

Тема 3. Загальні принципи побудови мікро-ЕОМ та ПЕОМ

3.1. Принципи побудови ЕОМ

Базові принципи організації мікропроцесора. Канонічна схема мікропроцесора. Системи, види і формати команд універсальних мікропроцесорів. CISC-і RISC-архітектури. Вибірка, дешифрування та виконання команд. Подання роботи обчислювального тракту процесора на мікроархітектурному рівні. Режими адресації пам'яті та пристроїв вводу-виводу. Система переривань. Механізми звернення до підпрограм. Мікропроцесор Intel 8086(88). Машинна мова. Архітектура процесора. Регістри загального призначення. Індексні регістри та регістри-вказівники. Регістри сегменту. Шини мікропроцесора. Зв'язок з магістраллю. Переривання (внутрішні та зовнішні, масковані та немасковані).

Архітектура ЕОМ залежить від типу центрального процесора чи мікропроцесора (МП). Визначальним моментом є розрядність МП (8, 16, 32, 64 і т. д.). ЕОМ, зібрані на базі цих МП, мають не тільки певний тип МП, а й функціональні розширювачі, контролери, гнучкі та жорсткі диски, клавіатуру, дисплей та інші вузли.

Відомо також, що розв'язування певної задачі на ЕОМ поділене на частини. Найпростіша частина – це машинна команда. Набір машинних команд, які записані у певній послідовності, становлять машинну програму. Для реалізації тої чи іншої програми необхідні арифметико-логічний пристрій, пристрій пам'яті, пристрій керування, пристрій введення/виведення інформації.

Взаємодія цих пристроїв відбувається через так звану системну магістраль (СМ), або шину. У переважній більшості випадків СМ – це розширення шини МП, однак, не завжди (напр. МП – шістнадцятирозрядний, а зовнішня шина – восьмирозрядна.)

Незалежно від архітектурного виконання структурну організацію взаємодії компонентів ЕОМ можна зобразити, як на рис. 3. 1.

Структурна схема може дещо видозмінюватися залежно від призначення ЕОМ: побутова, професійна чи спеціалізована. (Робоча станція, автоматизоване робоче місце тощо).

Між МП та ОП є зв'язок, тобто розрядність МП пов'язана з обсягом ОП. Наприклад, 8-розрядний МП – ОП 64 Кбайт, для 16 розрядної – від 1 до 16 Мбайт.

Швидкість виконання операцій визначена тактовою частотою МП та магістральної шини. Прості операції додавання цілих чисел типу "пам'ять – пам'ять" виконуються за 20-30 тактів, а множення-ділення – за 130-150. Знаючи тактову частоту процесора, легко визначити його швидкодію.

3.2. *Склад і характеристика ПЕОМ*

Базовий комплект сучасної ПЕОМ містить такі головні компоненти: системний блок, клавіатуру, монітор (дисплей), пристрій для друкування (принтер), маніпулятор. Засоби зв'язку з пристроями, що не належать до базового комплекту, містить також системний блок.

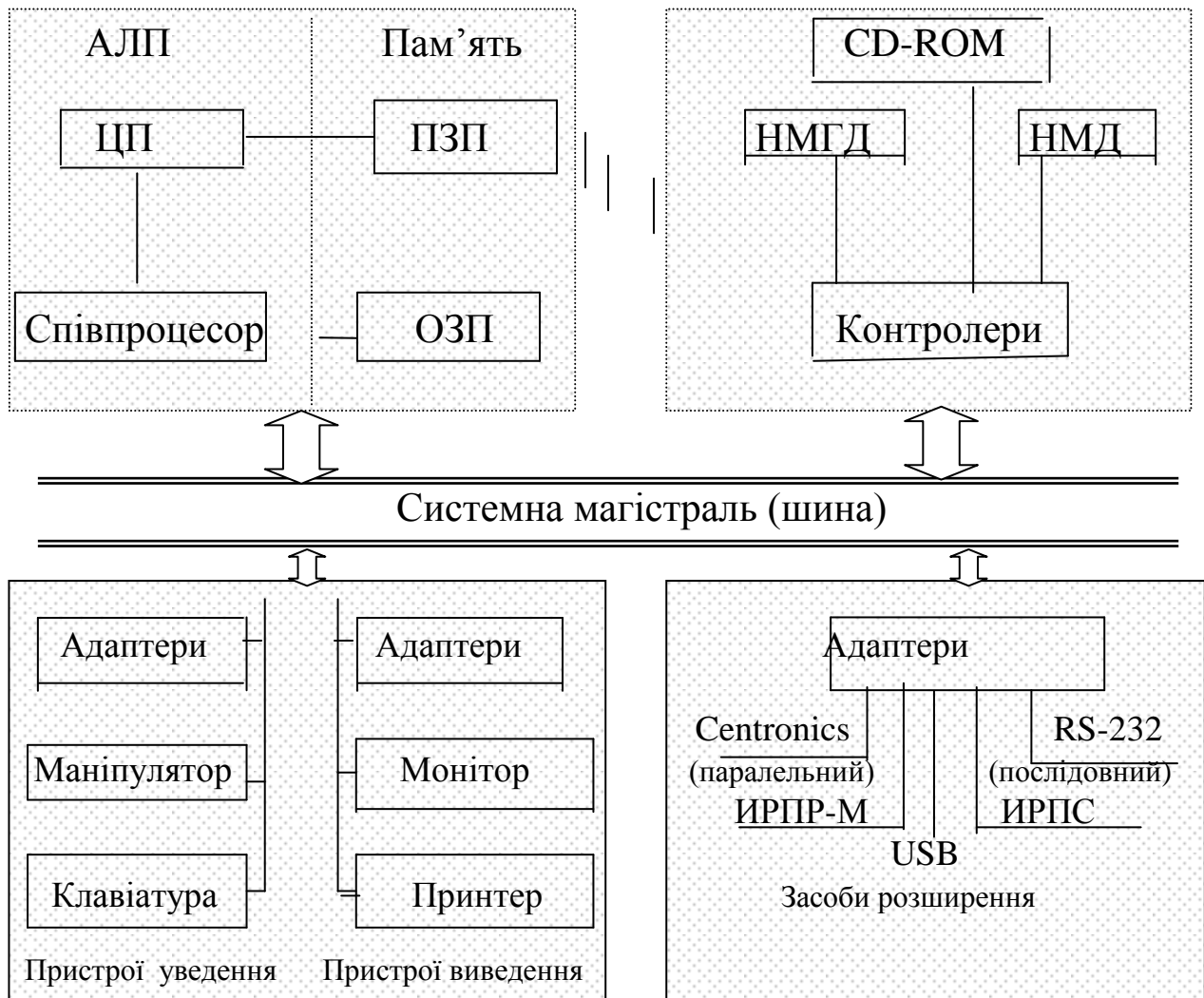


Рис. 3. 1. Структурна схема сучасної ЕОМ

Системний блок містить: електронні модулі (системна плата, оперативний запам'ятовувальний пристрій (ОЗП), постійний запам'ятовувальний пристрій (ПЗП), контролери пам'яті та зовнішні пристрої, блок живлення, спеціальні з'єднувачі (слоти) для модулів розширення ОЗП, адаптери для приєднання до мережі (LAN-карти), контролери складних зовнішніх пристроїв, наприклад, відеоадаптер. Залежно від типу корпусу системного блоку розташування окремих вузлів та рознять для під'єднання зовнішніх пристроїв не однакове. На рис. 3. 2 показано зовнішній вигляд передньої та задньої панелей системного блоку типу АТ та АТХ.

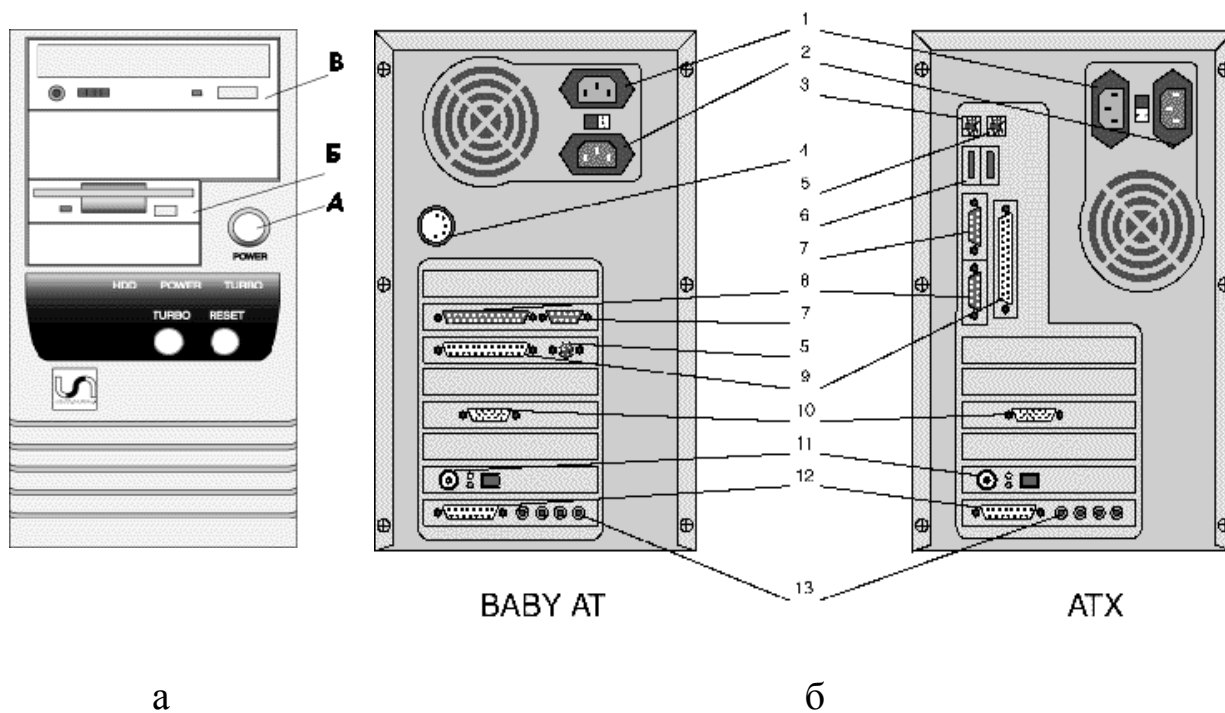


Рис. 3. 2. Зовнішній вигляд передньої (а) та задньої (б) панелей системного блоку ПЕОМ. А – кнопка для вмикання ПК, Б – пристрій для гнучких дисків, В – пристрій для компакт-дисків, 1 – роз'єм для під'єднання шнура живлення монітора, 2 – роз'єм для під'єднання ПК до мережі живлення, 3 – роз'єм для під'єднання клавіатури PS/2, 4 – роз'єм для під'єднання клавіатури, 5 – роз'єм для під'єднання маніпулятора (мишки), 6 – USB-порти, 7 – послідовний порт (COM2), 8 – послідовний порт (COM1), 9 – паралельний порт (LPT), 10 – вихід відеоадаптера, 11 – роз'єм для під'єднання до локальної мережі, 12 – ігровий порт, 13 – гнізда для під'єднання зовнішніх акустичних систем.

3.3. Типи даних МП

Для всіх сучасних МП базовою одиницею даних є байт, тобто 8-бітова величина. Після того, як дані локалізовані у пам'яті процесора, МП може інтерпретувати їх різними способами, залежно від змісту команди чи намірів програміста. Для кодування інформації використовують в основному двійкову, десяткову та шістнадцяткову системи числення. У сучасних МП зrealізована апаратна підтримка багатьох типів даних, які включають в себе числа, вказівники (pointer) та рядки (string):

1. *Біт* (bit) – поодинокий двійковий розряд.
2. *Бітове поле* (bit field) – група до 32 бітів.
3. *Рядок бітів* (bit string) – послідовність бітів довжини до 4 Гбіт.
4. *Байт* (byte) – ціле число, що займає 8 бітів (старший розряд відводиться під знак).
5. *Беззнаковий байт* (unsigned byte) – натуральне число, що займає 8 бітів.
6. *Ціле слово* (integer word) – ціле число, що займає 16 бітів.
7. *Беззнакове ціле* (unsigned integer, word) – натуральне число, що займає 16 бітів.
8. *Довге ціле* (long integer, long word) – ціле число, що займає 32 біти.
9. *Довге беззнакове ціле* (unsigned long integer) – натуральне число, що займає 32 біти.
10. *Чотирикратне ціле* (quad integer, quad word) – ціле число, що займає 64 біти.
11. *Чотирикратне беззнакове ціле* (unsigned quad integer) – натуральне число, що займає 64 біти.
12. *Символ* (char) – байтове подання керівних та алфавітно-цифрових символів ASCII.
13. *Рядок* (string) – неперервна послідовність, утворена із байтів слів чи подвоєних слів від 1 байта до 4 Гбайт.
14. *Двійково-десятковий код* (binary coded decimal, BCD) – байтове (незапаковане) подання десятикових цифр від 0 до 9.
15. *Запакований двійково-десятковий код* – байтове подання від 0 до 9 в одному байті.
16. *Короткий вказівник, зсув* (short pointer, offset) – 16- або 32-розрядна адреса, що непрямо визначає комірку пам'яті.
17. *Вказівник* (pointer) – повна адреса елемента пам'яті, яка містить 16-бітний або 32-розрядний зсув.

У тих випадках, коли МП взаємодіє з арифметичним співпроцесором, додатково використовують знакові 32-, 64- та 80-розрядні формати даних з плаваючою комою.

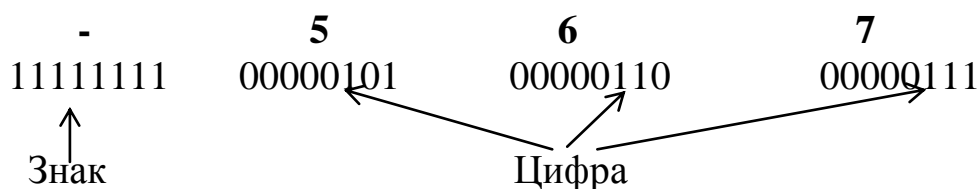
Арифметичні типи даних 32-розрядних ПК

Тип даних	Біти	Кількість десяткових значущих цифр	Діапазон значень
Коротке ціле	16	4	$-32768 \leq x \leq +32767$
Ціле	32	9	$-2 \cdot 10^9 \leq x \leq +2 \cdot 10^9$
Довге ціле	64	18	$-9 \cdot 10^{18} \leq x \leq +9 \cdot 10^{18}$
Запаковане десяткове	80	18	$-99..99 \leq x \leq +99..99$
Коротке дійсне	32	6-7	$8.43 \cdot 10^{-37} \leq x \leq 3.37 \cdot 10^{38}$
Подвійне дійсне	64	15-16	$4.19 \cdot 10^{-307} \leq x \leq +1.67 \cdot 10^{308}$
Довге подвійне дійсне	80	19	$3.4 \cdot 10^{-4932} \leq x \leq +1.2 \cdot 10^{4932}$

Пояснення запакованого та розпакованого формату:

а) У розпакованому форматі кожен байт містить одну десяткову цифру, яка зображується у молодшій тетраді. Знак числа кодують у різних ЕОМ по своєму. У ПК знак десяткового числа кодують у крайньому лівому байті, причому знак “-” кодується 1, знак “+” - кодується 0.

Наприклад, число -567 у розпакованому форматі буде мати вигляд



В неупакованому форматі цифри, як і символи найчастіше зображують у американському коді ASCII (American Standart Code for Information Interchange). Розрізняють дві частини таблиці кодів: 1 - основна, 2 - додаткова (з урахуванням національних стандартів) [5, 6]:

- 1) 0-127
- 2) 128-255.

Десяткові числа від 0 до 9 у кодуванні ASCII відповідають шістнадцятковим кодам від 30h до 39h відповідно.

б) В упакованому форматі десяткове число займає половину байта, тобто у кожному байті міститься дві десяткові цифри з вагами розрядів 8421 у кожній тетраді. Діапазон чисел, який може бути представлений у одному байті знаходиться у межах від 0 до 99.

Наприклад, число +963,8 в упакованому форматі буде таким:

+ 9 6 3 , 8
00000000 1001 0110 0011 1000

Знак Цифри

Положення десяткової коми може бути довільним, зокрема, для ПК вона знаходиться після третього зліва розряду числа.

Типи даних, що підтримуються архітектурою IA-64

Архітектура IA-64 64-розрядна. Тому в ній з'явилися нові типи даних. Цілі числа представлені 1, 2, 4 і 8 байтами. Дійсні — 4 (single— стандартна точність), 8 (double— подвійна точність) і 10 (double extended — «розширене» дійсне число подвійної точності).

Усі вказівники мають ділянки пам'яті ємністю 8 байт. За замовмочанням більшість інструкцій для роботи з цілими числами оперують даними з ємністю 64 біт. Запис у регістри здійснюється такими самими порціями. В 128-бітовий пакет інструкцій компілятор вставляє шаблон з кількох бітів з метою маркування команд, що виконуватимуться паралельно. Кожна інструкція містить три 7-бітових поля GPR (general-purpose register), які задають стан універсальних регістрів.

Всього процесори з архітектурою IA-64 мають 128 універсальних цілочислових регістрів і стільки ж регістрів для чисел з плаваючою комою. Всі регістри доступні програмісту в будь-якому порядку. Це принципово новий набір регістрів порівняно з регістрами архітектури IA-32.

Єдиний універсальний лінійний адресний простір процесорів архітектури IA-64 має ємність 264 байт. Дані та інструкції можуть зберігатися одночасно в одному діапазоні пам'яті. В цьому сутність єдиного адресного простору (single). «Універсальний» (uniform) в даному випадку означає, що за діапазонами пам'яті не закріплюються наперед визначені функціональні обов'язки. «Лінійний» (linear)

означає, що в адресному просторі не виділяються сегменти. Всі 264 байт пам'яті можуть розподілятися довільними порціями.

Загалом інструкції зберігаються в сервісному центрі пам'яті так, щоб байт з найменшою значущістю був першим (так званий little-endian byte order). Але такий байт може розташовуватися в пам'яті як першим, так і останнім (так званий bigendian byte order).

Серед нових в архітектурі IA-64 — інструкції, призначені для роботи з числами з плаваючою комою. Їх називають «множинно-акумулюючими» (Multiply and Accumulate), наприклад множення і додавання (fma) — операція, яку вважають основною для всієї арифметики з плаваючою комою. Якщо цю операцію записати у вигляді $y = a*b+c$, то при $c=0$ виконується множення, при $b=1$ — додавання, а при $b = 1$ і $c = 0$ — нормалізація.

Множинні операції дають змогу виключити одну операцію округлення, тому точність операції вища, ніж точність при використанні двох стандартних операцій mul і add.

3.4. Мікропроцесор – основа ЕОМ

Тепер детальніше ознайомимося з архітектурою та принципом функціонування МП. Як тільки з'явилась технологічна можливість створювати схеми великої інтеграції (ВІС), які дали змогу стикувати МП з ОЗП, ПЗП та інтерфейси периферійних пристроїв, то відразу виникли так звані мікропроцесорні комплекти (МПК). Одним з перших таких комплектів був Intel 4001/4004 (1971 р). До нього входили:

- центральний 4-бітовий МП ЦП-4004, який розміщувався на одному кристалі;
- ПЗП (4001) ємністю 256 8-бітових слів, які можна було використовувати для зберігання програм і констант. Він містився на одній мікросхемі і підлягав перепрограмуванню;
- оперативна пам'ять 4002 ємністю 80 4-бітових слів. Це була окрема МС, через яку виконувалася ще й програма введення/виведення (чотири вхідні лінії і система керування введення/виведення);

- зсувовий регістр 4003. Це 10-бітовий регістр з послідовним входом, а виходи можуть бути або один послідовного коду, або паралельний з усіх розрядів регістра. Головно відігравав роль буфера для вхідних і вихідних даних.

МПК 4004 використовував одну двонапрямлену 4-розрядну шину даних, а для сигналів керування між ЦП, ОЗП, ПЗП ще п'ять додаткових ліній.

Один ЦП міг керувати від 1 до 16 ПЗП і кількома регістрами зсуву. Разом з портами 4001 та 4002 обслуговувалося 128 ліній входу/виходу.

Система команд ЦП складалася з 45 команд, розподілених на три групи:

- 16 машинних команд (5 подвійної довжини);
- 14 акумуляторних команд;
- 15 команд уведення/виведення і адресування портів.

На наступному етапі розвитку ЕОМ розрядність МП зростала до 8 (Z80, КР580ВМ80А), 16(8086,8088,80186), 32(80386,80486) і т. д.

Архітектура сучасних мікропроцесорів.

Центральний процесор (англ. CPU - Central processing unit) – функціональна частина ЕОМ, що призначена для виконання програм на ЕОМ. Для побудови сучасних ЕОМ використовують мікропроцесори – процесори, які розташовані в корпусі однієї мікросхеми.

На рис. 3.2 зображена спрощена блок-схема мікропроцесора

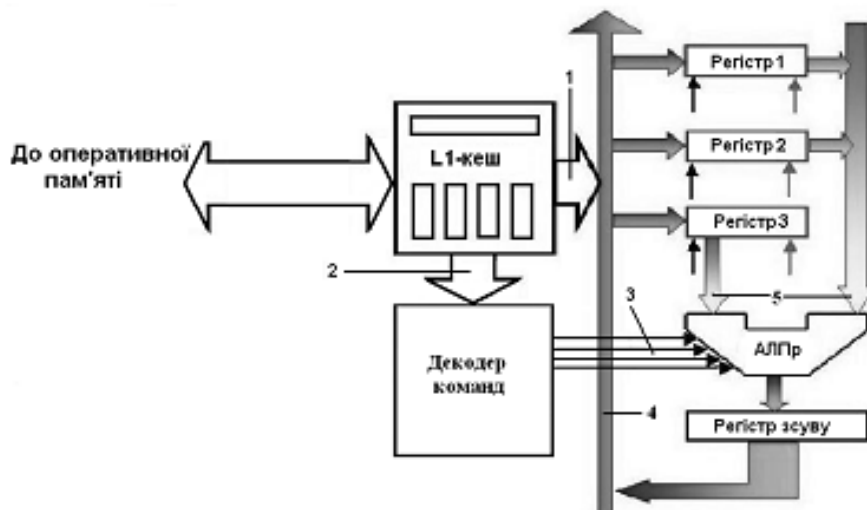


Рис. 3.2. Спрощена блок-схема мікропроцесора:

1 – внутрішня шина мікропроцесора, 2 – шина команд, 3 – шини зв'язку регістрів із арифметико-логічним пристроєм (АЛПр), Регістр 1, Регістр 2, Регістр 3 – регістри загального призначення, L1-кеш (див. далі) – швидка пам'ять для інструкцій і даних.

Через блок спраження із зовнішньою шиною мікропроцесора (Front Side Bus, FSB) у кеш-пам'ять мікропроцесора завантажують певну кількість даних і команд програми. Під час виконання програми чергова команда через шину команд завантажується в АЛПр, а дані через внутрішню шину мікропроцесора – у регістри загального призначення. Виконання команд програми здійснюється за сигналами тактового генератора, який розташовують на системній платі. На рис. 3.3 зображено операції, які виконує мікропроцесор під час виконання одного такту.

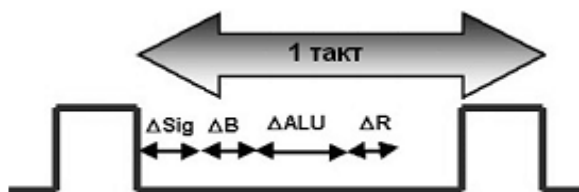


Рис.3.3.

ΔSig – час декодування команди мікропроцесором,
 ΔB – час завантаження даних із регістрів загального призначення в АЛПр,
 ΔALU – час виконання команди арифметико-логічним пристроєм,
 ΔR – час запису результатів виконання команди (в регістри загального призначення або у L1-кеш).

Для підвищення швидкодії мікропроцесорів у першу чергу підвищують тактову частоту (процесор Intel8086 мав тактову частоту 4.77 МГц). Для збільшення тактової частоти мікропроцесора можна:

- зменшити час декодування команди ΔSig ;
- зменшити часи ΔB , ΔR (збільшити швидкість обміну даними);
- зменшити час виконання команди ΔALU .

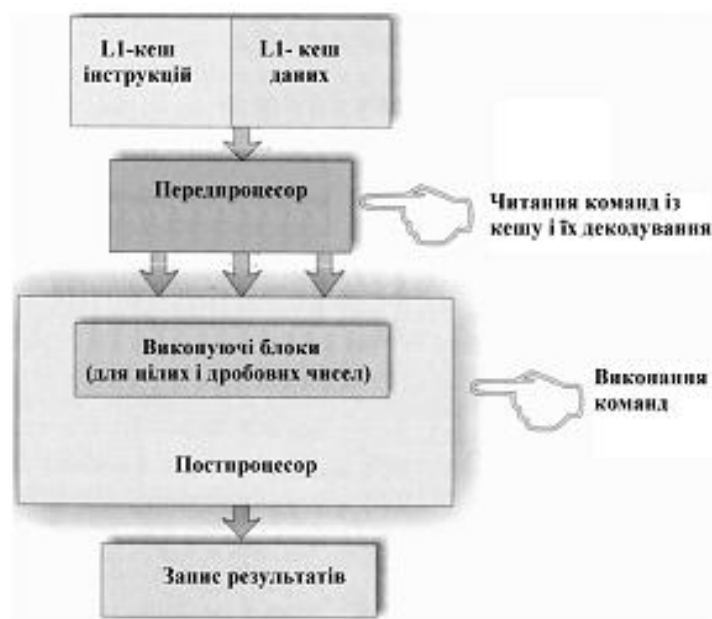
Для підвищення швидкості обміну даними виробники мікропроцесорів покращують розташування компонент мікропроцесора і зменшують їх геометричні розміри (чим менший шлях електрона, тим швидше він його проходить) – сучасні мікропроцесори виготовляють за техпроцесом 32, 22 нм, очікується перехід на 15 нм.

Час декодування команди ΔSig можна зменшити за рахунок скорочення числа команд мікропроцесора. Ще у вісімдесятих роках минулого сторіччя у Кебріджському і Стенфордському університетах були проведені числові експерименти із моделювання процесора зі скороченою системою команд (Reduced Instruction Set Computing – RISC), які довели перспективність RISC-архітектури. Мікропроцесори фірми Intel наразі проектуються за CISC-архітектурою (CISC – Complex Instruction Set Computing). Використання мікропроцесорів із великим набором команд ускладняє їх декодування (збільшує час ΔSig), але спрощує розробку програмного забезпечення для них.

Мікропроцесори фірми AMD мають RISC-архітектуру, а для використання їх в IBM PC- подібних ПЕОМ використовують програмну емуляцію системи команд $Ix86$. Набір команд мікропроцесорів Intel 8086, Intel 8088, Intel 80286 містив команди лише цілочисельної арифметики. Для обчислень з плаваючою комою використовували або програмну реалізацію (збільшувався час виконання команди), або зовнішній співпроцесор плаваючої арифметики.

Починаючи з мікропроцесора Intel 80486DX співпроцесор плаваючої арифметики розташовують в одному корпусі з мікропроцесором. Окрім підвищення тактової частоти розробники використовують різноманітні вдосконалення архітектури мікропроцесора (конвейєр, блок передбачення розгалуження, попереднє завантаження операндів, збільшення розмірів кеш-пам'яті [див. далі] різних рівнів [L1, L2, L3] і ін.). Конвейєр – це пристрій, у якому одночасно кілька команд програми проходять різні стадії обробки. В класичному п'ятирівневому конвейєрі п'ять команд одночасно проходять різні стадії виконання команди:

- читання [вибірка] команди з буфера;
- дешифрування команди;
- читання операндів;
- виконання команди;
- запис результатів виконання команди.



Виконання фрагменту програми мікропроцесором із конвейєром виглядає таким чином:

Такт	Читання команди	Дешифрування команди	Читання операндів	Виконання команди	Запис результатів виконання команди
1	Інструкція 1				
2	Інструкція 2	Інструкція 1			
3	Інструкція 3	Інструкція 2	Інструкція 1		
4	Інструкція 4	Інструкція 3	Інструкція 2	Інструкція 1	
5	Інструкція 5	Інструкція 4	Інструкція 3	Інструкція 2	Інструкція 1
6	Інструкція 6	Інструкція 5	Інструкція 4	Інструкція 3	Інструкція 2
7	Інструкція 7	Інструкція 6	Інструкція 5	Інструкція 4	Інструкція 3
8	Інструкція 8	Інструкція 7	Інструкція 6	Інструкція 5	Інструкція 4
9	Інструкція 9	Інструкція 8	Інструкція 7	Інструкція 6	Інструкція 5

Виконання п'яти команд на процесорі без конвейєра тривало би 5 тактів процесора, виконання цих же п'яти команд на процесорі з конвейєром триває один такт (після завантаження усіх сходинок конвейєра – у цьому прикладі на п'ятому такті і далі. В сучасних мікропроцесорах кількість „сходинок” конвейєра значно більша (операції читання команди з буфера, дешифрування команди, виконання команди, запису результатів виконання команди розділені на більш дрібні етапи).

Список літератури до теми

- 1 *Аладьев В.З., Хунт Ю.Я., Шишаков М.Л.* Основы информатики: Учеб. пособие. – М., 1999.
- 2 *Григорьев В.Л.* Микропроцессор i486. Архитектура и программирование. Кн.1. – М., 1993.
- 3 *Злобін Г.Г, Рикалюк Р.Є.* Архітектура та апаратне забезпечення ПЕОМ: Навч.посіб. –К., 2006., -К.,2012.
- 4 *3. Мячев А.А., Степанов В.Н.* Персональные ЭВМ и микроЭВМ. Основы организации: Справочник. – М., 1991.
- 5 *4. Каган Б.М.* Электронные вычислительные машины и системы: Учеб. пособие для вузов. – М., 1991.
- 6 *5. Фигурнов В. Э.* IBM PC для пользователя. Краткий курс. 7-е изд. – М., 1997.
- 7 *6. Дао Л.* Программирование микропроцессора 8088. Пер. с англ. – М., 1988.

Контрольні запитання до теми

1. Які компоненти входять до системного блоку?
2. Яка розрядність системної магістралі?
3. Якими типами даних оперує мікропроцесор?
4. Яка відмінність між запакованим і розпакованим форматом чисел?
5. Що таке командний цикл і як він пов'язаний з тактовою частотою процесора?
6. Яка функція тригера ознак?
7. Як використовуються тимчасові та акумуляторні регістри для організації виконання команд?
8. Як працює найпростіший мікропроцесор?