

Лекція 13. СТРУКТУРНА СХЕМА МІКРОПРОЦЕСОРА

- 13.1. Основні складові мікропроцесора.
- 13.2. Арифметично-логічний пристрій (АЛП).
- 13.3. Акумулятор.
- 13.4. Буферні регістри .
- 13.5. Регістр стану .
- 13.6. Регістри загального призначення.
- 13.6.1. Регістри загального призначення В, С, Д.
- 13.6.2. Лічильник команд.
- 13.6.3. Регістр адресу пам'яті.
- 13.7. Пристрій керування мікропроцесора.
- 13.7.1. Регістр команд .
- 13.7.2. Схема керування.
- 13.8. Процес виконання команди в МП.

13.1. Основні складові мікропроцесора

У кожному конкретному випадку наявність такої схеми полегшує розуміння архітектури МП обчислювальної системи. Згідно з рис.13.1. МП складається з трьох основних блоків: АЛП, декількох регістрів і схеми керування. Для передавання даних між цими блоками МП використовується внутрішня шина даних.

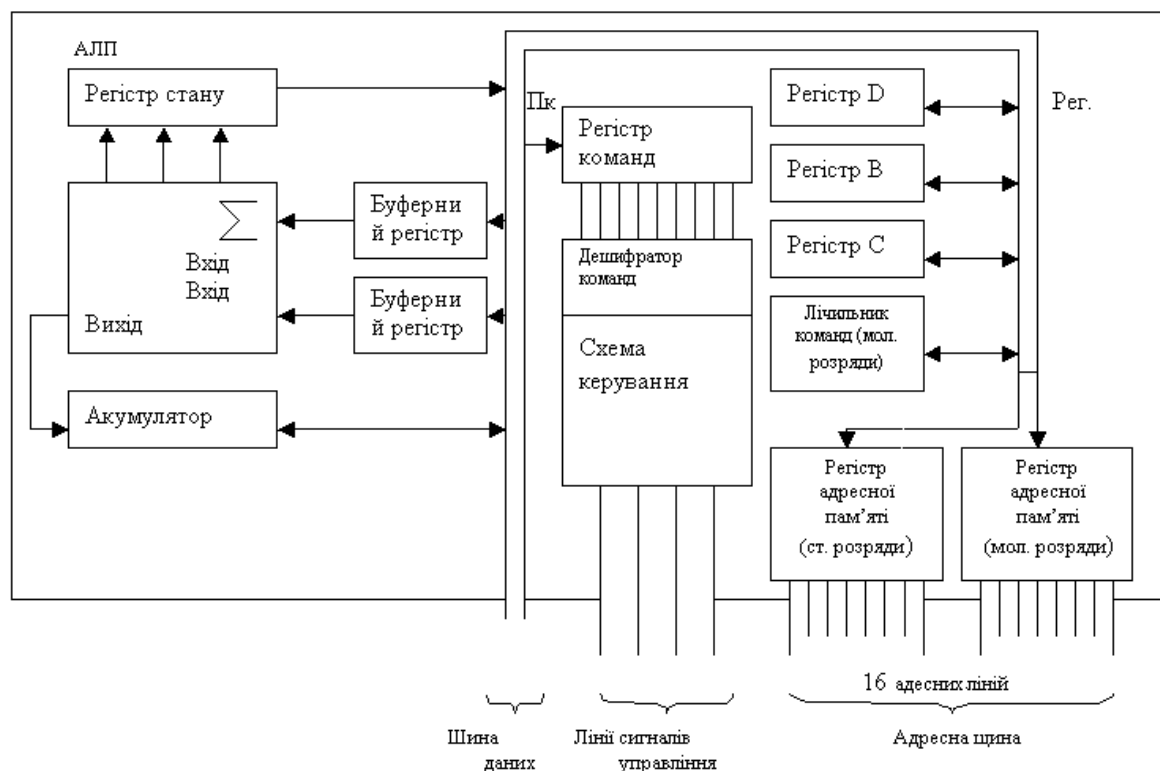


Рис. 13.1. Структурна схема 8-розрядного мікропроцесора

АЛП – арифметично-логічний пристрій, що має в своїй основі Σ – суматор, який виконує першу функцію – обробку даних.

Регістри є важливою складовою частиною МП. Вони беруть участь в реалізації основних логічних функцій МП незалежно від кількості регістрів. На початку розглянемо 6 регістрів, інші розглянемо пізніше.

Кожен регістр МП-ом може використовуватись для тимчасового зберігання одного слова даних. Деякі регістри мають спеціальне призначення, другі – багатоцільові – регістри загального призначення, можуть використовуватись програмістом як завгодно. Кількість і призначення регістрів в МП залежить від архітектури МП. Однак всі МП мають основні регістри: 1) стану; 2) буферні; 3) команд; 4) адреси пам'яті; 5) лічильник команд; 6) акумулятор. Інші регістри призначені для полегшення роботи програміста.

Тільки розуміння впливу кожного основного регістру на потік і переміщення даних в МП дозволяє отримувати правильну уяву про функціонування МП.

Пристрій керування – ПК – служить для підтримання необхідної послідовності роботи всіх ланок МП і виконує другу функцію – організоване маніпулювання даними.

Розглянемо ці складові частини структурної схеми мікропроцесора детальніше.

13.2. Арифметично-логічний пристрій (АЛП)

АЛП виконує одну із головних функцій МП – обробку даних. АЛП складається з суматора Σ , що має два вхідних порти – “вхід” і один вихідний – “вихід” та інших регістрів. Призначення вхідного порта – ввід слова даних в АЛП, а вихідного порта – вивід такого слова. Обидва вхідних порти мають буфери, роль яких виконують регістри тимчасового зберігання даних (буферні регістри). Кожен порт з'єднано зі своїм буферним регістром, який здатний зберігати для АЛП одне слово даних. Два вхідних порти дозволяють АЛП приймати дані або з внутрішньої шини даних МП, або з спеціального регістра, якого називають акумулятором. Єдиний вихідний порт АЛП дає можливість АЛП пересилати слово даних в акумулятор. Акумулятор призначений для зберігання слова даних, посланого в нього з вихідного порта АЛП або взятого з пам'яті. Коли, наприклад, АЛП сумує два слова даних, одне з них знаходиться в акумуляторі. Після виконання додавання результат - слово даних посилается в акумулятор на зберігання.

АЛП оперує одним або двома словами залежно від виду виконуваної операції; відповідно він використовує і вхідні порти. Наприклад, оскільки для додавання потрібно два слова даних, то цю операцію АЛП виконує, користуючись обома вхідними портами. А при інвертуванні слова даних АЛП обмежується одним вхідним портом. Ця операція веде до зміни в слові даних всіх двійкових нулів на двійкові одиниці, а всіх двійкових одиниць на двійкові нулі. Оскільки операція виконується над одним словом, достатньо під'єднати до акумулятора один вхідний порт.

Кількість функцій АЛП залежить від типу МП і різна для машин різних типів. Деякі АЛП використовують велику кількість операцій, а деякі не велику. Функції АЛП визначають архітектуру МП в цілому. Типовими операціями, які виконує АЛП є: додавання, віднімання, І, АБО, виключно АБО, Інверсія, Зсув вправо, Зсув вліво, Приріст додатній, Приріст від'ємний.

13.3. Акумулятор

Акумулятор - головний регістр МП при різних маніпуляціях з даними. Більшість арифметичних і логічних операцій здійснюється з допомогою АЛП і акумулятора. Будь-яка з таких операцій над двома словами (операндами) веде до розміщення одного із них в акумуляторі, а другого в пам'яті або ще в якому-небудь регістрі. При з'єднанні двох слів А і В розміщених в акумуляторі і пам'яті відповідно, результуюча сума С завантажується в акумулятор, при цьому попередня величина стирається.

Операцією другого типу, яку використовує акумулятор, є програмована передача даних із однієї частини МП в другу - пересилання даних між портом вводу-виводу і областю пам'яті, між областями пам'яті. Це здійснюється в два етапи: спочатку виконується пересилка даних із джерела в акумулятор, а потім із акумулятора - в пункт призначення. Було показано, що МП дозволяє використати АЛП для об'єднання даних в акумуляторі з іншими даними. Однак МП може виконувати деякі дії над величинами безпосередньо в акумуляторі.

Наприклад, акумулятор може бути очищений шляхом запису двійкових нулів у всі його розряди, встановлений в одиничний стан через запис двійкових одиниць у всі розряди. Вміст акумулятора можна зсувати вліво або вправо, отримувати його інвертоване значення.

Акумулятор є найбільш універсальним регістром МП і для виконання операцій над даними, їх перш за все необхідно розмістити в акумулятор. Із рис. 11.1. бачимо, що дані в акумулятор надходять із внутрішньої шини даних МП. А в свою чергу акумулятор може посилати дані на цю шину. Кількість розрядів акумулятора відповідає довжині слова МП, тобто 8 біт. Однак деякі МП мають акумулятори *подвійної довжини*. Такий акумулятор можна розглядати як один пристрій, або як два окремих акумулятори. В першому випадку другий акумулятор можна розглядати як один пристрій, або як два окремих акумулятори. В першому випадку другий акумулятор пари використовується для запису додаткових бітів, які з'являються при виконанні деяких арифметичних операцій.

Наприклад, при множенні двох 8-бітових слів, - результат 16-бітове слово, - розміщується в акумуляторі подвійної довжини. У деяких МП є група акумуляторів. Якщо їх, наприклад, два - акумулятори А і В, то МП повинен мати дві різні команди для завантаження в них даних з виходу АЛП: однієї команди - для запису даних в акумулятор А, другої - для запису даних в акумулятор В. Крім того, повинні бути дві відповідні команди очищення цих акумуляторів. Перевага "багатоакумуляторних" МП в порівнянні з "одноакумуляторними" в тому, що перші представляють можливість виконання операцій з передаванням даних від акумулятора до акумулятора.

Дані будуть тимчасово зберігатися в одному акумуляторі, поки другий використовується для виконання інших дій. Коли знову виникає необхідність в даних, які містяться в першому акумуляторі, пересилати їх не потрібно, оскільки вони вже є там.

В деяких випадках наявність двох акумуляторів може скоротити число необхідних операцій.

13.4. Буферні регістри

Буферний регістр (БР), що з'єднаний з акумулятором, називається *буфером акумулятора*, другий буферний регістр призначений для тимчасового зберігання даних, що надходять з внутрішньої шини

МП. АЛП одержує дані з внутрішньої шини, модифікує їх, а потім розміщує оброблені дані в акумулятор, що неможливо без цих реєстрів, що тимчасово зберігають дані.

На вхід буфера акумулятора дані можуть надходити із виходу акумулятора, що забезпечує таку послідовність виконання арифметичних і логічних операцій, при якій один з операндів поступає з акумулятора, проводиться операція, результат операції буде розміщуватись в акумуляторі, що дозволяє виключити ситуацію, при якій вхід і вихід суматора одночасно під'єднаний до одної і тої частини схеми МП.

13.5. Регістр стану

Саме наявністю реєстр стану (РС) ЕОМ відрізняється від калькулятора. РС призначений для збереження результатів деяких перевірок, які здійснюються в процесі виконання програми. Розряди РС приймають те чи інше значення при виконанні операцій, використовуючи АЛП і деякі реєстри. Запам'ятовування результатів перевірок дозволяє використовувати програми, які містять переходи. При наявності в програмі переходу, виконання команд починається із деякої нової області пам'яті, тобто лічильник команд завантажується новим числом. У випадку умовного переходу така дія має місце, якщо результати певних перевірок співпадають із очікуваними значеннями. Вказані результати знаходяться в РС.

РС надає програмісту можливість організувати роботу МП так, щоб при певних умовах змінювався порядок виконання команд. МП приймає рішення про те чи інше продовження ходу обчислень залежно від вказаних умов.

В АЛП можуть знаходитись і інші допоміжні реєстри, пристрої двійко-десятькової корекції тощо, або в АЛП можуть знаходитись декілька акумуляторів, в залежності від функцій і конфігурації конкретного мікропроцесора, про що треба дивитись в спеціальній літературі.

13.6. Регістри загального призначення

13.6.1. Регістри загального призначення В, С, Д

Ці реєстри можуть бути використані в якості запам'ятовуючих пристроїв або акумуляторами, тобто вони використовуються в залежності від необхідності і можуть бути використані як незалежні реєстри, такі як лічильник від'ємних пристроїв і т. д., в залежності від тих завдань, які ставляться перед режимом роботи і об'ємом виконання операцій. РЗП можуть бути доступними для програміста і програмно недоступними.

13.6.2. Лічильник команд

Лічильник команд є одним з найбільш важливих реєстрів МП. Як відомо, програма - це послідовність команд, які зберігаються в пам'яті мікро ЕОМ і призначені для того, щоб інструктувати машину, як розв'язувати поставлену задачу. Для коректного виконання задачі команди повинні надходити в строго визначеному порядку. На лічильнику команд лежить відповідальність слідкувати за тим, яка команда виконується, а яка буде виконана наступною. Часто лічильник команд має набагато більше розрядів, ніж довжина слова даних МП. Так, в більшості 8-розрядних МП, які адресуються до пам'яті об'ємом 65 К, число розрядів лічильника дорівнює 16.

Згідно з рис.11.1 лічильник команд з'єднано з внутрішньою шиною даних МП. Теоретично цей лічильник може отримувати дані про адреси команд програми із блоку МП, підключеного до внутрішньої шини. Однак на практиці дані надходять в лічильник команд із пам'яті мікро ЕОМ. Коли МП починає працювати, по команді початкового встановлення в лічильник команд завантажуються дані із області пам'яті, заданої проектувальником МП. Перед записом програми необхідно помістити початковий адрес для команд програми в область пам'яті, вказану проектувальником.

На відміну від акумулятора, лічильник команд не може виконувати операції різного типу. Набір команд для лічильника обмежений, на відміну від акумулятора. Перед виконанням програми, лічильник необхідно завантажити числом – адресою області пам'яті, яка містить першу команду програми.

Слід звернути увагу, що на рис. 13.1 реєстр адреси пам'яті і адресна шина розміщені нижче лічильника команд. Адреса області пам'яті, яка містить першу команду програми, посилається із лічильника команд в реєстр адреси пам'яті, після чого вміст обох реєстрів стає однаковим. Довжина реєстра адреса пам'яті рівна 16 розрядам.

Адреса місцезнаходження першої команди програми посилається по адресній шині до схем керування пам'яттю, в результаті чого зчитується вміст області з вказаною адресою. Пам'ять пересилає цю команду в спеціальний реєстр блоку МП, які називають реєстром команд.

Після отримання команд із пам'яті МП автоматично дає приріст вмісту лічильника команд. Цей приріст лічильник команд (ЛК) отримує якраз в той момент, коли МП починає виконувати програму, яка отримана з пам'яті. Відповідно, починаючи з цього моменту лічильник команд "вказує", якою буде наступна команда. Лічильник команд містить адресу наступної команди протягом всього часу виконання попередньої команди. Отже, в кожен момент ЛК вказує команду, яка наступна за виконуваною.

ЛК може бути завантажений іншим вмістом при виконанні особливої групи команд. Може виникнути необхідність виконати частину програми, яка "випадає" у послідовності команд основної, або головної програми. Наприклад, таку частину програми, яку слід багатократно повторювати в процесі виконання всієї програми. Замість того, щоб записувати цю частину програми кожен раз, коли в ній виникає необхідність, такий запис можна зробити один раз і повертатись до її повторного виконання, відступаючи від вказаної послідовності.

Частина програми, яка виконується шляхом відступу від строгої послідовності команд головної програми, називається *підпрограмою*. Після того, як в лічильник команд записано початкову адресу підпрограми, лічильник отримує приріст в міру виконання команд цієї програми. Так продовжується доти, поки не зустрінеться команда повернення в головну програму.

13.6.3. Реєстр адресу пам'яті

При кожному звертанні при пам'яті мікро ЕОМ, реєстр адреси пам'яті (РАП) вказує адресу області пам'яті, яка підлягає використанню МП. Реєстр адресу пам'яті містить двійкове число – адресу області пам'яті. Вихід цього реєстру називають адресною шиною і використовується для вибору області пам'яті або в деяких випадках для вибору порту вводу-виводу.

Протягом підциклу виконання команди вміст регістру адреси пам'яті залежить від виконуваної програми. Якщо відповідно з командою МП повинен виконати ще одне звернення до пам'яті, то регістр адреси пам'яті підлягає повторному використанню в процесі обробки цієї команди. Для деяких команд адресація до пам'яті не потрібна: наприклад, команда очищення акумулятора. При обробці таких команд регістр пам'яті використовується лише один раз – протягом підциклу вибірки команди із пам'яті.

А більшість МП РАП і лічильник команд мають однакове число розрядів. Як і лічильник команд, РАП повинен мати кількість розрядів для адресації області пам'яті мікро ЕОМ. У більшості 8-розрядних мікропроцесорів кількість розрядів регістра адреси пам'яті рівне 16. Такий регістр можна розділити на два окремих регістри, кожен з яких має незалежне підключення до шини даних МП. Одним із цих регістрів називають регістром старшого байта (СБ), а другий – регістр молодшого байта (МБ).

Оскільки РАП підключений до внутрішньої шини даних МП, він може завантажуватись від різних джерел. Більшість МП мають команди, які дозволяють завантажувати цей регістр вмістом лічильника команд, регістрів загального призначення або з області пам'яті. Деякі команди дають можливість змінювати вміст РАП шляхом виконання обчислень: нове значення вмісту цього регістру отримується шляхом додавання або віднімання вмісту лічильника команд з числом, вказаним в самій команді. Адресація такого типу отримала назву адресації з використанням розміщення.

13.7. Пристрій керування мікропроцесора

Пристрій керування мікропроцесора (ПКМП) служить для підтримання необхідної послідовності роботи всіх ланок МП. По “розпорядженню” схеми керування чергова команда витягується з регістру команд, визначається, що необхідно зробити з даними, а потім генерується послідовність керуючих сигналів по виконанню поставленої задачі.

Звичайно робота керування мікропрограмується і ця мікропрограма заноситься в пам'ять МП при виготовленні цього конкретного типу процесора.

Пристрій керування МП складається з регістра команд РК, дешифратора команд (ДШК) та схеми керування (СК), що має в свою чергу блок синхронізації (БС) та блок обміну інформації (БОІ).

13.7.1. Регістр команд

Регістр команд (РК) призначений для прийому в машинному такті ТЗ і зберігання на час виконання команди, яка виконується, причому ця функція реалізується МП автоматично з початком циклу "вибірка-виконання", який називається *машинним циклом*. Машинний цикл складається із двох підциклів: вибірки і виконання. Згідно з схемою рис.13.1., цей регістр з'єднаний з внутрішньою шиною даних, однак він тільки приймає дані, посилати дані на шину він не може. Вихід цього регістра є частиною дешифратора команд.

Пригадаємо ще раз послідовність реалізації циклу вибірка-виконання. Спочатку команду беруть із пам'яті, потім лічильник команд настроюється на вказання наступної команди, яку потрібно виконати. При отриманні команди із відповідної області пам'яті копія команди розміщується на внутрішній шині даних і пересилається в регістр команд. Після цього починається підцикл виконання

команди, протягом якого дешифратор команд “читає” вміст регістру команд, повідомляючи МП, що робити для реалізації операції команди.

Число розрядів РК залежить від типу МП: інколи воно співпадає з числом розрядів слова даних, в інших випадках – дорівнює лише 3 або 4.

Таким чином, всяка запрограмована фізична операція може бути перетворена в послідовність електричних сигналів, які в цифровій формі можуть перетворюватись (тобто в часі можуть над ними проводитись дії логічного характеру), результат цих логічних дій може бути перетворений в електричний сигнал, який призведе до виконання будь-який механізм. Ми маємо справу з технічним засобом, керованим людиною.

Дешифратор команд розшифровує код операції команди і виробляє мікронакази у відповідності з мікропрограмою виконання команд.

13.7.2. Схема керування

Схема керування з допомогою БС і БОІ формує 4 вхідних і 6 вихідних ліній зв'язку, тобто шину керування, по якій передаються всі необхідні керуючі сигнали, що забезпечують покомандне виконання програми, яку повинен виконувати конкретний МП. Для забезпечення зв'язку МП з повільнодіючими пристроями, зовнішньою пам'яттю, клавіатурою тощо, або ж для роботи пам'яті тільки з контролерами прямого доступу до пам'яті передбачено реалізацію за допомогою вузлів МПК станів – “Чекання”, “Переривання”, “Захоплення шин”, “Зупинка”, про що йтиме далі.

(БС) - блок синхронізації призначений для синхронної роботи всіх внутрішніх вузлів МП і синхронізує прийом і передачу потоків інформації з допомогою періодичних імпульсів, що називаються машинними тактами і машинними циклами (МЦ).

БС складається із схем формування МТ, МЦ, схем формування сигналу "Скид" і схеми формування сигналу синхронізації.

Схема формування МТ - призначена для вироблення тактів М1 - М5, що дорівнюють по довжині періоду тактової частоти. За час одного такту виконується тільки одна мікрокоманда. Схема МЦ необхідна для вироблення машинних циклів (М1 - М5), що дорівнює 3, 4, 5 МТ.

(БОІ) - блок обміну інформації - складається зі схем аналізу переривань, схем аналізу захоплення шин і схем аналізу готовності.

Схема аналізу переривань забезпечує можливість роботи МП в реальному часі. Вона складається з тригера дозволу переривань і тригера запиту переривань. Запит переривання аналізується в останньому такті останнього машинного циклу кожної команди.

Схема АНАЛІЗУ ЗАХОПЛ. ШИН організовує режим прямого доступу до пам'яті з допомогою вхідного сигналу ЗАХОПЛ. ШИН і вихідного сигналу Підтвердження ЗАХОПЛ.

Схема АНАЛІЗУ ГОТОВНОСТІ дозволяє МП працювати з пам'яттю і зовнішніми пристроями будь-якої швидкодії.

13.8. Процес виконання команди в МП

Для розуміння процесів, що відбуваються в мікропроцесорному комплекті, необхідно знати як і в якій послідовності виконується команда МП. Для виконання програми, яку повинен виконати МП

необхідно в пам'ять МПК занести коди машинних команд і операнди (дані), з якими буде оперувати МП. Розглянемо логіку роботи МП, уявивши, що МП, рис. 13.1, під'єднаний до оперативної пам'яті з адресним полем, куди попередньо була записана якась команда.

РК (регістр команд), ДШК (дешифратор команд), програмний лічильник (ПЛ) і пристрій керування (ПК) використовуються для вибору команд з пам'яті і їх виконання.

Записана в пам'ять команда має початковий адрес (напр. А000). Програмний лічильник через шину адреси видає на регістр адреси пам'яті (РАП) адресу команди. З пам'яті на шину даних надійде вміст цієї комірки (код операції) і МП запише його в регістр (РК). Цей регістр передасть інформацію дешифратору коду операції, який розпізнає цей код і видасть керуючі сигнали для схем синхронізації і керування. (Схеми синхронізації і керування - це процесор всередині процесора, який керується мікропрограмою, яка знаходиться в ПЗУ. Мікропрограма точно вказує МП, що робити, щоб виконати кожен команду.)

Наприклад, для команди MVI,А з дешифратора команди керування і синхронізації надходить дешифрований код ЗЕ. Потім збільшується адреса на 1 в програмному лічильнику. Пристрій керування (ПК) визначає, що за цим кодом операції йде байт даних, так що вміст комірки пам'яті, вказаний програмним лічильником, зчитується в акумулятор. Програма повинна складатись таким чином, щоб після коду операції (КОП) команди завжди йшли дані.

При виконанні операцій МП може виконувати основні типи операцій, - читання даних з порту вводу або пам'яті, - запис даних в порт виводу або пам'ять, - виконання внутрішніх операцій, - передача керування другій комірці пам'яті.

Для організації роботи МП треба подати на БС два тактові імпульси $\emptyset 1$ і $\emptyset 2$, з допомогою яких ця схема на початку кожного машинного циклу МЦ, виробляє сигнал "Синхр", а під час дії цього сигналу на шину даних видається інформація про стан МП. Мікропрограма вкаже керуючій логіці, що команда виконана, вміст програмного лічильника збільшується на 1, і наступний байт команди (наступний код) зчитується в регістр команд. Потім починається процес виконання наступної команди.