

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

Національний Університет

"Запорізька Політехніка"

Конспект лекцій

з дисципліни «Електроніка та основи схемотехніки»

для студентів спеціальності 173 «Авіоніка»

усіх форм навчання

Частина 4

Запоріжжя 2023

Конспект лекцій з дисципліни «Електроніка та основи схемотехніки» для студентів спеціальності 173 «Авіоніка» усіх форм навчання частина 4 / Укл. М.Л. Антонов, Є.В.Васильєва – Запоріжжя: НУЗП, 2023 – 55 с.

Укладачі: М.Л. Антонов , канд. техн. наук

Є.В.Васильєва, старш. викл.;

Рецензент: А.Є. Казурова, канд. техн. наук, доцент

Відповідальний за випуск: Є.В.Васильєва

Затверджено
на засіданні кафедри ЕПА
П протокол № 10
від 26.05.2023
Рекомендовано до видання
НМК ЕТФ
Протокол № 9
від 18.05.2023

Зміст

	стр
Розділ 8 Цифрові мікроелектронні пристрої	242
8.1. Поняття про цифрові мікроелектронні пристрої	242
8.2. Реалізація складних логічних функцій	242
8.3. Дешифратори	246
8.4. Мультиплексори	248
8.5. Лічильники імпульсів	248
8.6. Регістри	254
8.7. Цифро-аналогові (ЦАП) і аналого-цифрові (АЦП) перетворювачі	255
8.7.1. Будова і робота ЦАП	255
8.7.2. АЦП	258
8.8. Мультивібратори і одновібратори на логічних елементах і тригерах	260
8.9. ІМС таймера К1006ВИ1	263
8.10. Мікропроцесорні пристрої	266
8.10.1. Загальні положення. Дещо з історії процесорів	266
8.10.2. Особливості роботи і використання мікропроцесорів. Двійкова система числення. Структура мікропроцесорної системи	269
8.10.3. Структура мікропроцесора	273
8.10.4. Запам'ятовувальні пристрої. Загальні відомості	274
8.10.5. Запам'ятовувальні пристрої для зберігання оперативної інформації (ОЗП)	275
8.10.6. Запам'ятовувальні пристрої для зберігання постійної інформації (ПЗП)	276
8.10.7. Алгоритм. Способи представлення алгоритму	278
8.10.8. Задання і виконання програм МПП	278
8.10.9. Шістнадцяткова система числення	280
8.10.10. Асемблер	281
8.10.11. Програмовані логічні контролери	283
8.10.12. Мікроконтролери	286
8.10.13. Структура мікроконтролера	289

8.10.14. Зіставлення основних етапів створення пристроїв керування на основі цифрових автоматів з жорсткою логікою і програмованих

РОЗДІЛ 8

ЦИФРОВІ МІКРОЕЛЕКТРОННІ ПРИСТРОЇ

8.1. Поняття про цифрові мікроелектронні пристрої

Цифрові мікроелектронні пристрої являють собою дискретні цифрові автомати, виконані на ІМС і призначені для опрацювання інформації, що представлена у вигляді цифрового коду. Вони використовуються для створення цифрових інформаційних, вимірювальних систем та систем керування.

Усі цифрові пристрої поділяються на два великих класи: комбінаційні і послідовнісні.

Комбінаційні пристрої реалізують функції, які залежать тільки від комбінації змінних, що до них входять, у даний момент часу і не залежать від стану пристрою у попередній момент часу. Найпростішими прикладами таких пристроїв є логічні елементи (див. пп. 6.2).

Послідовнісні (від слова «послідовність») **пристрої** реалізують функції, що залежать не тільки від комбінації входних змінних у даний момент часу, а ще й від стану пристрою у попередній момент часу: вони мають пам'ять. Найпростішими прикладами таких пристроїв є тригери (див. розділ 7).

Основними (найбільш вживаними) мікроелектронними цифровими пристроями є:

- а) дешифратори;
- б) мультиплексори;
- в) лічильники імпульсів;
- г) регістри;
- д) цифро-аналогові та аналого-цифрові перетворювачі.

Будуються ці пристрої на логічних елементах і тригерах.

8.2. Реалізація складних логічних функцій

Складні логічні функції реалізують на ІМС простих логічних елементів.

Мінімальний набір логічних елементів, що реалізують деякі прості логічні функції і за наявності необмеженої кількості яких можна реалі

зувати наскільки завгодно складну логічну функцію, називають **функціонально повною системою логічних елементів** або базисом.

Найбільш відомими функціонально повними системами є:

- а) елементи, що реалізують функції алгебри Буля – І, АБО, НІ;
- б) елемент, що реалізує **функцію штрих Шеффера** – І-НІ;
- в) елемент, що реалізує **функцію стрілка Пірса** – АБО-НІ.

Якщо уважно подивимось на таблицю істинності логічних елементів (див. рис. 6.2), то побачимо, що, наприклад, елемент І для одиниць є елементом АБО для нулів. Тобто, якщо для прямих значень

$$y = x_1 \cdot x_2 \text{ то для інверсних } y = x_1 + x_2. \quad (8.1)$$

Виходячи з правил алгебри логіки (див. 6.18), для елемента І-НІ можна записати

$$y = \overline{x_1 \cdot x_2} = \overline{x_1} + \overline{x_2} \quad (8.2)$$

а для елемента АБО-НІ

$$y = \overline{x_1 + x_2} = \overline{x_1} \cdot \overline{x_2} \quad (8.3)$$

Якщо задати, наприклад, відповідно $x_2 = 1$ або $x_2 = 0$, то матимемо

$$y = \overline{x_1 \cdot x_2} = \overline{x_1} \cdot 1 = \overline{x_1} \quad (8.4)$$

$$\text{або } y = \overline{x_1 + x_2} = \overline{x_1 + 0} = \overline{x_1} \quad (8.5)$$

Тобто, ми можемо використовувати багатовходові логічні елементи з інверсією на виході як інвертори, задаючи на всіх, крім одного входу, сигнал 1 (const 1) або 0 (const 0).

Якщо на всі входи цих елементів подати одну й ту ж змінну, також матимемо інвертори:

$$\begin{aligned} y &= \overline{x_1 \cdot x_1} = \overline{x_1}, \\ y &= \overline{x_1 + x_1} = \overline{x_1} \end{aligned} \quad (8.6)$$

У разі, якщо кількості входів конкретного логічного елемента не вистачає, можна вчинити таким чином. Наприклад, потрібен тривходовий елемент, а маємо тільки двовходові, то тривходовий можна реалізувати на основі наступних співвідношень.

Оскільки очевидно, що

$$\overline{\overline{x}} = x \quad (8.7)$$

то

$$y = \overline{x_1 \cdot x_2 \cdot x_3} = \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} = x_1 \cdot x_2 \cdot 1 \cdot x_3, \quad (8.8)$$

або

$$y = \overline{x_1 + x_2 + x_3} = \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} = x_1 \cdot x_2 + 1 \cdot x_3. \quad (8.9)$$

Отже, реалізувати тривходовий елемент можна на трьох двовходових, один з яких повинен працювати як інвертор.

З наведених прикладів видно, що елементи І-НІ чи АБО-НІ дійсно дозволяють реалізувати логічну функцію будь-якої складності.

Нехай треба реалізувати функцію

$$y = (x_1 + x_2 + x_3) \cdot x_4 + x_4 + x_5 \quad (8.10)$$

У загальному випадку це можна зробити за допомогою схеми, наведеної на рис. 8.1

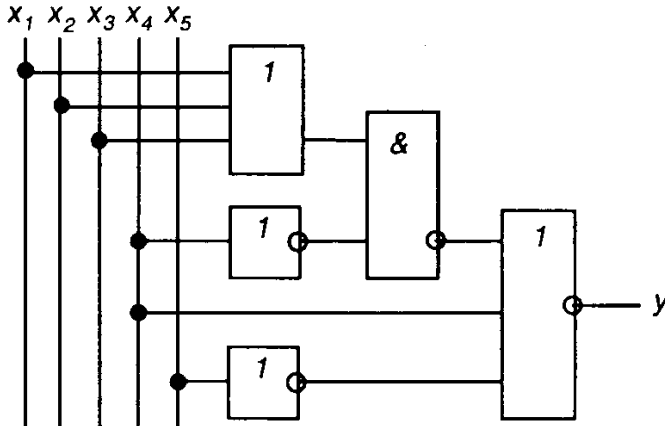


Рисунок. 8.1 – Приклад схемної реалізації логічної функції

Для забезпечення отримання простої (економічної в реалізації), швидкодіючої схеми, складні комбінаційні пристрої спочатку описують за допомогою логічних функцій (у вигляді математичних формул або таблиць істинності).

Потім ці функції мінімізують на основі законів алгебри логіки з урахуванням специфіки стандартних ІМС логічних елементів, що будуть використані для схемної реалізації. Це можна робити як вручну, що досить складно, так і з

використанням спеціальних програм на ЕОМ.

Так, якщо, наприклад, вираз (8.10) необхідно реалізувати на дво-
входових елементах І-НІ, то спочатку отримаємо еквівалентний міні-
мальний вираз у базисі І-НІ:

$$\begin{aligned}
 y &= (x_1 + x_2 + x_3) \cdot x_4 + x_4 + x_5 = x_1 \cdot x_2 \cdot x_3 \cdot x_4 + x_4 + x_5 = \\
 &= x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_4 \cdot x_5 = x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_5 = \\
 &= x_1 \cdot 1 \cdot x_2 \cdot 1 \cdot 1 \cdot x_3 \cdot 1 \cdot x_4 \cdot 1 \cdot x_5 \cdot 1. \quad (8.11)
 \end{aligned}$$

Отриманому виразу відповідає схема, наведена на рис. 8.2. Вона
виконана на трьох ІМС К561ЛА7

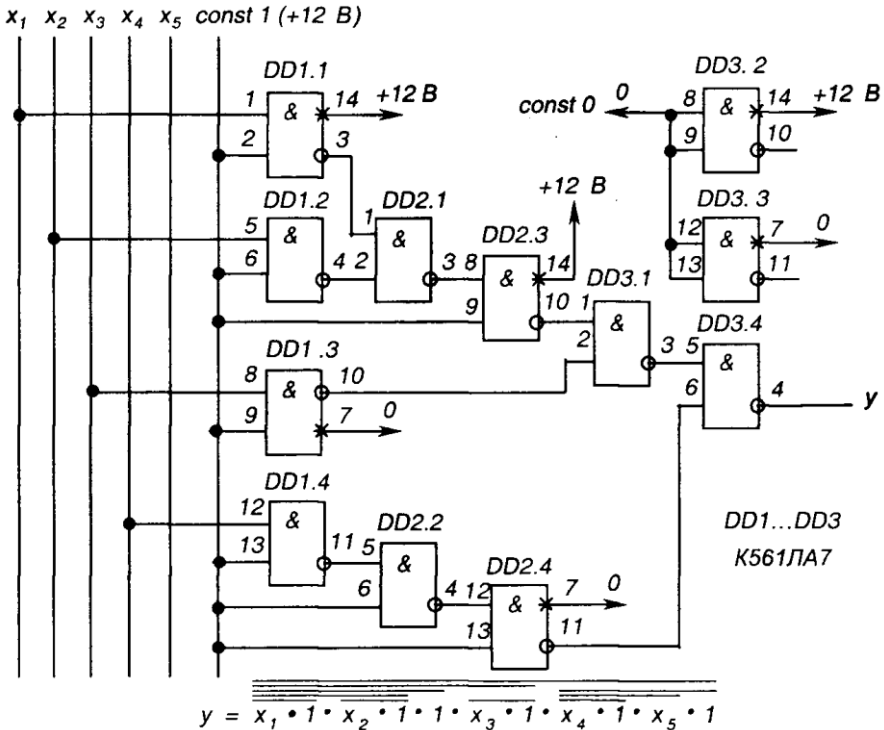


Рисунок. 8.2 – Схемна реалізація логічної функції на однотипних елементах

Зауважимо, що на таких схемах відносно ланцюгів живлення ІМС
на вільному полі в правій частині схеми у технічних вимогах дають

вказівки щодо підмикання відповідних контактів або вказують відповідні контакти ІМС як нелогічні виводи елементів і позначають місця їхнього підмикання (див. рис. 8.2 – контакти 7 і 14). На входи елементів, що не задіяні, подають сигнали const 1 або const 0, а їхні виходи залишають вільними (див. рис. 8.2 – елементи DD3.2 і DD3.3).

8.3. Дешифратори

Дешифратори (декодери) є комбінаційними пристроями, призначеними для перетворення кодованих двійкових вхідних сигналів у сигнали керування виконавчими пристроями, пристроями відображення інформації і т.п.

У загальному випадку дешифратор має декілька входів (за кількістю розрядів двійкових чисел, які необхідно декодувати) і декілька виходів.

Кожній комбінації вхідних сигналів відповідає певна комбінація вихідних (зрозуміло, що дешифратори як комбінаційні пристрої будуються на логічних елементах і їх випускають у вигляді ІМС).

Наприклад, **двійковий дешифратор**, умовне позначення якого наведено на рис. 8.3, має чотири входи ($n=4$) з ваговими коефіцієнтами 1, 2, 4, 8, що відповідає чотирьом розрядам послідовного двійкового коду ($2^0, 2^1, 2^2, 2^3$) – див. пп. 8.10.2, і шість надцять виходів: від 0 до 15 ($N=2^n=2^4=16$). Кожній комбінації нулів і одиниць на входах відповідає одиниця на відповідному виході.

Наприклад,

$$Y_5 = x_1 \cdot x_2 \cdot x_4 \cdot x_8$$

$$(5 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 0 \cdot 2^3). \quad (8.12)$$

У **двійково-десятьковому** дешифратора з інверсними виходами, умовне позначення якого наведено на рис. 8.4, кожній з перших десяти двійкових комбінацій (двійково-десятьковий код) відповідає нуль на відповідному виході. Такі дешифратори у вигляді ІМС застосовують для керування десятичними неоновими індикаторами, у яких десятикові знаки являють собою фігурні катоди неонові лампи.

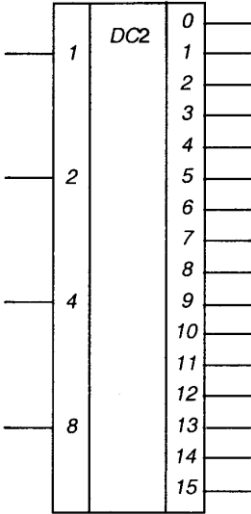


Рисунок. 8.3 – Двійковий
Двійково- дешифратор

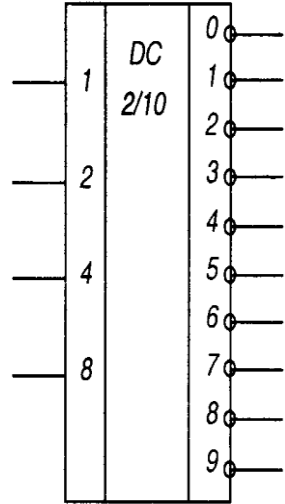
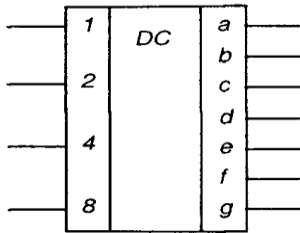
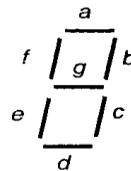


Рисунок 8.4 – десятиковий
дешифратор

Двійково-семисегментний дешифра- тор, зображений на рис. 8.5,а, перетворює двійкову комбінацію входних сигналів у комбінацію вихідних, необ- хідну для вмикання відповідної комбінації сегментів семисегментного десятикового індикатора – рис. 8.5,б.



а)



б)

Рисунок. 8.5 – Двійково-семисегментний
дешифратор (а) і схема розміщення сегментів індикатора (б)

8.4. Мультимплексори

Мультимплексори (комутатори) – це комбінаційні пристрої, що підмикають до виходу вхід (передають на вихід інформацію з входу), номер якого задає комбінація нулів і одиниць на адресних входах.

Схема чотиривходового мультимплексора та його таблиця істинності наведені на рис. 8.6

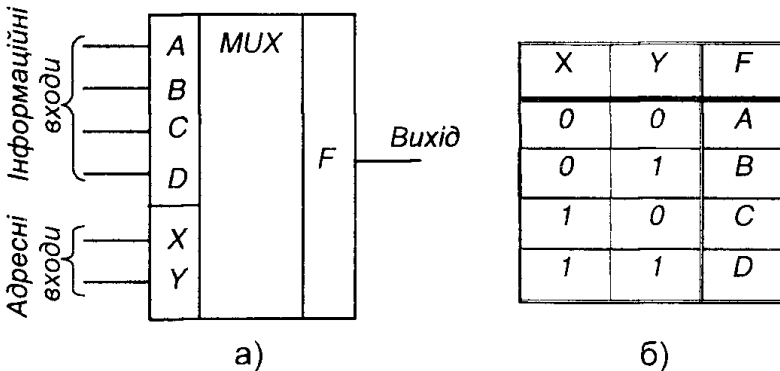


Рисунок. 8.6 – Чотиривходовий мультимплексор (а) та його таблиця істинності (б)

Логічна функція, що її реалізує цей мультимплексор, така:

$$F = Axy + Bxy + Cxy + Dxy. \quad (8.13)$$

Зауважимо, що К-МОН мультимплексори забезпечують комутацію не тільки цифрових, а й і аналогових сигналів.

8.5. Лічильники імпульсів

Одною з найрозповсюдженіших операцій у пристроях інформаційно-обчислювальної і цифрової вимірювальної техніки є фіксування кількості імпульсів – підрахунок їхньої кількості. Реалізують таку операцію **лічильники імпульсів**. Лічильники також забезпечують представлення інформації про кількість імпульсів у вигляді двійкового коду (завдяки принципу побудови).

Лічильники бувають **прості (підсумовуючі)**, у яких код збільшується

на одиницю після надходження на вхід кожного імпульсу; **віднімаючі**, у яких код відповідно зменшується після надходження на вхід кожного імпульсу) і **реверсивні** (суміщують властивості підсумовуючих і віднімаючих – можуть працювати в тому або іншому режимі за зовнішньою командою).

Як правило, лічильники будують на основі тригерів.

Схему чотирирозрядного підсумовуючого послідовного двійкового лічильника, виконаного на комбінованих RST-тригерах з імпульсними інверсними входами синхронізації наведено на рис. 8.7, його умовне позначення – на рис. 8.8, часові діаграми роботи – на рис. 8.9, таблиця переходів – у табл. 8.1.

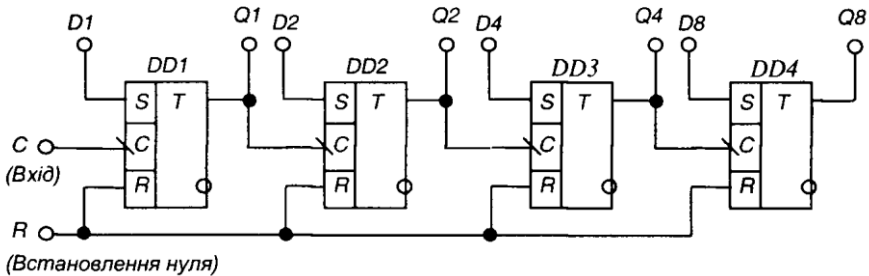


Рисунок. 8.7 – Чотирирозрядний послідовний двійковий лічильник

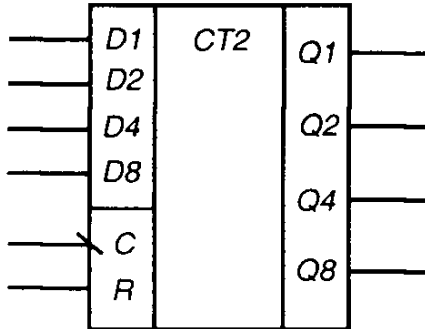


Рисунок. 8.8 – Умовне позначення чотирирозрядного послідовного двійкового лічильника

Таблиця 8.1 – Таблиця переходів
чотирирозрядного послідовного двійкового
лічильника

Стан	Q ₈	Q ₄	Q ₂	Q ₁	Стан	Q ₈	Q ₄	Q ₂	Q ₁
0	0	0	0	0	8	1	0	0	0
1	0	0	0	1	9	1	0	0	1
2	0	0	1	0	10	1	0	1	0
3	0	0	1	1	11	1	0	1	1
4	0	1	0	0	12	1	1	0	0
5	0	1	0	1	13	1	1	0	1
6	0	1	1	0	14	1	1	1	0
7	0	1	1	1	15	1	1	1	1

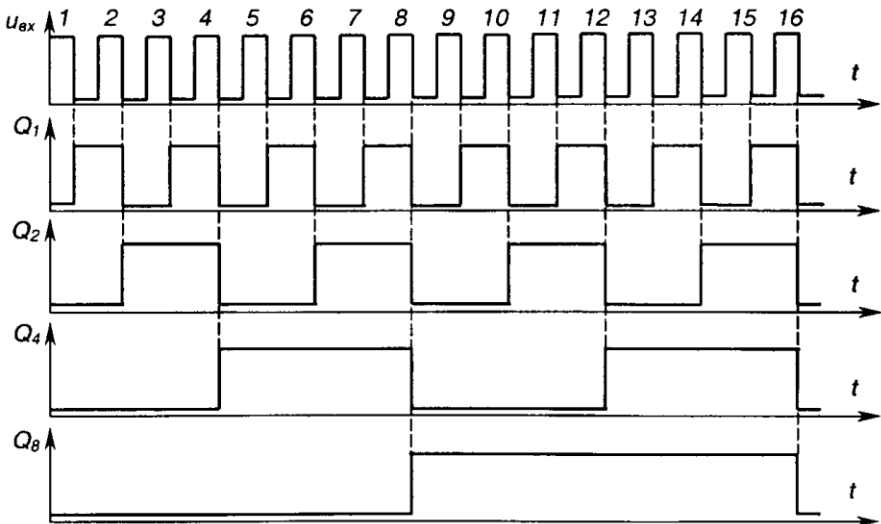


Рисунок. 8.9 – Часові діаграми роботи чотирирозрядного підсумовуючого послідовного двійкового лічильника з імпульсним інверсним лічильним входом

Лічильник називається **послідовним** тому, що вихід тригера кожного попереднього розряду з'єднано з лічильним входом (входом синхронізації) тригера наступного розряду, в результаті чого передача інформації – перемикання тригерів розрядів лічильника – відбувається послідовно одного за одним. Це визначає низьку швидкодію лічильника.

У **паралельних** лічильників інформація з розряду в розряд переда-

ється за допомогою спеціальної комбінаційної схеми, а входи синхронізації тригерів з'єднано разом, і перемикання усіх тригерів відбувається одночасно.

Як видно з таблиці переходів і часових діаграм, при безперервній роботі лічильника на його виходах Q_1 , Q_2 , Q_4 , Q_8 формується послідовний двійковий код.

Вхід R призначений для установки лічильника в нульовий стан (у даному разі – подачею сигналу логічної 1).

Входи D_1 , D_2 , D_4 , D_8 призначені для паралельного занесення чисел у лічильник – для задання початкового стану, що відрізняється від нульового.

Послідовну роботу лічильника можна порушити, обмеживши кількість його станів. Це можна зробити, вводючи зворотний зв'язок, як, наприклад, показано на рис. 8.10. Тепер при досягненні десятого стану лічильник одразу переходить у нульовий стан – ми отримали

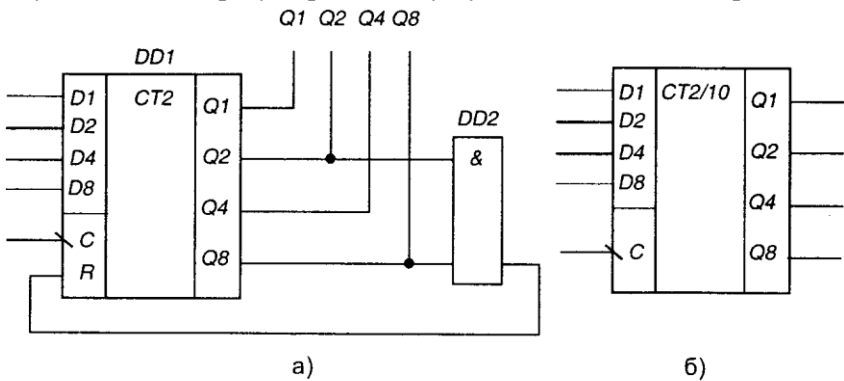


Рисунок. 8.10 – Отримання двійково-десятьового лічильника з двійкового (а) і умовне позначення двійково-десятьового лічильника (б)

Двійково-десятьовий лічильник, що має не 16, а 10 станів і формує на виходах двійково-десятьовий код. Його таблицю переходів наведено у табл. 8.2, а часові діаграми роботи – на рис. 8.11.

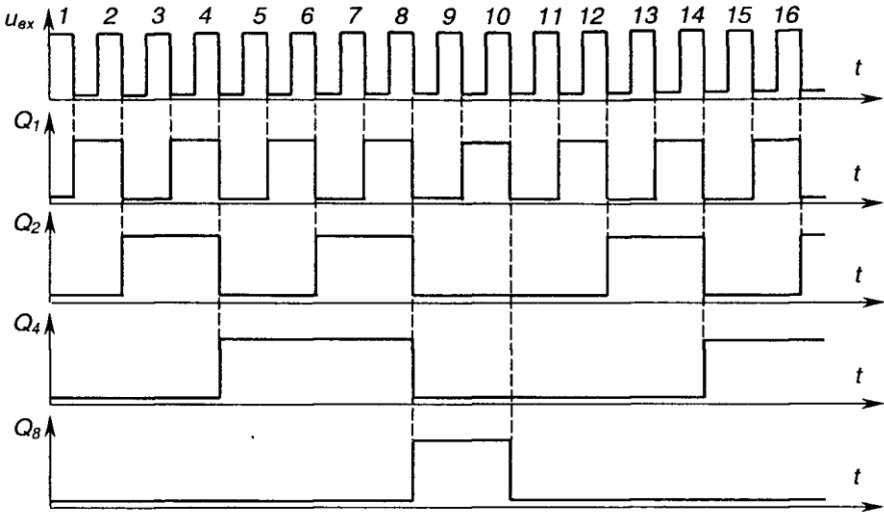


Рисунок. 8.11 – Часові діаграми роботи чотирирозрядного двійково-десятькового лічильника

Застосування таких лічильників разом із двійково-десятьковими або двійково - семисегментними дешифраторами дозволяє на основі стандартних ІМС будувати схеми керування багаторозрядними десятьковими індикаторами (розряди десяткові, а у межах десяткового розряду рахунок двійковий).

На рис. 8.12 наведено схему лічильника, що також має десять станів, але працює не в послідовному двійковому коді, бо після надходження восьмого імпульсу переходить з 7 стану у 14, після дев'ятого – у 15, а після десятого – у 0.

У вигляді ІМС випускається досить широка номенклатура лічильників. На рис. 8.13, як приклад, наведено умовне позначення чотирирозрядного реверсивного лічильника типу К561ІЕ14

Стан	Q_8	Q_4	Q_2	Q_1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	0	0	0	0

Таблиця 8.2 – Таблиця переходів чотирирозрядного послідовного двійково-десятькового лічильника

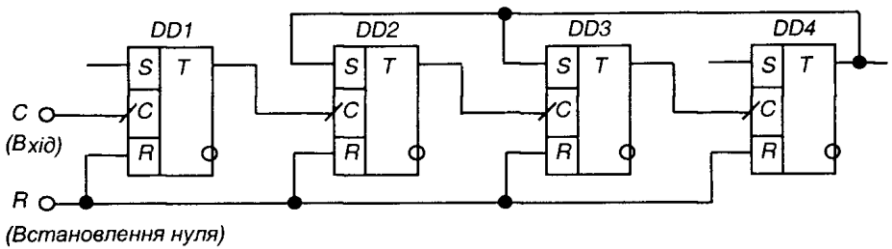


Рисунок. 8.12 – Чотирирозрядний лічильник, що має десять станів

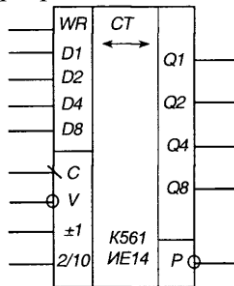


Рисунок. 8.13 – Реверсивний лічильник K561IE14

Він має виходи

- 1) Запису інформації з входів паралельного вводу інформації $D_1, D_2, D_4, D_8 - WR$
- 2) Паралельного вводу інформації, D_1, D_2, D_4, D_8
- 3) синхронізації (тактовий) – C
- 4) дозволу лічення – V
- 5) напрямку лічення (підсумовування чи віднімання) – ± 1
- 6) задання роботи в двійковому чи двійково-десятковому коді – $2/10$

Виходи

- 1) інформаційні – Q_1, Q_2, Q_4, Q_8
- 2) переносу (займу) в старший (із старшого) розряду – P

Напруга живлення подається на виводи 16 (плюс джерела живлення $+U_{ж}$) і 6 (нуль).

8.6 Регістри

Регістри призначені для запам'ятовування і зберігання інформації, представленої у вигляді багаторозрядних двійкових чисел (двійкового коду) та їхньої видачі за зовнішньою командою – це елементи короткочасної (оперативної) пам'яті.

Залежно від способу запису і видачі інформації регістри бувають:

- 1) послідовні – запис інформації в них виконується послідовно одного двійкового розряду за іншим через один вхід
- 2) паралельні – запис інформації в них виконується одночасно (паралельно) у всі розряди
- 3) послідовно-паралельні – можуть працювати як послідовні або паралельні, залежно від сигналу на спеціальному вході керування

На рис. 8.14 як приклад наведено умовне позначення універсального регістру типу 564ИР9

Він має входи:

- 4) синхронізації (тактовий) – C ;
- 5) задання режиму роботи (паралельний – послідовний) – P/S
- 6) послідовного вводу інформації (входи JK-тригера першого розряду) – $J, \bar{K}$;

- 7) паралельного вводу інформації – D_1, D_2, D_3, D_4
- 8) задання видачі інформації в прямому або інверсному коді – T/C
- 9) встановлення нульового стану – R

Виходи: Q_1, Q_2, Q_3, Q_4

При видачі інформації у послідовному коді, останній знімається з виходу Q_4 .

Напруга живлення подається на виводи 16 (плюс джерела живлення $+U_{ж}$) і 8 (нуль).

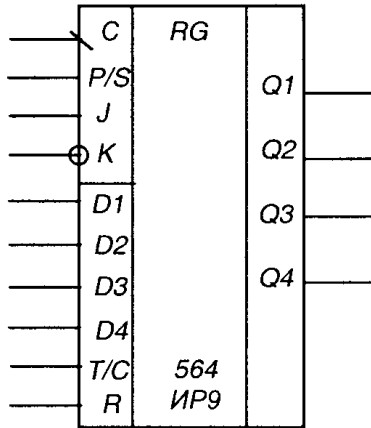


Рисунок. 8.14 – Універсальний регістр

8.7 Цифро-аналогові (ЦАП) і аналого-цифрові (АЦП) перетворювачі

Для узгодження цифрових пристроїв вимірювання і керування, що працюють з інформацією, представленою у двійковому коді, з виконавчими пристроями і датчиками, які мають аналогові відповідно вхідні та вихідні сигнали, застосовують **цифро-аналогові** і **аналого-цифрові** (ЦАП і АЦП) перетворювачі.

8.7.1 Будова і робота ЦАП

ІМС **ЦАП**, як правило, являють собою резисторні матриці. Матриця може бути виконана як дільник напруги з співвідношенням

опорів кра- тним 2^n : $R-2R-4R-8R-16R$ і т.д. (застосовують рідко, бо технологічно важко забезпечити точні значення великих опорів резисторів, що від- повідують старшим розрядам). Частіше матриця ЦАП являє собою ді- льник струмів з співвідношенням опорів $R-2R$. Також до ІМС, зазви- чай, входять транзисторні ключі (наприклад, на польових К-МОН- транзисторах), що забезпечують вмикання потрібної комбінації резис- торів за сигналами двійкового коду.

Так ІМС К572ПА1 (виконана за К-МОН-технологією) призначена для перетворення 10-розрядного прямого паралельного двійкового коду на цифрових входах у струм на аналоговому виході. Її спрощену структуру наведено на рис 8.15. Видно, що, окрім власне резисторної матриці $R-2R$, вона має ключі і вузли керування ними ВК. Вхідним аналоговим сигналом ЦАП є опорна напруга $U_{оп}$, що визначає значення напруги, яка відповідає молодшому двійковому роз- ряду. Вхідний струм I_0 , що протікає під дією $U_{оп}$, ділиться за двійковим законом у вузлах резисторної **матриці $R-2R$** . З рис. 8.16 можна бачити: приведені до місць перетинів А-А, В-В та С-С опори частини резисторів, що відтинаються у напрямку стрілок, дорівнюють $2R$ і тому зна чення струмів, які протікають через резистори $2R$, зменшуються удвічі зліва направо

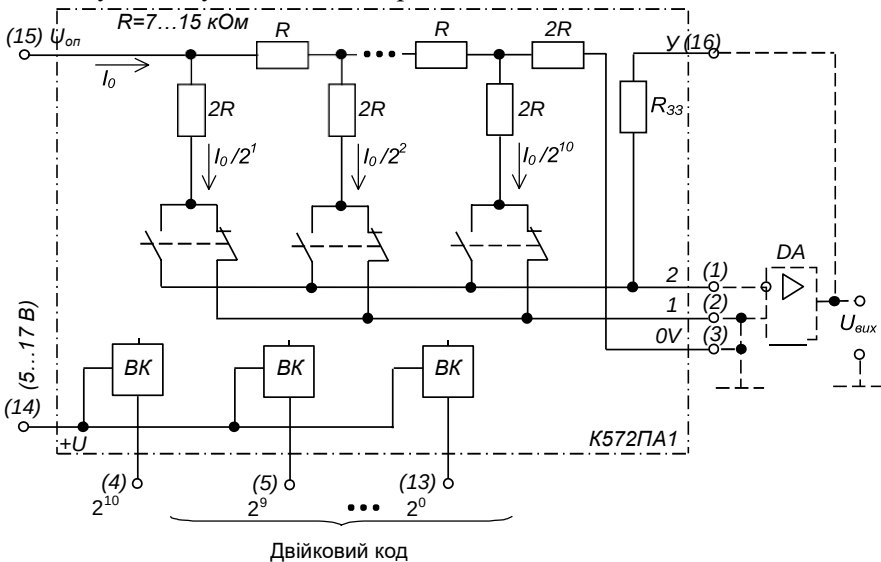


Рисунок. 8.15 – Структура ІМС К572ПА1

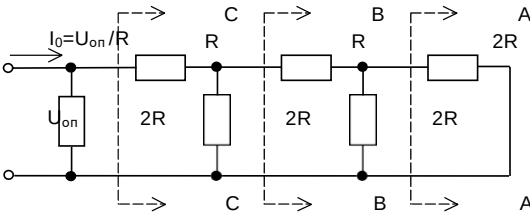


Рисунок. 8.16 – Резисторна матриця R-2R

Метод перетворення полягає в підсумовуванні у відповідності до заданих значень двійкового коду усіх розрядних струмів, зважених за двійковим законом і пропорційних значенню опорної напруги.

Більш точними є **ЦАП на основі методу перерозподілу заряду**. Вони будуються на основі конденсаторної матриці з співвідношенням ємностей кратним 2^n .

Такий ЦАП має три етапи роботи.

На першому етапі ОП, що також входить до його складу, працює як повторювач, а усі конденсатори матриці під'єднано до вхідного контакту пристрою і накопичують заряд, пропорційний вхідній напрузі.

На другому етапі схема керування перемикає конденсатори від входу на нульовий провідник. Тепер на вхід ОП, що працює вже як нуль-орган, подається напруга, що дорівнює вхідній $U_{вх}$ але з протилежним знаком.

Ну і на третьому етапі розпочинається перерозподіл заряду, коли до джерела опорної напруги $U_{оп}$ підмикається спочатку конденсатор старшого розряду матриці. На вході нуль-органу отримуємо напругу, що дорівнює $(0,5 U_{оп} - U_{вх})$. Схема керування, залежно від того, змінився стан на виході нуль-органу чи ні, залишає конденсатор у попередньому стані або знову перемикає його на нульовий провідник. Надалі те ж саме робиться з іншими конденсаторами, доки напруга на вході нуль-органу не знизиться до нуля і зміниться стан на його виході. Тоді відношення сумарної ємності, під'єднаної до нульового провідника, до ємності, що була під'єднана до $U_{оп}$, буде еквівалентна відношенню $U_{вх} / U_{оп}$.

Підвищена точність ЦАП забезпечується тим, що пристрій час від часу переводиться у режим самоналаштування. У цьому режимі до кожного з конденсаторів матриці підмикаються допоміжні конденса-

тори малої ємності доти, доки сумарна ємність не стане більшою за суму ємностей конденсаторів молодших розрядів на ємність конденсатора самого молодшого розряду. При цьому точність налаштування складає 0,25 ємності цього розряду.

8.7.2 АЦП

Світ, що нас оточує, є аналоговим. До аналогових належать процеси, що змінюються за законом безперервної функції. Так, звуки та зображення надходять до наших органів відчуття у вигляді коливань – звукових або електромагнітних. Ці коливання сприймаються органами відчуттів (слух, зір) і у вигляді імпульсів передаються до мозку. Але інформація, що передається аналоговим способом, легко викривляється у тракті передачі та потребує величезних обсягів пам'яті при використанні в техніці. Спосіб «оцифровування» інформації спрощує процеси передачі і опрацювання інформації. Оцифровування – це процес перетворення аналогової інформації в цифрову. До цифрових належать процеси, що змінюються за законом дискретної функції – зазвичай, це двійковий цифровий код. Техніка, що працює з такою інформацією називається цифровою. Цифрова інформація легко контролюється, дає стабільну і регульовану якість опрацювання і представлення процесів. Вона потребує менших обсягів пам'яті для зберігання. Отже, для світу техніки цифрова інформація підходить набагато більше, ніж аналогова.

Процес оцифровування аналогової інформації проходить два основні етапи. На першому аналогова інформація розбивається на невеликі рівні частини. На другому етапі кожна частина аналізується і зашифровується спеціальними алгоритмами у коди з послідовності одиниць і нулів

АЦП може бути побудовано на основі ЦАП, лічильника імпульсів і компаратора. Спрощену структурну схему такого АЦП наведено на рис. 8.18.

Цикл перетворення аналогового сигналу, представлений як напруга $U_{\text{вх}}$, у двійковий код складається з таких операцій.

Напруга $U_{\text{вх}}$ подається на вхід пристрою – один з входів компаратора К. Сигнал з виходу компаратора дозволяє роботу генератора імпульсів ГІ.

Сигнал Пуск встановлює нульовий стан і дозволяє роботу лічильника імпульсів ЛІ, що починає заповнюватись імпульсами ГІ.

Код з виходу ЛІ подається на цифрові входи ЦАП (входи керування ключами). У результаті з виходу ЦАП ступінчасто зростаюча напруга надходить на другий вхід компаратора. Після досягнення цієї напругою значення $U_{вх}$ компаратор забороняє роботу генератора, а на виході ЛІ маємо прямий паралельний двійковий код, що відповідає значенню $U_{вх}$.

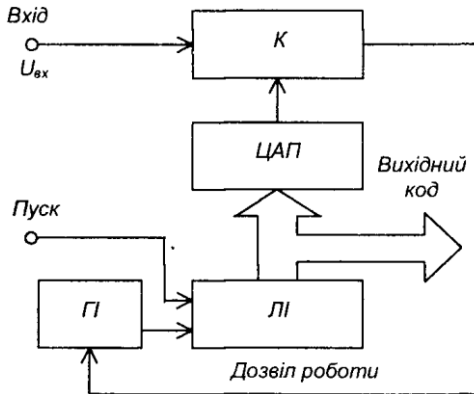


Рисунок. 8.17 – Структурна схема АЦП

Реально ж для збільшення швидкості перетворення застосовують більш складний алгоритм. До входів ЦАП замість лічильника підмикається регістр пам'яті. Після пуску схема керування встановлює регістр у стан, якому відповідає одиниця у старшому розряді і нулі у всіх інших. При цьому на виході ЦАП формується напруга, що дорівнює половині діапазону перетворення. Якщо вона менша за $U_{вх}$, то в старший розряд регістра записується нуль, а у другий за старшинством одиниця. Це відповідає напрузі на виході ЦАП, рівній половині попередньої. У разі перевищення $U_{вх}$ цієї напруги одиниця встановлюється у третьому за старшинством розряді і вихідна напруга ЦАП збільшується у 1,5 рази. Описана процедура повторюється доти, доки на виході ЦАП не сформується напруга, що відрізняється від $U_{вх}$ не більш ніж на ту, що відповідає одиниці молодшого розряду ЦАП.

Перетворювачі, що працюють за таким алгоритмом, називають АЦП послідовного наближення.

8.8 Мультивібратори і одновібратори на логічних елементах і тригерах

При побудові цифрових мікроелектронних пристроїв необхідні генератори імпульсів часто будують на таких же ІМС, що і весь пристрій

у цілому: на логічних елементах або тригерах. При цьому є велика кількість схемних рішень. Наведемо деякі з них.

Оскільки для забезпечення генерації треба мати коефіцієнт підсилення відповідного пристрою, більший за одиницю, і фазовий зсув вхідного сигналу на 360 електричних градусів, то мультивібратор може бути побудовано на двох логічних елементах з інверсією (НІ) на виході.

На рис. 8.20 наведено одну з найпростіших схем мультивібратора, виконаного на елементах К-МОН-логіки

До цього часу наголошувалось, що у логічного елемента, залежно від комбінації вхідних сигналів (що являють собою 0 або 1) отримуємо певне значення сигналу на виході (також 0 або 1)

Виникає питання, при повільній зміні вхідного сигналу від низького рівня напруги високого – від 0 до 1 (або навпаки), коли саме логічний елемент перестає сприймати вхідний сигнал як 0 і починає сприймати його як 1? Яке значення **напруги порогу перемикавання** $U_{пор}$ він має? Це залежить від типу елементної бази, на якій виконано елемент. Так, наприклад, для елементів К-МОН-логіки поріг перемикавання становить приблизно половину напруги джерела живлення.

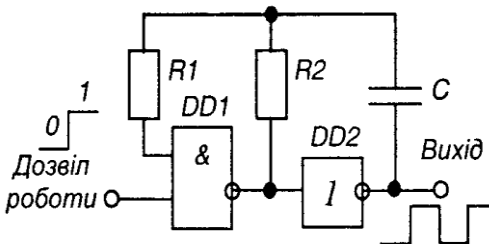


Рисунок. 8.18 – Мультивібратор на логічних елементах

З урахуванням цього робота схеми, наведеної на рис. 8.18, полягає в заряді конденсатора C по колу: вихід елемента $DD1$, R_2 , C , вихід елемента $DD2$ – коли на вході $DD1$ маємо 1 (а на виході $DD2$, як наслідок, 0), і наступного розряду по колу: вихід $DD2$, C , R_2 , вихід $DD1$ – коли на виході $DD1$ маємо 0 (на виході $DD2$ – 1). Зміна сигналу на виході $DD1$ відбувається з 0 на 1, якщо рівень напруги на резисторі R_2 досягає значення $U_{пор}$ і з 1 в 0 – якщо рівень напруги на R_2 знижується до $U_{пор}$. Значення напруги на R_2 визначається струмом заряду або розряду C (див. пп. 5.2: інтегровальний ланцюжок).

У результаті на виході пристрою маємо прямокутні імпульси з періодом надходження

$$T \approx 1,4R_2C. \quad (8.21)$$

Резистор R_1 обмежує струм розряду конденсатора C через внутрішні входні діодні захисні ланцюги елемента $DD1$ при вимиканні живлення.

Якщо у якості $DD1$ маємо двовходовий елемент І-НІ (а не просто інвертор), то другий його вхід можна використати для дозволу (1) або заборони (0) роботи мультивібратора.

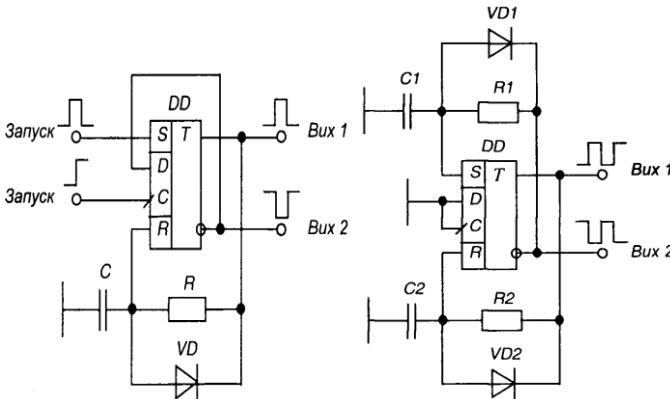


Рисунок. 8.19 – Одновібратор (а) та мультивібратор (б) на RSD-тригері. На рис. 8.19,а наведено схему одновібратора, побудованого на основі комбінованого К-МОН RSD-тригера. Тригер, як відомо, є, наприклад, двокаскадним підсилювачем із ПЗЗ або, як у даному випадку, побудований на логічних елементах.

Часозадаючий RC-ланцюжок підімкнено до прямого виходу (Вих

1) тригера. Напруга з конденсатора С подається на вхід встановлення тригера у нульовий стан R.

Можливі два способи запуску цього одновібратора.

Перший – подачею імпульсу запуску на асинхронний вхід S. При цьому тривалість імпульсу повинна бути меншою за тривалість генерованого ($t_{\text{зап}} < t_i$).

Другий – подачею імпульсу будь-якої тривалості на вхід синхронізації С (тригер реагує тільки на передній фронт імпульсу). На вхід S при цьому необхідно подати 0.

У вихідному стані на прямому виході маємо 0. Конденсатор розряджений.

Після подачі імпульсу запуску, тригер переходить в одиничний стан (залежно від виду запуску: як асинхронний RS-тригер або як синхронний D-тригер, на D-вході якого зафіксовано 1, що подається з інверсного виходу тригера – Вих 2). На прямому виході отримаємо 1.

Тепер конденсатор почне заряджатися і, коли напруга на ньому, а значить, і на вході R, досягне значення порогу перемикання, тригер повернеться в нульовий стан, а конденсатор швидко розрядиться через діод VD.

У результаті, на виходах тригера (прямому – Вих 1 і інверсному – Вих 2) ми отримали імпульс, тривалість якого

$$t_i \approx 0,7RC, \quad (8.22)$$

а одновібратор готовий до повторного запуску.

На такому ж тригері можна побудувати і автоколивальний мультивібратор, як показано на рис. 8.21,б. Бачимо, що тут до тригера під'єднано два часозадаючі ланцюги. Його робота зрозуміла з пояснень роботи одновібратора, а період генерованих імпульсів становить

$$T_1 \approx 0,7(R_1C_1 + R_2C_2). \quad (8.23)$$

За використання мультивібраторів у якості тактових для керуючих пристроїв або у годинниках важлива стабільність частоти генерованих імпульсів. Забезпечують це, застосовуючи у часозадаючих ланцюгах кварцеві резонатори.

Принцип дії кварцевого резонатора базується на п'єзоелектричному ефекті: резонатор є електромеханічною коливальною системою. Він являє собою пластинку, вирізану з кристала кварцу під певним кутом до кристалічних ґраток. Це

визначає його електромеханічні властивості – прикладення напруги до одної з осей пластинки призводить до зміни геометричних розмірів по другій. І навпаки, зміна розмірів – до виникнення напруги. На пластинку наносять металізовані шари контактів і вона поміщається у герметичний корпус. Резонансна частота такої електромеханічної коливальної системи надзвичайно стабільна завдяки малій залежності геометричних розмірів кристала кварцу від дестабілізуючих факторів, перш за все, від температури. По суті кварцевий резонатор є аналогом резонансного LC-контур, але значно перевищує його за стабільністю – у тисячі разів більш стабільний ніж контури на дискретних LC-елементах (відповідно, добротність складає 105...106 і 10...100, і якщо зміна температури на 15° у LC-контурі викликає зміну частоти, наприклад, на 1 %, то у кварцевого резонатора – на тисячні долі процента).

8.9. ІМС таймера K1006ВИ1 (555)

Для побудови генераторів імпульсів використовують також спеціалізовані ІМС. Одна з них – це ІМС **таймера** універсального призначення K1006ВИ1 (міжнародний код 555). Його умовне позначення наведено на рис. 8.23, а схему – на рис. 8.24. До складу ІМС входять два операційних підсилювачі, тригер, комбінаційна логічна схема, два транзистори (один з яких досить потужний) і вихідний підсилювач.

Таймер дозволяє будувати одно-вібратори з тривалістю імпульсів від десяти мікросекунд до однієї години і мультивібратори з частотою імпульсів до 500 кілогерц при точності 1 %. При цьому напруга живлення може задаватися в межах від 4,5 до 18 вольт, а струм навантаження сягає 100 міліампер (200 міліампер у ІМС 555).

Схеми одно-вібратора і мультивібратора, побудованих на ІМС K1006ВИ1, наведено на рис. 8.25 і 8.26 відповідно.

У цих пристроях ємність конденсатора можна задавати більшою за 100 пікофарад, а опори резисторів – від 1 кілоому до 10 мегаом (для забезпечення працездатності ІМС і виключення впливу паразитних ємностей і опорів на параметри генерованих імпульсів).

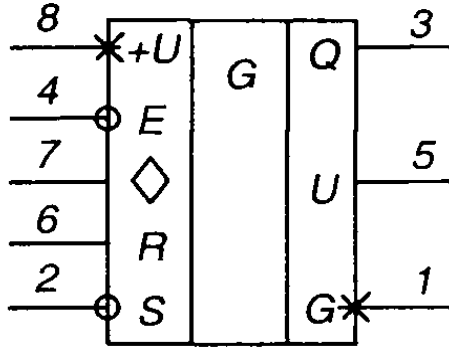


Рисунок. 8.20 – Умовне позначення таймера K1006BII

Тривалість імпульсу одновібратора становить

$$t_i \approx 1,1RC, \quad (8.24)$$

а період генерованих мультивібратором імпульсів:

– без діода

$$T \approx 0,7(R_2 + 2R_1)C; \quad (8.25)$$

– з діодом

$$T \approx 0,7(R_2 + R_1)C. \quad (8.26)$$

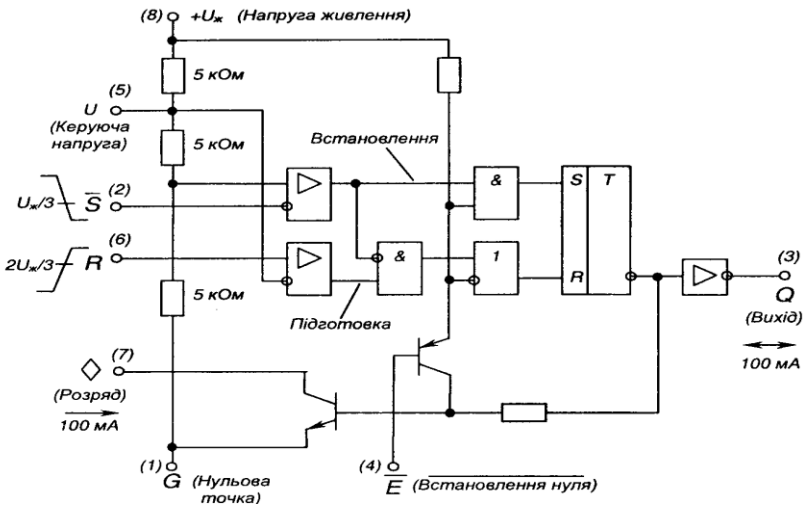


Рисунок. 8.21 – Схема ІМС таймера універсального призначення K1006BII

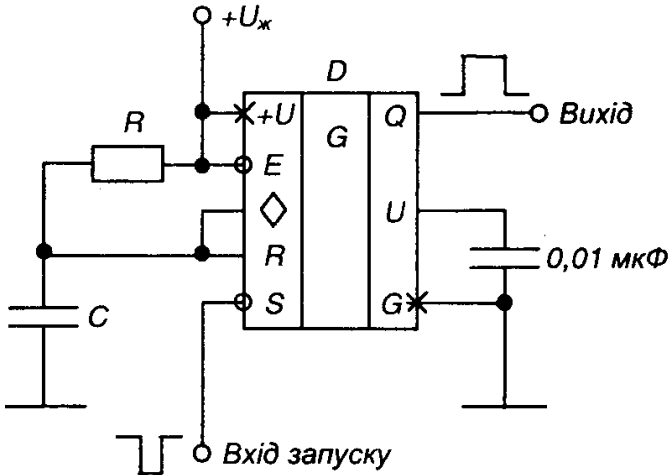


Рисунок. 8.22 – Одновібратор на ІМС К1006ВІ1

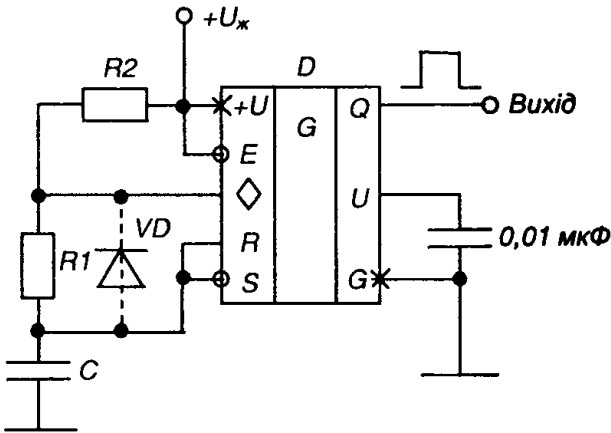


Рисунок. 8.23 – Мультивібратор на ІМС К1006ВІ1

Змінюючи співвідношення опорів R_1 і R_2 , можна змінювати шпаруватість послідовності генерованих імпульсів

$$Q = \frac{R_1 + R_2}{2R_1 + R_2} \quad (8.27)$$

ІМС таймера 555 знайшла якнайширше використання при побудові імпульсних пристроїв.

8.10. Мікропроцесорні пристрої

8.10.1. Загальні положення. Дещо з історії процесорів

Із розвитком технології і схемотехніки ІМС виникла можливість створення на одному кристалі цілого електронного пристрою, наприклад, радіоприймача, вимірювального приладу чи пристрою керування. Але, зрозуміло, що при цьому ІМС (ВІМС) стають вузькоспеціалізованими. І якщо у випадку застосування пристроїв масового призначення, що продукуються десятками тисяч, (радіоприймач, вимірювальний прилад) з цим можна змиритися, то з пристроями керування виникають великі проблеми.

Перш за все, **алгоритм** (послідовність виконання дій, що веде до успіху) керування часто є індивідуальним (наприклад, для кожного з автоматичних маніпуляторів, що працюють на складальному конвеєрі з виробництва автомобілів). Крім того, у процесі експлуатації технологічного устаткування часто виникає необхідність у заміні всього алгоритму керування або його частин (добавлення або вилучення операцій, зміна послідовності їхнього виконання). Застосування спеціалізованих ВІМС у такому разі з економічної точки зору недоцільне.

Але й побудова пристроїв керування на основі універсальних ІМС середнього ступеня інтеграції, хоча це й менше коштує, теж не є вдалим виходом, бо зміни алгоритму при цьому ведуть до зміни електричної схеми пристрою керування (повністю або у деяких її частинах). До речі, пристрої керування з незмінним алгоритмом називають пристроями з **жорсткою логікою**.

Усунути протиріччя між ступенем складності ВІМС і можливістю її універсального використання вдалося за рахунок застосування програмування виконуваних функцій (у найпростішому випадку це, наприклад, програмування коефіцієнта перерахунку лічильника імпульсів). Зараз основним типом програмованих ВІМС, що мають найбільшу інформаційну потужність, є мікропроцесор.

Мікропроцесор (МП) – це програмно керований пристрій опрацювання цифрової інформації, виконаний у вигляді однієї (рідше декількох) ВІМС. Функції МП аналогічні до функцій центрального процесора цифрової ЕОМ. Його можна вважати одним з найбільших

досягнень сучасної мікроелектроніки.

Тут слід зазначити, що спроби автоматизувати обчислювальні роботи робились дуже здавна і займалися цим найвидатніші мислителі і дослідники. Але рівень техніки і технології не давав змоги створити відповідні пристрої. Лише з середніх віків починаються зрушення у цій справі.

Так у 1642 році фізик Б. Паскаль побудував механічну машину, що виконувала операції додавання та віднімання (надалі було застосовано до п'ятдесяти таких машин). Подібні пристрої розробляли німецький математик Г.В. Лейбніц і російський математик П. Л. Чебишев (орієнтовані на розв'язання рівнянь вищої математики), а пізніше російський інженер В. Т. Орднер. «Колесо Орднера» стало основою арифмометрів. Далі були електромеханічні пристрої – на електромагнітних реле – та, нарешті, електронні – спочатку на лампах, а потім на транзисторах та ІМС.

Найбільш близьким прообразом сучасних цифрових ЕОМ є механічна «аналітична машина» англійського математика Ч. Беббіджа, запропонована ним у 1833 році і пізніше реалізована його сином (додавання двох чисел тривало 2 секунди, а множення – хвилини). У цій машині вперше було запропоновано і реалізовано ідею **процесора**, що по чергово виконує у заданій послідовності наперед задані математичні операції над числами. **Структура процесора Ч. Беббіджа**, наведена на рис. 8.27, не зазнала суттєвих змін і до сьогодні.

Арифметико-логічний пристрій (АЛП) тут виконує арифметичні або логічні дії над числами (кодами) у порядку, що задається **пристроєм керування (ПК)** згідно з **програмою**, яка вводиться до нього (наприклад, з перфоративної стрічки).

Практичне застосування процесора можливе за наявності **пристрою вводу і виводу чисел ПВВ і запам'ятовувального пристрою (ЗП)**, необхідних для вводу вихідних даних, виводу результатів та збереження проміжних результатів.

Повну механічну обчислювальну машину, що мала процесор, ПВВ і ЗП, побудовано 1936 року у Німеччині під керівництвом К. Цузе. Тоді ж з'являється й перша алгоритмічна мова для написання програм для цієї машини. З цього часу розпочинається стрімкий розвиток процесорів як основної ланки обчислювальної машини.

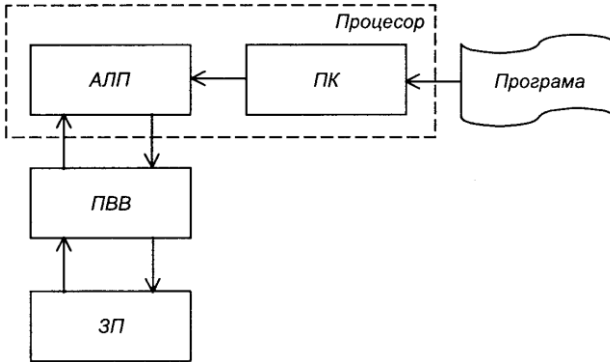


Рисунок. 8.24 – Структура процесора

У 1938 р. під керівництвом К. Цузе створено процесор на електромагнітних реле, що збільшило швидкість виконання операцій у десятки разів.

1946 року у США побудовано першу електронну обчислювальну машину ENIAC, що містила 18000 електронних ламп (додавання і віднімання тривало 200 мікросекунд, а множення 2800 мікросекунд).

Одну з перших радянських ЕОМ, призначену для серійного виробництва, «Стрела», було створено у 1953 році. Вона містила 6000 ламп, споживала потужність 150 кВА і розміщувалася у приміщенні площею 300 м²: уніфіковані її комірки (логічні елементи, тригери та ін.) містилися у 6 вертикальних стійках висотою 2,5 метри та у пульті оператора. Вона виконувала 2000 операцій за секунду. Її постійно обслуговували 5-7 техніків.

Вже у 1955 році з'являються транзисторні процесори, а з початку 60-х років – виконані на ІМС середнього ступеня інтеграції (процесор виконувався десь на п'ятдесяти ІМС).

Для появи мікропроцесора (процесора у мікровиконанні – у вигляді ІМС) необхідна була не тільки технологічна база, що вже з'явилася на кінець 60-х років, а й усвідомлення, розуміння того, що програмно керована логіка більш ефективна і простіша у виконанні при реалізації складних функцій, ніж схеми з жорсткою логікою.

Так, 1969 р. співробітник фірми INTEL (США) М. Хофф отримав пропозицію від однієї з японських фірм розробити комплект ІМС для виконання усіх математичних дій і розрахунку усіх стандартних функцій універсальним калькулятором BUSICOM. Передбачалося створення дванадцяти ІМС на 2000 транзисторів кожна, що мали б виконувати

відповідні групи з задуманих функцій. Однак М. Хофф запропонував використати лише одну універсальну ІМС з програмним керуванням, що виконувала б усі задані функції, а за необхідності – й інші.

Єдиним недоліком було збільшення часу обчислень, що незворотне за програмної реалізації (порівняно з жорсткою логікою). Але для калькулятора це було несуттєво: збільшення тривалості обчислень з десятків мікросекунд до десятків мілісекунд користувачем не помічалось.

Отже, у 1971 році фірма INTEL створила мікропроцесор INTEL 4004, що виконував операції з чотирирозрядними двійковими числами, вміщував 2250 транзисторів та мав тактову частоту 108 кГц.

З'явившись у мікрокалькуляторах, мікропроцесори (**програмна логіка**) стали застосовуватись у найрізноманітніших обчислювальних і керуючих пристроях, витісняючи схеми з жорсткою логікою.

8.10.2. Особливості роботи і використання мікропроцесорів. Двійкова система числення. Структура мікропроцесорної системи

Особливістю сучасних процесорів є те, що вони працюють з числами, представленими не у **десятковій**, а у **двійковій системі числення**: число представляється не як сума ступенів числа 10, помножених на відповідні числа від 0 до 9

$$N_{10} = a_1 \cdot 10^0 + a_2 \cdot 10^1 + a_3 \cdot 10^2 + \dots + a_{n+1} \cdot 10^n, \quad (8.28)$$

де $a_i = \{0; 1; 2; 3; \dots; 9\}$,

а як сума ступенів числа 2, помножених на 0 або 1

$$N_2 = a_1 \cdot 2^0 + a_2 \cdot 2^1 + a_3 \cdot 2^2 + \dots + a_{n+1} \cdot 2^n, \quad (8.29)$$

де $a_i = \{0; 1\}$.

У результаті основою такого процесора є двостабільний (такий, що має два стани), а не десятистабільний елемент.

Прикладом двостабільного елемента, призначеного для запам'ятовування інформації є тригер, а десятистабільного – може бути двійково-десятковий лічильник. Двійково-десятковий лічильник, що має 10 станів, побудовано на чотирьох тригерах. В той же час на чотирьох тригерах може бути побудовано пристрій (наприклад, також лічильник або регістр), що має 16 станів. Звідси видно, чому за основу взято двійкову систему числення. Її застосування при побудові цифро-

вих електронних засобів з опрацювання інформації дає змогу уникнути зниження їхніх потенційних можливостей більш ніж на третину. Отже, двійкова система при технічній реалізації є більш економною з точки зору апаратних затрат.

У 1946 році ВМС США замовили двом університетам – Гарвардському і Принстонському – розробку цифрової обчислювальної машини для систем керування артилерією лінкорів. Гарвардська структура мала два пристрої пам'яті: програм і даних. Це забезпечувало високу швидкодію, але вимагало використання двох трактів передачі інформації. Оскільки такі машини тоді будували на електронновакуумних лампах, то введення до їхнього складу будь-якого вузла призводило до значного збільшення габаритів і зменшення надійності всієї системи. Отже, перемогла принстонська структура – структура фон Неймана, за якої дані і програма зберігаються в одному і тому ж ЗП і для обміну даними між АЛП й ЗП та між ПК й ЗП використовуються однакові процедури. Після звернення МП до ЗП перший код вважається кодом команди, а наступні – адресами, даними чи командами у контексті виконання першої команди. Структуру фон Неймана застосовують при побудові і сучасних універсальних процесорів. Але зараз, особливо в мікроконтролерах, про які піде мова дещо пізніше, застосовують і гарвардську структуру, що значно підвищує їхню швидкодію. А надійність систем визначається дуже високою надійністю ВМС – надійність мікросхеми значно вища за надійність навіть однієї лампи.

На основі МП виконуються **мікропроцесорні пристрої** (МПП), перш за все мікро-ЕОМ – пристрої, що містить МП, запам'ятовувальні пристрої, органи керування і засоби зв'язку з зовнішніми пристроями – інтерфейс.

Інтерфейс (англ. interface – засіб спряження, сполучення) є сукупністю уніфікованих технічних і програмних засобів, необхідних для підключення зовнішніх пристроїв. Він забезпечує перетворення сигналів МП у сигнали, що сприймаються зовнішніми пристроями і навпаки, підсилення сигналів та становить собою апаратні засоби і набір програм передачі даних (уніфікований протокол обміну інформацією). Якщо мікро-ЕОМ призначена для керування деяким об'єктом, то вона доповнюється засобами сполучення (узгодження) з об'єктом (датчики, аналого-цифрові і цифро-аналогові перетворювачі, виконавчі пристрої і т.п.). Сукупність мікро-ЕОМ і засобів сполучення називають **мікропроцесорною системою**. Структуру такої системи приведе-

но на рис. 8.25

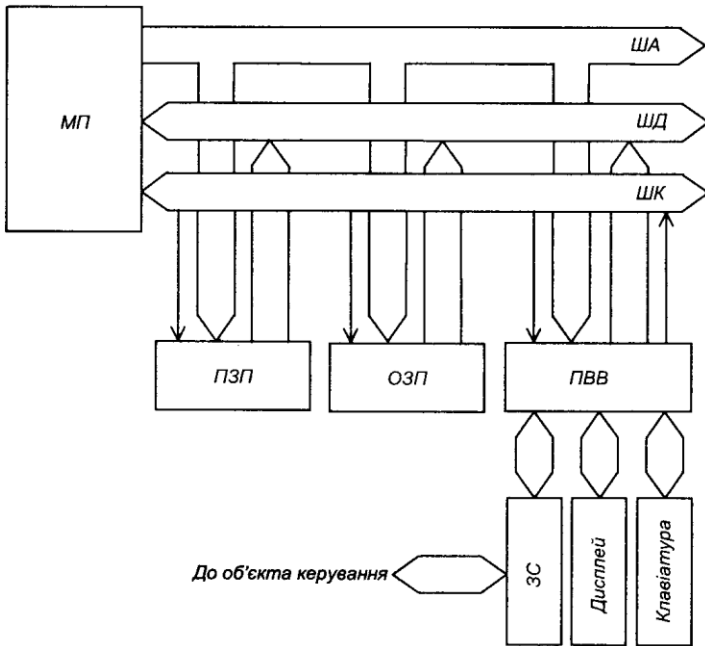


Рисунок. 8.25 – Мікропроцесорна система

МП є мозком МПП, у якому відбувається опрацювання інформації (команд і даних), що представляються у двійковому коді.

МП може виконувати дуже складні завдання з обчислень і керування, уміючи виконувати лише елементарні логічні й арифметичні операції, операції пересилання даних, порівняння двох чисел і деякі інші, за рахунок багаторазового їхнього повторення відповідно до заданої користувачем програми (визначеної послідовності команд).

Для забезпечення виконання операцій МП містить пристрої вибірки інформації з пам'яті і її дешифрування, арифметико-логічний пристрій, що є сукупністю схем, що реалізують арифметичні і логічні операції над даними, пристрій керування, який забезпечує виконання операцій МП, різні регістри для тимчасового зберігання (надоперативний ЗП) і перетворення даних і команд, тактовий генератор, що задає темп роботи МП.

Для зберігання інформації є два види ЗП: **постійний запам'ятовувальний пристрій (ПЗП)** та **оперативний за-**

пам'ятовувальний пристрій (ОЗП).

ПЗП призначений для зберігання інформації, що заноситься в нього при виготовленні МПП і зберігається при відключенні живлення. Завдяки цьому МПП після вмикання знає, як підготувати себе до роботи і що потрібно робити при одержанні тих чи інших зовнішніх команд і дій. ПЗП може працювати тільки у режимі видачі інформації. Вміст ПЗП можна змінити заміною його ІМС на інші з новим набором програм або перепрограмуванням ІМС, якщо вони це допускають.

ОЗП призначений для тимчасового зберігання даних і програм користувача, проміжних результатів роботи і працює як у режимі запису, так і у режимі видачі інформації. Якщо інформацію, що міститься в ОЗП, необхідно зберегти довгостроково, то її треба записати у зовнішній пристрій пам'яті.

ПВВ забезпечує зв'язок МП із зовнішніми пристроями – дисплеєм, клавіатурою (засоби сполучення з оператором) та **засобами сполучення (ЗС)** з керованим об'єктом – датчики, виконавчі пристрої, АЦП, ЦАП – і т.п.

Зв'язок між розглянутими модулями МПП здійснюється на основі **магістральної схеми**. Відповідно до неї всі модулі підключені до **системної магістралі**, що складається із **шини адреси (ША)**, **шини даних (ШД)** і **шини керування (ШК)**. Кожна із шин являє собою набір з деякого числа провідників – ліній.

По ША МП повідомляє, з яким з модулів чи з якою коміркою пам'яті (вказується їхній код – адреса) він буде працювати у даний момент часу.

Розрядність ША (число провідників) визначає число модулів, з яким може взаємодіяти МП. Зазвичай вона складає 16 (число модулів до 2^{16} – комірок пам'яті ПЗП чи ОЗП, модулів вводу, виводу і т. ін.).

По ШК повідомляється характер взаємодії: введення даних чи їхній вивід (запис чи читання).

По ШД дані надходять у процесор і виводяться із нього, тобто ШД двонаправлена. У кожний конкретний момент часу ШД використовується тільки для вводу або тільки для виводу.

Оскільки до ШД приєднано всі модулі одночасно, то, щоб виключити їхній взаємний вплив, підмикання до провідників шини виконується за допомогою елементів, що мають три стани: **два – 1; 0 (логічні)** і **третій – стан відключення від шини (нелогічний)**. МП вибирає

один модуль для виводу даних на ШД, задаючи його адресу на ША. Виходи інших модулів при цьому знаходяться у третьому стані – відімкнені від ШД.

Розрядність ШД визначає розрядність двійкових даних (розрядність слова даних), з якими може оперувати МП. Зазвичай ШД має 4, 8, 16 або 32 розряди: МП опрацьовує двійкові числа (слова) з 4, 8, 16 або 32 розрядів – біт (8 біт складають байт, $2^{10} = 1024$ біт – кілобайт).

Типовий МП зазвичай виконують у вигляді ВІМС, що має до 40 електричних виводів. Так у восьмирозрядного МП: 8 виводів забезпечують підмикання до ШД, 16 – до ША, 5-7 – до тактового генератора і джерела живлення, решта – до ШК. Є й мікро-ЕОМ у вигляді ВІМС. Їх називають однокристальними.

8.10.3. Структура мікропроцесора

Структурну схему МП наведено на рис. 8.26.

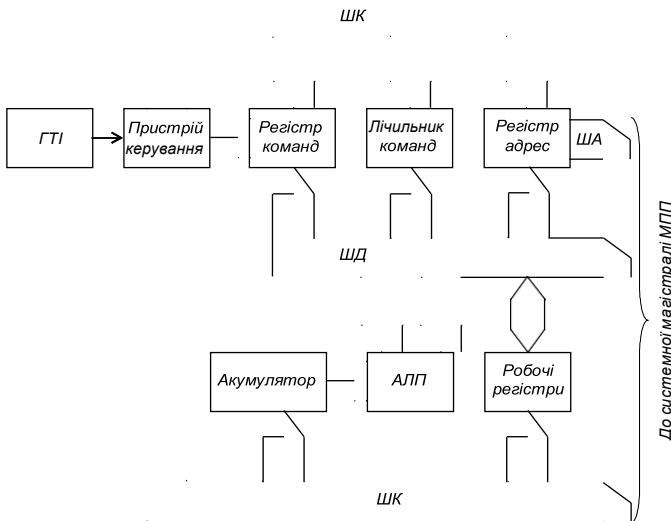


Рисунок. 8.26 – Структура мікропроцесора

МП містить три основні пристрої: арифметико-логічний пристрій (АЛП), регістр даних (РД) і пристрій керування (ПК). Ці та інші пристрої, що приймають участь у пересиланні даних, з'єднано шинами

внутрішньої магістралі: ША, ШД і ШК.

Синхронізує і задає темп роботи (швидкодію) МП **генератор тактових імпульсів** (ГТІ).

АЛП є комбінаційним пристроєм, у якому відбувається виконання арифметичних і логічних операцій над даними.

Регістри забезпечують виконання функцій МП за рахунок зберігання кодів даних, адрес, команд та ін.

Акумулятор є основним регістром МП при діях з даними. Більшість арифметичних і логічних операцій з двома словами даних здійснюється з використанням АЛП і акумулятора.

У **регістрі стану** (ознак або прапорців) зберігаються ознаки результату побіжної операції АЛУ (нульовий, додатний, негативний та ін.). Ці ознаки використовуються при виконанні наступних команд, наприклад, для розгалуження програми.

Лічильник команд зберігає номер команди, що виконується, і тієї, що буде виконуватись наступною. Перед початком виконання програми до нього заноситься адреса її першої команди.

Регістр адреси пам'яті містить адресу пам'яті, що буде використана МП у наступному такті роботи. Його виходом є ША.

Регістр команд зберігає код команди, що виконується у даний момент.

Робочі регістри є внутрішньою надоперативною пам'яттю МП і застосовуються для проміжного зберігання кодів даних і адрес, забезпечуючи підвищення швидкодії.

Підключення внутрішньої магістралі МП до системної магістралі МПУ відбувається через **буферні регістри інтерфейсу і вихідні підсилювачі**.

Пристрій керування узгоджує роботу всіх вузлів МП.

8.10.4. Запам'ятовувальні пристрої. Загальні відомості

Запам'ятовувальні пристрої (ЗП) забезпечують зберігання програми роботи, вихідних даних і результатів опрацювання, обмін інформацією між окремими частинами мікропроцесорної системи. Будуються вони на напівпровідникових ІМС. Мікросхеми пам'яті складають біля 40 % у загальному обсязі випуску ІМС. Також ЗП входять до складу ВІМС: мікропроцесорів, мікроконтролерів та ін.

Максимально можливий обсяг інформації ЗП визначається його **інформаційною ємністю** в бітах (**біт** – мінімальний обсяг цифрової інформації, може приймати значення 0 або 1) або **словах (байтах)**, що складаються з декількох біт, наприклад, з чотирьох або восьми). Біт зберігається у **запам'ятовувальному елементі (ЗЕ)**, а слово – у **запам'ятовувальній комірці**, що є групою ЗЕ, до яких можливе лише одноразове звернення.

Інформаційна ємність вимірюється, наприклад, в кілобітах ($1 \text{ Кбіт} = 2^{10} = 1024 \text{ біт}$), кілобайтах ($1 \text{ Кбайт} = 2^{10} = 1024 \text{ байт}$), або в мегабітах ($1 \text{ Мбіт} = 2^{20} = 1048576 \text{ біт}$), мегабайтах ($1 \text{ Мбайт} = 2^{20} = 1048576 \text{ байт}$) і т.п.

Організація ЗП визначає, скільки слів і якої розрядності він зберігає. Так, наприклад, при ємності 512 біт може зберігатися 64 восьмирозрядних слова, або 128 чотирирозрядних.

Швидкодія ЗП оцінюється часом зчитування і запису.

Цифрові обчислювальні пристрої мають шинну структуру, за якої звертання до комірки ЗП відбувається за номером, що називається **адресою**.

8.10.5 Запам'ятовувальні пристрої для зберігання оперативної інформації (ОЗП)

Такі ЗП **RAM-типу** (Random Acces Memory – ОЗП) поділяються на статичні і динамічні.

У **статичних ЗП – SRAM** (Static RAM) – як ЗЕ використовуються тригери з ланцюгами установки і встановлення нуля. При К-МОН-технології тригер складається з шести транзисторів. Тому такі ЗП досить дорогі і займають багато місця на кристалі, але мають велику швидкодію.

В **динамічних ЗП – DRAM** (Dynamic RAM) – дані зберігаються у вигляді зарядів ємностей МОН-структур. Основою ЗЕ тут є просто конденсатор невеликої ємності. Такий ЗЕ значно простіший за тригетний, що дозволяє розмістити на кристалі значно більше ЗЕ. Оскільки конденсатор з часом втрачає свій заряд, то зберігання даних вимагає їхньої періодичної регенерації (через кожні декілька мікросекунд) за допомогою спеціальних підсилювачів-регенераторів. Зараз ємність

динамічних ЗП становить до 128 Мбіт, у стані розробки ЗП на 256 Мбіт (з кількістю транзисторів на кристалі до 300 мільйонів. Динамічні ЗП у 4-5 разів дешевші ніж статичні, та у стількижразів мають більшу інформаційну ємність.

Обидва ці типи пам'яті є **енергозалежними** – при вимкненні джерела живлення ІМС інформація безповоротно губиться.

8.10.6 Запам'ятовувальні пристрої для зберігання постійної інформації (ПЗП)

Є декілька типів ЗП для зберігання постійної інформації – **ROM-типу** (Read On Memory – ПЗП). Як ЗЕ в них використовують перемички, діоди, біполярні і МОН-транзистори.

У **маскові ЗП типу ROM(M)** інформація записується при виготовленні ІМС за допомогою спеціального шаблону – маски. Матриця діодного ЗП такого типу є координатною сіткою з горизонтальних ліній вибірки слів і вертикальних ліній зчитування. Код слова визначається наявністю діода (що відповідає одиниці) або його відсутності (відповідає нулю) у вузлах координатної сітки.

Такі ЗП компактні і дешеві. Їх застосовують для зберігання інформації масового призначення – кодів літер і цифр, таблиць типових функцій, стандартного програмного забезпечення і т.п. Користувач не може змінити інформацію, що зберігається.

Мікросхеми ЗП типу PROM (Programmable ROM – програмовані ПЗП) програмують одноразово видаленням або створенням перемичок у вузлах координатної сітки. У вихідній заготовці наявні (або відсутні) всі перемички. При програмуванні користувачем за допомогою спеціального програмуючого пристрою залишаються (або видаляються) тільки необхідні. Перемички можуть бути або плавкими елементами, увімкненими послідовно з діодом, або двома зустрічно увімкненими діодами, один з яких пробивається при програмуванні.

ЗП типу EPROM дозволяють не тільки записувати в них інформацію, а й стирати її та замінювати на нову – вони є **репрограмовуваними**. Як ЗЕ в них використано **транзистори ЛІЗМОН-типу** (МОН-транзистори з лавинною інжекцією заряду). Такі транзистори мають так званий плаваючий затвор – обмежену з усіх боків діелектриком провідну зону. Введений у неї як у пастку в результаті лавинного про-

бою під дією імпульсу напруги у 20-25 В заряд зберігається дуже тривалий час. Цей заряд забезпечує закритий стан транзистора. Стирання інформації відбувається під дією ультрафіолетового опромінення, для чого корпус ІМС має спеціальне прозоре віконце. Виникнення фото- і теплових струмів дозволяє заряду покинути плаваючий затвор. Стирання триває декілька хвилин, одразу стирається вся інформація. Опромінення веде до змін властивостей матеріалів транзисторів, тому число циклів перепрограмування складає 10-100.

Новітні **ЗП типу EEPROM** також є репрограмовуваними. Їхньою основою є МНОН-транзистори, що, на відміну від звичайних МОН-транзисторів мають двошаровий підзатворний діелектрик – окрім тонкого шару SiO_2 є ще більш товстий шар нітриду кремнію Si_3N_4 (звідси літера Н в аббревіатурі). Під дією електричного поля достатньо високої напруженості носії заряду проходять через тонкий шар і скопичуються на межі розділу шарів. Після зняття поля заряд залишається у приграничному шарі нітриду кремнію і вже не розсмоктується, що й забезпечує зберігання інформації на протязі десятків років. Для стирання інформації необхідно видалити носії заряду з приграничного шару, для чого подається напруга, що створює електричне поле протилежної направленості. При цьому інформацію можна стирати не зі всього кристалу, а вибірково. Тривалість процесу досить коротка. Кількість циклів перепрограмування складає 10^4 - 10^6 разів.

Найвищим досягненням у розробці напівпровідникових програмованих ЗП є створення **флеш-пам'яті**. Її ЗЕ подібні елементам пам'яті типу EEPROM, але в схемах флеш-пам'яті не передбачене стирання окремих слів. Інформація може стиратися або вся одразу, або достатньо великими блоками за єдиним сигналом, миттєво (flash – спалах). За своєю дією флеш-пам'ять вже подібна до пам'яті ОЗУ але є **енергонезалежною**. З розвитком технології і збільшенням місткості інформації пристрої флеш-пам'яті поступово витісняють такі накопичувачі, як жорсткі магнітні диски, оптичні диски та інші носії. Вони у сотні разів скорочують споживану потужність, мають велику надійність, не- великі розміри і вагу, в декілька разів збільшують швидкодію пристроїв пам'яті. Однією з найважливіших властивостей є також відсутність в ЗП цього типу рухомих механічних частин, в тому числі таких, що обертаються, і повна технологічна сумісність з ІМС. Флеш-пам'ять може бути частиною

багатьох типів ІМС мікроконтролерів (що забезпечує універсальність їхнього застосування за рахунок можливості багаторазового перепрограмування) та ін.

Також зазначимо, що це є одним з прикладів поступової відмови людства від колеса у багатьох застосуваннях.

8.10.7 Алгоритм. Способи представлення алгоритму

Алгоритм (від латинської трансліції імені математика аль-Хорезмі) – це спосіб (програма) розв’язання обчислювальних та інших задач, що точно вказує, як і в якій послідовності отримати результат, однозначно визначений вихідними даними.

Отже, послідовність виконання дій пристрою керування і є алгоритмом. Задати алгоритм можна різними способами.

Перш за все – це **описове представлення алгоритму** на «людській» мові. Така форма представлення прийнятна навіть для неспеціаліста у пристроях керування, але громіздка, допускає тлумачення. А отже, не годиться для створення більш-менш складного пристрою керування, як з жорсткою логікою, так і програмованого.

Більш впорядкованим є **представлення алгоритму у вигляді таблиці**.

Досить популярним є **графічне представлення алгоритму** – у вигляді граф-схеми. Воно забезпечує наочність, оперативність огляду.

На основі алгоритму складають схему електричну принципову пристрою керування з жорсткою логікою або програму роботи програмованого. Це зрештою теж є способами представлення алгоритму.

8.10.8 Задання і виконання програм МПП

Для забезпечення виконання МПП необхідних функцій у його ПЗП (іноді у ОЗП) записується набір програм (стандартних, що реалізують, наприклад, арифметичні і логічні операції, обчислення тригонометричних функцій і т.п., а також і спеціальних, що реалізують, наприклад, алгоритми керування конкретним об’єктом).

При складанні програм програміст повинен дати МП детальний опис його дій. МП може працювати з великою швидкістю і точністю протягом тривалого часу, але тільки за чітким завданням, що не

допус- кає тлумачень. МПП може реагувати на зміну умов, якщо в пам'яті у нього є програма, що повідомляє МП, як йому поводитись у нових об- ставинах. МП діє логічно, але не має творчих здібностей (принаймні, сьогодні). Удавана розумність різних МПП є результатом великого числа програм, закладених у них. Хоча слід зазначити, що в наш час інтенсивно ведуться роботи з створення так званого штучного інтелек- ту (набору програм, що імітують інтелектуальну діяльність людини). І на цьому шляху, як відомо, здобуто чимало досягнень. Вважається, що створення «розумних» машин буде у першому наближенні завершено тоді, коли вони навчатися створювати собі подібних.

Для ініціалізації виконання програми в МП заноситься її початкова адреса (адреса комірки ПЗП чи ОЗП) і подається команда ПУСК. При роботі МП забезпечує таку послідовність дій:

- а) витяг даних (коду команди, числа) з пам'яті (ПЗП чи ОЗП);
- б) розшифровка команди;
- в) виконання команди і т.д. знову.

При цьому, наприклад, щоб переслати число з ПЗП до ОЗП, МП, будучи тактованим (синхронним) пристроєм, у найближчому такті роботи за сигналами тактового генератора задає згідно з програмою на ША код адреси комірки пам'яті ПЗП. Після завершення перехідних процесів, пов'язаних з зарядом паразитних ємностей провідників (ліній) ША (від чого, до речі, у значній мірі залежить швидкодія пристрою), у другому такті роботи МП по ШК дає команду на видачу коду з ПЗП на ШД (читання), а у наступному такті – команду по ШК на прийом коду у свій внутрішній регістр пам'яті, що називається акумулятором.

Надалі, у подальших тактах роботи, відповідно задається на ША номер комірки ОЗП та подається по ШК команда запису і число, на- решті, потрапляє на визначене програмою місце.

З опису цієї послідовності дій зрозуміло, чому **програмні пристрої обчислення і керування значною мірою поступаються у швидкодії порівняно з пристроями з жорсткою логікою**, де код з одного вузла до іншого зазвичай передається за один такт по окремо виділених лініях зв'язку.

На робочому рівні МП реагує на список команд, представлений, як

уже зазначалося, у вигляді набору двійкових кодів, що називається **машинною програмою**. Таке представлення програм є винятково важким для оперативного сприйняття людиною.

8.10.9 Шістнадцяткова система числення

Деяке спрощення при написанні і читанні команд дає використання **шістнадцяткової системи числення**. Вона містить 16 символів, кожному з яких відповідає чотирирозрядне число у двійковому коді: число представляється як сума ступенів числа 16, помножених відповідно на 0, 1, 2, ..., 9, A, B, C, D, E, F

$$N_{16} = a_1 \cdot 16^0 + a_2 \cdot 16^1 + a_3 \cdot 16^2 + \dots + a_{n+1} \cdot 16^n, \quad (8.30)$$

де $a_i = \{0; 1; 2; 3; \dots; 9; A; B; C; D; E; F\}$.

Табл. 8.3 показує відповідність запису чисел у десятковому, двійковому і шістнадцятковому кодах.

Таблиця 8.3 – **Запис чисел у різних кодах (системах числення)**

Десятковий код	Двійковий код	дцятковий код
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Перетворити двійковий код у шістнадцятковий дуже легко: двійкове число поділяється на групи по чотири розряди, починаючи з молодшого – на **тетради**. Кожна група замінюється відповідним шістнадцятковим символом. При зворотному перетворенні кожен

символ замі- нюється двійковою тетрадою.

Наприклад, $0101\ 0111\ 1101\ 1111\ 0001_2 = 57DF1_{16}$.

Таким чином, шістнадцятковий код, з точки зору застосування для запису програм для МПП, можна вважати більш компактним записом двійкового.

8.10.10

Асемблер

Програма, записана у шістнадцятковому коді, усе-таки залишається складною для розуміння.

Виходом є використання мови простого програмування – **асемблера**. Відповідна **програма-ретранслятор** переводить слова і фрази програми, представлені у вигляді **мнемонік** (англійських скорочень назв) асемблера, у машинний код, який потім буде занесено у ІМС ПЗП, що підмикаються для цього до спеціального адаптера машини-програматора.

Приклади мнемонік асемблера та їхня відповідність у шістнадцятковому коді наведено у табл. 8.4.

Таблиця 8.4 – Приклади команд асемблера

Команда	Мнемоніка	Шістнадцятковий код
1. Читання даних з об'єкта з адресою F у внутрішній регістр пам'яті A	INF F	DB FA
2. Порозрядне логічне множення чисел, що знаходяться у внутрішніх регістрах пам'яті A і B	ANA B	A0
3. Додавання числа, що зберігається у внутрішньому регістрі C до числа, що зберігається у внутрішньому регістрі A	ADD C	81
4. Вивід числа, що зберігається у внутрішньому регістрі A до об'єкта, адреса якого знаходиться у внутрішніх регістрах пам'яті B і C	AX B	02

Відзначимо також, що кожній команді асемблера відповідає **мікрокоманда** МП – набір дій МП (**мікрооперацій**), запрограмованих при виробництві МП у його внутрішньому ПЗП. Ініціює виконання мікропрограми пристрій керування МП після отримання коду команди.

При розробці програм для МПП, що виконує функції керування, складають узагальнений алгоритм виконання програми – послідов-

ність дій з опитування вхідних даних, їхнього опрацювання та формування команд керування на відповідних виходах.

Далі алгоритм розбивають на окремі фрагменти і деталізують дії у кожному з них до рівня, що відповідає **системі команд** вибраного типу МП (зазвичай – від декількох десятків і десь до півтори сотень команд).

Програма спочатку складається у вигляді **списку мнемонік**, що називається **вихідною програмою**. Кожному коду присвоюється номер комірки пам'яті. Потім вручну або за допомогою ЕОМ програма переводиться на машинну мову. Тепер вона називається **об'єктною програмою** і заноситься в ПЗП чи ОЗП МПП.

Використання алгоритмічної мови високого рівня (BASIC, FORTRAN або ін.) недоцільне, бо веде до втрати програмістом контролю за використанням ресурсів пам'яті. Це може призвести після її транслявання у машинні коди до значного перевищення оптимального обсягу ПЗП та втрати швидкодії. А це, в свою чергу, веде до збільшення апаратних витрат (числа ІМС ПЗП) та загрожує втратою можливості керуючого МПП працювати у **реальному масштабі часу**, коли реакція МПП на зміну ситуації у об'єкті керування повинна відбуватися зі швидкістю, не меншою за ту, з якою відбуваються процеси в об'єкті.

При використанні МПП для керування можливість зміни програми ПЗП заміною ІМС чи їхнім перепрограмуванням (у залежності від типу) забезпечує використання даного МПП у різних застосуваннях для вирішення різних задач без будь-яких схемних змін в апаратурі – функція МПП задається його програмуванням.

Поліпшення техніко-економічних характеристик МП дозволило надзвичайно ефективно використовувати їх у різних пристроях цифрового опрацювання інформації і керування (як у якості центральних процесорів, так і у периферійних пристроях ЕОМ, у мікрокалькуляторах, у системах керування і регулювання потоковими лініями, верстатами-автоматами, роботами-маніпуляторами, стабілізованими джерелами живлення, в електронних пристроях – телевізорах, радіостанціях, вимірювальних приладах, цифрових фотоапаратах, іграшках і т.п.).

Це підняло на новий рівень ефективність роботи технічних пристроїв і систем.

Завдяки використанню МПП ЕОМ у наш час перетворилися у предмет

побутової техніки. Взагалі МПП немає різниці чим керувати: ракетою, світлофором чи пральною машиною. Але, зазвичай, для конкретних застосувань вибирають з широкої номенклатури МП необхідної інформаційної потужності і будують МПП у відповідному конструктивному та кліматичному виконанні.

Потужність МП визначається його здатністю опрацьовувати дані і оцінюється трьома параметрами: довжиною слова даних (кількість розрядів ШД), довжиною слова пам'яті (кількість розрядів ША – визначає кількість об'єктів, до яких може звертатися МП), швидкістю виконання команд.

8.10.11

Програмовані логічні контролери

Зрозуміло, що продуктивність устаткування та точність виконання технологічних операцій значною мірою залежать від ступеню автоматизації. Також зрозуміло, що сьогодні найкраще забезпечують це електронні пристрої керування, особливо цифрові.

Тут слід зазначити, що цифрові пристрої керування реалізуються як автомати з жорсткою логікою або програмовані.

Апаратні методи побудови цифрових пристроїв керування з жорсткою логікою близькі до методів побудови широко розповсюджених раніше аналогових систем керування: для виконання кожної функції використовується окремий елемент або група елементів і введення нової функції вимагає введення нових елементів. Системи керування, реалізовані такими методами, мають високу швидкодію та надійність, традиційні принципи їхньої побудови не викликають труднощів при розробці та експлуатації (з точки зору застосовуваного при регламентних та ремонтних роботах обладнання, приладів, з огляду на рівень кваліфікації персоналу). В той же час, вони складаються з великої кількості електронних компонентів (в основному інтегральних мікросхем – ІМС – середнього ступеню інтеграції), громіздкі, мають малу гнучкість при необхідності змін алгоритму керування (або частіше взагалі не дозволяють цього робити) та дорого коштують. Вони вимагають індивідуальної розробки, причому, кількість типів функціональних модулів, що створюють функціонально закінчену систему, яка слугує для побудови різних систем керування, виявляється значним. Отже використання пристроїв керування з жорсткою логікою вимагає

у кожному окремому випадку розробки спеціального унікального пристрою. Для створення МПП керування технічними пристроями зручно використовувати програмовані мікроконтролери, що випускаються промисловістю.

Програмовані пристрої будуються на основі процесорів, як правило, у мікровиконанні – мікропроцесорів.

Слід зазначити, що побудова пристрою керування класу, що нами розглядається, на основі мікропроцесорного набору (власне процесор, оперативний та постійний запам'ятовувальні пристрої, інтерфейс та ін.) за величиною апаратних затрат мало відрізняється від його реалізації на ІМС середнього ступеня інтеграції. Вартість розробки та реалізації такого пристрою виявляється навіть більшою за вартість пристрою з жорсткою логікою на ІМС середнього ступеню інтеграції. Крім того, мікропроцесорні пристрої висувають зовсім інші вимоги до кваліфікації як розробників апаратури, так і до тих, що забезпечують експлуатацію. Але гнучкість програмованих пристроїв керування при необхідності змін алгоритму, що має величезне значення як на етапах розробки так і при експлуатації одиночно або малосерійно продукovanого технологічного устаткування, робить їхнє застосування вкрай перспективним.

Тут могли б придатися програмовані логічні контролери. Зазначимо, що раніше контролерами називали багатопозиційні перемикачі для комутації електричних кіл (наприклад, як ті, що застосовуються для керування електродвигуном трамваю).

Програмований логічний контролер – це програмований логічний пристрій керування з мовою програмування, доступною для неспеціалістів з інформатики (спочатку, наприклад, мали мову драбинчастих діаграм і призначалися для заміни релейних шаф керування послідовними логічними процесами в умовах промисловості у реальному масштабі часу).

Малогабаритні і гнучкі у програмуванні ПЛК спочатку використовувалися для заміни логічних пристроїв керування, виконаних на електромагнітних реле, що займали цілі шафи. Надалі, з появою мікроелектронних ЦАП і АЦП, їх почали застосовувати і для розв'язання задач, пов'язаних з обчисленням та порівнянням аналогових сигналів. ПЛК монтуються на одній або декількох платах (у вигляді одного або декількох модулів) і доповнюються пристроями програмування і

індикації з клавіатурою та цифровим дисплеєм.

Більшу універсальність мають **багатоплатні** ПЛК, що являють собою набір модулів: процесора, ПЗП, ОЗП, ЦАП, АЦП, вводу та виводу даних, підсилення потужності, джерела живлення та ін. З цих модулів створюється необхідна у конкретному випадку конфігурація ПЛК, для чого необхідний їхній набір встановлюється у так звану корзину – конструкцію з основою, на якій розміщено рознімачі для підмикання модулів до системної магістралі і джерела живлення. Підмикання до джерел сигналів та до виконуючих пристроїв об'єкту керування здійснюється окремими кабелями.

Таким чином, багатоплатний ПЛК є універсальним конструктором для створення програмованих пристроїв керування.

Такі ПЛК широко застосовувались до тих пір, доки з розвитком мікросхемотехніки стало можливим усі вузли (до 95-98 % слабкоструктурних елементів) контролера розміщувати у одній ВМС – **одноплатному** ПЛК. Силові елементи і джерело живлення при цьому розміщують на окремих платах. Одноплатні ПЛК мають значно менші габаритні розміри, ніж багатоплатні, простіші у виготовленні і дешевші. Вони більш надійні, бо не мають корзини з великою кількістю дорогих і малонадійних рознімачів. Але вони не є універсальними. Тому фірми-виробники зазвичай випускають набори таких ПЛК з різними можливостями для забезпечення керування типовими для деякої галузі техніки об'єктами.

Виходячи з недоліків чи переваг одно- і багатоплатних ПЛК у конкретних застосуваннях використовують як одні, так і інші. Побудовані на мікропроцесорах контролери, мають спрощену систему команд, випускалися серійно і були достатньо дешеві, порівняно з вартістю розробки та виготовлення пристрою керування з жорсткою логікою. Але як пристрої керування технологічного устаткування програмовані логічні контролери широкого розповсюдження не знайшли, бо були спочатку громіздкими, все ж досить дорогими, і для багатьох можливих застосувань (наприклад, для керування електронними ключами джерел живлення технологічного устаткування) повільно діючими. Також важливим чинником у обмеженні їхнього розповсюдження були обмежена доступність для пересічного споживача як власне контролерів, так і інформації про користування ними

8.10.12 Мікроконтролери

Починаючи десь з середини 80-х років минулого сторіччя на світовому ринку електронної продукції з'являються ІМС мікроконтролерів. Так перші восьмирозрядні мікроконтролери були створені фірмою Intel (США) у 1982 році. Зараз такі мікроконтролери у світі продукують більш як тридцять фірм (деяку незручність створює несумісність мікроконтролерів різних фірм-виробників).

До речі, подібні ІМС з'явилися у кінці 80-х років минулого сторіччя і в СРСР. Вони називалися однокристальними електронними обчислювальними машинами (серія КР1816). Знову ж таки, вони були дорогими, малодоступними, бракувало інформації про їхнє застосування.

На територію країн СНГ мікроконтролери, виготовлені у Південно-Східній Азії, поставляються, в основному, фірм Atmel та Microchip Technology Inc. (США). Ці мікроконтролери мають особливу популярність також і у розробників електронної техніки малої та середньої складності. Їхня номенклатура надзвичайно велика. Так тільки фірма Atmel у кінці 90-х років минулого сторіччя випустила 140 моделей на протязі одного року (різної інформаційної потужності, у корпусах для різного типу монтажу та ін.).

Наведемо сучасне розуміння термінів у цій галузі.

Мікропроцесор – є центральним обчислювальним ядром комп'ютерної системи.

Контролер – закінчений електронний пристрій, зазвичай виконаний на друкованій платі і призначений для прийому та опрацювання сигналів від датчиків, а також для керування зовнішніми пристроями на основі результатів опрацювання прийнятих сигналів.

Мікроконтролер – програмно керована ІМС, що застосовується для побудови контролерів. Мікроконтролер, окрім ядра має оперативний та постійний запам'ятовувальні пристрої, таймери, лічильники, канали введення та виведення інформації, інші пристрої – це самостійна комп'ютерна система, що вміщує процесор, допоміжні схеми та пристрої введення-виведення даних, розміщені у спільному корпусі.

Мікрокомп'ютер має ще й відео- та звуковий процесори. Нагадаємо, традиційні великі комп'ютери, що будувались ще з сорокових років минулого століття спочатку на лампах, а потім на ІМС середнього

ступеня інтеграції, мали так звану принстонську архітектуру (запропоновану у 1949 році співробітником Принстонського університету фон Нейманом), за якої як для програм, так і для даних використовується єдиний простір пам'яті. Це, внаслідок зменшення апаратних затрат, підвищувало надійність машин, але за рахунок зниження швидкодії.

Мікроконтролери, у яких за рахунок інтегральної технології виконання відпала проблема впливу кількості складових мікроелементів на надійність виробу, мають гарвардську архітектуру (запропоновану тоді ж таки Гарвардським університетом, але відхилену із зрозумілих міркувань), за якої для даних і команд використано окремі області пам'яті та шини. Це забезпечує високу швидкодію за рахунок можливості до- ступу за один цикл як до пам'яті, так і до даних за одночасного вико- нання побіжної команди і вибірки наступної.

Мікроконтролери мають низьке споживання (струм живлення складає кілька міліампер за рахунок побудови на К-МОН-структурах), мінімальні габарити (випускаються залежно від інформаційної потужності у 8-, 18-, 20-, 28-, 40-, 44- та 64-вивідних корпусах, та найчастіше застосовують контролери у перших чотирьох типах корпусів), потребують мінімум (буквально декілька) зовнішніх дискретних компонентів.

Пам'ять даних у них складає від 36 до 368 байт, а пам'ять програм від 0,5 до 128 кілобайтів.

Кількість ліній введення-виведення, що можуть за вибором передавати інформацію у одному з напрямків, становить від 6 до 53.

Робоча частота становить 1 – 24 МГц.

Напруга живлення може знаходитись у межах 1,8 – 6 В (номінальне значення 5 В).

Вони можуть мати у своєму складі аналогові компаратори (наприклад, два), аналого-цифрові перетворювачі (наприклад, у деяких мікроконтролерів AVR фірми Atmel від 4 до 11), таймери (1-2), широтно-імпульсні модулятори (1-8), різні інтерфейси зв'язку з зовнішніми пристроями.

У мікроконтролерів PIC (Peripheral Interface Controller – периферійний контролер інтерфейсу) фірми Microchip є можливість захисту коду програми від несанкціонованих змін або копіювання (один або два бі- ти захисту).

За технологією виконання пам'яті програм є три типи мікро-

контролерів

а) масково-програмовані – мають найменшу вартість, але застосовуються лише при масовому виробництві, бо програмуються на заводі-виробнику ІМС у процесі виготовлення кристалу;

б) з ультрафіолетовим стиранням – дозволяють перепрограмування, але процес цей досить тривалий і після ультрафіолетового опромінення можлива нечітка робота контролера, особливо за зниженої напруги живлення;

в) з флеш-пам'яттю – дозволяють багаторазове електричне стирання (дозволяють здійснювати розробку і відпрацьовування програм на готовому пристрої), але у декілька разів дорожчі від контролерів першого типу.

Та все ж вартість масово застосовуваних мікроконтролерів останнього типу сягає лише чотирьох доларів США. Саме такі мікроконтролери є найприйнятнішими для застосування при розробці пристроїв керування технологічного устаткування, виходячи з таких їхніх особливостей:

а) практично необмежена кількість циклів перепрограмування, що в умовах одиничного та малосерійного виробництва забезпечує виконання робіт з налагодження і доведення, а також оперативну зміну алгоритму керування при експлуатації устаткування;

б) достатньо високі робочі частоти;

в) наявність простих, дешевих і доступних для самостійної реалізації моделей програматорів;

г) мінімальні габарити і вартість, висока надійність;

д) доступність і вкрай прийнятна вартість;

е) наявність моделей з вбудованими аналоговими компараторами та аналого-цифровими перетворювачами (з розвитком інтегральної технології з'явилась можливість розміщення на одному кристалі як цифрових, так і аналогових елементів);

ж) можливість захисту програм від несанкціонованих змін та копіювання;

з) можливість отримання інформації від сучасних інтегральних датчиків та передачі інформації комп'ютеру у відповідних протоколах інтерфейсу;

и) велика кількість доступної довідкової та іншої технічної інформації, програмного забезпечення (для складання програм,

їхнього відлагодження, занесення у пам'ять мікроконтролера), інформації з типових застосувань та досвіду використання як у друкованому вигляді, так і у INTERNET-джерелах.

На жаль, сучасні мікроконтролери втратили початкову особливість програмованих логічних контролерів – доступність мови програмування для неспеціалістів з інформатики. Хоч процесори мікроконтролерів, на відміну від CISC-процесорів (Complex Instruction Set Computer) ЕОМ з складною системою команд, відносять до RISC (Reduced Instruct Set Computers) процесорів, у яких набір виконуваних команд скорочено до мінімуму (залежно від типу мікроконтролера його асемблер має від 33 до 133 команд), їхнє програмування на асемблері особливо складне і доступне власне спеціалістам з мікроконтролерів. Дещо простіше програмування на алгоритмічній мові С+ або на спеціальній версії Basic, але також вимагає впевненого знання комп'ютера та веде до збільшення обсягу програм до тридцяти відсотків.

Це висуває вимоги високої кваліфікації персоналу (вміння користування комп'ютером, програмування), використання при обслуговуванні й ремонті пристроїв керування на основі мікроконтролерів досить дорогого комп'ютерного устаткування. Але сучасний рівень виробництва все більше висуває саме такі вимоги як до персоналу, так і до власне устаткування. Окрім того, зрештою, зниження строків виробництва устаткування, зниження його вартості, збільшення надійності роботи, зручності експлуатації того варті.

8.10.13

Структура мікроконтролера

Отже, перші мікроконтролери з'явилися у 1976 році, коли у одній ВМС стало можливим сумістити процесор, ОЗП, ПЗП і елементи інтерфейсу. На відміну від універсальних мікро-ЕОМ, в мікро-контролерах невелика за розміром пам'ять (декілька сотень байт ОЗП та декілька десятків кілобайт ПЗП) і простий інтерфейс зв'язку з зовнішніми пристроями. Це витікає із специфіки використання мікро-контролерів. Їх не застосовують в універсальних обчислювальних системах. Мікроконтролери призначені для створення високоефективних і дешевих систем керування і регулювання. Вони реалізують відносно нескладні алгоритми, тому потребують обсягів пам'яті на декілька порядків менших, ніж ЕОМ

універсального призначення. Їх застосовують у системах керування маніпуляторами, вимірювальних приладів технологічного устаткування, станків, автомобілів, побутової техніки та ін.

Типову структуру мікроконтролера наведено на рис 8.27.

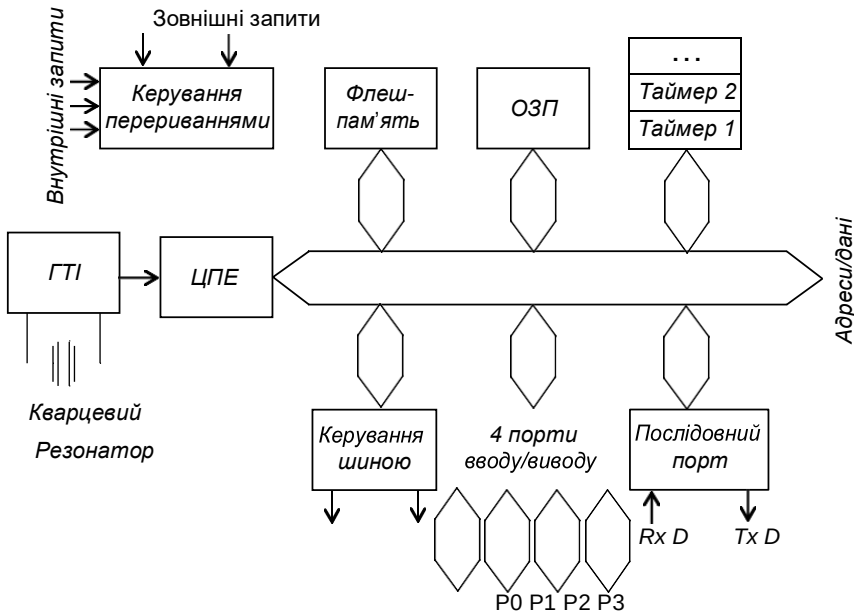


Рисунок. 8.27 – Типова структура мікроконтролера

Ємності ОЗП у декілька сотень байт цілком достатньо для тимчасового зберігання даних та проміжних результатів при виконанні програм керування. Програми зберігаються у флеш-пам'яті ємністю у десятки кілобайт, чого, як правило, цілком вистачає. За необхідності передбачено можливість нарощування обсягів пам'яті за рахунок підключення зовнішніх ВІМС оперативної та постійної пам'яті.

Зрозуміло, що роботою всіх вузлів керує **центральный процесорний елемент ЦПЕ**. Синхронізують його роботу імпульси ГТІ з частотою 12-24 МГц, що задається зовнішнім кварцевим резонатором. Це забезпечує високу стабільність частоти, що необхідно при відпрацюванні проміжків часу.

Адреси і дані передаються по **шині адреси/дані** з розрядністю слів, що опрацьовуються мікроконтролером. Введення і виведення

інформації здійснюється через **чотири паралельних і один послідовний порти введення/виведення**. Через порти P0-P3 обмін з зовнішніми пристроями відбувається по 8-розрядних шинах. Зараз також випускають 16- і 32-розрядні мікроконтролери. Кожен з портів забезпечує взаємодію з одним пристроєм. Функції ліній шин в різних режимах можуть мати різне значення (введення або виведення).

Через вхід Rx D послідовного порту відбувається прийом даних у послідовному коді, через вхід Tx D – передача.

Мікроконтролер може обслуговувати декілька як внутрішніх так і зовнішніх **запитів на переривання** програми для переходу до виконання більш приоритетних підпрограм.

Таймери-лічильники призначені для відпрацьовування інтервалів часу або підрахунку кількості імпульсів.

Систему команд мікроконтролера орієнтовано на виконання задач керування, тому поряд зі звичайними командами, характерними для всіх мікропроцесорів, є й специфічні.

Оскільки всі вузли мікропроцесорної системи тут знаходяться всередині однієї ВІМС, то непотрібні звертання до зовнішніх пристроїв. Тому більшість команд короткі з часом виконання, наприклад, біля 1 мкс за тактової частоти 12 МГц.

8.10.14 Зіставлення основних етапів створення пристроїв керування на основі цифрових автоматів з жорсткою логікою і програмованих

У табл. 8.5 наведено зіставлення основних етапів створення пристроїв керування з жорсткою логікою і програмованих. З неї видно, що при застосуванні типових програмованих пристроїв керування (таких, що можна придбати готовими), у циклі створення готового пристрою відсутні процедури, пов'язані з розробкою конструкторської і технологічної документації та власне виробництвом. Усі процедури фактично пов'язано виключно з розробкою та відлагодженням програм.

Насамкінець зазначимо, що специфіка роботи з МПП відносить їх скоріш до галузі інформатики, бо у більш-менш серйозних застосуваннях вартість розробки програмного забезпечення складає до 80 % вартості всього пристрою керування. Виходячи з цього ми тут обмежи-

лись лише наданням загальних понять, особливо у тому, що відносить- ся до питань програмування МПП.

Таблиця 8.5 - Зіставлення основних етапів створення пристроїв керування

Пристрій керування з жорсткою логікою	Програмований пристрій керування
1. Розробка алгоритму керування	
2. Розробка електричної принципової схеми пристрою керування на основі обраного типу ІМС	2. Розробка програми роботи пристрою керування на основі обраного типу МПП (наприклад, ПЛК)
3. Створення макету пристрою керування	3. Занесення програми в ОЗУ МПП
4. Відладка макету пристрою керування і відповідне коригування його електричної принципової схеми	4. Відладка і відповідне коригування програми
5. Розробка конструкторської і технологічної документації для виробництва пристрою керування	5. Придбання типового МПП (наприклад, ПЛК)
6. Виробництво пристрою керування	6. Занесення програми в ІМС постійної пам'яті і встановлення їх в ПЗУ МПП
7. Поставка пристрою керування замовнику	