

## Лабораторна робота №2

### Тема: Вивчення архітектури, конструктивного виконання та системної синхронізації мікро контролерів серії AVR

#### Основні характеристики сімейства мікроконтролерів AVR

Здебільшого всі мікроконтролери побудовані по одній схемі. Система управління, що складається з лічильника команд і схеми декодування, бере на себе зчитування і декодування команд з пам'яті програм, а операційний пристрій відповідає за виконання арифметичних і логічних операцій; інтерфейс вводу/виводу дозволяє обмінюватися даними з периферійними пристроями; і, нарешті, необхідно мати запам'ятовуючий пристрій для зберігання програм і даних (рис. 2.1).

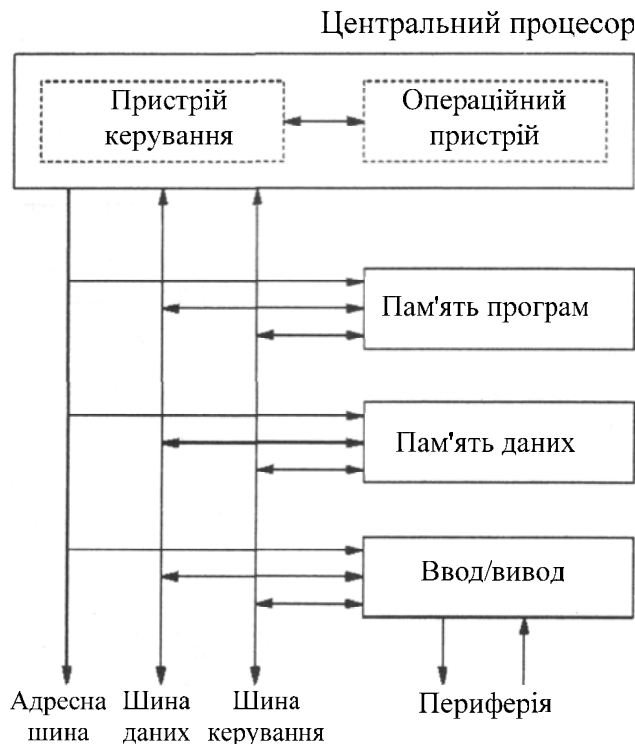


Рис. 2.1. Базова структура мікроконтролера

Операційний пристрій, як правило, складається з арифметико-логічного пристрою (АЛП), накопичуючого суматора і декількох допоміжних регістрів. У класичній програмі майже половина всіх команд — це команди пересилки (move) для передачі даних від допоміжних регістрів до накопичуючого суматора і назад. У сімействі мікроконтролерів AVR накопичуючий суматор, що є "тонким місцем", стає не так критично важливим, завдяки застосуванню 32-х робочих регістрів, зв'язаних з блоком АЛП. В результаті арифметичні і логічні операції можуть бути виконані протягом єдиного такту.

#### Арифметико-логічний пристрій

Операції АЛП можна розділити на три основні категорії: арифметичні, логічні і порозрядні. Кожній з цих категорій відповідають свої команди. Деякі мікроконтролери сімейства AVR мають також апаратні помножувачі в арифметичному блоці АЛП.

#### Структура команд

Запас команд мікроконтролера AT90S8515, базової серії, має 118 команд, надзвичайно оптимізованих з погляду доцільності і ефективності, а "найслабкіший" представник сімейства — AT90S1200 — 89 команд. У мікроконтролерах AVR базової серії всі команди мають однакову ширину слова 16 біт (тобто, 2 байти). Виключенням є тільки дві команди

для прямої адресації статичної пам'яті даних *lds* і *sts*, що складаються з двох слів і, відповідно, з 4 байт.

За деякими винятками (команда переходу, операції з безпосередньою адресацією і пересилки в пам'ять /з пам'яті) всі команди обробляються протягом одного такту системної синхронізації.

### Час виконання команди

На рис. 2.2 наведено зіставлення кількості необхідних тактових імпульсів осцилятора для виконання двох послідовних команд процесорами 68HC05, 80C51, а також сімействами мікроконтролерів PIC і AVR.

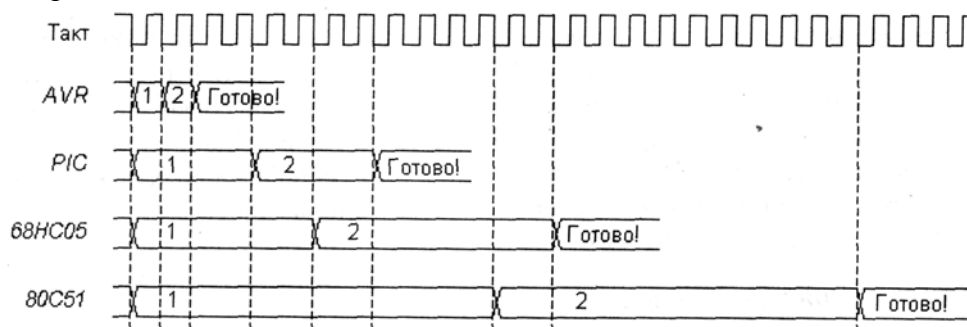


Рис. 2.2. Порівняння часу виконання команд різними процесорами

### Архітектура пам'яті

У запам'ятовуючих пристроях, які відповідають класичній концепції фон Неймана, дані і команди зберігаються в одній пам'яті. На противагу цьому, пам'ять по Гарвардській архітектурі, яка використовується і в мікроконтролерах AVR, складається з декількох компонентів. В даному випадку розділені пам'ять команд і пам'ять даних, тобто, звернення до команд здійснюється незалежно від доступу до даних.

У мікроконтролерах AVR окремі сегменти пам'яті влаштовані по-іншому також і фізично.

- Пам'ять програм реалізована на основі програмованої флеш-технології і стирається електрично. У всіх мікроконтролерах AVR пам'ять 16-розрядна (двобайтова), і завжди знаходиться "на кристалі". Розширення пам'яті програм за допомогою блоків EPROM або флеш неможливе.
- Внутрішня пам'ять для енергозалежних даних (тобто, даних, які будуть втрачені після відключення робочої напруги), являє собою статичну пам'ять RAM (SRAM). Перевага цього полягає у тому, що відпадає всяка необхідність у внутрішній регенерації як у випадку з деякими іншими процесорами, які використовують динамічну пам'ять. З цієї причини мікроконтролери AVR можуть працювати з тактами аж до 0 Гц. Крім того, деякі мікроконтролери AVR для збільшення об'єму оброблюваних даних можуть працювати із зовнішньою пам'яттю SRAM, що підключається. Для цього доводиться жертвувати дорогоцінними контактами вводу/виводу портів A і C.
- Для даних, які повинні зберегтися після відключення робочого напруги, в розпорядженні є мікросхема EEPROM (Electrically EPROM — електрично стирає ППЗУ). У пам'ять EEPROM можна записувати дані під час нормального виконання програми. Для EEPROM також немає обов'язкової необхідності в програмуючому пристрої.

### Область вводу/виводу

У мікроконтролерах AVR область вводу/виводу займає 64 байти і знаходиться в блоці статичної пам'яті SRAM за адресами від \$20 до \$5F. У мікроконтролерах AVR функції вводу/виводу для взаємодії із іншими пристроями застосовні не тільки для портів (від A до D), але також і для всіх регістрів стану і управління таких периферійних функцій як таймер,

пристрій UART, сторожовий таймер, доступ на запис і читання до пам'яті EEPROM, інтерфейс SPI і т.д.

#### Переривання і підпрограми

Техніка підпрограм і переривань в мікроконтролерах AVR може бути застосована звичайним способом. Перехід до частин програми, які в процесі її виконання обробляються багато разів або повинні міститися окремо унаслідок структуризації, здійснюється за допомогою команди *rcall*. Це — команда переходу, яка поміщає в стек наступну команду нормального виконання програми як адресу повернення. Після виконання підпрограми адреса витягується зі стека за допомогою команди *ret*, і програма продовжується.

Стек для адрес повернення знаходиться в пам'яті SRAM. Виняток становить тільки мікроконтролер AT90S1200, який застосовує для цієї мети апаратний стек за принципом LIFO (Last In, First Out — "останнім увійшов, першим вийшов").

У прикладі на рис. 2.3 показник стека ініціалізувався, починаючи з мітки *Start*, після чого слідує мітка *Call1* першого виклику підпрограми *UP1*. Як адреса повернення в стек поміщається адреса наступної команди, тобто, *\$78A*. Сама підпрограма *UP1* у представленому прикладі не виконує ніяких дій, і відбувається негайне повернення до головної програми. Для цього із стека витягується адреса, і виконання програми продовжується з команди *rjmp Call2*. З мітки *Call2* відбувається повторний виклик підпрограми *UP1*. В цьому випадку в стек поміщається адреса повернення *\$78D*.

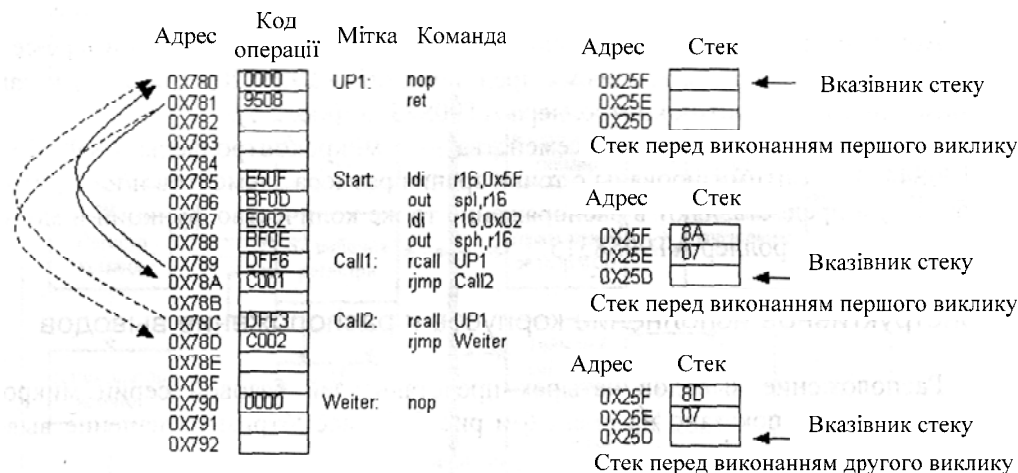


Рис. 2.3. Два виклики підпрограми

У правій частині рис. 2.3 показаний фрагмент статичної пам'яті SRAM, відведеної для стека. Вказано вміст стека перед виконанням першого і після виконання першого і другого викликів. Підпрограми також можуть бути вкладеними (тобто, з підпрограми можна викликати іншу програму).

Особливою формою підпрограм є підпрограми обробки переривань, за допомогою яких реалізовані наперед не заплановані звернення з програм. Іншими словами, програма асинхронна по відношенню до події, що викликає переривання. Тут, наприклад, може йтися про роботу таймера, прийомі байта через приймач-передавач UART або запиті на переривання від зовнішнього пристрою.

Залежно від побудови мікроконтролера, у розпорядженні базової серії AVR є до 12 векторів переривання (адрес входу).

#### Периферійні функції

З погляду апаратної частини, в мікроконтролерах AVR реалізовано безліч периферійних функцій. Так, наприклад, AT90S8515 має такі особливості:

- 32 програмованих лінії вводу/виводу;
- програмований пристрій UART;
- синхронний інтерфейс SPI;
- 8-розрядний таймер/лічильник;

- 16-розрядний таймер/лічильник з функціями порівняння, захоплення і можливістю реалізації двох виходів ШІМ (широко-імпульсний модулятор);
- інтегрований сторожовий таймер;
- аналоговий компаратор.

### Блок-схема мікроконтролерів AT90S1200 і AT90S8515

Архітектура сімейства мікроконтролерів AVR показана на прикладі її найбільш простого в даний час представника AT90S1200 (рис.2.4) і найпродуктивнішого мікроконтролера AT90S8515 (рис. 2.5).

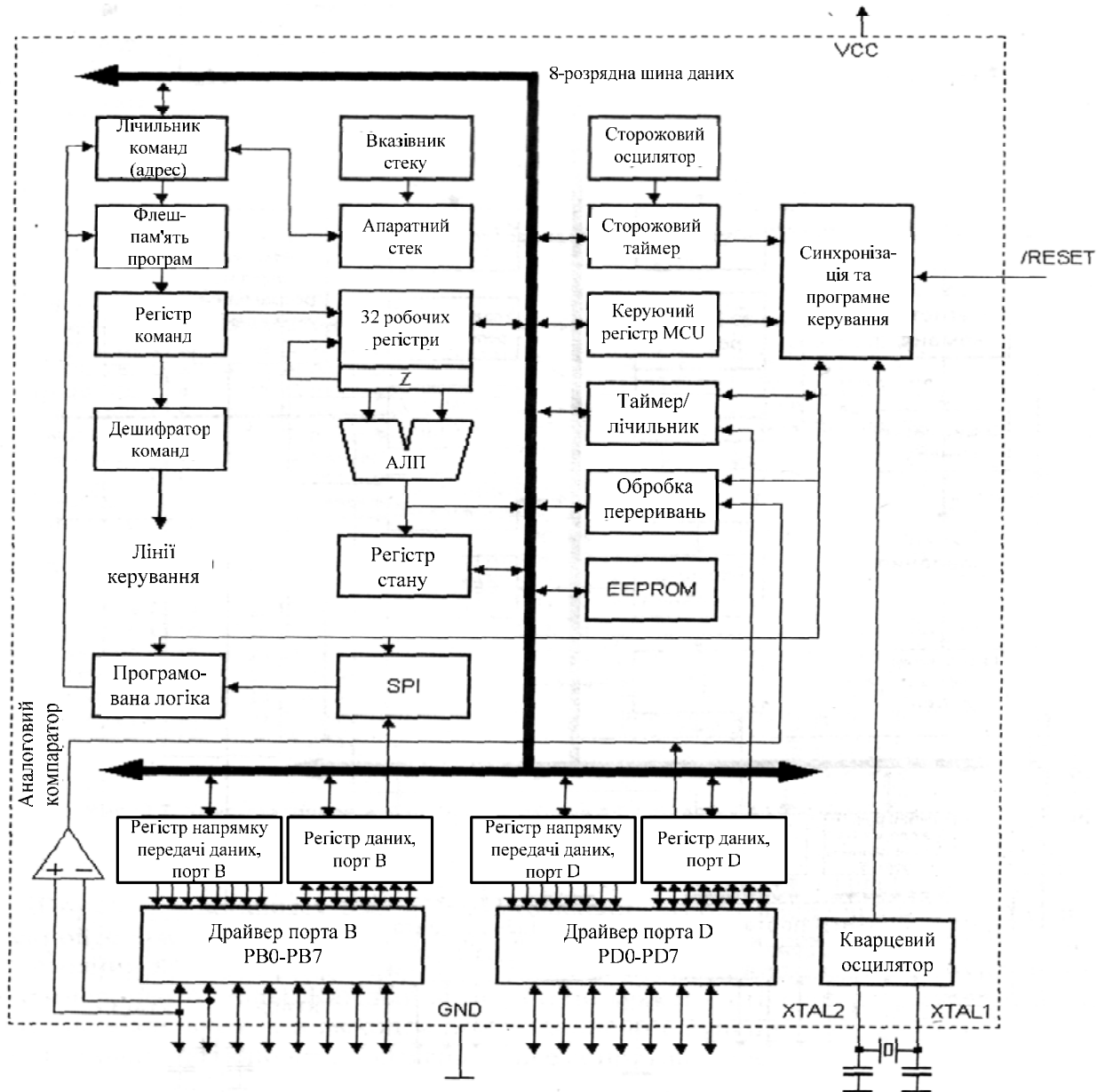


Рис. 2.4. Блок-схема мікроконтролера AT90S1200

Два інші представники сімейства — мікроконтролери AT90S2313 і AT90S4414 — оптимізовані з погляду розміру елементів і/або ціни і мають ту ж кількість функцій і модулів, що і мікроконтролер AT90S8515. У табл. 2.1 наведено характеристики представників базової серії мікроконтролерів сімейства AVR.

Таблиця 1.1.

Характеристики мікроконтролерів AVR базової серії

Характеристика	AT90S8515	AT90S4414	AT90S2313	AT90S1200
Кількість команд	118	118	118	89
Флеш-EPROM	8 Кбайт	4 Кбайта	2 Кбайта	1 Кбайт
Програмування через інтерфейс SPI	так	так	так	так
EEPROM	512 байт	256 байт	128 байт	64 байти
SRAM	512 байт	256 байт	128 байт	тільки стек
Робочі регістри	32	32	32	32
Контакти вводу/ виводу	32	32	15	15
Приймач-передавач UART	так	так	так	ні
Інтерфейс SPI	так	так	ні	ні
Сторожовий таймер	так	так	так	так
8-розрядний таймер/лічильник з попереднім дільником частоти	так	так	так	так
16-розрядний таймер/ лічильник з попереднім дільником частоти	так	так	так	ні
ШІМ-модулятор	2	2	1	ні
Джерела переривання	12	12	10	3
Аналоговий компаратор	так	так	так	так
Чекаючий режим	так	так	так	так
Режим зниженого енергоспоживання	так	так	так	так
Робоча напруга	2,7...6 В	2,7...6 В	2,7...6 В	2,7...6 В
Тактова частота при $V_{CC} = +5 В$	0...8МГц	0...8МГц	0...10МГц	0...12 МГц
Тактова частота при $V_{CC} = +3 В$	0...4МГц	0...4МГц	0...4МГц	0...4МГц
Корпус	PDIP40 PLCC44	PDIP40 PLCC44	PDIP20 SOIC20	PDIP40 SOIC20 SSOP20

**Конструктивне виконання корпусів і розташування виводів**

Розташування виведень чотирьох представників базової серії мікроконтролерів AVR показано на рис. 2.6 і рис. 2.7. Розглянемо призначення виводів докладніше.

**VCC** – підведення живлячої напруги

**GND** – заземлення

**Port A(PA0...PA7)**

Порт А є двонаправленим портом вводу/виводу з пропускнуою спроможністю 8 біт. Буфер виведення порту А в режимі прийому даних у змозі приймати струм силою до 20 мА і, завдяки цьому, безпосередньо живити, наприклад, світлодіоди. Кожне вивід порту може бути конфігурований індивідуально як вхід або вихід, а при виконанні функції введення до нього, при бажанні, можна підключати узгоджуючий опір.

Як особлива функція через порт А працює демультиплексована шина передачі даних і адрес, якщо до мікроконтролера AVR повинна бути підключена зовнішня пам'ять RAM.

**PortB(PB0...PB7)**

Порт В є двонаправленим портом вводу/виводу (I/O) з пропускнуою спроможністю 8 біт. Буфер виведення порту В в режимі прийому даних в змозі приймати струм силою до 20 мА і, завдяки цьому, безпосередньо живити, наприклад, світлодіоди. Кожне вивід порту

може бути конфігурований індивідуально як вхід або вихід, а при виконанні функції введення до нього, при бажанні, можна підключати узгоджуючий опір.

Альтернативно, через порт В можуть виконуватися також різні спеціальні функції (таймер, підключення входів аналогового компаратора, інтерфейс SPI).

### PortC(PC0...PC7)

Порт С є двонаправленим портом вводу/виводу (I/O) з пропускнуою спроможністю 8 біт. Буфер виведення порту С в режимі прийому даних в змозі приймати струм силою до 20 мА і, завдяки цьому, безпосередньо живити, наприклад, світлодіоди. Кожний вивід порту може бути конфігурований індивідуально як вхід або вихід, а при виконанні функції введення до нього, при бажанні, можна підключати узгоджуючий опір.

Як особлива функція через порт С виводиться старший байт адресної шини, якщо до мікроконтролера AVR повинна бути підключена зовнішня пам'ять RAM.

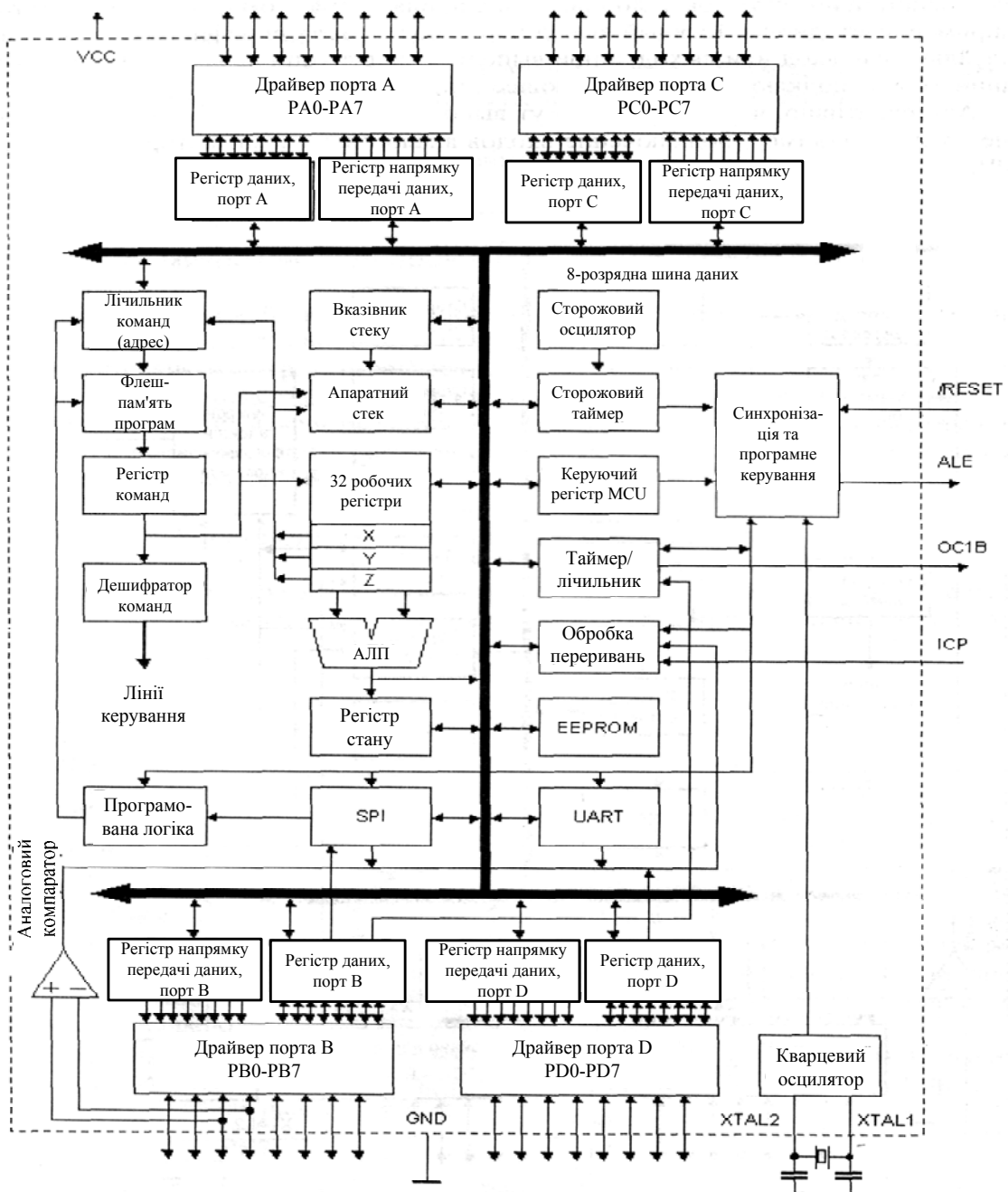


Рис. 2.5. Блок-схема мікроконтролера AT90S8515

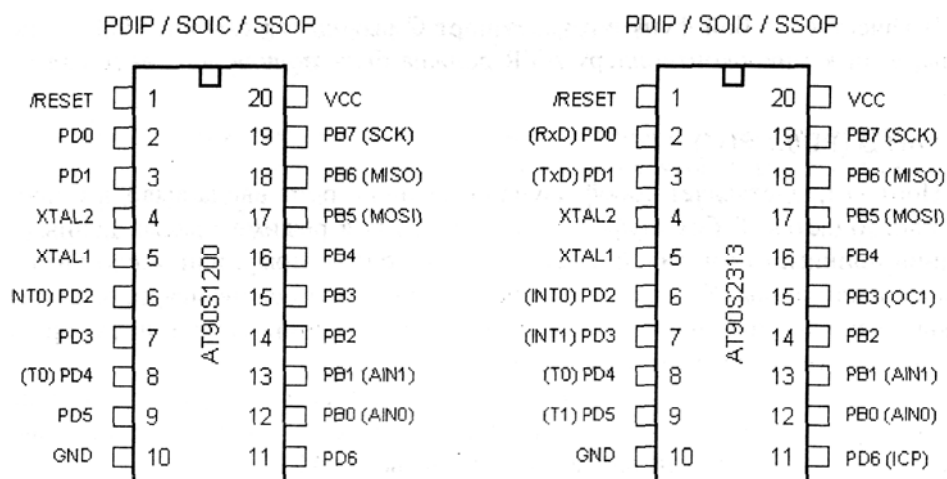


Рис. 2.6. Розташування виводів мікроконтролерів AT90S1200 і AT90S2313

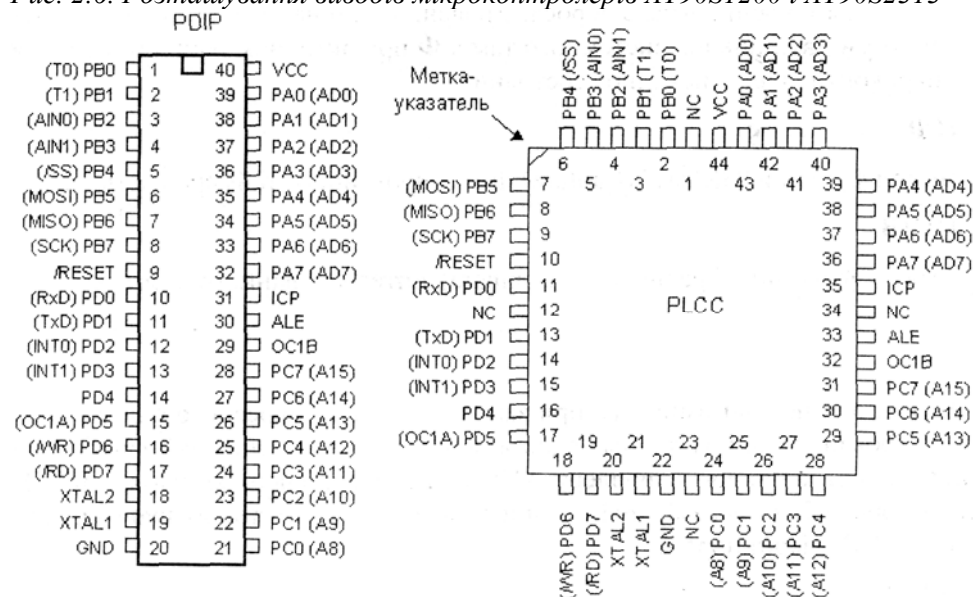


Рис. 2.7. Конструктивне виконання корпусів AT90S4414 і AT90S8515

### PortD(PD0...PD7)

Порт D є двонаправленим портом вводу/виводу з пропускнуою спроможністю 8 біт. Буфер виводу порту D в режимі прийому даних в змозі приймати струм силою до 20 мА і, завдяки цьому, безпосередньо живити, наприклад, світлодіоди. Кожне вивід порту може бути конфігурований індивідуально як вхід або вихід, а при виконанні функції введення до нього, при бажанні, можна підключати узгоджуючий опір.

Альтернативно, через порт D можуть виконуватися також різні додаткові функції (наприклад, надходження запитів на переривання, передача вихідних даних таймера, інтерфейс з пристроєм UART).

### /RESET

Вивід для подачі сигналу скидання. Рівень лог. 0 на цьому виводі на протязі мінімум двох циклів системного такту  $\Phi$  при активному осциляторі переводить мікроконтролер в початковий стан.

### ICP

Виведення функції "Захоплення" (Capture) інтегрованого таймера/лічильника T/C1.

### OC1B

Виведення функції "Порівняння" (Compare) інтегрованого таймера/лічильника T/C1.

## ALE

Вивід для подачі імпульсу при записі молодшого адресного байта з демультіплексованої шини даних/адреси через порт A в зовнішній фіксуючий регістр, коли до мікроконтролера AVR підключена зовнішня пам'ять RAM. Байт даних передається на другому кроці звернення до пам'яті RAM через порт A.

## XTAL1

Вхід інтегрованого осцилятора для вироблення такту системної синхронізації  $\Phi$  і вхід для зовнішнього тактового сигналу, якщо внутрішній осцилятор не застосовується.

## XTAL2

Вихід інтегрованого осцилятора для вироблення такту системної синхронізації  $\Phi$ .

## Генерування такту системної синхронізації у мікроконтролерах AVR

Для генерування тактів системної синхронізації в мікроконтролерах сімейства AVR використовується інтегрований осцилятор (виводи XTAL1 і XTAL2), що виробляє такти  $\Phi$ . Як альтернативний варіант може також використовуватися зовнішній тактовий сигнал. У разі мікроконтролера AT90S1200 можливий ще один варіант, оскільки в цьому мікроконтролері для генерування такту системної синхронізації може бути застосований інтегрований RC-осцилятор, що виробляє тактовий сигнал для сторожового таймера і схеми скидання.

## Інтегрований кварцовий осцилятор базової серії мікроконтролерів AVR

Інтегрований кварцовий осцилятор використовується всіма представниками базової серії мікроконтролерів AVR.

XTAL1 і XTAL2 — це входи/виходи інвертуючого підсилювача, який може бути застосований як вбудований осцилятор для генерування такту системної синхронізації  $\Phi$ . Генерування коливань може здійснюватися кварцовим або керамічним резонатором. Ємності конденсаторів C1 і C2 на рис. 2.8, які разом з кварцом і внутрішнім інвертором утворюють генератор Пірса, звичайно складають 22 пФ.

Якщо мікроконтролер AVR повинен працювати при тактуванні від зовнішнього джерела, то сигнал підводиться на вхід XTAL1, а вихід XTAL2 вбудованого осцилятора в цьому випадку залишається відкритим (рис. 2.9).

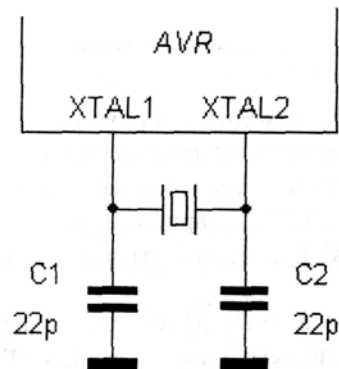


Рис. 2.8. Вироблення такту системної синхронізації  $\Phi$  з допомогою вбудованого осцилятора

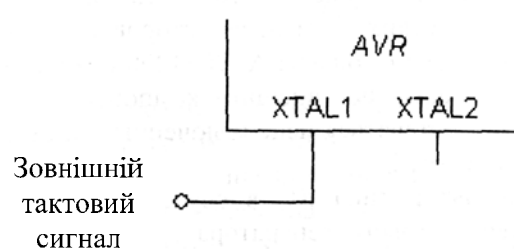


Рис. 2.9. Застосування зовнішнього тактового сигналу як такт системної синхронізації  $\Phi$

## Генерування такту системної синхронізації за допомогою контура RC-осцилятора

Генерування такту системної синхронізації за допомогою контура RC-осцилятора реалізоване тільки в моделі AT90S1200. Для випадків застосування, особливо чутливих до швидкодії, коли не пред'являються великі вимоги до точності такту системної синхронізації, як тактове джерело мікроконтролера AT90S1200 замість кварцового осцилятора може бути використаний RC-осцилятор, присутній на кристалі всіх мікроконтролерів AVR.

Цей осцилятор служить в першу чергу для подачі тактів на сторожовий таймер Він коливається з постійною частотою близько 1,1 МГц при робочій напрузі  $V_{CC} = 5$  В. Частоти для інших значень робочої напруги можуть бути узяті з рис. 2.10.



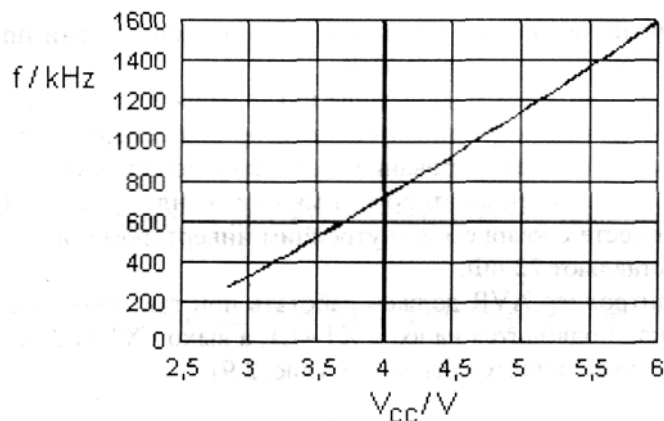


Рис. 2.10. Частота вбудованого RC-осцилятора залежно від живильної напруги  $V_{cc}$  при температурі навколишнього середовища  $T_A = 25^\circ\text{C}$

Саме собою зрозуміло, точність RC-осцилятора знаходиться в рамках звичайних електричних схем осциляторів типу RC, і залежить від температури. Для роботи мікроконтролера AT90S1200 спільно з RC-осцилятором немає необхідності в яких-небудь зовнішніх компонентах, вивід XTAL2 кварцового осцилятора може залишатися непідключеним, а вхід XTAL1 повинен бути встановлений в лог. 0 або лог. 1.

Осцилятор типу RC (мікроконтролер AT90S1200) вибирається як тактовий генератор за допомогою керуючого розряду RCEN (RC Enable — "RC-осцилятор активний") у флеш-пам'яті. Якщо розряд RCEN запрограмований (тобто, знаходиться в стані лог.0), то вибирається вбудований в кристал RC-осцилятор; а в тому випадку, якщо RCEN = лог.1, такт системної синхронізації виробляється кварцовим осцилятором.

Змінити розряд RCEN можна тільки в паралельному режимі програмування. Якщо мікроконтролер AT90S1200 повинен працювати з RC-осцилятором, а дані для пам'яті флеш і EEPROM повинні бути запрограмовані в послідовному режимі, то розряд RCEN повинен бути наперед запрограмований у паралельному режимі.

### Зміст звіту

1. Блок-схема базової структури мікроконтролера та призначення її основних елементів.
2. Організація архітектури пам'яті мікроконтролерів AVR.
3. Блок-схема мікроконтролера AT90S8515.
4. Призначення основних виводів мікроконтролера AT90S8515.

### Контрольні запитання

1. Які функції виконує арифметико-логічний пристрій?
2. Чим, на Вашу думку, визначається кількість команд мікроконтролера?
3. Поясніть відмінність архітектури пам'яті побудованої за концепцією фон Неймана від Гарвардської архітектури. За якою концепцією побудована архітектура пам'яті мікроконтролера AT90S8515?
4. У чому відмінність двох видів пам'яті SRAM та EEPROM?
5. Поясніть, яким чином здійснюються переривання в мікроконтролерах AVR?
6. Розкажіть, для чого призначені виводи зображені на рис.2.5.
7. Поясніть, як здійснюється формування системної синхронізації за допомогою інтегрованого кварцового осцилятора.
8. Поясніть, як здійснюється формування системної синхронізації за допомогою RC-осцилятора.
9. Поясніть призначення основних елементів мікроконтролера AT90S8515 (за блок-схемою).
10. Які відмінності в архітектурі мікроконтролерів AT90S1200 від AT90S8515