

ОСНОВИ ІНФОРМАТИКИ.

СКЛАДОВІ ЧАСТИНИ МІКРОПРОЦЕСОРНИХ СИСТЕМ

- Як передавати, зберігати та обробляти інформацію?
- Що таке «біт», «байт»?
- З яких «цеглинок» складаються комп'ютери?
- Що таке постійна та оперативна пам'ять? Як функціонують «ROM» та «RAM»?
- Що таке «тригер» та «регістр»?
- Що таке шина? Які бувають шини?
- Як працює арифметико-логічний пристрій? Які бувають логічні операції?
- Що таке мікропроцесор?
- Як виглядає класична архітектура мікропроцесорної системи?

Якщо ви цікавитесь новітніми технологіями та мрієте отримати [сучасну інженерну спеціальність](#), вам неодмінно доведеться мати справу з комп'ютерами (мікропроцесорними системами). Інженерна діяльність будь-якого напрямку неможлива без застосування інформаційних технологій. Тож вам доведеться використовувати, а можливо програмувати й навіть розробляти мікропроцесорні системи. Якщо ви не змогли відповісти на запитання, що поставлені вище, ця стаття доступною мовою з використанням анімованих ілюстрацій дозволить вам зменшити прогалини у своїх знаннях.

Будь-який *комп'ютер* – це машина для обробки інформації, не зважаючи на те, яку конкретну задачу він виконує. Ви можете грати у комп'ютерну гру, переглядати 3D-фільм, створювати музичну композицію або анімаційне кіно (мультфільм), можете програмувати, писати реферат або спілкуватися з другом; комп'ютер при цьому робить лише одне – обробляє *інформацію*.

Щоб зробити комп'ютер треба навчитися *передавати, зберігати та обробляти інформацію*.

Найменший об'єм інформації, який можна подати – це *біт* (скорочення від “**b**inary **d**igit” – двійкова цифра). Біт дає можливість у даний конкретний момент часу втілити один з двох можливих станів: “0” або “1”.

Для передачі одного біта інформації достатньо скласти таку просту схему (*рис. 1*).

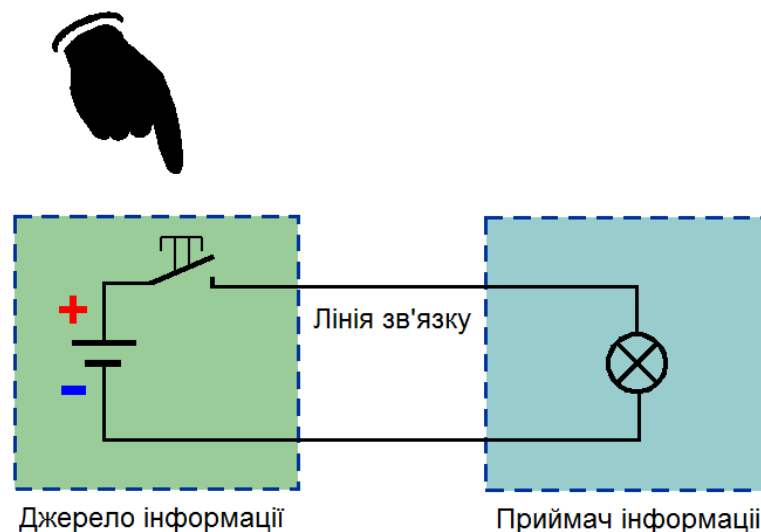


Рис. 1. Схема для передачі одного біта інформації ([анімація](#))

Якщо натиснути кнопку, замкнеться електричне коло, лампочкою потече струм і вона запалиться (буде світитись). Тим самим ми передаємо один з двох можливих станів. Назвемо цей стан «активним» або одиничним (інші назви цього стану: «1», «Істина», «True», «On»).

Якщо кнопка НЕ натиснута, коло розімкнене, напруга НЕ дістає лампочки, електричний струм НЕ протікає, лампа НЕ горить. Тим самим ми передаємо другий, «неактивний» стан (інші назви цього стану: «0», «Хибність», «False», «Off»).

Схема (рис. 1) складається з джерела інформації, приймача інформації та лінії зв'язку. Інформація передається лінією зв'язку за допомогою електричного сигналу. Але електричне поле розповсюджується у просторі зі швидкістю світла, а реальна швидкість передачі електричних сигналів в технічних системах незрівнянно менша. Таким чином можна зробити два висновки:

1. Для передачі інформації у просторі з точки А до точки Б необхідний деякий час, тобто інформацію неможливо передавати миттєво.
2. За певний проміжок часу можна передати лише обмежений об'єм інформації.

Ці два природні обмеження є нездоланими для класичної обчислювальної техніки. (Слід відзначити, що разом з іншими недоліками сучасних комп'ютерів та систем, ці обмеження спонукають науковців та інженерів розробляти принципово нові інтелектуальні системи, які у дечому «копіюють» нервові системи та мозок біологічних «прототипів» – тварин та людини. Але це – тема іншої статті).

Джерело інформації обов'язково має у своєму складі джерело енергії (у нашій схемі – це умовно гальванічний елемент, або «батарейка»). Саме джерело енергії надає сигналу ту енергію, яка передається у просторі (на жаль із втратами) та має впливати на приймач інформації. Ось чому будь-який комп'ютер (і не тільки комп'ютер, а й мозок біологічної істоти) потребує живлення та виділяє тепло, «гріється».

До речі, інформацію можна передавати також за допомогою пневматичних або гідравлічних сигналів («1» – є тиск, «0» – немає тиску), за допомогою механіки («1» – піднятий важіль, «0» – опущений важіль), за допомогою оптики («1» – є світло, «0» – немає світла) та іншими способами (наприклад, гідропідсилювач автомобіля – це система для передачі інформації в одному напрямку: від керма до коліс авто). Проте, з суто технічних причин реалізація електронних пристроїв для передачі та обробки інформації є найбільш вдалою.

Але повернемося до електричних схем. Схема, що наведена на *рис. 2*, дозволяє у даний конкретний момент часу передати 8 бітів інформації (а це вже 1 *байт*).

Якщо натиснути усі кнопки одразу, загоряться усі 8 ламп, тим самим ми передамо 8 логічних одиниць. Якщо ж не натискати жодної кнопки – то ми передаємо 8 логічних нулів. Це так звана 8-розрядна схема. Можна натискати/відпускати кнопки у будь-якій довільній комбінації. Якщо розглядати ці 8 розрядів не окремо, а у сукупності, то ми отримаємо так зване 8-розрядне *двійкове число* (двійкове, тому що кожен розряд може приймати один з двох станів: «0» або «1»).

Усього можливо передати 256 комбінацій нулів та одиниць.

Якщо кожній комбінації співставити знак (букву або цифру або ж музичну ноту) чи колір, то за допомогою схеми (*рис. 2*) можна передавати тексти, музику, зображення (візуальні картини) і в загалі – інформацію будь-якого роду ([анімація](#)).

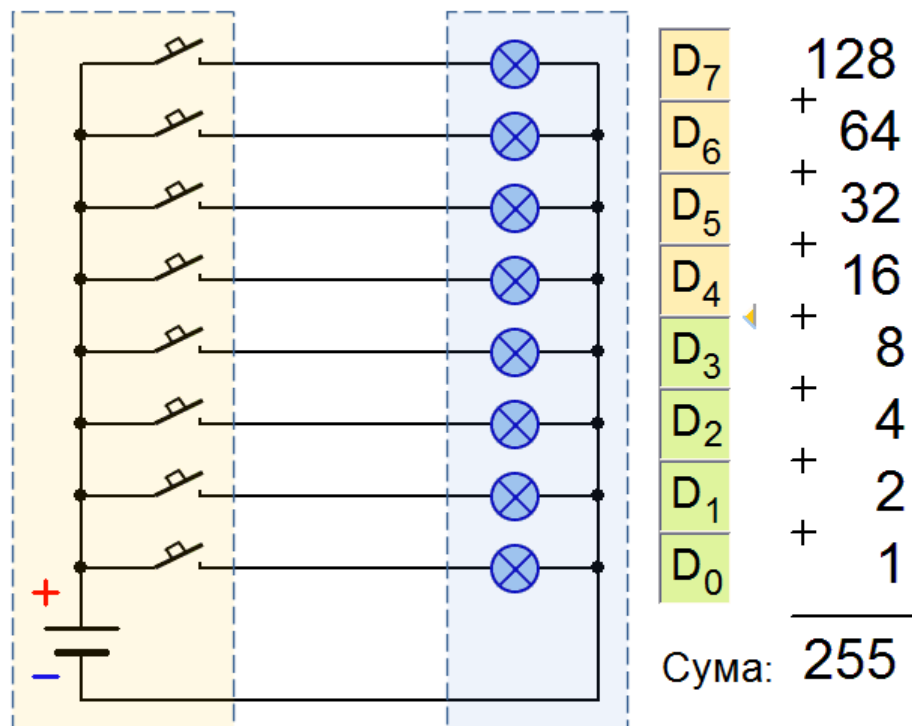


Рис. 2. Схема для передачі 8-ми бітів (1-го байта) інформації ([анімація](#))

Вісім проводів між кнопками та лампами складають так звану «*шину даних*», якою передається «корисна» інформація, тобто саме та інформація, заради якої і розробляли всю цю систему. Існують також поняття «*шина адреси*» та «*шина керування*». Це допоміжні шини у мікропроцесорних та комп'ютерних системах. *Шиною адреси* передається (також у двійковому форматі) адреса того пристрою системи, для якої призначена інформація. *Шиною керування* передаються команди, які вказують, що робити з переданою інформацією.

Шина даних – це класичний поштовий лист, який ми пишемо другу та вкладаємо до конверта (у листі ми подаємо «корисну» інформацію, те, що ми хочемо передати адресату). Шина адреси – це конверт, на якому вказуємо адресу одержувача. А шина керування – це марка, без якої лист не дійде до адресата. Не можна сказати, що адреса на конверті та цінник на марці – це «марна» або «непотрібна» інформація. Але у поштовому відправленні головне – лист, а не конверт.

Якщо усі провідники шини даних, шини адреси та шини керування зв'язати до одного джгута, то отримаємо «*системну шину*».

І ще одна технічна деталь. На *рис. 2* крім 8-ми проводів між кнопками та лампами є ще й дев'ятий провід, який необхідний тільки для того, щоб замкнути контур (електричне коло). Цей провід загальний для усієї схеми, тому його так і називають: «*загальний провід*».

Перша схема (*рис. 1*) дає можливість передати «чорно-білу» картину світу (тільки дві комбінації). Друга схема (*рис. 2*) дає «кольорову» картину (256 комбінацій сигналів). Перейшовши від першої схеми до другої, ми збільшили кількість апаратних витрат (кнопок, ламп, провідників, потужність джерела живлення) у вісім разів, а кількість «колірів» (тобто комбінацій сигналів) збільшилась у 128 разів. А якщо ж взяти 16-розрядну схему, то кількість можливих комбінацій буде становити 65535, тобто збільшиться ще у 256 разів. Тобто кількість комбінацій зростає швидше ніж витрати апаратних ресурсів комп'ютерних систем. Ось чому розрядність мікропроцесорів та комп'ютерів постійно нарощується (див. статтю «[Історія створення та розвитку мікропроцесорних засобів автоматизації](#)»).

Отже, ми навчилися передавати інформацію. Але ще необхідно зберігати та обробляти її. Для збереження інформації використовують *запам'ятовуючі пристрої*.

Найпростіший електронний запам'ятовуючий пристрій – це конденсатор (*рис. 3*). Якщо приєднати конденсатор до джерела живлення постійного струму (наприклад, до «батарейки»), він зарядиться. Коли від'єднати конденсатор від джерела, він залишиться зарядженим. Тим самим ми «записали» до цього запам'ятовуючого пристрою логічну одиницю («1» або «Істина»). Далі ми можемо за допомогою вольтметра або гальванометра упевнитися в тому, що конденсатор заряджений. Тим самим ми «зчитуємо» логічну одиницю, яку було записано раніше, із пристрою пам'яті.

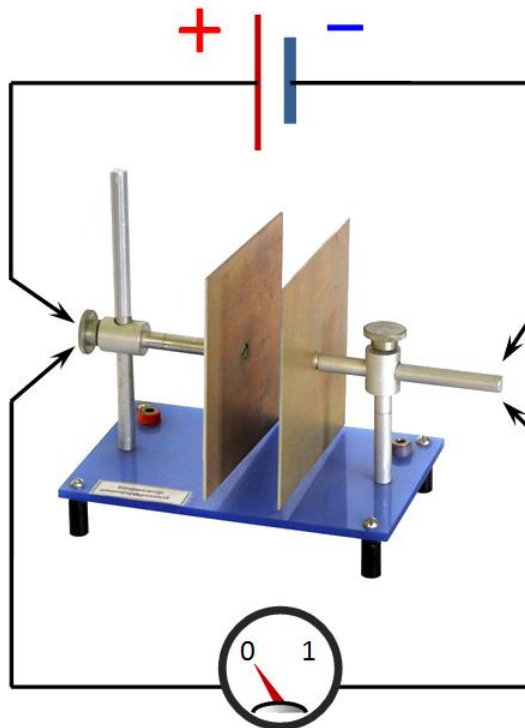


Рис. 3. Оперативна пам'ять ємністю 1 біт на основі конденсатора ([анімація](#))

Якщо звичайним металевим проводом з'єднати (закоротити) дві обкладинки конденсатора, то він розрядиться. Тим самим ми записали до пам'яті логічний нуль («0» або «Хибність»). У будь-який наступний момент часу за допомогою вольтметра або гальванометра можна упевнитись, що конденсатор розряджений, тобто можна зчитати із пам'яті логічний нуль, який туди було записано раніше.

Причому «записувати» інформацію до конденсатора (заряджати або розряджати його) може одна людина, а зчитувати (вимірювати напругу на конденсаторі) – зовсім інша.

Тобто конденсатор – це пристрій пам'яті ємністю в один біт (дає можливість зберігати один з двох станів: «0» або «1»).

Слід зазначити, що заряджений конденсатор з часом розряджається, навіть, якщо його не чіпати. Тобто з часом «1», яка була записана раніше, буде втрачена. Тому інформацію в конденсаторі слід регулярно «*регенерувати*», відновлювати за таким алгоритмом:

1. Виміряти напругу на конденсаторі.
2. Якщо напруга є (записана «1»), то дозарядити конденсатор.
3. Якщо напруги немає (записаний «0»), то нічого не робити.

Пристрої пам'яті, які виконуються на конденсаторах (у тому числі у мікроелектроніці) називаються «Оперативними Запам'ятовувючими Пристроями» або «ОЗП» (російською мовою – «Оперативное Запоминающее Устройство», «ОЗУ»).

Як говорять спеціалісти – це *пам'ять з довільним доступом* (з «доступом до запису» та «доступом для читання»). Іншими словами до пам'яті такого типу можна необмежену кількість разів записувати нову інформацію і зчитувати її. Тому англomовні спеціалісти називають такий тип пам'яті «**Random Access Memory**» або «**RAM**».

Отже, знаючи фізичну реалізацію пристроїв оперативної пам'яті, можна зробити висновки щодо їх властивостей:

1. Ця пам'ять називається «ОЗП» (російською «ОЗУ») або «RAM».
2. Це пам'ять з довільним доступом для запису й для читання (необмежена кількість циклів запису).
3. Це швидка («оперативна» – значить швидка) пам'ять (конденсатори маленької ємності можна дуже швидко розряджати та заряджати).
4. Вона потребує постійного живлення для регенерації інформації (тому що конденсатори з часом саморозряджаються). Тобто це – енергозалежна пам'ять (при збої живлення записана інформація втрачається).

Ще один можливий спосіб збереження інформації – за допомогою елементів, які можна руйнувати, наприклад, перемичок.

Уявіть собі пристрій (*рис. 4*), який складається з дерев'яної дощечки (ізоляційної основи), в яку встромлені пари гвіздків, що з'єднані між собою тонкими перемичками. Будемо вважати так: якщо перемичка ціла, то у цей розряд записана логічна одиниця, якщо ж перемичку зруйнувати (наприклад, спалити, приклавши високу напругу), то у цей розряд записаний логічний нуль. Як визначити стан перемички (ціла чи зруйнована)? За допомогою тестера, тобто, пропустивши нею електричний струм (протікає чи ні).

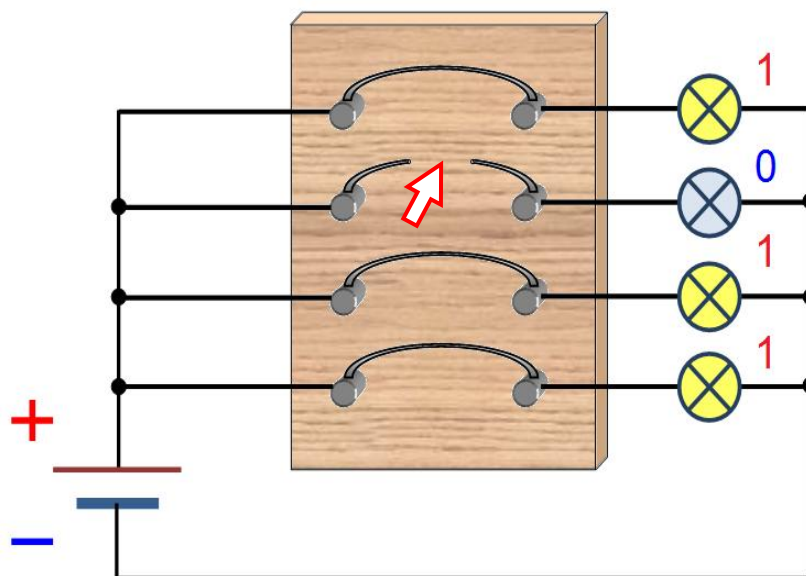


Рис. 4. Постійна пам'ять ємністю 4 біти на основі перемичок

Одна пара гвіздків з однією перемичкою дає можливість зберігати один біт інформації, вісім перемичок – один байт.

Пристрої пам'яті, в яких реалізовано викладений спосіб збереження даних, називаються «Постійними Запам'ятовувачими Пристроями» або «ПЗП» (російською мовою – «Постоянное Запоминающее Устройство», «ПЗУ»).

Зрозуміло, що перемичку, яка вже зруйнована, неможливо відновити (принаймні, без механічного втручання). Тому інформацію до пристрою пам'яті такого типу можна записати лише один раз. Ось чому англомовні спеціалісти називають пам'ять такого типу «Read Only Memory», «ROM», що у перекладі означає «пам'ять тільки для читання». Записати інформацію до неї можна лише один раз, а зчитувати – скільки завгодно. Ще одна можлива назва «One Time Programmable Memory» або «ОТР» – одноразово програмована пам'ять.

До того ж, для запису інформації до постійної пам'яті у загальному випадку необхідний спеціальний пристрій (*програматор*), який короткими імпульсами струму підвищеної напруги «перепалює» ті перемички, до яких слід записати «0».

І ще одна технічна деталь. Якщо на заводі-виробнику заздалегідь відомо, які перемички слід залишити, а які зруйнувати (тобто заздалегідь відома інформація, яку необхідно записати до пристрою пам'яті), то це можна зробити на етапі виготовлення пристрою. У тих розрядах, де треба мати логічну одиницю, зробити перемички, а у тих розрядах, де необхідно мати логічний нуль – просто не ставити їх. Такий тип пристрою називають «масковий ПЗП» за назвою технології, коли вміст пам'яті задається спеціальною маскою на етапі виготовлення.

Знаючи фізичну реалізацію пристроїв постійної пам'яті, можна визначити їх класичні властивості:

1. Ця пам'ять називається «ПЗП» (російською «ПЗУ») або «ROM».
2. Це пам'ять з доступом для тільки для читання (записати можна один раз, зчитувати – скільки завгодно). Для запису даних у загальному випадку необхідно мати спеціальний програматор.
3. Це швидка пам'ять (хоча її НЕ називають «оперативною»).
4. Це – енергоНЕзалежна пам'ять. (ці пристрої НЕ потребують живлення для зберігання даних, а лише для сервісних функцій під час зчитування даних). При збої живлення інформація НЕ втрачається.

У мікропроцесорних системах пристрої постійної пам'яті (ПЗП, ROM) найчастіше використовують для збереження програм, табличних даних (наприклад, кодових таблиць знакогенераторів для дисплея), констант (наприклад, коефіцієнтів для регуляторів), тощо. А оперативна пам'ять (ОЗП, RAM) зазвичай використовується для збереження поточних технологічних даних, даних користувача, а також програм, які дозавантажують з інших пристроїв (наприклад, з дисків).

Звичайно, це спрощена картина. Типів пам'яті й особливостей їх побудови та використання значно більше.

Зокрема, один з найважливіших мікроелектронних пристроїв для збереження інформації – це *регістр*. Регістр (*рис. 5*) будується на *тригерах* – транзисторних елементах, здатних запам'ятовувати та зберігати один біт даних. Якщо використати 8 тригерів одночасно, отримаємо 8-розрядний *паралельний регістр*, за допомогою якого можна зберігати 1 байт (8 бітів) даних. З точки зору користувача регістр – це елемент пам'яті, схожий на елемент ОЗУ (оперативної пам'яті). Проте, якщо кожний елемент пам'яті комп'ютера має свою адресу, яка задається числом, то кожний регістр у загальному випадку має свою особисту назву (наприклад, «А», «В» або «PSW», тощо).

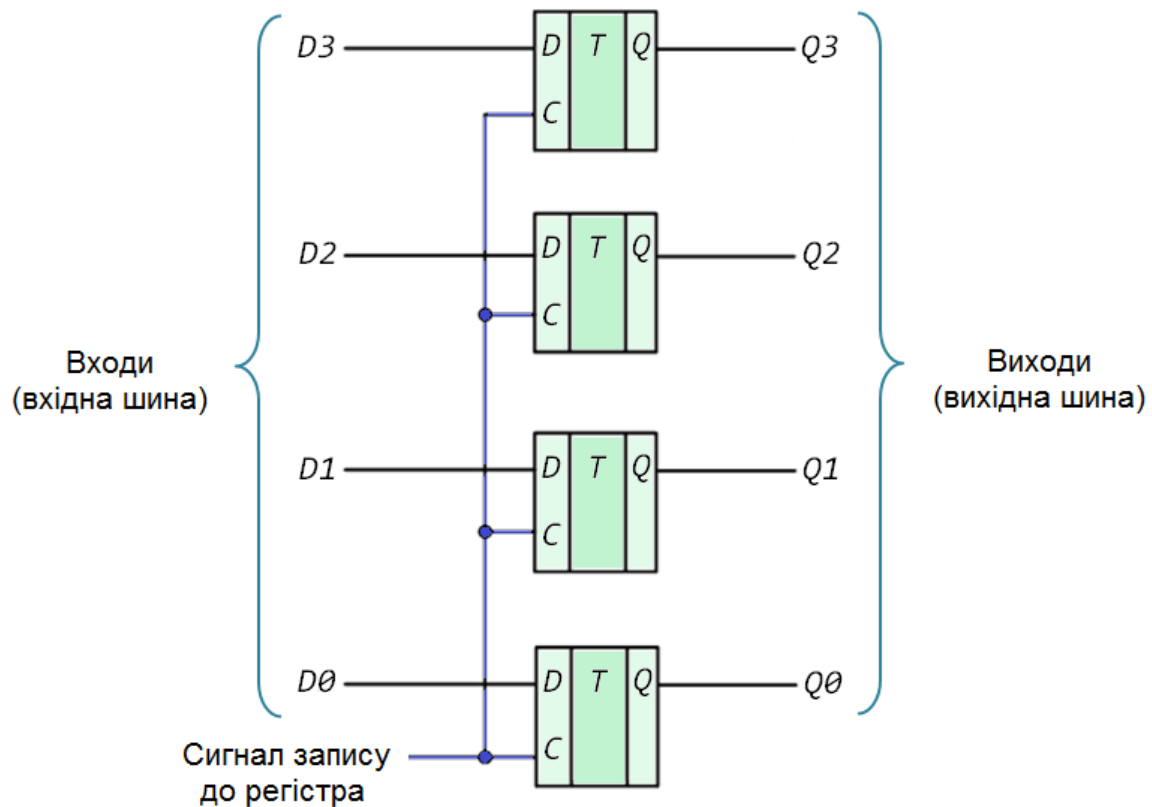


Рис. 5. Структурна схема регістра ємністю 4 біти

Отже, тепер ми вміємо передавати та зберігати інформацію. А як її обробляти?

Як ми вже знаємо, будь-яку інформацію можна подавати у вигляді масивів чисел (логічних нулів та одиниць). Тож обробка інформації полягає у виконанні арифметичних та логічних операцій над цими числами.

Основні *арифметичні операції* відомі нам зі шкільної лави:

1. Додавання;
2. Віднімання (операція, зворотна до додавання);
3. Множення (багаторазове додавання);
4. Ділення (операція, зворотна до множення).

Усі складні методи обробки інформації розкладають та зводять до цих чотирьох операцій. Наприклад, обчислення тригонометричних функцій можна за допомогою математичних рядів звести до послідовного виконання базових арифметичних функцій.

Основні *логічні операції*:

1. Логічне додавання («диз'юнкція» або «OR»);
2. Логічне множення («кон'юнкція» або «AND»);
3. Логічне заперечення («інверсія» або «NOT»).

На *рис. 6* наведені електричні схеми, що реалізують основні логічні операції. Таблиці істинності ілюструють стан вихідних сигналів (на лампах) за різних комбінацій сигналів на входах (кнопках). Якщо ви бажаєте краще зрозуміти, у чому полягає суть логічних операцій, скористуйтеся демонстраційною програмою «[Логічні операції](#)».

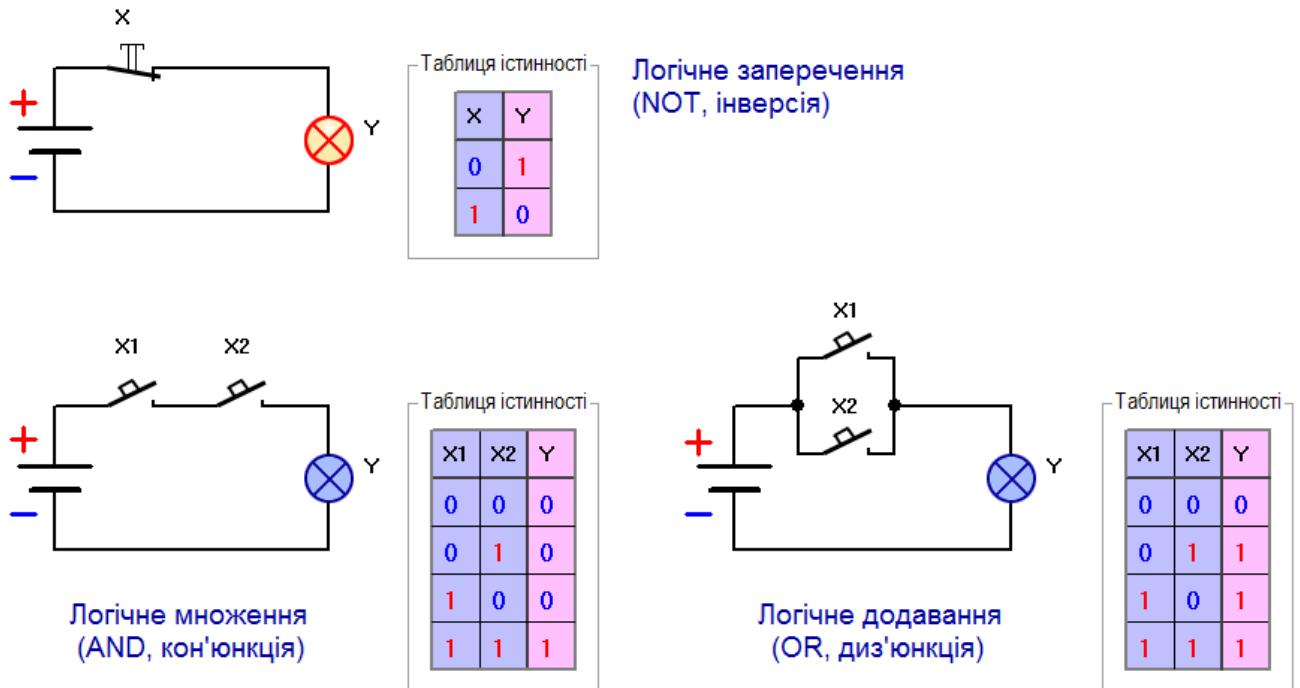


Рис. 6. Основні логічні операції

Для виконання арифметичних та логічних операцій над числами, що подані у двійковому форматі (логічними нулями та одиницями), в електроніці використовують так звані «Арифметико-Логічні Пристрої» або «АЛП» (російською мовою – «Арифметико-Логическое Устройство», «АЛУ»). В англomовному світі цей пристрій називається «Arithmetic Logic Unit» або «ALU». Слід зазначити, що арифметико-логічний пристрій є основою будь-якого мікропроцесора. Структурна схема АЛП наведена на рис. 7. Два числа «X1» та «X2» (операнди) подаються до обчислювальних входів, а на виході маємо результат «Y» арифметичної дії або логічної операції.

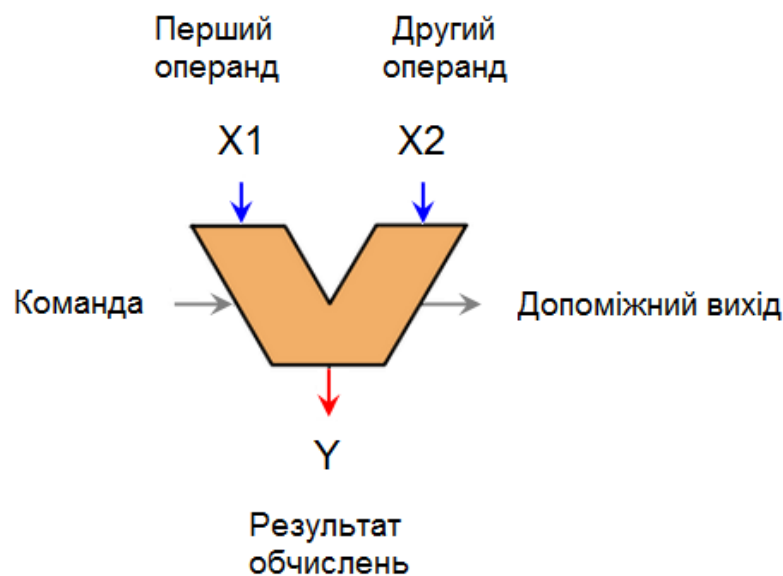


Рис. 7. Арифметико-логічний пристрій ([анімація](#))

Для того, щоб забезпечити функціонування арифметико-логічного пристрою, необхідно додати регістри, в яких будуть зберігатися операнди «X1», «X2» та результат обчислень «Y».

Також необхідно додати допоміжні схеми керування та синхронізації. В результаті отримаємо мікропроцесор.

Мікропроцесор – пристрій для обробки інформації, який виконує арифметичні та логічні операції і керує процесами обміну інформацією згідно з програмою, що записана у вигляді машинного коду.

На основі мікропроцесора можна побудувати класичну мікропроцесорну систему (комп'ютер), див. *рис. 8*.

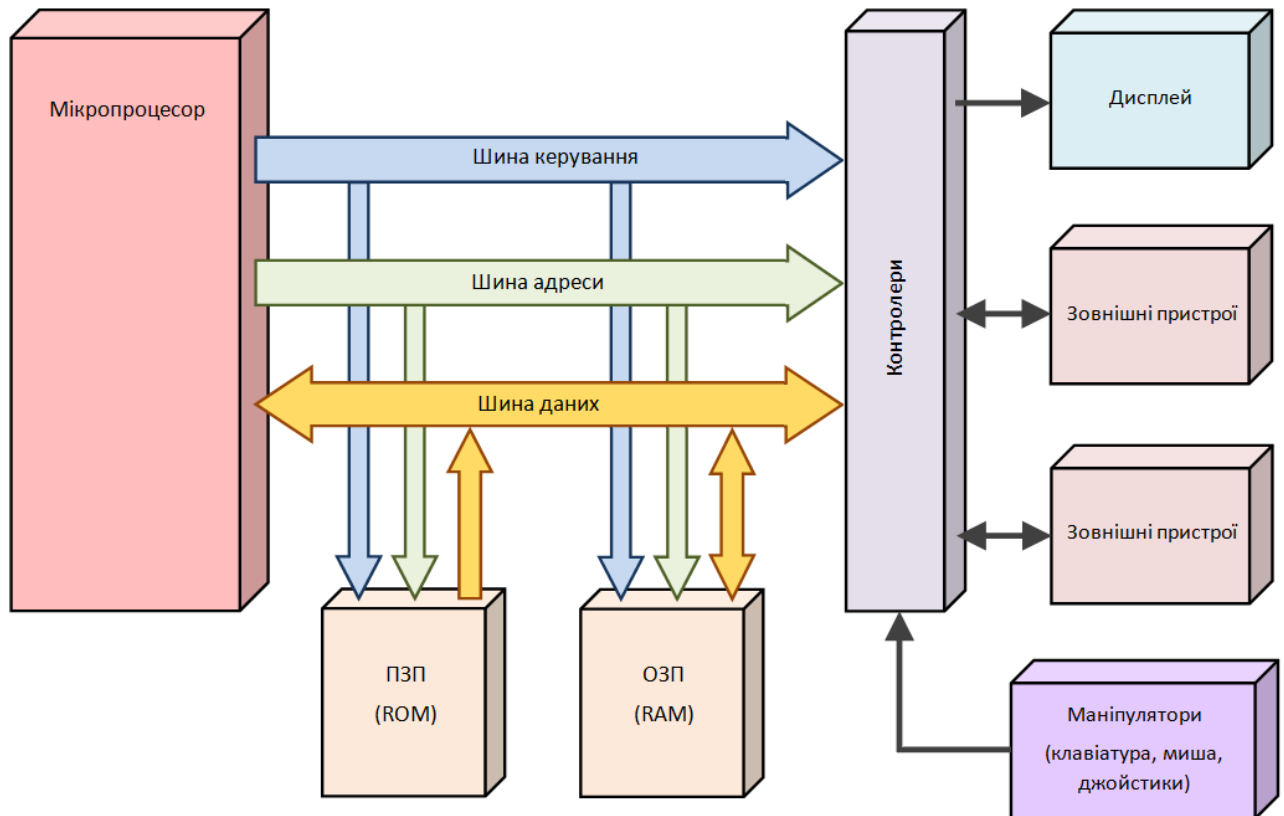


Рис. 8. Трьохшинна архітектура мікропроцесорної системи (комп'ютера)

Докладно вивчити роботу мікропроцесорних систем, навчитися розробляти та програмувати прикладні міні-комп'ютери для задач автоматизації можна на [кафедрі електропривода](#) Національного гірничого університету (м. Дніпропетровськ).

Текст, ілюстрації: Яланський О. А.