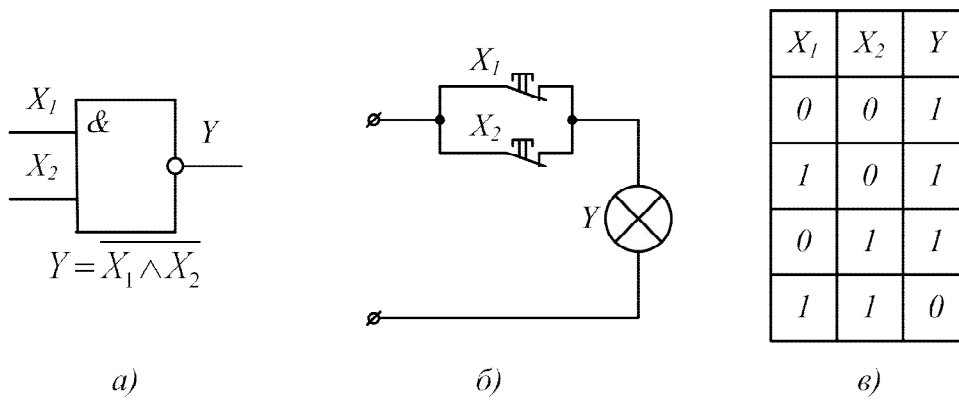


Рис.2.7. Операція І-НІ: а – зображення; б – інтерпретація, в – таблиця істинності

Відповідний логічний елемент (рис. 2.8 а) має два входи та один вихід. Таку функцію звичайно використовують для порівняння двох змінних за значенням.

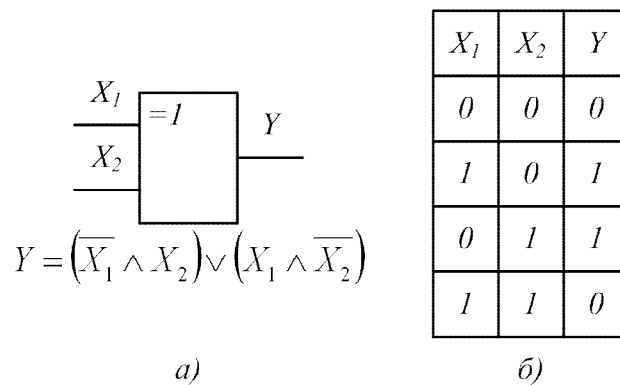


Рис. 2.8. Операція виключне АБО: а – зображення; б –таблиця істинності

На основі розглянутих типів логічних елементів можна реалізувати яку завгодно логічну функцію.

Пристрої, вихідні функції яких однозначно визначаються вхідними логічними змінними, названо комбінаційними. Для побудови комбінаційних пристроїв звичайно використовують два типа базових логічних елементів – елемент АБО-НІ (елемент Шеффера) та елемент І-НІ (елемент Пірса).

Найпоширеніші електронні реалізації бітових операцій реалізуються за допомогою транзисторних ключів на основі технологій: резистивно-транзисторної логіки (РТЛ), діодно-транзисторної логіки (ДТЛ), емітерно-зв'язаної логіки (ЕЗЛ), транзисторно-транзисторної логіки (ТТЛ), N-МОП логіки, КМОП логіки та ін.

2.3 Спеціальні цифрові схеми – асинхронні тригери

Складні пристрої, які формуються з розглянутих логічних елементів, не «запам'ятовують» попередній стан чергової комбінації вхідних сигналів. Такою властивістю «запам'ятовування» володіють спеціальні цифрові схеми, які називають **тригерами**.

Тригер — це пристрій з двома стійкими станами, в кожному з яких він має змогу знаходитись як завгодно довго. Електричний аналог тригера — вимикач. На відміну від кнопки, вимикач залишається у ввімкненому або вимкненому стані після короткочасної дії на нього (ручного вмикання або вимкнення), тобто, «запам'ятовує» цю дію.

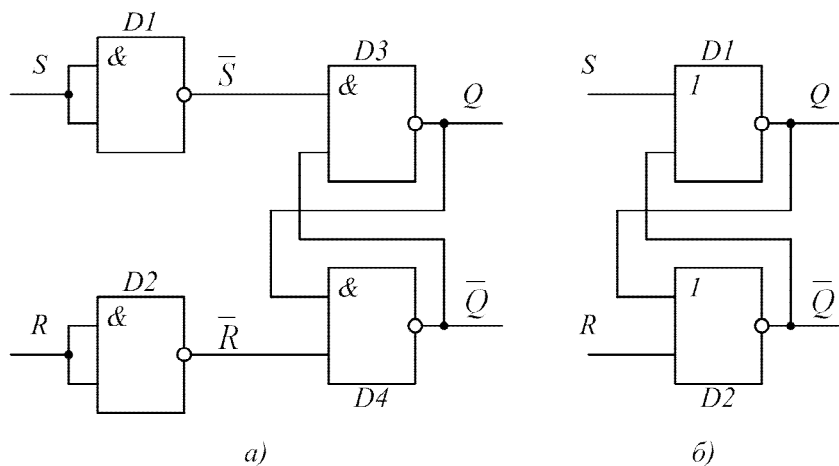


Рис.2.9. RS-тригер на елементах: а- 2І-НІ, б- 2АБО-НІ

Розглянемо основні типи тригерів.

RS-тригер. Це найпростіший тригер, схема якого на основі елементів 2І — НІ наведена на рис.2.9 а. Він має два входи — S і R і два виходи — Q та \bar{Q} . Завдяки зворотнім зв'язкам, що є в схемі (вихід елемента D3 підключений до одного з входів елемента D4, і навпаки), схема має змогу тривалий час знаходитись в одному з стійких станів з $S = R = 0$. Наприклад, якщо із вмиканням схеми випадково встановиться $Q = 0$, то на виході \bar{Q} встановлюється логічна одиниця, тобто сигнал 1 з виходу елемента D3 надходить на один з входів елемента D4. Змінити стан схеми можна, якщо подати сигнал $S = 1$ разом із $R = 0$. Тоді на перший (верхній за рис.2.9а) вхід елемента D3 подано логічний нуль, внаслідок чого на його виході з'являється логічна 1. При цьому на обох входах елемента D4 є одиниці, а так на його виході встановлюється логічний нуль. Цей сигнал утримує елемент D3 у стані $Q = 1$ і після переходу сигналу в нуль $S = 0$. Подання на входи комбінації $S = 0, R = 1$ знов переводить тригер у початковий стан. Таким чином, робота тригера аналогічна роботі вимикача: комбінація $S = 1, R = 0$ — «вмикання» (тобто, $Q = 1$), комбінація $S = 0, R = 0$ — зберігання попереднього (тобто, «ввімкненого» стану), комбінація $S = 0, R = 1$ — «вимкнення» ($Q = 0$), і т.д. З принципу роботи витікає, що стан виходів тригера завжди протилежний. Ці обставини (відповідно і для других типів тригерів) відображено в його позначенні: Q і \bar{Q} відповідно. Аналогічний пристрій можна виконати на елементах АБО-НІ (рис.2.9 б).

Входи тригера умовно названі згідно з описаним вище алгоритмом його роботи: вхід S – від англ. SET – встановлювати, вмикати, і вхід R – від англ. RESET — повернення, вимкнення. Позначення RS-тригерів, що виготовляються звичайно у вигляді відповідної ІМС, наведено в табл.2.1.

Для опису можливих станів тригера в залежності від різних

комбінацій вхідних сигналів використовують спеціальну **таблицю перемикань** (стовпчик 4 табл.2.1). Згідно з цією таблицею, комбінації $S = 1, R = 0$ і $S = 0, R = 1$ є інформаційними, вони викликають відповідні перемикання тригера (запис інформації), а комбінація $S = 0, R = 0$ — зберігання інформації (стан тригера після її появи не змінюється). Таким чином, тригер — це найпростіша комірка пам'яті. Комбінація $S = 1, R = 1$ є забороненою, тобто, по-перше, встановлює на обох виходах одиницю, що йде всупереч прийнятому статусу виходів (протилежність значень), а по-друге, після зняття цієї комбінації тригер рівно імовірно встановлюється в один з двох стійких станів, що не може бути передбачено заздалегідь. На основі RS-тригера створені інші, більш складні за структурою тригери.

RS-тригер реагує на вхідні сигнали одразу з надходженням їх на входи S і R , згідно з таблицею перемикань, тому *такий тригер названо асинхронним*.

2.4 Синхронні тригери

При створенні складних цифрових систем, часто необхідно, щоб тригер реагував на вхідні сигнали лише у визначені моменти часу. Такі тригери існують, їх названо *синхронними*.

Ці тригери мають додатковий вхід C , який названо входом синхронізації (табл. 2.1). Синхронний тригер реагує на сигнали інформаційних входів лише у момент зміни сигналу (з 0 в 1, або з 1 в 0 в залежності від конкретного типу тригера) на вході C .

Перехід з 0 в 1 (рис. 2.10 а) прийнято називати *позитивним синхроімпульсом*, а з 1 в 0 (рис. 2.10 б) - *негативним синхроімпульсом*.

Таблиця 2.1. Позначення тригерів та режими роботи

№ п/п	Тип тригера	Позначення	Таблиця перемикачів				Режим	
1	2	3	4				5	
1	RS-тригер (асинхронний)		S	R	Q_{i+1}	\bar{Q}_{i+1}	зберігання запис інформації запис 0 заборонена комбінація	
			0	0	Q_i	\bar{Q}_i		
			1	0	1	0		
			0	1	0	1		
2	D-тригер (синхронний)		D		Q_{i+1}	\bar{Q}_{i+1}	синхронний запис інформації	
			0		0	1		
			1		1	0		
3	JK-тригер (синхронний)		J	K	Q_{i+1}	\bar{Q}_{i+1}	зберігання синхронний запис рахування	
			0		0	Q_i		\bar{Q}_i
			1		0	1		0
			0		1	0		1
			1		1	\bar{Q}_i		Q_i
1		1	Q_i	\bar{Q}_i				

D-тригер є найпростішим синхронним тригером. Його позначення і таблиця перемикачів наведені в табл. 2.1. D-тригер виконує перенесення стану інформаційного входу D на вихід Q у момент надходження синхроімпульсу.

Розглянемо роботу D-тригера по діаграмі на рис. 2.11.

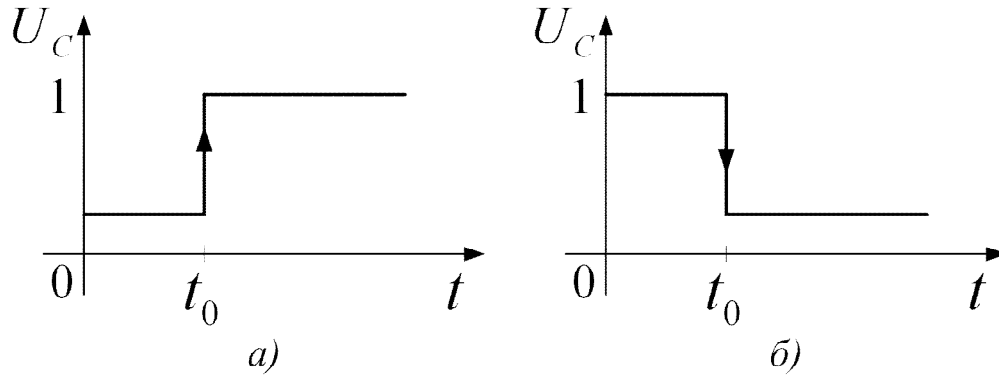


Рис.2.10. Фронти синхроімпульсів: а – позитивний; б - негативний

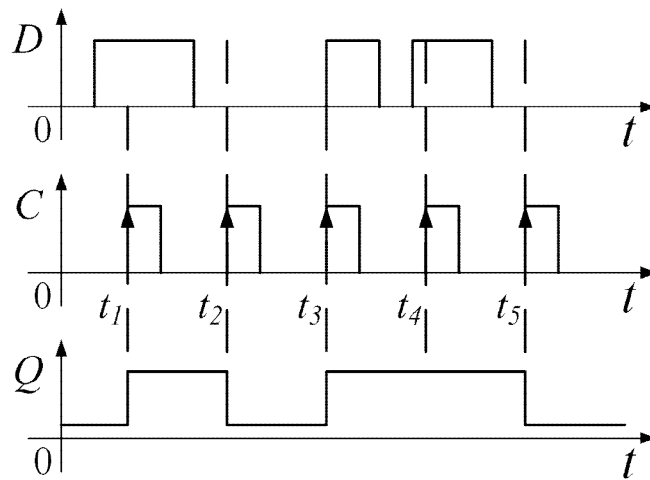


Рис.2.11. Діаграма роботи D-тригера

Припустимо, що тригер реагує на позитивні синхроімпульси (показані стрілками на діаграмі). У момент часу t_1 виконується перенесення стану входу D-логічної 1 на вихід Q. Цей стан зберігається до моменту часу t_2 , тобто до надходження наступного синхроімпульсу. В момент часу t_2 на виході тригера встановлюється $Q = 0$ (хоча сигнал на вході D установлений на нуль значно раніше). Аналогічне перемикання стається в момент часу t_3 , причому одиничний стан виходу Q зберігається і після синхроімпульсу в момент часу t_4 тому, що в цей момент $D = 1$. В момент часу t_5 тригер перекидається в нульовий стан, виконується перенесення сигналу $D = 0$ на вихід Q. Алгоритм функціонування D-

тригера наведений в його таблиці перемикань (стовпчик 4 табл.1). Для таблиць перемикань синхронних тригерів прийняті позначення: Q_i , Q_{i+1} — стан виходу відповідно до i після надходження синхроімпульсу.

D-тригер широко використовується як елемент пам'яті, а також у складі більше складних пристроїв — лічильників і регістрів, що розглядаються нижче.

JK-тригер — є більш складним універсальним синхронним тригером. Його позначення і таблиця перемикань подані в табл. 1. Як витікає з таблиці перемикань (окрім вхідної комбінації $J=1, K=1$), JK-тригер функціонує так, як і RS-тригер, але з синхронізацією за входом С (тобто, виконує функцію синхронного RS-тригера): при $J=0$ і $K=0$ в момент надходження синхроімпульсу стан тригера не змінюється, комбінації $J=1, K=0$ і $J=0, K=1$ за синхроімпульсом надходять на відповідні виходи тригера. Комбінація $J=1, K=1$ для JK-тригера не є забороненою. У цьому випадку на кожний синхроімпульс тригер перекидається у протилежний стан. (Наприклад, якщо був стан $Q=1, \bar{Q}=0$, то після першого синхроімпульсу стає $Q=0, \bar{Q}=1$, після наступного — $Q=1, \bar{Q}=0$ і т.д.). Такий режим називається *режимом рахування* і JK-тригери в такому режимі використовуються для побудови лічильників. Тригери всіх розглянутих типів виготовляються в складі серії цифрових мікросхем. При цьому конкретні пристрої набувають особливостей, що не розглядаються у цьому розділі. Наприклад, D-тригер окрім входів D і С може мати інверсні асинхронні входи S і R. Сигналом $S=0$ (коли $R=1$) тригер, незалежно від стану інших входів, встановлюється у стан $Q=1$, а сигналом $R=0$ (коли $S=1$) — у стан $Q=0$. Тобто, цей конкретний тригер виконує функції як D, так і RS-тригера. Додаткові асинхронні входи S і R використовуються у багатьох пристроях в якості входів для попереднього встановлення стану.

2.5 Генератори імпульсів в мікропроцесорних системах

В якості генераторів тактових імпульсів в МПС використовуються мультівібратори. Вони призначені для формування прямокутних імпульсів.

Мультівібратори - це автоколивальні електронні пристрої, що періодично змінюють свій стан — перекидаються з "0" в "1" і навпаки.

Схема мультівібратора на елементах 2І-НІ приведена на рис. 1, а, часові діаграми роботи — на рис. 2.12 б. За побудовою схема нагадує RS-тригер і використовує два зворотні зв'язки з виходу одного з елементів D1 і D2 на вхід іншого через конденсатори C_1 і C_2 . Входи елементів через резистори R_1 і R_2 з'єднані зі спільною точкою (СТ) схеми. Останні утворюють коло заряду конденсаторів ("+" джерела живлення, C_1 , R_2 , СТ) або ("+", C_2 , R_1 , СТ) якщо на відповідному виході Y_1 або Y_2 високий рівень напруги — "1" і він з'єднаний з виводом "+" джерела. При перемиканні у стан "0" виходи Y_1 або Y_2 з'єднуються з СТ, утворюючи коло розряду відповідного конденсатора. Елементи D1 і D2 знаходяться у протилежних станах $Y_1="1"$, $Y_2="0"$, коли $X_1="0"$, $X_3="1"$ і навпаки, $Y_1="0"$, $Y_2="1"$ коли $X_1="1"$, $X_3="0"$. Час перебування в кожному з станів визначається опором R і ємністю C (сталю часу перезаряду $\tau = RC$). При вмиканні живлення схема випадковим чином встановлюється в один із вказаних станів. Припустимо, що це стан, коли $Y_1="1"$ (U_{Y1} дорівнює напрузі джерела U). Конденсатор C_1 починає заряджатися через R_2 з напруги $U_{C1} = 0$. У подальшому U_{C1} збільшується за експотенційним законом (стала часу $\tau_1=R_2*C_1$). Відповідно напруга на резисторі R_2 (вхід X_3) спочатку дорівнює напрузі джерела $U_{X3} = U$ і по мірі заряду зменшується.

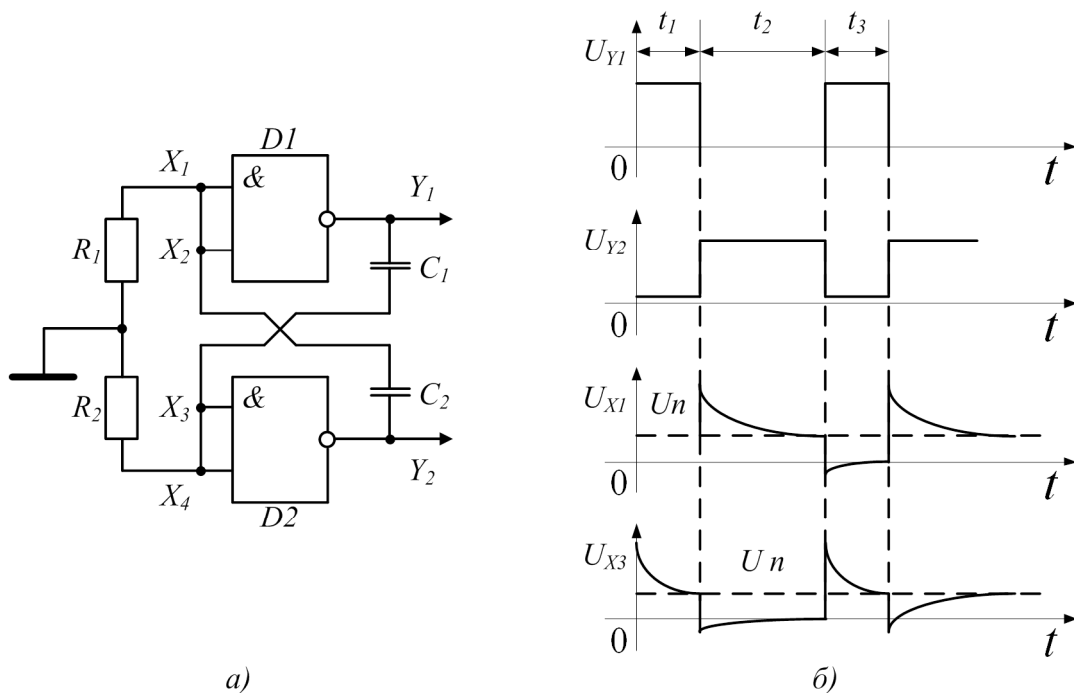


Рис.2.12. Мультивібратор на елементах 2І-НІ: а – схема, б – діаграма

Таким чином, деякий час t_1 на вході елемента D2 утримується високий рівень напруги, що сприймається як $X_3 = "1"$, а на виході $Y_2 = "0"$. Коли U_{x3} стає менша за поріг U_n (рис. 2.12 б), що сприймається як $X_3 = "0"$ елемент D2 переходить у стан $Y_2 = "1"$ і знаходиться у ньому продовж часу t_2 . Встановлення $Y_2 = "1"$ призводить до початку заряду C_2 через R_1 – процеси аналогічні заряджанню C_1 (стала часу $\tau_2 = R_1 \times C_2$). Одночасно елемент D1 переходить у стан $Y_1 = "0"$, що призводить до розряду C_1 – струм через R_2 і, відповідно, напруга U_{x3} змінюють напрямок і поступово зменшуються до нуля. Далі процеси в колі періодично повторюються ($t_3 = t_1$). Таким чином на виходах кожного з елементів формується імпульсна напруга з частотою, що визначається параметрами відповідних резисторів і конденсаторів. Відношення тривалості інтервалів: $t_1/t_2 = \tau_1/\tau_2$

Кварцовий генератор. Стабільність частоти розглянутих RC — генераторів невисока. Вона залежить від температурних коефіцієнтів зміни параметрів елементів (R, C). Суттєвого підвищення стабільності частоти досягають використанням кварцових резонаторів, що перетворюють енергію електричного поля в енергію механічних коливань пластини з кварцу. Частота коливань визначається власною резонансною частотою коливань пластини. При цьому резонатор являє собою високостабільний коливальний контур з малими витратами енергії. Позначення кварцового резонатора C і схема використання наведені на рис. 2.13. У якості джерела енергії для збудження коливань і їхнього підсилення до необхідного рівня використано відповідний електронний підсилювач (П).

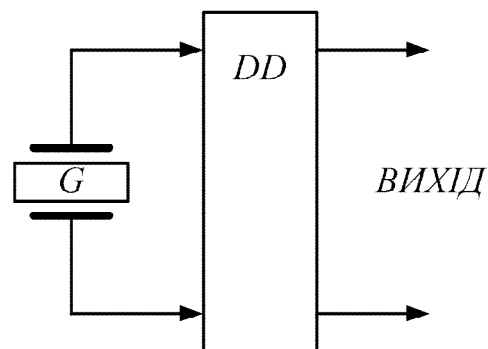


Рис.2.13. Кварцовий генератор

Контрольні запитання

1. Дати визначення ключовому транзисторному підсилювачу.
2. Дати визначення поняттю – відеоімпульс.
3. Дати визначення поняттю – радіоімпульс.
4. Що таке режим насичення транзисторного ключа?
5. Що таке режим відсікання транзисторного ключа?
6. Назвіть основні параметри відеоімпульса.
7. Дати визначення операції «АБО».

8. Дати визначення операції «І».
9. Дати визначення операції виключне «АБО».
10. Який математичний апарат використовується для логічних змінних?
11. Дати визначення операції логічне множення.
12. Дати визначення операції логічне додавання.
13. Дати визначення синхронного тригера.
14. Принцип роботи D-тригера.
15. Принцип роботи JK-тригера.
16. Дати визначення позитивному та негативному синхроімпульсу.
17. Який режим називають режимом рахування?
18. Дати визначення мультівібратора.
19. Принцип роботи мультівібратора.
20. Що таке кварцовий генератор?
21. Завдяки чому формується ширина імпульсу.
22. Якими елементами задається стала часу мультівібратора?

Розділ третій

ПРЕДСТАВЛЕННЯ ТА ДІЇ НАД ДВІЙКОВИМИ ЧИСЛАМИ В МІКРОПРОЦЕСОРНИХ СИСТЕМАХ

3.1 Представлення чисел в цифрових системах

У цифрових пристроях сигнал має лише два значення — 0 і 1 (низький і високий рівень напруги, вимк./вимкн.), тому числа у таких пристроях можна подати лише у бінарній системі обчислення.

З таблиці видно, що числам від 0 до 9, які записані звичайними десятковими цифрами 0,1,2...9 відповідають двійкові числа від 0 до 1001, тобто ці числа мають чотири двійкові розряди. При цьому згідно з табл.3.1. таким чотирирозрядним двійковим числом можна представити десяткові числа і більші, ніж 9, аж до числа 1111, якому відповідає десяткове число 15.

Двійкові числа можна записати в шістнадцятковій системі обчислювання. У цій системі цифри (правий стовпчик табл.1) від 0 до 9 — ті ж, що і в десятковій, а цифри від 10 до 15 представлені першими шістьма літерами латинського алфавіту.

Щоб відрізнити шістнадцяткове число (скорочено Нех — число від слова hexadecimal) від десяткового, наприкінці його пишуть букву Н або на початку \$). Наприклад, 47 — десяткове число, 47Н — шістнадцяткове. Якщо Нех — число починається з букви, то попереду пишуть 0: 0АН, 0С3Н та ін. Аналогічно, наприкінці двійкового числа іноді (якщо є вірогідність сплутати його з десятковим) пишуть букву В (від слова binary): 101 — десяткове число, 101В — двійкове. Переведення чисел з однієї системи в іншу і дії над ними виконуються за відомими

правилами арифметики (користуючись вагою відповідних розрядів).

Таблиця.3.1. Переведення чисел

Десяткове число	Двійкове число				Шістнадцяткове число
	Розряди				
	$2^3=8$	$2^2=4$	$2^1=2$	$2^0=1$	
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	2
3	0	0	1	1	3
4	0	1	0	0	4
5	0	1	0	1	5
6	0	1	1	0	6
7	0	1	1	1	7
8	1	0	0	0	8
9	1	0	0	1	9
10	1	0	1	0	A
11	1	0	1	1	B
12	1	1	0	0	C
13	1	1	0	1	D
14	1	1	1	0	E
15	1	1	1	1	F

У загальному випадку в будь-якій позиційній системі з основою q довільне число n -розмірності $A = a_{n-1}a_{n-2} \dots a_2a_1a_0$ записується поліномом

$$A = a_{n-1}q^{n-1} + a_{n-2}q^{n-2} + \dots + a_2q^2 + a_1q + a_0.$$

Число q служить ваговим коефіцієнтом для кожного розряду і називається основою системи числення. Основа системи числення може бути довільним числом, навіть дробом.

Наприклад:

$$347 = 3 * 10^2 + 4 * 10^1 + 7 * 10^0 = 3 * 100 + 4 * 10 + 7 * 1.$$

Аналогічно:

$$\begin{aligned} 10101\text{B} &= 1 * 2^4 + 0 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 = \\ &= 1 * 16 + 0 * 8 + 1 * 4 + 0 * 2 + 1 * 1 = 21; \end{aligned}$$

$$135\text{H} = 1 * 16^2 + 3 * 16^1 + 5 * 16^0 = 256 + 48 + 5 = 309.$$

Двійкове число легко перевести у шістнадцяткове (і навпаки), користуючись наступним правилом: двійкове число розподіляють на частини по чотири розряди, починаючи справа. Якщо крайня частина, що знаходиться ліворуч при цьому стає меншою, ніж чотири розряди, то додають необхідну кількість нулів. Потім кожен частину замінюють шістнадцятковою цифрою згідно з таблиці.

Наприклад: $1011010011\text{B} = 10.1101.0011 = 0010.1101.0011 = 2\text{D}3\text{H}$.

Аналогічно виконується і зворотне перетворення:

$$7\text{САН} = 0111.1100.1010 = 11111001010\text{В}.$$

Десяткове число переводять у двійкове або шістнадцяткове шляхом ділення на 2 або на 16 відповідно. Наприклад:

$$\begin{array}{r|l}
 434 & 2 \\
 \hline
 434 & 217 \\
 \hline
 0 & 216 \\
 & \underline{1} \\
 & 108 \\
 & \hline
 & 108 \\
 & \hline
 & 54 \\
 & \hline
 & 54 \\
 & \hline
 & 27 \\
 & \hline
 & 26 \\
 & \hline
 & 13 \\
 & \hline
 & 12 \\
 & \hline
 & 6 \\
 & \hline
 & 6 \\
 & \hline
 & 3 \\
 & \hline
 & 2 \\
 & \hline
 & 1
 \end{array}$$

Потім записують, починаючи з останнього добутку, справа наліво, залишки, що отримані від ділення: $434 = 110110010\text{В}$

Аналогічно:

$$\begin{array}{r|l}
 435 & 16 \\
 \hline
 432 & 27 \\
 \hline
 3 & 16 \\
 & \hline
 & 11
 \end{array}$$

У цьому випадку, якщо деякий залишок знаходиться в межах від 10 до 15, його записують шістнадцятковою цифрою згідно з таблицею. У даному випадку другий залишок $11 = 0\text{ВН}$, тоді $435 = 1\text{ВЗН}$.

Цифрові системи на основі мікропроцесорів оперують з двійковими числами, що мають вісім розрядів (або кратним 8, 16, 24, 32, 64 розряди і т.д.). Таке восьмирозрядне двійкове число названо **байтом**, а кожний розряд байта названо **біт**. Для скорочення запису байт звичайно

записують дворозрядним Нех — числом: $10110011 = 0B3H$, $01100111 = 67H$ та ін.

3.2 Основні дії над двійковими числами

Арифметичні дії над двійковими числами виконуються подібно відповідним діям з десятковими, порозрядно (побітно), починаючи з молодшого розряду (біту), що записаний у останній позиції праворуч.

Приклади виконання основних операцій з двійковими числами (операндами A і B), що мають вісім розрядів, представлені в табл.3.2.

Додавання ($A + B$) — починаємо з біту 0: $0 + 1 = 1$. Аналогічний результат отримуємо для біту 1: $1 + 0 = 1$. Для біту 2: $1 + 1 = 0$ і отримуємо перенесення 1 у старший розряд (біт 3). Відповідно для біту 3 з урахуванням перенесення:

$$(0 + 0) + 1 = 1.$$

Далі діємо аналогічно і у підсумку отримуємо суму з перенесенням 1 у старший (біт перенесення) розряд.

Віднімання ($A - B$). Оскільки біт 0 операнда B більший ніж у операнда A , то необхідно запозичити 1 з старшого розряду A (біт 1) при цьому результат дорівнює 1. З урахуванням цього для біту 1 маємо $0 - 0 = 0$. Аналогічні результати отримуємо для бітів 2-4, що мають однакові значення. Для бітів 5 і 6 необхідно також запозичувати 1 з старшого розряду.

Аналогічним чином можна виконати множення і ділення. Так при множенні послідовно перемножуємо байт A на біт 0 операнда B , потім на біт 1 і так далі. Надалі для отриманих результатів множень, починаючи з другого, здійснюємо зсув на 1 біт ліворуч і виконуємо їхнє додавання. Очевидно, що в підсумку отримуємо число, що має 16 бітів.

Таблиця 3.2. Основні операції над двійковими числами

Операція	Біт перенесення	Біт 7	Біт 6	Біт 5	Біт 4	Біт 3	Біт 2	Біт 1	Біт 0
Операнд А		1	0	0	1	0	1	1	0
Операнд В		0	1	1	1	0	1	0	1
Арифметичні операції									
Сума (А+В)	1	0	0	0	0	1	0	1	1
Різниця (А-В)		0	0	1	0	0	0	0	1
Логічні операції									
Логічне додавання OR (АБО)		1	1	1	1	0	1	1	1
Виключне АБО (XOR)		1	1	1	0	0	0	1	1
Логічне множення AND (І)		0	0	0	1	0	1	0	0
Інверсія А-NOT (НІ)		0	1	1	0	1	0	0	1
Спеціальні операції									
Зсув ліворуч (А)	1	0	0	1	0	1	1	0	1
Зсув праворуч (В)	1	1	0	1	1	1	0	1	0

Операції зсуву звичайно виконуються циклічно через біт перенесення. Так при зсуві ліворуч старший розряд через біт перенесення заноситься до молодшого, інші розряди зсовуються на розряд ліворуч.

Зсув ліворуч еквівалентний множенню на 2, що у двійковій формі

дорівнює 00000010, зсув праворуч - відповідно діленню на 2.

3.3 Комбінаційні пристрої в мікропроцесорних системах

Логічні елементи і цифрові пристрої побудовані на їхній основі (без зворотних зв'язків) можна розподілити на два класи:

- **комбінаційні пристрої** — вихідні функції яких однозначно визначаються вхідними логічними сигналами в момент їх надходження;
- **послідовні пристрої** — вихідні функції котрих, окрім вхідних логічних сигналів, визначаються попереднім їх станом, а перехід у кожний наступний стан здійснюється в момент надходження спеціального синхроімпульсу.

Розглянемо комбінаційні пристрої, що використовуються у складі серій цифрових ІМС, які найчастіше використовуються при створенні цифрових мікропроцесорних систем.

Дешифратор — це пристрій, що має n (від 2 до 4) входів і 2^n виходів. Дешифратори перетворюють **двійковий код у позиційний** і бувають таких видів: 2x4, 3x8, 4x10 и 4x16 (наприклад, 2x4 — 2 входи і чотири виходи).

На рис. 3.1 наведено позначення дешифратора 3x8 і його таблиця істинності. З таблиці витікає, що сигнал логічної одиниці формується на тому виході, номер якого представлено двійковим числом, які формуються вхідними сигналами.

Наприклад, з вхідною комбінацією 101 одиниця є на виході Q5 (тому, що $101_2 = 5$), і т.д. Входи дешифратора звичайно позначають згідно з вагою розрядів двійкового числа (1, 2, 4, ...).

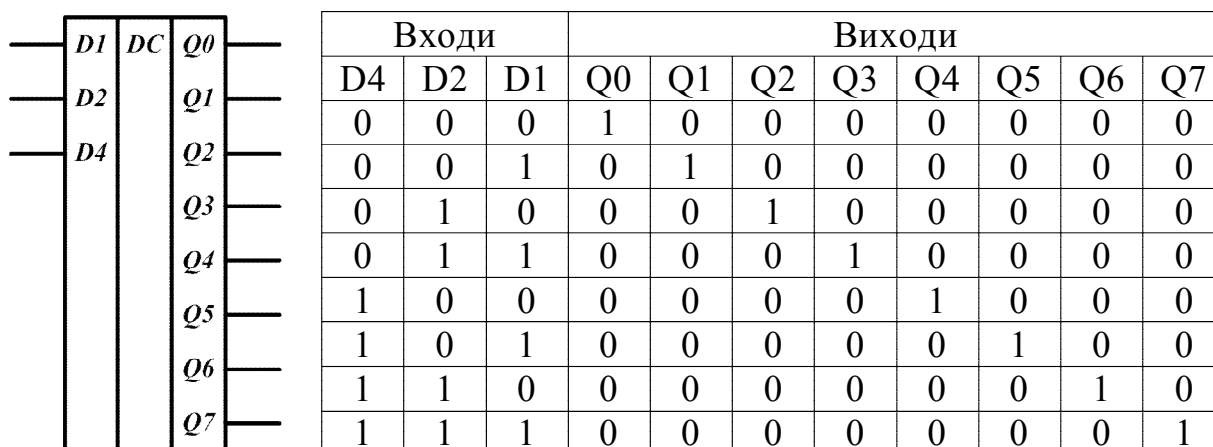


Рис.3.1. Позначення дешифратора 3x8

Конкретні дешифратори у складі серій ІМС набувають деяких особливостей: інверсні виходи (замість одиниці на обраному виході є нуль, а на інших — одиниці), наявність спеціальних додаткових входів, що дозволяють роботу схеми і т.д.

Дешифратори використовують у схемах індикації, а також у складі складних обчислювальних систем, у тому числі мікропроцесорних, для обирання (адресації) конкретного елемента (комірки пам'яті) та ін. Наприклад: індикатор відображення цифрової інформації — чисел від 0 до 9 має десять входів. Висвітлювання відповідної цифри стається з подаванням 1 (0) на відповідний вхід. Для керування індикатором двійковий код, що відповідає цифрам, подається крізь дешифратор 4x10 (скорочений варіант дешифратора 4x16, де виходи 10-15 і відповідні їм входні коди не використовуються)

Мультиплексор— це своєрідний електронний перемикач, який має n (1-4) адресних входів, інформаційні входи і один вихід. При цьому на вихід надходить сигнал з того входу, номер якого представлено двійковим числом на адресних входах. На рис. 3.2 наведено позначення

мультиплексора 4x1 (4 входи на 1 вихід) і його спрощена схема, що пояснює роботу пристрою. Так, сигнал з входу D2 надходить на вихід (стан «перемикача» представлено на рис. 3.2 б), якщо на адресних входах є комбінація $10_{\text{B}} = 2$ (тобто, «A2» = 1, «A1» = 0). Як і дешифратор, мультиплексор використовується в схемах відображення інформації і в складі обчислювальних систем. Наприклад: мультиплексор можна використовувати для послідовного опитування n — комірок пам'яті з наступним передаванням інформації по одній лінії. При цьому в конкретний момент часу інформація зчитується лише з однієї комірки, адреса(номер) якої представлено двійковим числом на адресних входах.

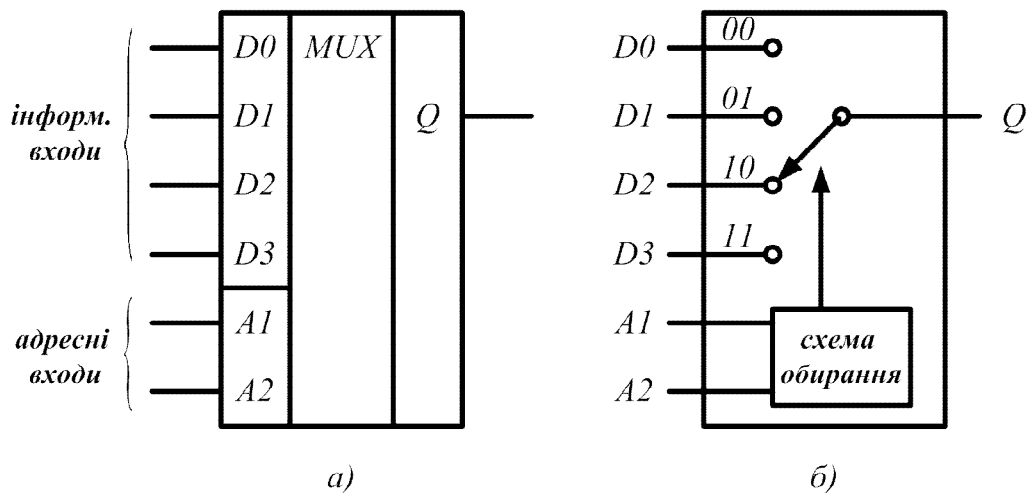


Рис. 3.2. Позначення та спрощена схема мультиплексора

Суматор — здійснює складання двох чисел, що представлені у двійковій формі, а також значення на спеціальному вході перенесення (рис. 3.3).

Суматор має входи $A1...An$ для першого числа (як правило, $n = 1 - 4$), $B1...Bn$ — для другого числа, вхід перенесення а також виходи $S1...Sn$ і вихід перенесення P . Позначення дворозрядного суматора наведено на рис. 3.3.

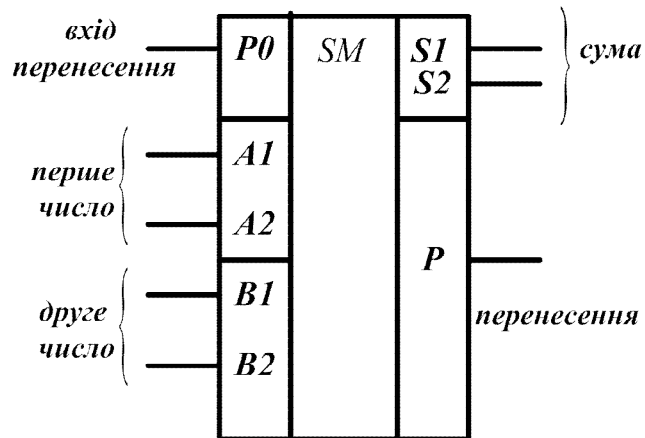


Рис. 3.3. Суматор

Скорочена таблиця істинності для випадку двох дворозрядних чисел представлена в табл.3.3.

Таблиця 3.3. Таблиця істинності

Входи						Виходи			
Десяткове значення	A1	A2	Десяткове значення	B1	B2	Десяткове значення	P	S1	S2
0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	1	0	0	1
1	0	1	1	0	1	2	0	1	0
1	0	1	2	1	0	3	0	1	1
3	1	1	1	0	1	4	1	0	0
3	1	1	2	1	0	5	1	0	1
3	1	1	3	1	1	6	1	1	0

Як видно з таблиці, на виходах S1 і S2 формується двійковий код числа, що дорівнює сумі чисел, які подані двійковими кодами на входах відповідно. Якщо сума цих чисел перевищує три (11В), то відбувається перенесення одиниці в третій розряд — розряд перенесення P. Наприклад,

сума чисел $2 = 10_B$ та $3 = 11_B$ дорівнює $5 = 101_B$ (шостий рядок таблиці істинності).

Наявність входу перенесення $P0$ дозволяє об'єднувати суматори для підсумку чисел з більшою кількістю розрядів. Так, якщо брати два суматора, що аналогічні наведеному на рис. 3.3, і з'єднати вихід P першого з входом $P0$ другого, то отримуємо чотирирозрядний суматор (рис. 3.4). На вхід $P0$ першого суматора при цьому необхідно подати нуль. Таким способом можна отримати потрібний суматор іншої розрядності.

Необхідно відзначити, що ми розглянули лише важливіші комбінаційні пристрої з багатьох, що існують у складі серій ІМС, про які згадувалось вище.

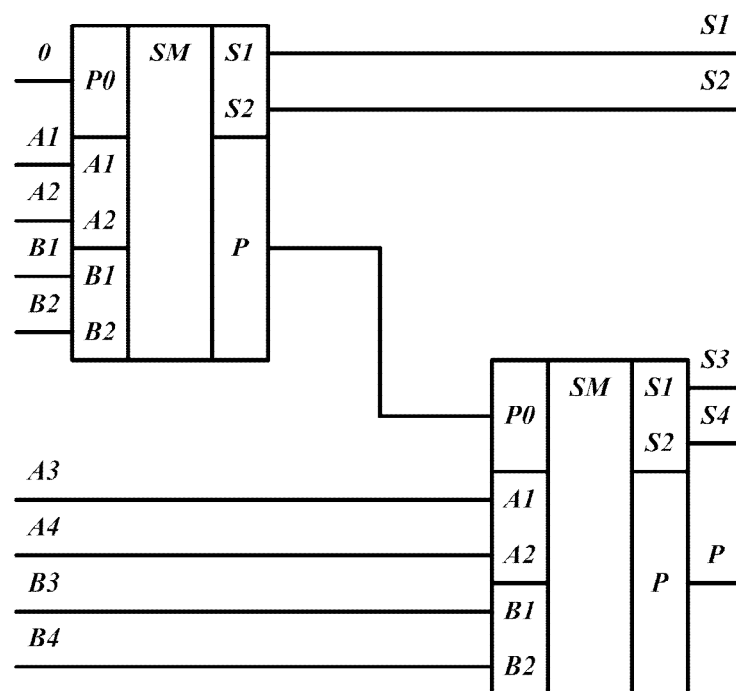


Рис. 3.4. Об'єднання суматорів

3.4 Послідовні пристрої в мікропроцесорних системах

Окрім комбінаційних, існує і ряд послідовні цифрових пристроїв, що є основою цифрових мікропроцесорних систем. Розглянемо деякі найбільш важливі їхні види.

Лічильник. Цей пристрій має один вхід підрахунку і n виходів, на яких формується n -розрядне двійкове число, що дорівнює кількості імпульсів, які надходять на вхід. Лічильник будується на основі синхронних тригерів. Розглянемо роботу найпростішого послідовного дворозрядного лічильника на основі JK-тригерів(рис. 3.5). Для переводу тригерів у режим рахування на входи J і K подано сигнали 1. Імпульси, що рахуються, подаються на вхід синхронізації першого тригера, вихід якого з'єднано з входом синхронізації другого тригера.

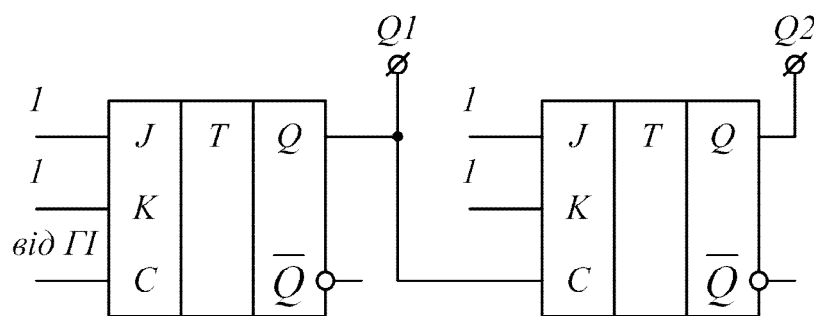


Рис. 3.5. Двійковий двох розрядний лічильник

Нехай тригери реагують на негативний синхроімпульс і знаходяться у стані $Q1=0$ і $Q2=0$. З кожним синхроімпульсом тригери перекидаються у протилежний стан. Так, в момент часу t_1 (рис.3.6) першим синхроімпульсом (що подано від генератора імпульсів ГІ) на виході $Q1$ першого тригера встановлюється рівень $Q1 = 1$.

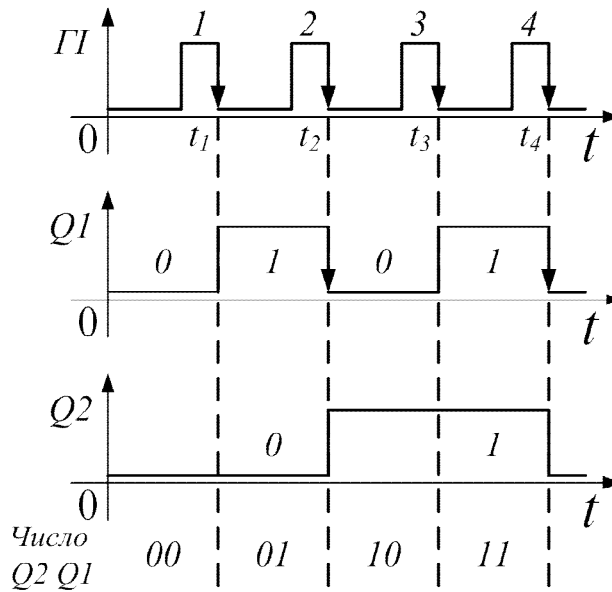


Рис. 3.6. Діаграми роботи лічильника

Другим синхроімпульсом перший тригер встановлюється у стан $Q1 = 0$. Одночасно згідно з сигналом на вході $C = Q1$ другий тригер перемикається у стан $Q2 = 1$. Третій синхроімпульс знову встановлює $Q1 = 1$, а четвертий — $Q1 = 0$, і одночасно $Q2 = 0$, повернувши схему у первісний стан. Четвертим синхроімпульсом всі виходи лічильника переводяться в нульовий стан (лічильник обнуляється) і рахування повторюється.

Таким чином, на виходах пристрою формується двійкове число, що дорівнює кількості синхроімпульсів, які надходять на його вхід. Схема містить лише два тригера, отриманий лічильник фіксує числа від 0 до 3 (від $00B = 0$ до $11B = 3$).

Очевидно, що збільшення розрядності шляхом додавання тригерів відповідно збільшує і ємність лічильника. За допомогою трирозрядного двійкового лічильника можна рахувати до 7, а за допомогою чотирирозрядного - до 15.

Лічильники, зазвичай, виготовляють у вигляді завершених

пристроїв у складі серій ІМС. Наприклад, на рис.3.7 показаний чотирирозрядний лічильник (виходи лічильників прийнято позначати вагою відповідних розрядів). Окрім входу рахування C , лічильник має вхід R (скидання).

лічильника в нульовий стан. За допомогою спеціальних схем можна виконати рахування, з довільним коефіцієнтом перерахування (що відрізняється від 2^n , де n -кількість тригерів у схемі). Вони широко використовуються в ролі дільників частоти і схемах рахування у різних пристроях задавання часу та обчислювальних і інших пристроях.

Регістри — це функціональні вузли, що призначені для зберігання інформації, а в деяких випадках — її перетворювання. Існує три види регістрів:

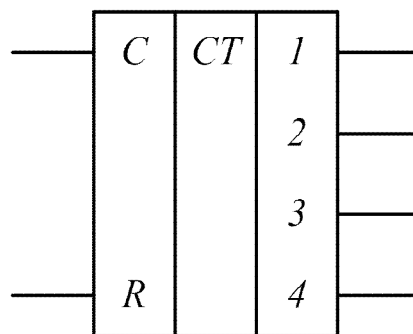


Рис.3.7. Графічне зображення чотирирозрядного лічильника

а) Паралельний регістр.

Це найбільш простий вид регістрів (рис. 3.8). Він являє собою n (наприклад, $n = 4$) D-тригерів, входи синхронізації яких з'єднані. Це дозволяє одночасно (за синхроімпульсом, що подано на вхід C) переносити на їхні виходи і зберігати необхідний час (визначається інтервалом до наступного синхроімпульсу) n — розрядне двійкове число,

що подано на входи. Таким чином, паралельний регістр — це n — розрядна комірка пам'яті. Паралельні регістри виготовляються на 4-8 розрядів. Паралельний регістр позначається (наприклад, чотирирозрядний) згідно з рис. 3.9 а.

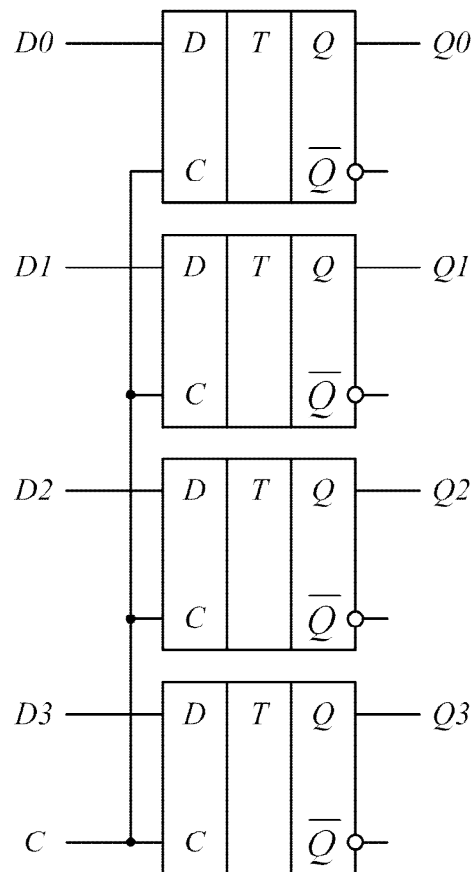


Рис. 3.8. Паралельний регістр

б) Паралельно — послідовний регістр зсуву.

Цей регістр (також виконаний на основі D- тригерів) має n входів для паралельного запису інформації (D_0, D_1, D_2, \dots), вхід синхронізації C , спеціальний вхід керування L і один вихід Q (рис. 3.9 б). Регістр має 2 режими роботи:

1. Режим паралельного запису — коли сигнал $L = 1$. У цьому режимі за синхроімпульсом на вході C інформація з входів D_0 - D_3 (для наведеного на

рис. 3.9,3.10 чотирирозрядного регістру), записується в тригери аналогічно паралельному регістру, що розглянуто вище. При цьому стан молодшого розряду D_0 з'являється на виході Q .

2. Режим зсуву. Якщо встановити на вході керування стан $L=0$ і подавати синхроімпульси на вхід C , то на вихід Q по чергово надходять записані раніше у режимі запису стани входів D_1 , D_2 і D_3 .

Наприклад, якщо було записано (при $L=1$) число 1011 (виходи D_3, D_2, D_1, D_0), то після встановлення $L=0$ на виході по чергово встановлюються: 1 (значення молодшого розряду встановлюється із записом), після першого синхроімпульсу — 1, після другого — 0, після третього — 1. Із подальшим подаванням синхроімпульсів, у залежності від конкретного типу регістра, стани на виході повторюються (тобто знову 1, 1, 0, 1 і т.д.) — циклічний зсув за колом, або дорівнюють 0.

Таким чином, розглянутий регістр, окрім збереження інформації, виконує її перетворювання з паралельної форми в послідовну. Такі пристрої часто використовуються в системах передавання цифрової інформації на відстані тому, що це дозволяє зменшити кількість з'єднувальних проводів.

в) Послідовно-паралельний регістр зсуву.

Цей регістр виконує зворотне перетворювання цифрової інформації — з послідовної форми в паралельну (рис. 3.10). Він має один інформаційний вхід D , вхід синхронізації C , n виходів (у даному випадку — 4) і вхід скидання R . Таблиця перемикань такого регістра наведена на рис. 3.10.

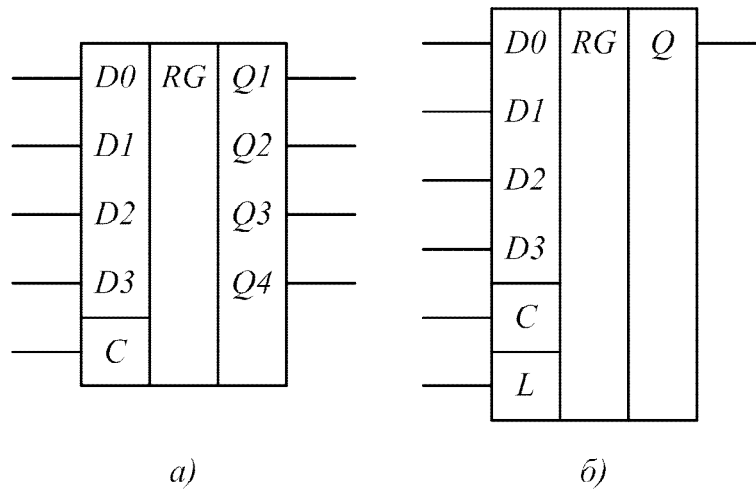


Рис.3.9. Позначення регістрів

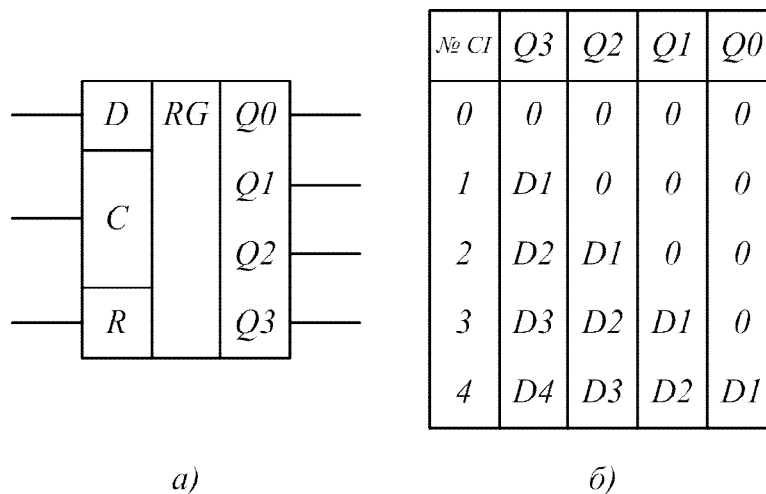


Рис. 3.10. Послідовно-паралельний регістр зсуву

Працює регістр таким чином. Нехай всі тригери регістра переведені в стан логічного нуля (подавання короткочасного сигналу $R=1$). Першим синхроімпульсом стан сигналу з входу D записується в старший ($Q3$) розряд. Другим синхроімпульсом цей стан зсувається з розряду $Q3$ в розряд $Q2$ (стан $Q2$ в $Q1$, $Q1$ в $Q0$), а нове значення з входу D знову записується в розряд $Q3$ і т.д. Якщо, наприклад, поєднати лінією зв'язку вихід регістра, який представлено на рис.3.9 б, з входом даного

регістра і синхронно подавати імпульси на їхні входи С, то обране за приклад число 1011 через 4 такти встановлюється на виходах розглянутого регістра. Цей регістр використовується також в системах передавання інформації і в інших випадках, коли необхідне перетворення інформації з послідовної форми в паралельну.

3.5 Цифро-аналогові і аналого-цифрові перетворювачі

На основі розглянутих раніше пристроїв, можна створити цифрову систему опрацювання інформації, що вирішує достатньо складні завдання, зокрема, цифрові керуючі машини, цифрові вимірювальні прилади і т.д. В цих випадках виникає необхідність перетворення аналогових величин (напруги, струму та ін.) у цифровий код, частіше за все у двійковий, і навпаки. Ці завдання виконують *аналого-цифрові (АЦП)* і *цифро-аналогові перетворювачі (ЦАП)*.

Цифро-аналоговий перетворювач. Схема найпростішого ЦАП наведена на рис. 3.11 а. Якщо припустити, що рівень логічної 1, який подано на всі цифрові входи, однаковий і дорівнює E , а рівень логічного нуля дорівнює 0, то на виході операційного підсилювача установлюється напруга на рівні:

$$U_{\text{вих}} = E * (R_{33}/R)(8 * Q_8 + 4 * Q_4 + 2 * Q_2 + 1 * Q_1) \quad (3.1)$$

де Q_8, Q_4, Q_2, Q_1 — значення цифрового коду (0 або 1) на входах 8,4,2,1.

З приведеної формули витікає, що із зміною коду числа на цифрових входах від 0000В (0) до 1111В (15) напруга на виході системи змінюється від $U_{\text{вих}} = 0$ до $U_{\text{вих}} = 15 * E(R_{33}/R) = 15e$ з дискретністю $E(R_{33}/R)=e$. Робота пояснюється діаграмою рис. 3.11 б, що відображає залежність (3.1).

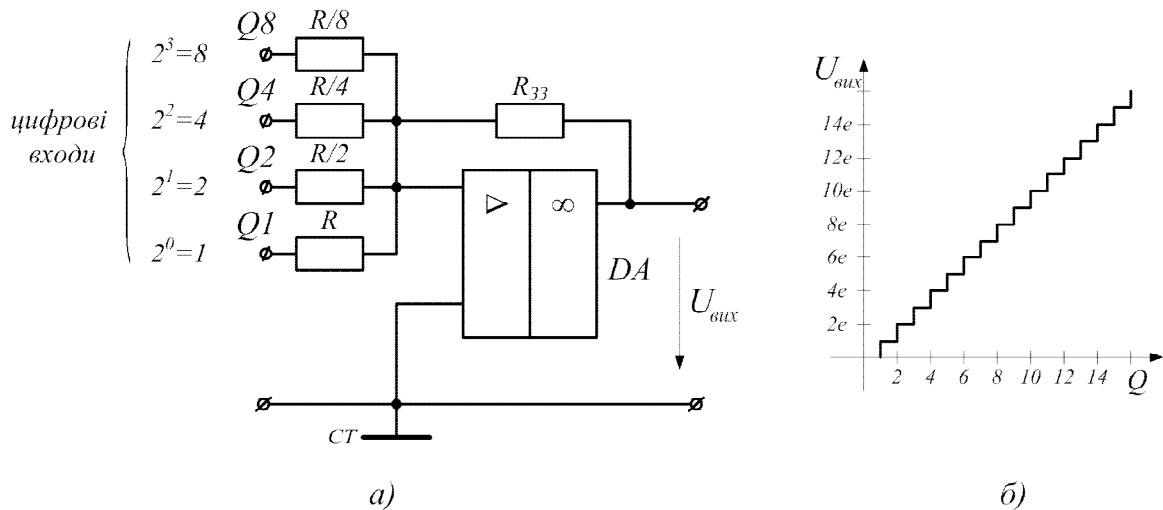


Рис. 3.11. Цифро-аналоговий перетворювач

Очевидно, що чим більше вхідних розрядів має ЦАП, тим меншою є дискретність, і тим вища точність перетворення. ЦАП, які використовуються в МПС, як правило, мають 10... 12 вхідних цифрових розрядів, а також ряд спеціальних схемних рішень, що забезпечують високу точність перетворювання.

Аналого-цифровий (АЦП) перетворювач. Один з принципів побудови АЦП проілюструємо пристроєм, схема якого наведена на рис. 3.12 а. АЦП містить генератор імпульсів ГІ, елемент І, двійковий лічильник, ЦАП і компаратор K . Працює пристрій таким чином.

В початковому стані лічильник обнулений (що здійснюється за входом скидання R), внаслідок чого на виході ЦАП напруга дорівнює нулю. З подаванням на прямий вхід компаратора K напруги, величину якої необхідно перетворити у двійковий код, на його виході встановлюється рівень логічної 1, і тим самим дозволяє проходження імпульсів від ГІ на вхід лічильника через елемент І. Код на його виходах при цьому починає збільшуватись (0000, 0001, 0010 і т.д.). Оскільки виходи лічильника підключено до входів ЦАП, напруга на виході останнього починає зростати як показано на рис.3.12 в до тих пір, коли в момент t_1 не стає більшою, ніж

$U_{вх}$. Так як напругу подано на інверсний вхід компаратора, то на його виході встановлюється рівень логічного 0 (рис. 3.12 г). Цей сигнал, що поданий на один з входів елемента І, встановлює логічний 0 на його виході і цим зупиняє надходження імпульсів від ГІ на вхід лічильника. При цьому на виходах лічильника (виходи АЦП) встановлюється код, який відповідає рівню вхідної напруги (у прикладі, наведеному на рис. 3.12, рахування припиняється після дев'ятого імпульсу від ГІ, тобто на виходах установлюється код 1001В).

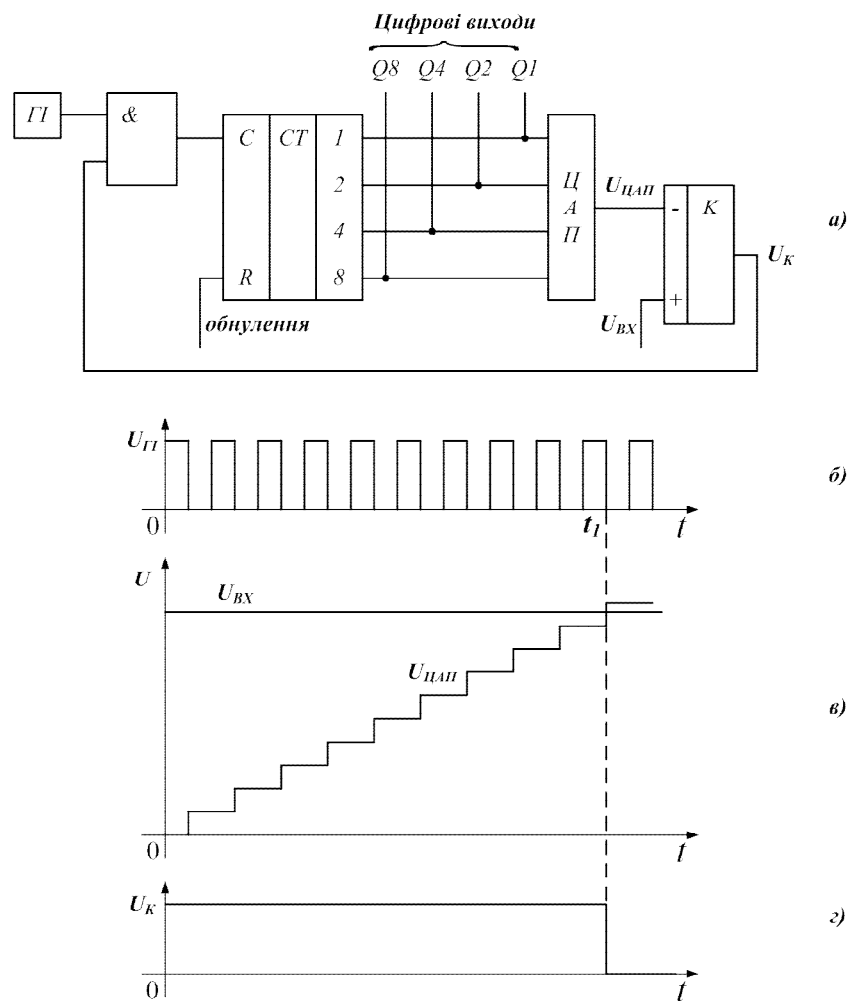


Рис. 3.12. Аналого-цифровий перетворювач

З діаграм (рис. 3.12 в, г) видно, що процес перетворення займає деякий час і здійснюється з деякою похибкою. Прискорити процес

перетворення можна, якщо підвищити частоту подавання імпульсів ГІ, а підвищити точність можна збільшенням розрядності лічильника і ЦАП.

Промислові АЦП мають від 8 до 20 розрядів і досить короткий час перетворення, який складає декілька мікросекунд.

Контрольні запитання

- 1. Як виконати переведення десяткового числа в двійкове?*
- 2. Як виконати переведення десяткового числа в вісімкове?*
- 3. Як виконати переведення двійкового числа в шістнадцяткове?*
- 4. Як виконати переведення двійкового числа в десяткове?*
- 5. Назвіть типи електронних цифрових пристроїв без зворотних зв'язків.*
- 6. Яким логічним елементом реалізується операція логічного додавання?*
- 7. Яким логічним елементом реалізується операція логічного множення?*
- 8. Назвіть логічні операції.*
- 9. Назвіть арифметичні операції.*
- 10. Назвіть спеціальні операції.*
- 11. Як виконуються арифметичні дії над двійковими числами?*
- 12. Назвіть типи електронних цифрових пристроїв без зворотних зв'язків.*
- 13. Основні пристрої комбінаційного типу.*
- 14. Дати визначення суматора.*
- 15. Дати визначення мультиплексора.*
- 16. Дати визначення дешифратора.*
- 17. Яким чином виконати збільшення розрядності суматора?*
- 18. Назвіть основні послідовні пристрої для МПС.*

19. Дати визначення регістра.
20. Типи регістрів.
21. Дати визначення паралельного регістра.
22. Дати визначення паралельно-послідовного регістра.
23. Дати визначення послідовного регістра
24. В чому полягають переваги цифрового способу подання і опрацювання інформації у порівнянні з аналоговим?
25. Для чого використовується ключовий підсилювач?
26. Поясніть сенс логічних операцій НІ, І, АБО.
27. Що таке таблиця істинності?
28. Наведіть приклади асинхронного і синхронного тригерів. Чому їх так названо?
29. Чому в цифрових електронних пристроях використовують двійкову і шістнадцяткову системи числення?
30. Назвіть основні види комбінаційних і послідовних пристроїв і скорочено дайте характеристику функцій, що ними виконуються.
31. В чому різниця між комбінаційними і послідовними пристроями?
32. Для чого використовують цифро-аналогові і аналого-цифрові перетворювачі.
33. Поясніть принципи побудови ЦАП і АЦП.

Розділ четвертий

ФУНКЦІОНАЛЬНІ СТРУКТУРИ І АРХІТЕКТУРА МІКРОПРОЦЕСОРНИХ СИСТЕМ

4.1 Структурна схема мікропроцесора

Структурна схема МП (архітектура МП) дає можливість наочно розглянути його роботу щодо виконання двох основних функцій: обробки і маніпулювання даними. Загальна структура 8-розрядного МП наведена на рис. 4.1.

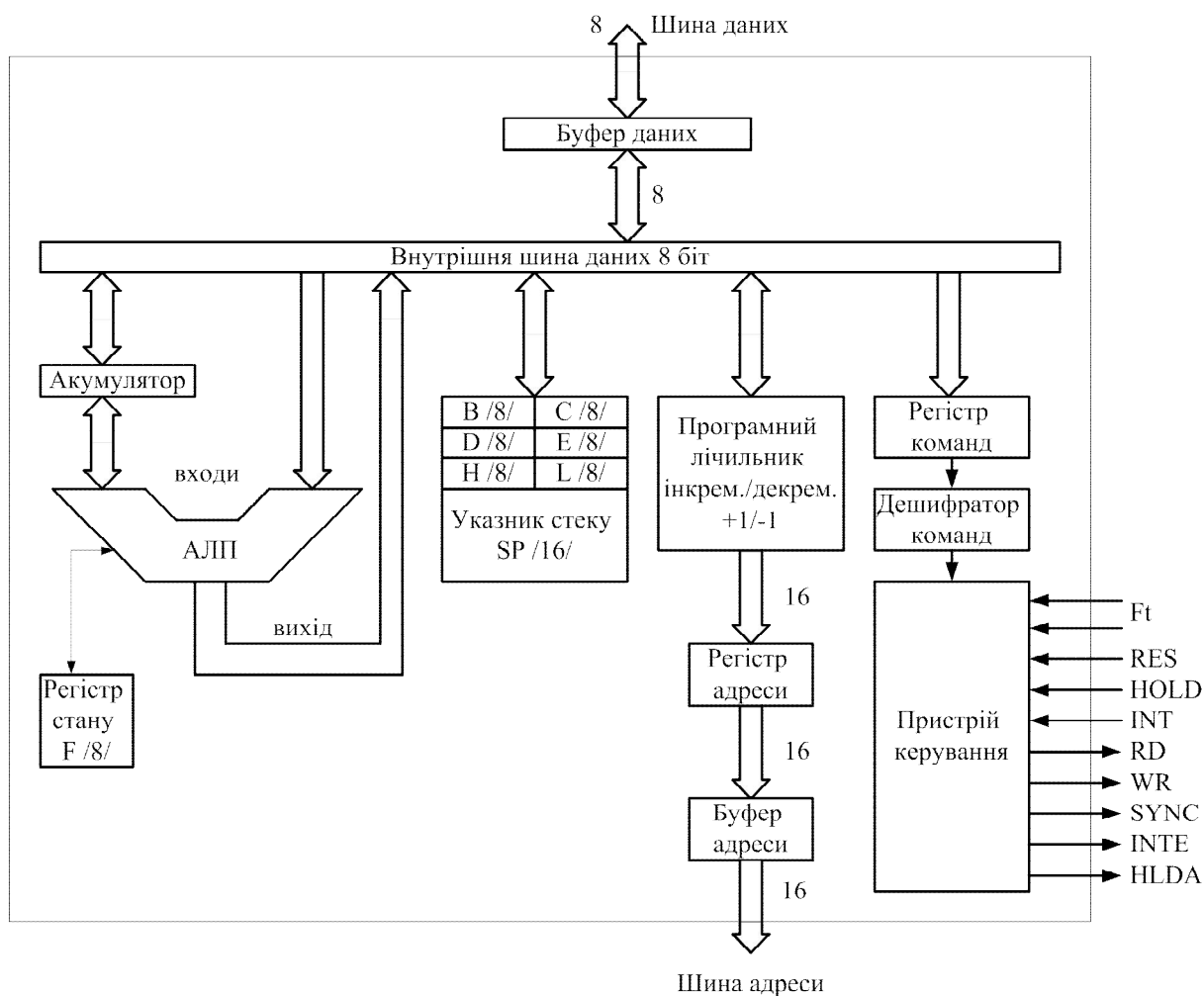


Рис. 4.1. Структура 8-розрядного МП

Арифметико-логічний пристрій (АЛП). АЛП виконує функцію обробки даних і має два *вхідних порти*, позначених як «входи», і один *вихідний порт* — «вихід» (*порт* — *схема, куди надходить вхідна або вихідна інформація*). Призначення вхідного порта — введення "слова" даних в АЛП, а вихідного порта — виведення "слова". Обидва вхідних порти обладнані регістрами тимчасового зберігання даних (*буферні регістри*). Два вхідних порти дозволяють АЛП приймати дані з внутрішньої шини даних МП і зі спеціального регістра, який називають *акумулятором (робочим регістром)*. Вихідний порт надає АЛП можливість пересилати "слово" даних в акумулятор або на шину даних.

АЛП оперує одним або двома словами в залежності від операції, що виконується. Наприклад, для додавання потрібно два слова даних і цю операцію АЛП виконує, користуючись обома вхідними портами. А при інвертуванні слова даних АЛП обмежується одним вхідним портом. До типових функцій, що виконуються АЛП, відносяться: арифметичні додавання (Add) та віднімання (Subtract), логічні І (AND), АБО (OR), виключне АБО (*XOR*), інверсія (Not), порівняння (Compare), позитивний приріст на одиницю (*Increment*) і негативний приріст (віднімання) (*Decrement*), зсув даних на розряд праворуч або ліворуч.

Регістри МП. Регістри призначені для тимчасового зберігання одного слова даних і на відміну від комірок зовнішньої пам'яті вибираються конкретною командою. Окремі регістри мають *спеціальне призначення (РСП)*, інші — багатocільове і називаються *регістрами загального призначення (РЗП)* і можуть використовуватися програмістом на його розсуд. Кількість і призначення регістрів конкретного МП залежить від його архітектури.

Акумулятор (робочий регістр) — основний регістр МП. Він застосовується при різноманітних маніпуляціях з даними та має 8 розрядів. Будь-яка з операцій над двома словами даних (*операндами*) передбачає розміщення одного з них в акумуляторі, а іншого в пам'яті або в якому-небудь регістрі. Результат операції АЛП звичайно міститься в акумуляторі. Так, при додаванні двох операндів A і B , розташованих відповідно в акумуляторі і пам'яті, результат $C = A + B$ завантажується в акумулятор, а його попереднє значення втрачається.

Виконання операції «програмована передача даних» також здійснюється з використанням акумулятора (рис. 4.2): спочатку дані пересилаються (копіюються) з джерела до акумулятора, а після цього з акумулятора - до регістру призначення (приймач). На рис. 4.2 стан до здійснення операції виділено темним кольором.

МП може виконувати деякі дії над даними безпосередньо в акумуляторі. Наприклад, акумулятор можна очистити шляхом запису 0 у всі його розряди, встановити в стан 1 шляхом запису одиниць до всіх розрядів. Вміст акумулятора можна зсувати на розряд ліворуч або праворуч, одержувати його інвертоване значення, а також виконувати інші операції.

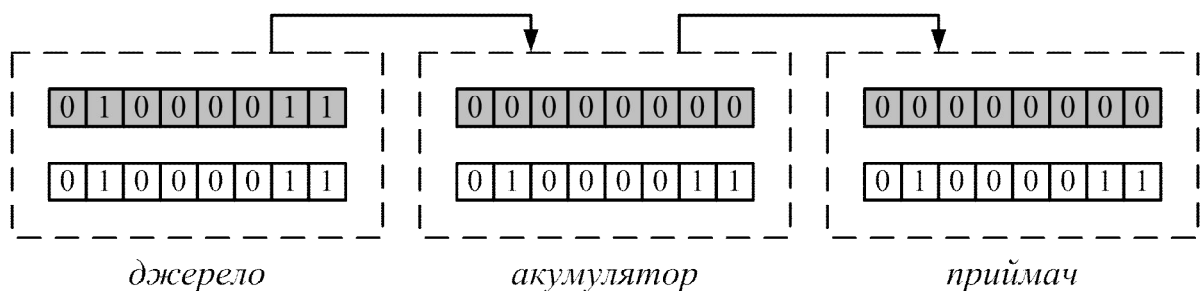


Рис. 4.2. Виконання операції «програмована передача даних»

Регістри загального призначення (РЗП) - це восьми розрядні

реєстри, які надані у розпорядження користувача. Представлений МП має декілька реєстрів загального призначення: *B*, *C*, *D,E*, *H*, *L*. При цьому реєстри *B* і *C*, *D* і *E*, *H* і *L* можуть об'єднуватися, створюючи реєстрові пари *BC*, *DE* і *HL*, еквівалентні одному шістнадцятирозрядному реєстру.

Шістнадцятирозрядний реєстр *SP* — показник стеку. За допомогою цього реєстра організуються спеціальні звернення МП до пам'яті.

Шістнадцятирозрядний лічильник команд (програмний лічильник - ПЛ) — це один з найбільш важливих реєстрів МП. Для коректного виконання команди повинні надходити у визначеному порядку. На ПЛ покладається відповідальність стежити за тим, яка команда виконується, а яка підлягає виконанню наступною. Кількість розрядів лічильника команд дорівнює 16, чим забезпечується можливість звернення на будь-яку адресу від 0 до 65535, де може знаходитися інформація про той чи інший крок програми — програма може початися і закінчитися в будь-якому місці.

Шістнадцятирозрядний реєстр адреси. При кожному зверненні до пам'яті мікро-ЕОМ *реєстр адреси* містить двійкове число, що вказує конкретну комірку пам'яті, що підлягає використанню МП. Вихід цього реєстру має буфер (*буфер адреси*) і утворює *шину адреси* (ША).

Реєстр стану (ознак) — спеціальний реєстр, окремі біти якого встановлюються в 0 або 1 в залежності від результату дій, що виконує АЛП.

Найбільш важливими і часто застосовуваними є біти — ознаки нуля (ознака 2) і переносу (ознака С). Перенос може виникнути, якщо в результаті дії (наприклад, додавання) отримано число, більше

11111111 = 0FFH = 255 (двійкове, шістнадцяткове та десятинне подання).

При цьому: $Z = 1$, якщо в результаті дій отримано 0 (відповідно $Z = 0$, якщо результат не дорівнює нулю).

$C = 1$, якщо в результаті дій (наприклад, додавання) виник перенос. Тобто, ознака C — фактично дев'ятий розряд акумулятора.

Багато МП володіють додатковими розрядами стану, застосування яких не «стандартизовано».

Регістр команд — спеціальний восьмирозрядний регістр, в який поміщується код поточної команди (звичайно, він вибирається з чергової комірки пам'яті). Ця функція реалізується МП автоматично з початком циклу вибірка-виконання, який називається також *машинним циклом*.

Дешифратор команд — логічна схема, що розпізнає двійковий код команди і формує відповідні сигнали для пристрою керування.

Пристрій керування — цифрова логічна схема, що формує сигнали керування внутрішніми вузлами мікропроцесора, а також зовнішніми пристроями за сигналами від дешифратора команд і внутрішніх вузлів МП.

Пристрій керування (ПрК) має декілька входів і виходів, сигнали на яких утворюють шину керування МП. Це наступні основні сигнали:

F_T — вхід тактових (синхро) імпульсів, що забезпечують синхронізацію роботи внутрішніх вузлів МП від зовнішнього генератора, відповідно, цими імпульсами здійснюється машинний цикл МП: вибірка - виконання команди. У якості джерела тактових імпульсів звичайно використовується зовнішній кварцовий або інший генератор;

RES — вхід "скидання". При поданні імпульсу на цей вхід всі вузли мікропроцесора встановлюються у вихідний стан, і він автоматично починає виконання програми з читання команди, записаної в комірку пам'яті з адресою 0000H;

RD, WR — виходи "читання" і "запис" відповідно. Подачею імпульсів на ці виходи, що підключені до однойменних входів відповідних комірок пам'яті або зовнішніх приладів, МП ініціює їх для відповідного обміну інформацією;

INT - вхід, *INTE* - вихід - відповідно запит і підтвердження переривання. Наявність цих сигналів дозволяє апаратним (а не програмним) засобом змінити порядок виконання програми. Детальніше функціонування системи переривання розглядається нижче;

HOLD - вхід, *HLDA* - вихід - відповідно запит і підтвердження звільнення шин. Ці сигнали дозволяють при необхідності (наприклад, при організації прямого доступу в пам'ять, минаючи процесор) відключити МП від шин мікро-ЕОМ.

Внутрішня шина даних (ШД) має 8 розрядів і з'єднує між собою АЛП і регістри, здійснюючи передачу даних всередині МП. Кожний блок МП завжди підключений до внутрішньої шини даних, однак скористатися нею може тільки після отримання відповідного сигналу від схем керування.

Майже всі функціональні вузли МП мають двосторонній зв'язок з внутрішньою шиною даних, тобто вони можуть і посилати дані на шину, і приймати з неї дані. Внутрішня шина даних являє собою лінію двостороннього зв'язку, має буфер для підключення зовнішніх пристроїв.

Буфери — підсилювачі сигналів шин. Ці вузли підсилюють сигнали з шин, здійснюють за сигналами від пристрою керування вибір напрямку передавання інформації або перемикають виходи шин МП в Z-стан (в режимі звільнення шин - виходи відключені).

У відповідності з розглянутою структурою цикл виконання команди мікропроцесором здійснюється наступним чином:

- а) після встановлення в початковий стан МП формує на ША адрес 0000H — це повинна бути комірка пам'яті з кодом першої команди програми, і за сигналом RD зчитує з цієї комірки її вміст — код команди (копія) розміщується на внутрішній ШД;**
- б) цей код по ШД заноситься в реєстр команд і на цьому завершується підцикл вибірки команди;**
- в) після цього починається підцикл виконання команди, під час якого вміст реєстру команд за допомогою дешифратора команд і пристрою керування перетворюється в послідовність дій, що здійснюються внутрішніми вузлами МП з виконання заданої команди. При цьому МП може зчитувати і записувати дані в пам'ять і зовнішні пристрої, здійснювати арифметичні і логічні операції, і т.п.;**
- г) після виконання команди вміст лічильника — реєстра адреси автоматично збільшується на одиницю, МП зчитує вміст пам'яті за цією новою адресою (це повинен бути код наступної команди), і далі процеси протікають аналогічно тому, що розглянуте в п. а і б.**

Такий порядок виконання програми може бути змінений як програмно, так і апаратно (шляхом подачі імпульсу запиту на переривання на вхід INT).

4.2 Типова структура мікропроцесорної системи

МП може функціонувати тільки спільно з іншими складовими частинами системи і розглядати МП у відриві від них не можна. Стосовно до узагальненого варіанту керування технологічним об'єктом (ТО) мікропроцесорна система (МПС) має вигляд у відповідності з рис.4.3. Схема містить в собі елементи, що стосуються ТО: виконавчі пристрої, керування якими здійснюється відповідними електричними апаратами (контактори, реле тощо), пристрій сполуки (ПС), давачі параметрів ТО, аналогово-цифровий перетворювач (АЦП). ПС необхідний для підсилення сигналів керування до необхідного рівня, АЦП — для перетворення сигналів з виходу давачів у цифровий вигляд.

Для обміну інформацією з МП використовують зовнішні, так звані, периферійні пристрої (дисплей, клавіатура).

Для функціонування системи необхідні також тактовий генератор, що забезпечує синхронізацію роботи системи у часі, пристрої пам'яті, пристрій вводу-виводу (порти 0-3).

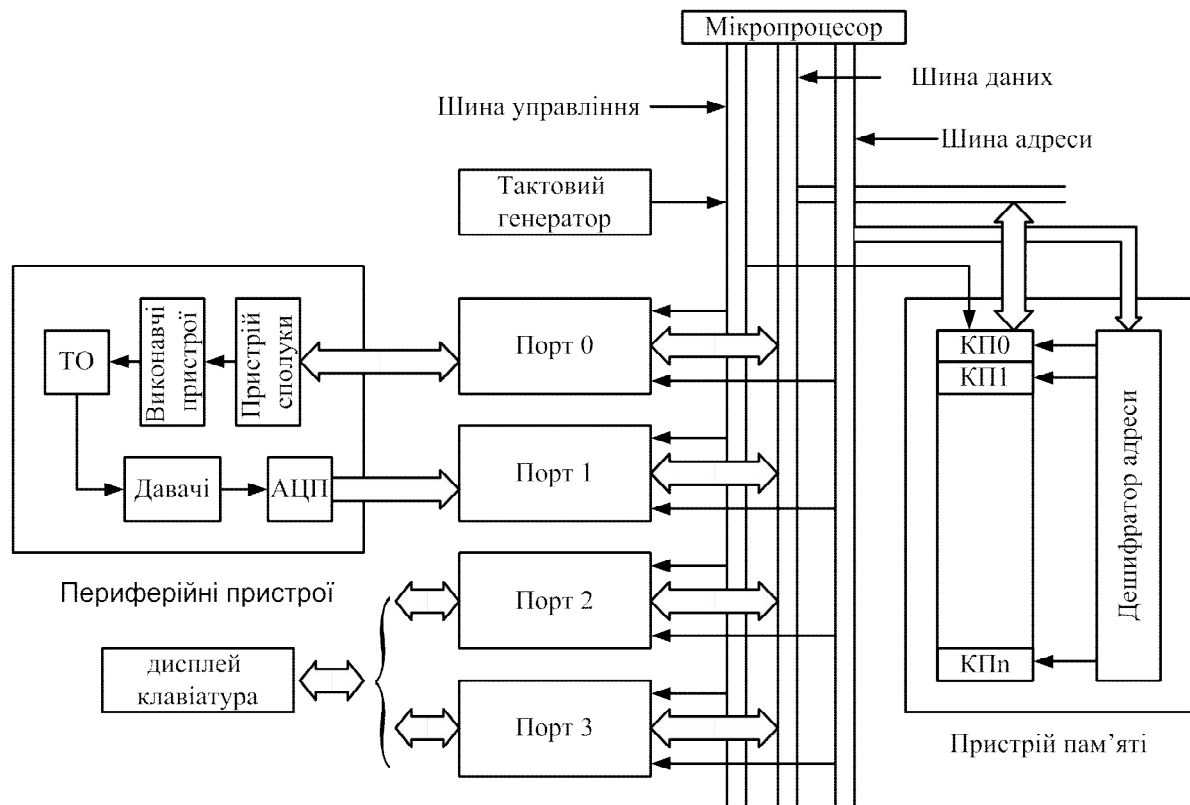


Рис. 4.3. Структурна схема мікропроцесорної системи

Пристрої пам'яті (ПрП) містять у собі постійний запам'ятовуючий пристрій (ПЗП) і оперативний запам'ятовуючий пристрій (ОЗП).

В ПЗП, як правило, записується (виробником) програма ініціалізації МПС (дії, що виконуються автоматично при вмиканні і забезпечують діалог з користувачем, а також роботу пристроїв вводу —

виводу інформації). Інформація в ПЗП зберігається постійно і незалежно від того, ввімкнутий прилад чи ні — її можна лише зчитувати.

ОЗП призначений для зберігання програм і даних користувача, а також проміжних результатів обчислень. Інформація в ОЗП може записуватися і зчитуватися. При вимкненні живлення інформація в ОЗП втрачається. Пам'ять звичайно з'єднана з МП трьома шинами: *адреси, даних і управління.*

Код, що надходить по шині адреси, використовується для ідентифікації областей пам'яті або пристроїв, з якими МП необхідно мати зв'язок. Інформація по шині адреси передається лише в одному напрямку з МП.

По шині даних в будь-який момент часу передається одне слово даних в МП або у зворотному напрямку. На відміну від адресних ліній, лінії даних є двонаправленими. За цими лініями передаються слова даних з внутрішньої шини МП до пам'яті, або — до блоку вводу-виводу, або — у зворотному напрямку.

Лінії керування (шина управління) дозволяють МП керувати роботою зовнішніх пристроїв або контролювати роботу останніх. За допомогою цих ліній МП повідомляє зовнішнім пристроям, коли подавати дані на шину даних або коли отримувати їх з шини. Завдяки сигналам, що надходять по лініях керування, передача даних відбувається в належному порядку і напрямку.

Кожне слово у ПрП розміщується в окремій комірці пам'яті (КП), яка має свій індивідуальний номер — адресу. Адреса (двійкове число) — з'являється на ША і перетворюється *дешифратором адреси* у сигнал, що ініціює відповідну КП до обміну даними з МП згідно сигналам «Читання пам'яті» або «Запис в пам'ять» на шині керування. Ці сигнали інформують МП про необхідність подавання даних на відповідну шину або отримання даних з шини для запису у вибрану область пам'яті. При

запису даних до пам'яті попередній вміст відповідної КП стирається і замінюється інформацією, що записується. Читання даних із пам'яті (точніше — копіювання) не змінює вміст області, де вони знаходяться.

Пристрої вводу-виводу МПС включають в собі *порти вводу-виводу* з певними номерами 0, 1 тощо. В даній схемі порти 0 і 1 використовуються для передавання сигналів керування ТО і інформації про його стан. Порти 2 і 3 — відповідно для обміну даними з периферійними пристроями. Кожний порт (рис. 4.4) має вісім двонаправлених (вхідних/вихідних) ліній, що можуть програмуватись на ввід або вивід окремо або разом. Також порт є з'єднаним з шинами адреси і керування. Для налаштування порта використовується спеціальний керуючий регістр (КР) — куди заздалегідь записується певний код. Стан кожної лінії вводу/виводу зберігається у буфері — регістрі даних. Кожний з регістрів має свою адресу. При читанні порту вхідні дані з регістра даних подаються на шину даних. Після цього МП поміщує ці дані в акумулятор.

У разі виводу інформації байт даних, наявний на шині даних, записується в буфер даних порту, до якого адресується МП. Ці дані буфер зберігає для можливого їхнього використання зовнішніми пристроями. Дані зберігаються до наступного звернення до цього порту або до часу, доки не відключать живлення мікро-ЕОМ. Вказані порти є *портами паралельного вводу-виводу*, оскільки дані передаються одночасно по восьми лініями (паралельно).

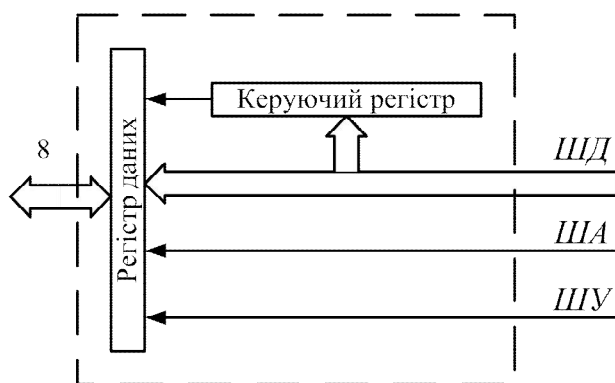


Рис. 4.4. Порт вводу-виводу

При *послідовному вводиті-виводі* відповідний послідовний порт передає/приймає дані послідовно біт за бітом, для чого використовується лише одна лінія. Послідовний порт вводу-виводу використовується, коли виникає необхідність передавати дані на великій відстані. У пункті приймання здійснюється зворотне перетворення послідовного коду в паралельний.

У відповідності з розглянутою структурою, в найбільш загальному представленні, МПС працює наступним чином: згідно з програмою, що записана в пристроях пам'яті, мікропроцесор здійснює операції над даними, забезпечує обмін інформацією між своїми внутрішніми вузлами, пристроєм пам'яті і зовнішніми об'єктами, використовуючи пристрої вводу-виводу. Інформація пересилається по шині даних, причому мікропроцесор заздалегідь встановлює на шині адреси комірки пам'яті або зовнішнього пристрою, з яким в даний момент часу здійснюється обмін, ініціюючи процес обміну сигналами "читання" або "запис" що пересилаються по шині управління.

4.3 Ідеологія побудови мікропроцесорних систем

Застосування мікропроцесорної техніки в різних галузях народного господарства, у тому числі і в електроенергетиці, розвивається по двох напрямках:

- розробка спеціалізованих обчислювальних засобів, орієнтованих на рішення відповідних прикладних задач електроенергетики, наприклад, автоматичний контроль, діагностика і управління в електричних мережах;
- розробка нових мікропроцесорних систем на основі універсальної елементної бази, що серійно випускається (як вітчизняній, так і зарубіжній).

В цілому сучасна мікропроцесорна техніка відрізняється від попередніх поколінь більш високою надійністю, щодо низькою вартістю і можливістю реалізації простими засобами складних алгоритмів управління і контролю.

Спеціалізовані МПС для електроенергетики повинні відповідати і додатковим специфічним вимогам, до яких відносяться:

- наявність достатньої обчислювальної потужності і розвинутого апаратно-програмного забезпечення для виконання арифметичних операцій з достатньо високою точністю;
- гнучкість обчислювальних засобів, можливість зміни їх конфігурації, розширення і модульна обчислювальних структур;
- наявність значних об'ємів пам'яті, можливість розширення і модульність оперативних і постійних ЗП;
- наявність універсальної системи обміну інформацією між обчислювальними пристроями, елементами пам'яті і об'єктом

управління в процесі рішення функціональних задач, у тому числі і при значній віддаленості від об'єкту управління і обчислювальної системи один від одного;

- швидкодія, не менш декілька мільйонів операцій в секунду, що дозволяє виконувати функціональні задачі в реальному часі;
- висока живучість системи, тобто стійкість апаратного і програмного забезпечення до збоїв і відмов апаратури, збереження працездатності при відмові частини апаратури;
- достатньо низька вартість елементної бази.

Щоб на базі МП реалізувати МПС, процесор необхідно доповнити його зовнішніми пристроями (пристроєм ЗП, що запам'ятовує; пристроями введення-виводу) і організувати їх взаємодію.

Розглянемо шинну структуру зв'язків.

Для досягнення максимальної універсальності і спрощення протоколів обміну інформацією в мікропроцесорних системах застосовується так звана шинна структура зв'язків між окремими пристроями, що входять в систему. Суть шинної структури зв'язків зводиться до наступного.

При класичній структурі зв'язків всі сигнали і коди між пристроями передаються по окремих лініях зв'язку. Кожний пристрій, що входить в систему, передає свої сигнали і коди незалежно від інших пристроїв. При цьому в система обтяжена великою кількістю ліній зв'язку і різних протоколів обміну інформацією.

При шинній структурі зв'язків всі сигнали між пристроями передаються по одних і тих же лініях зв'язку, але в різний час (це називається мультиплексорною передачею). Причому передача по всіх

лініях зв'язку може здійснюватися в обох напрямках (так звана двонаправлена передача). В результаті кількість ліній зв'язку істотно скорочується, а правила обміну (протоколи) спрощуються. Група ліній зв'язку, по яких передаються сигнали або коди якраз і називається шиною (англ. bus).

Зрозуміло, що при шинній структурі зв'язків легко здійснюється пересилка всіх інформаційних потоків в потрібному напрямі, наприклад, їх можна пропустити через один процесор, що дуже важливо для мікропроцесорної системи. Проте при шинній структурі зв'язків вся інформація передається по лініях зв'язку послідовно в часі, по черзі, що знижує швидкодію системи в порівнянні з класичною структурою зв'язків.

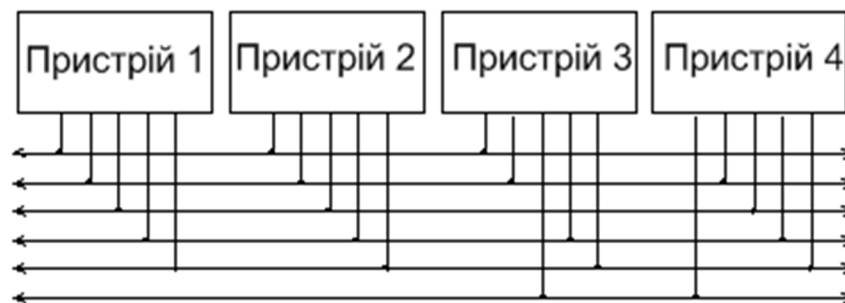


Рис. 4.5. Шинна структура зв'язків

Перевагою шинної структури зв'язків полягає в тому, що всі пристрої, підключені до шини, повинні приймати і передавати інформацію за одними і тими ж правилами (протоколам обміну інформацією по шині). Відповідно, всі вузли, що відповідають за обмін через шину в цих пристроях, повинні бути одно типовими та уніфікованими.

Недолік шинної структури пов'язаний з тим, що всі пристрої підключаються до кожної лінії зв'язку паралельно. Тому будь-яка несправність будь-якого пристрою може вивести з ладу всю систему, якщо

вона псує лінію зв'язку. З цієї ж причини налаштування системи з шинною структурою зв'язків досить складна і звичайно вимагає спеціального устаткування.

В системах з шинною структурою зв'язків застосовують всі три існуючі різновиди вихідних каскадів цифрових мікросхем (рис. 4.6 а):

- стандартний вихід або вихід з двома станами (позначається 2С, 2S, рідше ТТЛ, TTL);
- вихід з відкритим колектором (позначається ВК, ОС);
- вихід з трьома станами або (що те ж саме) з можливістю відключення (позначається 3С, 3S).

Спрощений ці три типи вихідних каскадів можуть бути представлено у вигляді схем на рис. 4.6.

У виходу 2С два ключі замикаються по черзі, що відповідає рівням логічної одиниці (верхній ключ замкнутий) і логічного нуля (нижній ключ замкнутий). У виходу ВК замкнутий ключ формує рівень логічного нуля, розімкнений — логічної одиниці. У виходу 3С ключі можуть замикатися по черзі (як у разі 2С), а можуть розмикатися одночасно, утворюючи третій, високоімпедансний стан. Перехід в третій стан (Z-стан) керується сигналом на спеціальному вході EZ.

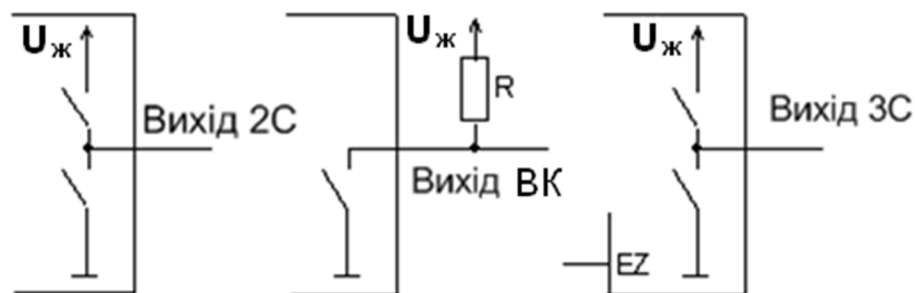


Рис. 4.6. Три типи виходів цифрових мікросхем.

Вихідні каскади типів ЗС і ВК дозволяють об'єднувати декілька виходів мікросхем для отримання мультиплексорних (рис. 4.7) або двонаправлених (рис.4.8) ліній.

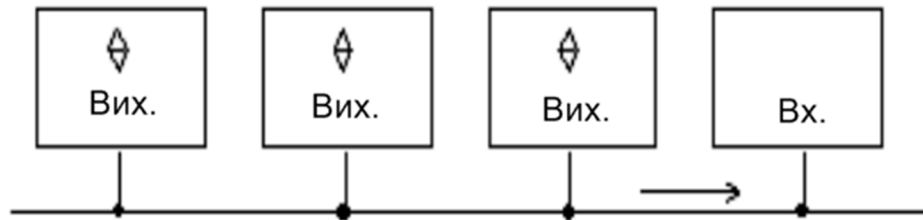


Рис. 4.7. Мультиплексорна лінія

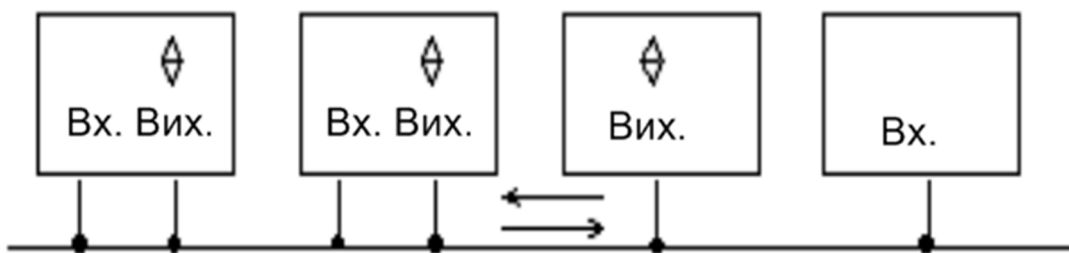


Рис. 4.8. Двонаправлена лінія

При використанні виходів ЗС необхідно забезпечити, щоб на лінії завжди працював тільки один активний вихід, а вся решта виходів знаходилася б в цей час в третьому стані, інакше можливі конфлікти. Об'єднані виходи ЗК можуть працювати всі одночасно, без жодних конфліктів.

Типова структура мікропроцесорної системи приведена на рис. 4.9.

Вона включає три основні типи пристроїв:

- процесор;

- пам'ять, що включає оперативну пам'ять (ОЗП, RAM — Random Access Memory) і постійну пам'ять (ПЗП, ROM — Read Only Memory), яка служить для зберігання даних і програм;
- пристрої введення/виводу, служби для зв'язку мікропроцесорної системи із зовнішніми пристроями, для прийому (введення, читання, Read) вхідних сигналів і видачі (виводу, запису, Write) вихідних сигналів.

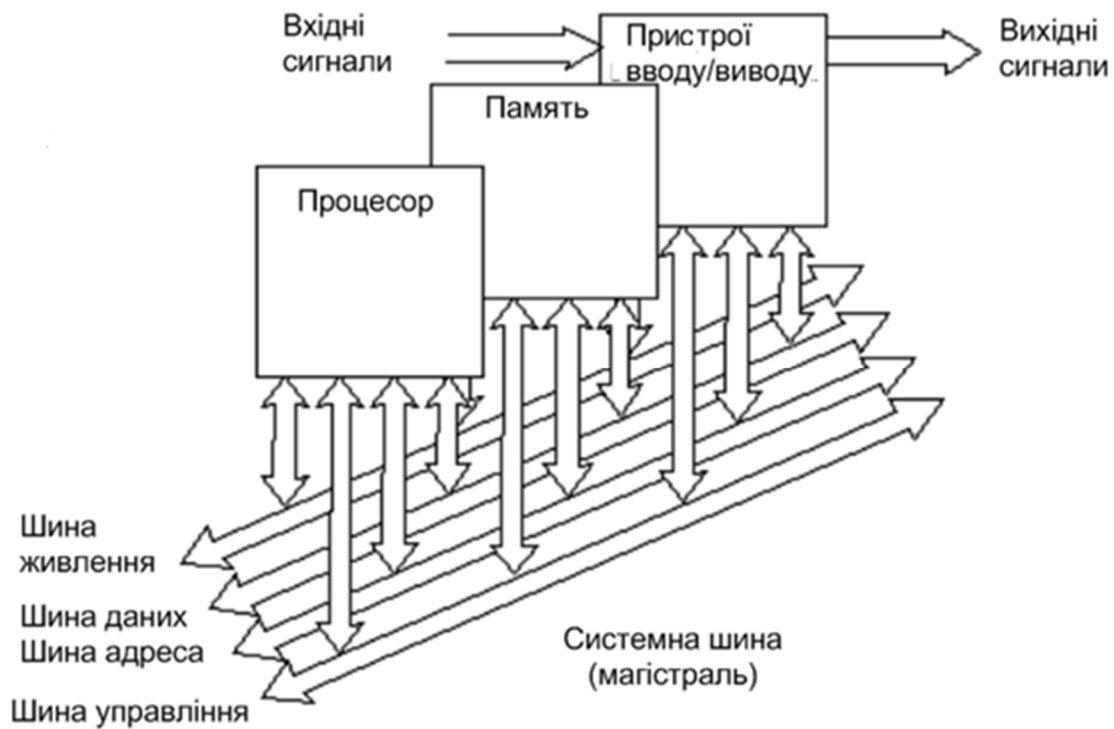


Рис. 4.9. Структура мікропроцесорної системи із загальною шиною

Всі пристрої мікропроцесорної системи об'єднуються загальною системною шиною (вона ж називається ще системною магістраллю або каналом). Системна магістраль включає чотири основні шини нижнього рівня:

- шина адреси (Address Bus);

- шина даних (Data Bus);
- шина управління (Control Bus);
- шина живлення (Power Bus).

Шина адреси служить для визначення адреси (номера) пристрою, з яким процесор обмінюється інформацією в даний момент. Кожному пристрою (окрім процесора), кожному елементу пам'яті в мікропроцесорній системі привласнюється власна адреса. Коли код якоїсь адреси виставляється процесором на шині адреси, пристрій, якому ця адреса приписана, розуміє, що йому належить обмін інформацією. Шина адреси може бути однонаправленою або двонаправленою.

Шина даних — це основна шина, яка використовується для передачі інформаційних кодів між всіма пристроями мікропроцесорної системи. Звичайно в пересилці інформації бере участь процесор, який передає код даних в якийсь пристрій або в елемент пам'яті або ж приймає код даних з якогось пристрою або з елемента пам'яті. Але можлива також і передача інформації між пристроями без участі процесора. Шина даних завжди двонаправлена.

Шина управління на відміну від шини адреси і шини даних складається з окремих управляючих сигналів. Кожний з цих сигналів під час обміну інформацією має свою функцію. Деякі сигнали служать для того, щоб стробувати передаючі або приймаючі дані (тобто визначають моменти часу, коли інформаційний код виставлений на шину даних). Інші управляючі сигнали можуть використовуватися для підтвердження прийому даних, для скидання всіх пристроїв в початковий стан, для тактування всіх пристроїв і т.д. Лінії шини управління можуть бути однонаправленими або двонаправленими.

Шина живлення призначена не для пересилки інформаційних сигналів, а для живлення системи. Вона складається з ліній живлення і загального дроту. В мікропроцесорній системі може бути одне джерело живлення (частіше +5 В) або декілька джерел живлення (звичайно ще –5 В +12 В і –12 В). Кожній напрузі живлення відповідає своя лінія зв'язку. Всі пристрої підключені до цих ліній паралельно.

Якщо в мікропроцесорну систему треба ввести вхідний код (або вхідний сигнал), то процесор по шині адреси звертається до потрібного пристрою введення/виводу і приймає по шині даних вхідну інформацію. Якщо з мікропроцесорної системи треба вивести вихідний код (або вихідний сигнал), то процесор звертається по шині адреси до потрібного пристрою введення/виводу і передає йому по шині даних вихідну інформацію.

Якщо інформація повинна пройти складну багатоступінчасту обробку, то процесор може берегти проміжні результати в системній оперативній пам'яті. Для звернення до будь-якого елемента пам'яті процесор виставляє її адресу на шину адреси і передає в неї інформаційний код по шині даних або ж приймає з неї інформаційний код по шині даних. В пам'яті (оперативній і постійній) знаходяться також і управляючі коди (команди виконуваної процесором програми), які процесор також читає по шині даних з адресацією по шині адреси. Постійна пам'ять використовується в основному для зберігання програми початкового пуску мікропроцесорної системи, яка виконується кожного разу після включення живлення. Інформація в неї заноситься виробником раз і назавжди.

Таким чином, в мікропроцесорній системі всі інформаційні коди і коди команд передаються по шинах послідовно, по черзі. Це визначає порівняно невисоку швидкодію мікропроцесорної системи. Воно обмежено звичайно навіть не швидкодією процесора (яке теж дуже

важливо) і не швидкістю обміну по системній шині (магістралі), а саме послідовним характером передачі інформації по системній шині (магістралі).

Важливо відмітити, що пристрої введення/виводу частіше за все є пристроями на «жорсткій логіці». На них може бути покладений частина функцій, виконуваних мікропроцесорною системою. Тому у виробника завжди є можливість перерозподіляти функції системи між апаратною і програмною реалізаціями оптимальним чином. Апаратна реалізація прискорює виконання функції, але має недостатню гнучкість. Програмна реалізація значно повільніше, але забезпечує високу гнучкість. Апаратна реалізація функцій збільшує вартість системи і її енергоспоживання, програмна — не збільшує. Частіше всього застосовується комбінування апаратних і програмних функцій.

Іноді пристрої введення/виводу мають в своєму складі процесор, тобто є невеликою спеціалізованою мікропроцесорною системою. Це дозволяє перекласти частину програмних функцій на пристрої введення/виводу, розвантаживши центральний процесор системи.

4.4 Функціональні структури і архітектура комп'ютерних систем в електроенергетиці

Функціонально класифікацію МПС можна представити трьома основними типами структур:

- програмно-логічного управління (ПЛУ);
- системи збору і обробки інформації (ЗОІ);
- системи цифрового автоматичного управління (ЦАУ).

Тип **ПЛУ** характеризується тим, що всі вимірювані (вхідні) сигнали X_i і видані на об'єкт сигнали управління Y_i мають логічний характер (так/ні, вкл/викл). Логічні сигнали управління Y_i обчислюються МП програмно, як вихідні сигнали логічних функцій або кінцевих логічних апаратів.

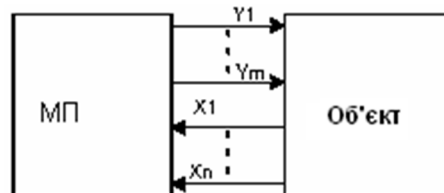


Рис. 4.10. Структура мікропроцесорної системи ПЛУ

Тип **ЗОІ** призначений для виконання трьох основних функцій:

- постійне опитування і вимірювання сигналів із групи датчиків (датчики тиску, температури, струму і т.д.), що установлені на об'єкті;
- первинна обробка вхідної інформації (усунення перешкод, перетворення формату даних і т.д.);
- збереження блоків отриманої інформації у пам'яті або передача її на ЕОМ верхнього рівня (ЕОМ ВР).

Загальна структура даного типу представлена на рис. 4.11.

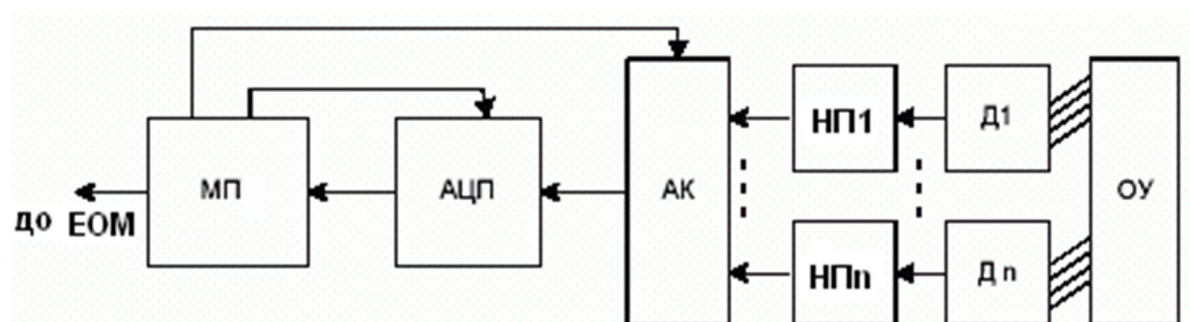


Рис. 4.11. Загальна структура мікропроцесорної системи типу ЗОІ

$Д_1, \dots, Д_n$ – датчики на об'єкті управління ОУ. $НП_1, \dots, НП_n$ – нормуючий пристрій, що перетворює величину сигналу з датчиків у необхідний діапазон для виміру АЦП. АК – аналоговий комутатор,

здійснює підключення одного з датчиків до АЦП для вимірювання сигналу.

Тип ЦАУ призначений для організації зворотного (замкненого) контуру управління об'єктом і реалізує функції автоматичного регулятора, заданого передатною функцією або рівнянням чи алгоритмами інтелектуального управління (нечітким, нейро і т.п.).

Загальна структура даного типу має вигляд:

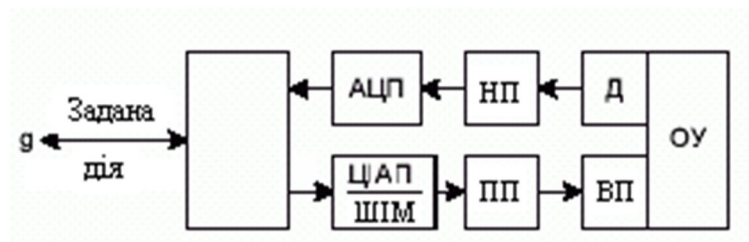


Рис. 4.12. Загальна структура мікропроцесорної системи типу ЦАУ

ВП – виконавчий пристрій; ПП – погоджувачий пристрій, що підсилює сигнал із ЦАП або ШІМ до рівня необхідного для ВП; ШІМ – широтно-імпульсний модулятор, використовується для перетворення коду в аналоговий сигнал, шляхом модуляції ширини імпульсу.

Сучасні електроенергетичні установки та об'єкти налічують сотні та тисячі різних датчиків (електромагнітні, електромеханічні, акустичні, світлові, лазерні і т.ін.), які формують на своєму виході напруги та струми різного роду та величини. Відповідно для опрацювання сигналів засобами МПС виникає проблема нормування, вимірювання сигналів та їх переведення в цифровий вигляд. Існують різні підходи до цього питання. На сьогодні розробляються як спеціалізовані МПС з своїми нормуючими пристроями та гальванічними розв'язками, так і модульні нормуючі пристрої вхідних сигналів різних частотних діапазонів: низькочастотного

від 0 до 50 Гц; середньо частотного від 50 до 20000 Гц; високочастотного вище 20 кГц.

Контрольні запитання

1. Назвіть основні пристрої 8-розрядного МП.
2. Для чого використовується АПП?
3. Поясніть як здійснюється цикл виконання команди мікропроцесором?
4. Які функції виконує основний регістр МП?
5. Які функції виконує стек.
6. Пристрій керування та його функції?
7. Назвіть основні пристрої МПС.
8. Для чого використовується порт вводу-виводу?
9. Поясніть як здійснюється перепрограмування порту вводу-виводу?
10. Які функції виконує тактовий генератор МПС?
11. Які функції виконує ОЗП та ПЗП в МПС?
12. В чому переваги шинної структури зв'язків?
13. Типи вихідних каскадів цифрових мікросхем.
14. Дати визначення поняттю – шина даних.
15. Дати визначення поняттю – шина адресу.
16. Дати визначення поняттю – шина управління.
17. Дати визначення поняттю – мультиплексорна лінія.
18. Дати визначення поняттю – двонаправлена лінія.
19. Назвіть функціональні структури МПС в електроенергетиці.
20. Характерні особливості систем програмно-логічного управління.
21. Наведіть структуру систем збору і обробки інформації.
22. Наведіть структуру систем цифрового автоматичного управління.