

Лекція 05

[5.1]

Передача даних між компонентами мікропроцесорних систем

Лекція 5

Судаков О.О, Радченко С.П.

«Сучасна мікропроцесорна техніка»

[5.2]

Проблема взаємодії компонентів процесорних систем

- Велика кількість пристроїв взаємодіє між собою і з процесором
- Всі ці пристрої треба з'єднати між собою
- Застосовуються спільні шини (BUS, bidirectional universal switch)
- Застосувати окремі шини для кожного пристрою - не можливо
- Спільні шини – конфлікт при доступі
- Застосовуються різні шини і різні протоколи
 - Паралельні
 - Послідовні
 - Синхронні
 - Асинхронні

[5.3]

Спеціальні стани процесора

- Стан **читання-запису пам'яті (memory read/write)**
 - По шині адреси передаються адреси пам'яті
 - По шині даних передаються команди або дані
 - Спеціальні сигнали керування вказують на доступ до пам'яті
- Стан **читання-запису портів введення виведення (port in/out)**
 - По шині адреси передаються номер порта введення-виведення
 - По шині даних передаються команди або дані
 - Спеціальні сигнали керування вказують на доступ до зовнішніх пристроїв
- Стан **зупинки (stop)**
 - Процесор нічого не виконує
 - Спеціальні сигнали керування вказують на зупинку
- Стан **утримання (hold)**
 - Процесор нічого не передає по шині і переводить виводи шин стан високого опору
 - Спеціальні сигнали керування вказують на утримання
- Стан **переривання (interrupt)**
 - Процесор перериває виконання за запитом зовнішнього пристрою
 - Спеціальні сигнали керування вказують на переривання
- Стан **скидання (reset)**
 - Початковий стан
- Стан **ввімкнення**
 - Відразу після ввімкнення живлення

[5.4]

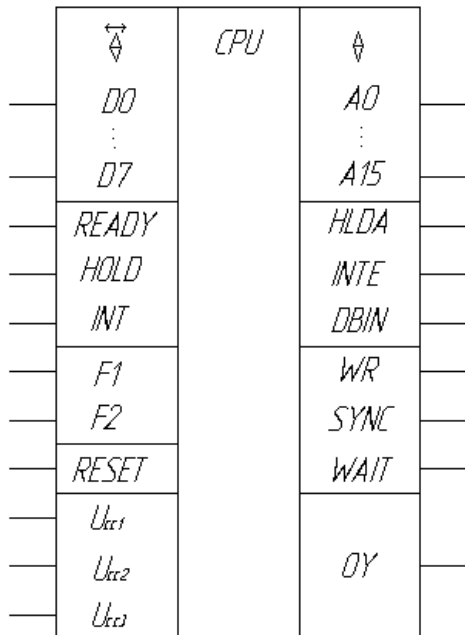
Стан ввімкнення (power on)

- Регістри процесора невизначені
- Процесор не можна використовувати поки він не буде переведений у стан ініціалізації
- При ввімкненні живлення деякі процесори вміють це обробляти
 - Переривання по ввімкненню живлення
- Деякі процесори вимагають ініціалізації
 - Штучно подати високий чи низький рівень на вивід reset
- Деякі процесори при збогах чи заниженні живлення виконують ініціалізацію

- Brown out reset

[5.5]

- Стан ініціалізації (скидання, reset)
Виникає при подачі сигналу на вивід reset
- Всі регістри, крім регістрів лічильника команд (IP, Code Segment) не змінюються
- Регістри лічильника команд (IP, CS) встановлюються у певне значення (reset vector)
- Виконання починається з машинної інструкції за цією адресою

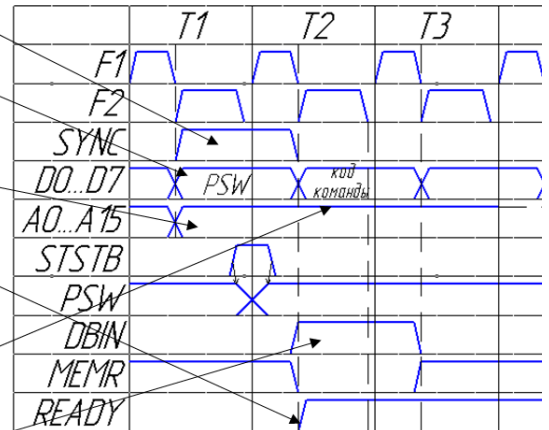


[5.6]

Цикл Читання (вибірка)-запис пам'яті (MEMORY)

- Такт 1
 - Проц. Сигнал синхронізації
 - Проц. Слово стану на шину даних
 - Проц. Спеціальний біт вказує читання/запис пам'яті
 - Проц. Адреса команди
- Такт 2 перевірка готовності
 - Проц. Перевірка сигналу ready
 - Проц. Перевірка сигналів переривання, захоплення
 - Очікування поки немає цих сигналів
- Такт 3
 - Пам'ять. Видає вміст пам'яті на шину даних
 - Пам'ять. Видає сигнал запису
- Аналогічні цикли можуть посторюватись

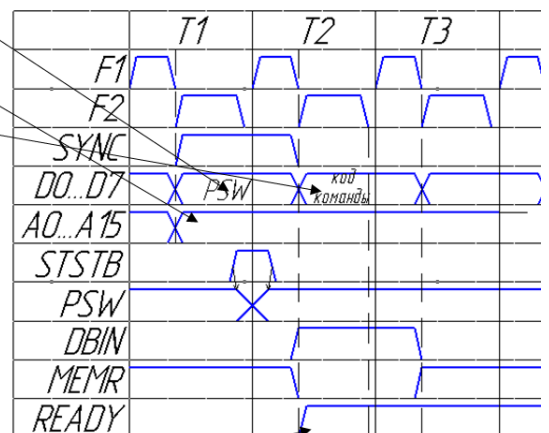
- Такт 1
 - Проц. Сигнал синхронізації
 - Проц. Слово стану на шину даних
 - Проц. Спеціальний біт вказує читання/запис пам'яті
 - Проц. Адреса команди
- Такт 2 перевірка готовності
 - Проц. Перевірка сигналу ready
 - Проц. Перевірка сигналів переривання, захоплення
 - Очікування поки немає цих сигналів
- Такт 3
 - Пам'ять. Видає вміст пам'яті на шину даних
 - Пам'ять. Видає сигнал запису
- Аналогічні цикли можуть посторюватись



[5.7]

Цикл читання порту вводу виводу (IO)

- Відрізняється від циклу читання пам'яті
 - бітами у слові стану
 - На шину адреси видається не адреса пам'яті а номер порта
 - Дані передаються не в пам'ять а ззовнішній пристрій
- Сама команда вводу- виводу одного байта триває декілька циклів
 - Цикл 1 - вибірка команди
 - Цикл 2 – вибірка номера порту
 - Цикл 3 – читання порту вводу виводу
- Пристрій повинен відслідковувати сигнали SYNC, шину даних, шину адреси, видавати сигнал готовності
- Відрізняється від циклу читання пам'яті
 - бітами у слові стану
 - На шину адреси видається не адреса пам'яті а номер порта
 - Дані передаються не в пам'ять а ззовнішній пристрій
- Сама команда вводу- виводу одного байта триває декілька циклів
 - Цикл 1 - вибірка команди
 - Цикл 2 – вибірка номера порту
 - Цикл 3 – читання порту вводу виводу
- Пристрій повинен відслідковувати сигнали SYNC, шину даних, шину адреси, видавати сигнал готовності



[5.8]

- Стан зупинки (stop, halt)
 - Процесор зупиняє роботу
 - При виконанні команди HLT

- Зупиняється в наступному циклі після видачі слова стану, де встановлений спеціальний біт зупинки
- Запустити процесор можна
 - Перериванням
 - Переведенням в стан захоплення
 - Переведенням в стан reset
- Використовується для налагодження програм, економії електроенергії

[5.9]

Стан захоплення (утримання,hold)

- Переведення всіх виводів шин у стан високого вихідного опору
 - Здійснюється в такті перевірки будь-якого циклу
 - Для переведення в hold зовнішній пристрій подає сигнал на вхід hold
 - Після переведення в стан hold процесор на виводі ahold встановлює сигнал підтвердження захоплення
 - Після цього зовнішні пристрої можуть передавати дані по шині
- Застосовується для обміну по шині без участі процесора
 - Прямий доступ до пам'яті (ПДП, Direct Memory Access, DMA)
- Зовнішній пристрій записує/читає дані в пам'ять без участі процесора
 - Швидко виконується ввід-вивід
 - Не споживається енергія процесором

[5.10]

Контролер ПДП

- Спрощує обмін ПДП
- Має декілька каналів
- Кожен канал – буферні регістри
 - Адреса
 - Дані
 - Керування
 - Стан
 - Лічильник
- При запиті на ПДП від пристрою
 - Запис в регістри номер каналу, адресу початку або кінця, кількість байт
 - Запит на ПДП
 - При підтвердженні захоплення шини пристрій передає дані контролеру
 - Контролер передає дані в пам'ять



[5.11]

Стан переривання (interrupt)

- Переривання – процесор перериває виконання програми і виконує запит від зовнішнього пристрою
 - ❑ Процесор повинен дозволити переривання команда IE
 - ❑ Для заборони переривання команда ID
 - ❑ В режимі ПДП переривання заборонені
 - ❑ З режиму STOP виводиться перериванням
 - ❑ Пристрій видає на вивід процесора INT сигнал
 - ❑ Процесор в кінці кожного циклу перевіряє запит на переривання
 - ❑ При наявності запиту на переривання процесор видає сигнал підтвердження на виводі INTA
 - ❑ Процесор чекає на команду на шині даних
 - ❑ При отриманні команди починається обробка переривання

[5.12]

- Після підтвердження переривання процесор зчитує на шині даних код команди, або число – індекс вектора переривань (номер переривання)
 - ❑ Вектор переривань – масив адрес за якими знаходяться підпрограми обробки переривань
 - ❑ Переривання 5 – елемент вектора 5
- Відбувається виклик підпрограми за адресою, яка відповідає номеру переривання у векторі (або виконання команди яку зчитали з шини)
- Після завершення обробки відбувається вихід із підпрограми
- Широко застосовується для обробки запитів від зовнішніх пристроїв
- Можна програмно змінювати код обробника, дозволяти і забороняти переривання

[5.13]

Контролер переривань

- Спрощує взаємодію із зовнішніми пристроями
- Приймає запити від пристроїв по виводам переривань (номер вивода номер переривання)
- Встановлює пріоритетність переривань
 - ❑ Коли обробляється переривання з меншим номером – переривання з більшим номером не обробляється
 - ❑ Таймер - найпріоритетніше

[5.14]

Спеціальні переривання

- Нештатні стани процесора можуть генерувати переривання
 - ❑ Переривання reset
- Немасковане переривання – переривання яке має найвищий пріоритет і забороняється, чи дозволяється окремо
 - ❑ Окремий вивід процесора (Non-Maskable Interrupt, NMI)
 - ❑ Генерується при фатальних помилках (пам'ять, тощо)
- Програмні переривання
 - ❑ Виклик вектора переривань

[5.15]

Шина ISA

- Industrial Standard Architecture
- Використовує порти вводу виводу, переривання, канали ПДП
- Пристрій має
 - ❑ Діапазон портів вводу-виводу IO RANGE
 - ❑ Номер переривання IRQ
 - ❑ Номер каналу DMA

- Діапазон адрес пам'яті DMA
- Лінії (62(ХТ)-98(АТ))
 - Адреса 20(ХТ)-24(АТ) ліній
 - Дані 8(ХТ)-16(АТ) ліній
 - Переривання до 11 ліній
 - Керування
- Швидкість 4-20 МГц
- Зараз використовується рідко в основному для серверів
- Проблема
 - Неможливо спільно використовувати переривання
 - Обмежена швидкість і кількість пристроїв
 - Проблеми з plug-and-play

[5.16]

Шина PCI (Peripheral Components Interconnect)

- Шина незалежна від типу процесора
- Швидкість 33, 66 та 133 МГц 32 та 64 біти
- Автоконфігурація
 - Пристрої визначаються автоматично
- Взаємодія з іншими пристроями та шинами за допомогою мостів
 - «Північний міст» – взаємодія з процесором, пам'яттю, швидкими пристроями
 - «Південний міст» – взаємодія з повільними пристроями вводу виводу
 - PCI-PCI міст
 - PCI-ISA міст
- Мости складні контролери, які забезпечують роботу шини
- Передача по шині без участі процесора
- Можлива паралельна передача даних і роботи процесора

[5.17]

Послідовні шини

- Паралельні шини швидкі, але є проблеми
 - Наводки від сусідніх ліній
 - Необхідність однаковості параметрів ліній
 - Необхідність гарантування синхронності передачі
 - Велика імовірність виходу з ладу
 - Дорого
 - Мала дальність передачі
- Послідовні шини
 - Прості
 - Можна масштабувати швидкість
 - Працюють на великій відстані і великій швидкості
 - Дешеві
- Дуже часто застосовуються
 - UART
 - SPI
 - I²C

[5.18]

UART (Universal Asynchronous Receiver Transceiver)

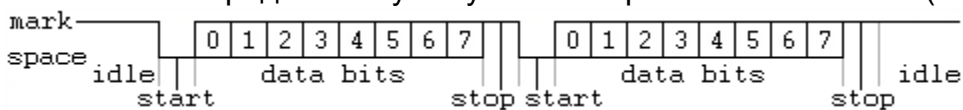
- Протокол передачі послідовною шиною
- Потрібно 2-3 проводи (одно двонаправлена передача)
 - Асинхронний – приймач і передавач не повинні мати спільного тактового генератора
 - Універсальний – підходить для будь-яких даних
 - Може приймати і передавати
- Дані представляються у вигляді потоку бітів (серіалізуються) і назад (десеріалізуються)
- Застосовується для керування пристроями
 - Консолі
 - Модеми
 - Програматори

- Недоліки
 - Повільний (стандарт до ~100 кБайт/с)
 - Великі накладні витрати (передається багато непотрібної інформації)

[5.19]

Передача-прийм

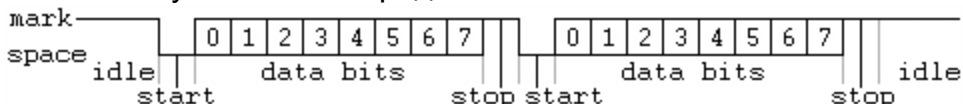
- Дані передаються по одному біту
- Приймач і передавач налаштовуються на однакову швидкість і однакові особливості протоколу передачі
- Біти кодуються рівнем 0- низький 1 –високий
- Передача кожного біту відбувається впродовж однакового фіксованого проміжку часу
 - Час передачі біта = $1/(\text{швидкість передачі})$
- Дані передаються порціями по 1 байту
 - 1байт = 8-9 біт (налаштову)
- Перед порцією даних можуть бути стартові біти (завжди 0)
- Після передачі можуть бути біти парності і стопові біти (завжди 1)



[5.20]

UART – передавач

- В режимі коли нічого не передається передавач видає на лінію високий рівень (1)
- Передача байта починається з передачі стартового біту (0)
 - Передавач перемикається з високого на низький рівень
- Передавач чекає впродовж часу передачі біту
- Передавач перемикає на лінію у високий чи низький стан в залежності від значення бітів даних, які передаються і чекає впродовж часу передачі бітів
 - Так 8 або 9 разів
- Після передачі останнього біта з байта даний передавач передає стоповий біт
 - Перемикає лінію у високий стан і чекає час передачі біту
- Наступний байт передається аналогічно



[5.21]

UART-приймач

- Приймач чекає перепаду з високого на низький рівень
- Коли такий перепад трапляється приймач чекає половину часу передачі біту
 - Якщо все ще нуль (стартовий біт), то приймає далі
- Приймач чекає час передачі біту і зчитує поточне значення біту
 - Повторює так 8-9 разів і записує прийняті дані
- Приймач чекає час передачі біту і перевіряє наявність 1
- Якщо стартовий або стоповий біт неправильні – помилка
- Далі все повторюється



[5.22]

Швидкість передачі

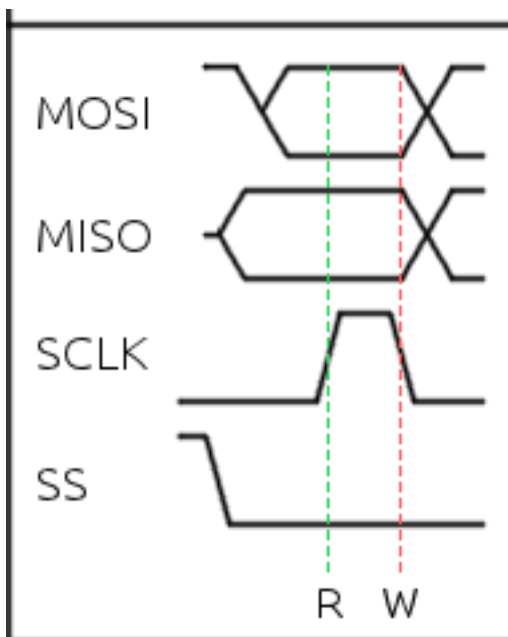
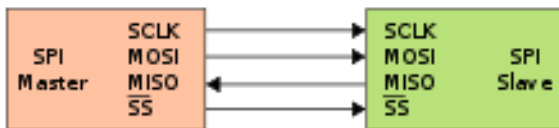
- Приймач і передавач повинні мати тактові генератори з **приблизно** однаковою частотою
 - Точність така, що за період передачі даних не повинна набігти похибка більша, ніж 0.5 часу передачі біта
 - 0.5біт/9.5 біт~5% (реально 1-3 %)

- ❑ Це обмежує максимальну швидкість, бо складніше забезпечити стабільність
- Швидкість передачі (бод, baud rate)
 - ❑ Кількість фізичних бітів за секунду
 - ❑ Швидкість передачі корисних даних менша
- Є стандартні швидкості
 - ❑ 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600
- Є стандарти на рівні напруг і протоколи
 - ❑ RS-232, RS-485, LIN, IrDa

[5.23]

Шина SPI (до 50 МГц)

- 1 Головний - багато ведених
 - ❑ 4 проводи
- Головний видає тактовий сигнал на всі ведені пристрої
- За допомогою виводу SS=0 головний вибирає ті пристрої з якими від взаємодіє
- Передача даних двонаправлена
 - ❑ Головний з виводу MOSI на вивід MISO веденого
 - ❑ Ведений з виводу MISO на вивід MOSI головного
- Виведення даних по спадтактового імпульсу
- Введення даних по фронту тактового імпульсу



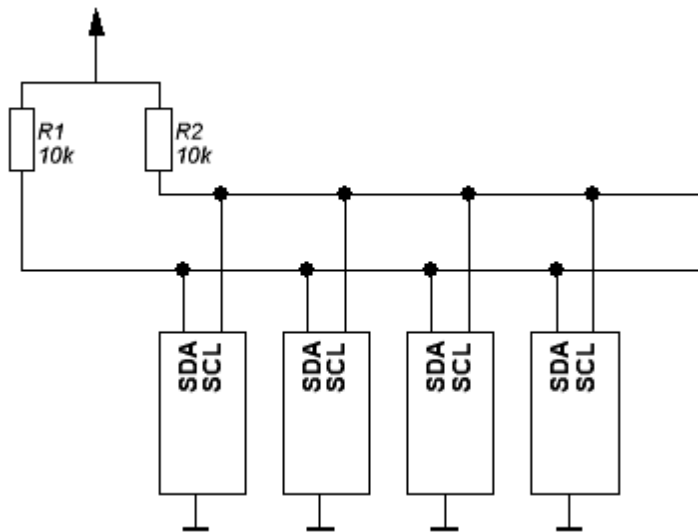
[5.24]

2

IC - Inter-Integrated Circuit (10 кбіт/с-3.4 Мбіт/с)

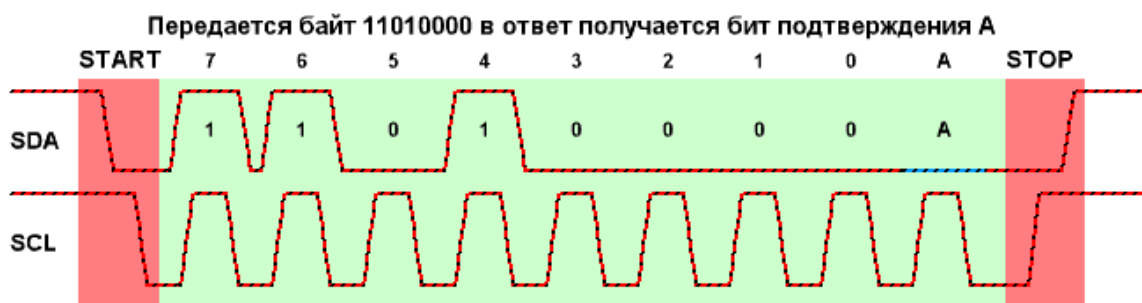
- Багато головних і багато ведених пристроїв
- 3 проводи
 - ❑ SDA – дані
 - ❑ SCL- такт
- Відкритий колектор, або стік
 - ❑ Низький рівень на шині виграє
- Тактовий сигнал видається лише головними, а читається всіма

- Дані видаються всіма, читаються всім
- Арбітраж – коли декілька одночасно
 - Поки всі дані однакові – нормально
 - Відключається той, хто бачить не свої дані



[5.25] 2

- І С – передача -прийом
- Запис на шину – при низькому такті
- Читання шину – при високому такті
- Старт – перехід старту і даних з високого в низьке значення
- Стоп – при високому такті відбувається запис на шину спочатку низького а потім високого значення
- Можна регулювати швидкість змінюючи тривалість нуля на тактовому проводі
- Потужний протокол, широко застосовується



[5.26]

Шина USB Universal Serial Bus

- USB 1.x,2.x,3.x
 - 4 проводу
 - Швидкість 1,5 Мбіт/с- 4,8 Гбіт/с).
 - Струм до 0.5 А
 - Дальність 3-5 м
- Гаряче підключення-відключення
 - Спочатку відключаються два проводу даних, потім два проводу живлення
- Ініціатором завжди є хост – комп'ютер тощо
- Дані передаються способом схожим на UART
 - Викорстовується декілька рівнів сигналу