

ЛЕКЦІЯ 4

Послідовний інтерфейс передачі даних SPI

Характеристика та структура інтерфейса SPI (Serial Peripheral Interface). Послідовний периферійний інтерфейс SPI розроблений, як повнодуплексний чотирипровідний інтерфейс з шинною конфігурацією вузлів (пристроїв), що підключаються, для систем з одним головним вузлом. Первинна базова версія інтерфейсу SPI дозволяє підключати до одного головного (або ведучого - Master) вузла декілька ведених (Slave) вузлів через загальну шину. Окремий сигнал вибору веденого пристрою NSS (Slave Select signal) використовується для вибору веденого пристрою при здійсненні з ним обміну даними. Схема підключення двох МК по інтерфейсу SPI наведена на рис.1.1.

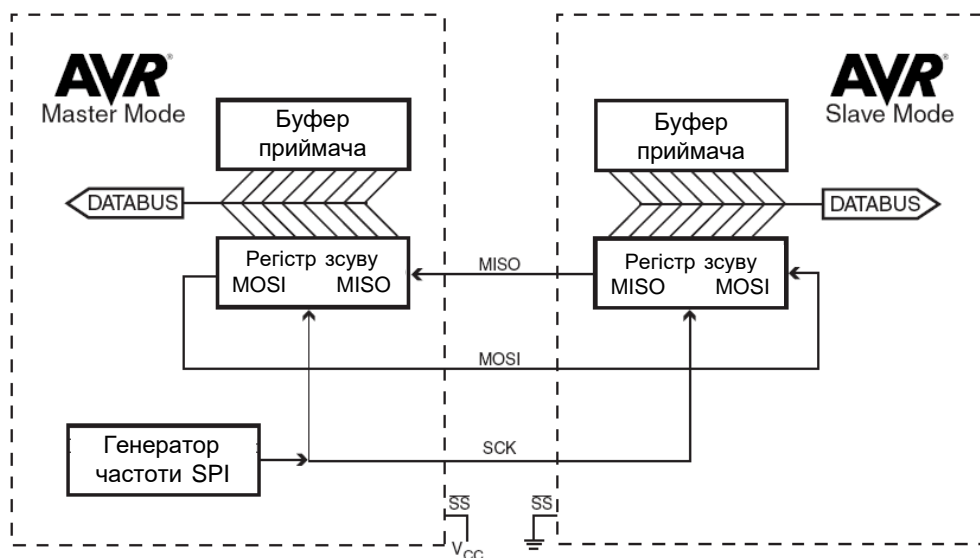


Рисунок 1.1 Схема підключення двох МК по інтерфейсу SPI

Інтерфейс SPI має чотири сигнальні лінії:

- Лінія MOSI (Master-Out, Slave-In) - вихідна лінія даних ведучого інтерфейсу і вхідна лінія даних веденого інтерфейсу. З назви виходить, що лінія призначена для передачі даних від ведучого (Master) інтерфейсу (або вузла мережі) до веденого (Slave) інтерфейсу (або вузлу мережі).
- Лінія MISO (Master-In, Slave-Out) - вхідна лінія даних ведучого інтерфейсу і вихідна лінія даних веденого інтерфейсу. Лінія призначена для

передачі даних від веденого інтерфейсу до ведучого. Дані передаються байтами, побітно, починаючи із старшого біта. Слід пам'ятати, що вивід MISO веденого інтерфейсу знаходиться в стані високого імпедансу, якщо ведений інтерфейс не вибраний по лінії NSS

- Лінія NSS (Slave Select) – (\overline{SS}) лінія вибору веденого.
- Лінія SCK (Serial Clock) - вихідна лінія тактових імпульсів

ведучого вузла і вхідна лінія тактових імпульсів веденого вузла. Лінія SCK використовується для синхронізації передачі даних між ведучим і веденим інтерфейсами по лініях MOSI і MISO.

Діаграми обміну по інтерфейсу SPI (рис 1.2) пояснюють вибір полярності (CPOL) та фази (CPHA) тактових імпульсів.

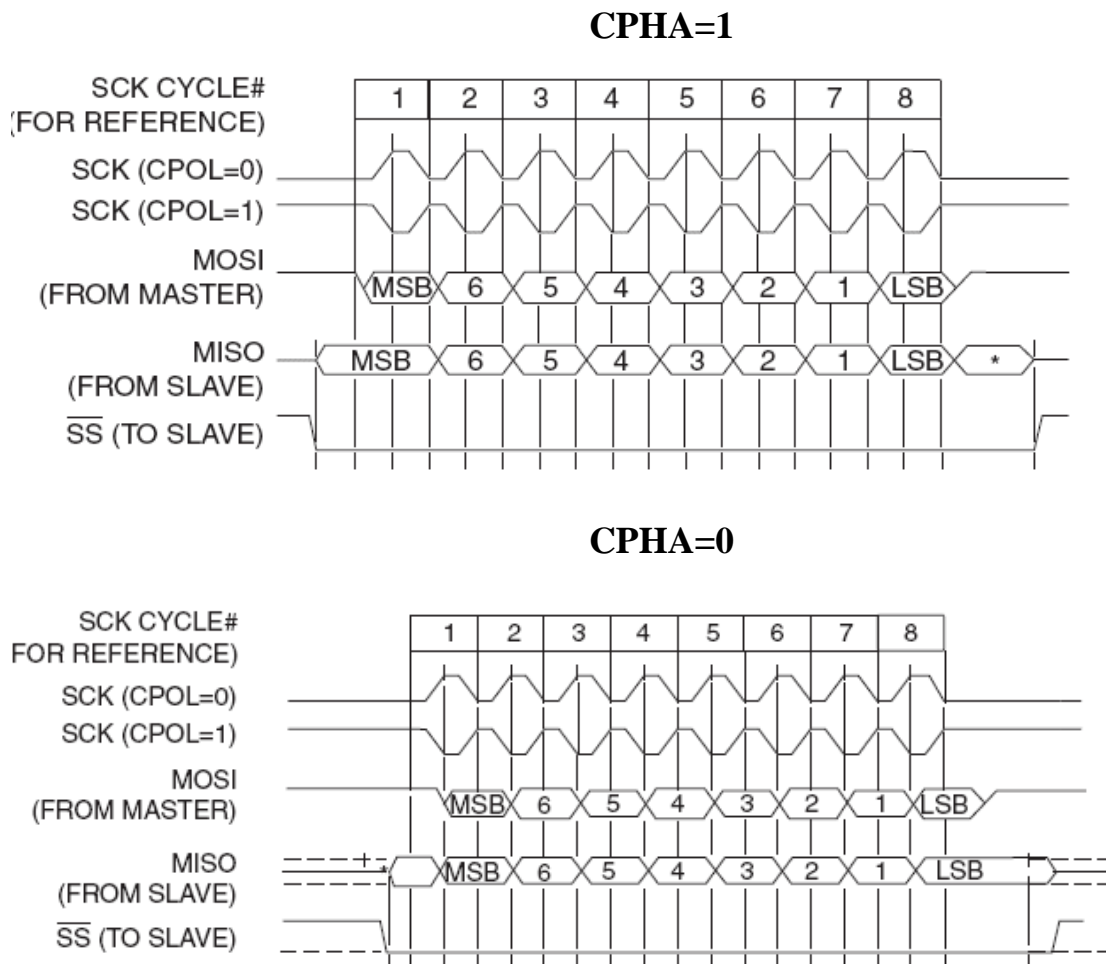


Рис. 1.2 – Діаграми обміну по інтерфейсу SPI, MSB – Most Significant Bit, LSB – Less Significant Bit

Регістри SPI. Зазвичай при апаратній реалізації інтерфейсу SPI, він доступний програмістові через чотири спеціалізованих функцій регістра SFR (Special function registers). Назви цих регістрів можуть трохи відрізнятися для мікроконтролерів різних виробників, проте їх функціональне призначення при цьому залишається незмінним. Надалі використовуватимемо найменування SFR регістрів, вживаних в документації фірми Silicon Laboratories Corp., яка в даний час є лідером в розвитку інтерфейсу SPI.

Інтерфейс SPI містить всього чотири SFR регістра:

1. SPIODAT - регістр даних;
2. SPIOCKR - регістр управління швидкістю;
3. SPIOCFG - регістр конфігурації;
4. SPIOCN - регістр управління шиною SPI.

Перші два регістри, призначення яких виходить з назви, не представляють особливого інтересу при розгляді архітектурних особливостей інтерфейсу.

Зазначимо, що код, записуваний в регістр SPIOCKR, - управління швидкістю, дозволяє визначити швидкість роботи інтерфейсу SPI (частоту тактових імпульсів F SCLK) по наступній формулі:

$$F SCLK = 0.5 * SYSCLK / (SPIOCKR + 1)$$

З приведеної формули виходить, що максимальна тактова частота SPI інтерфейсу може бути рівна половині системної тактової частоти SYSCLK в режимі ведучого (майстера). У режимі веденого швидкість передачі інтерфейсу SPI визначається тактовою частотою ведучого інтерфейсу SPI.

Третій регістр - регістр конфігурації інтерфейсу – SPIOCFG, містить біти СКРНА (SPI Clock Phase) - управління фазою тактування, і СКPOL (SPI Clock Polarity) - управління полярністю тактуючих імпульсів. Ці біти дозволяють вибрати фазу і полярність імпульсів тактування (Див. рис. 1.2).

У базовій версії мережі на базі SPI інтерфейсів лише один інтерфейс може бути ведучим. Інтерфейс встановлюється в режим ведучого установкою прапора MSTEN (Master Enable Flag) - біта SPIOCN.1. Якщо інтерфейс встановлений в режим ведучого, то запис байта даних в регістр даних SPIODAT приводить до початку передачі. Ведучий інтерфейс негайно побітно зсуває дані і видає їх на лінію MOSI у супроводі тактових імпульсів

на лінії SCK. Після завершення передачі встановлюється прапор SPIF (SPIOCN.7). Якщо дозволені переривання, видається відповідне переривання. Крім того, інтерфейс може бути запрограмований на видачу від одного до восьми бітів для здійснення зв'язку по SPI з приладами, що мають різну довжину слова. Довжина передачі (кількість передаваних бітів) може бути задана бітами SPIFRS в регістрі конфігурації SPIOCFG.[2:0] (SPI Configuration Register).

Режими роботи

Інтерфейс може працювати в повнодуплексному режимі, тобто можлива одночасна передача даних по лініях MOSI від ведучого до веденого, і MISO від веденого до ведучого. Дані, отримані від веденого інтерфейсу, замінюють дані в регістрі даних ведучого інтерфейсу. Цей регістр двічі буферизований на введення, але не на виведення. Тобто якщо в регістр даних SPIODAT відбувається спроба запису даних під час передачі попереднього байта, встановлюється прапор WCOL (SPIOCN.6) і спроба запису ігнорується.

Таким чином, поточна передача даних продовжується безперервно. Читання з регістра даних SPIODAT приводить до читання приймального буфера.

Якщо прийом не закінчений, встановлюється прапор RXOVRN (SPIOCN.4). Нові дані не передаються в регістр читання, поки попередній прийнятий байт не буде прочитаний. Вочевидь, що при затримці читання прийнятих байтів може статися втрата даних. Якщо SPI інтерфейс не налагоджений, як ведучий, він працюватиме в режимі веденого.

Не дивлячись на те, що базова версія інтерфейсу SPI розроблена для мікроконтролерних систем з одним ведучим, можливий все ж режим мережі з багатьма ведучими. Прапор MODF (SPIOCN.5 - Mode Fault flag) встановлюється в логічну одиницю, якщо інтерфейс визначений як ведучий (MSTEN=1) і вивід NSS переведений в низький логічний рівень, тобто SPI інтерфейс намагаються використовувати як ведений. Якщо при цьому встановлений прапор MODF, біти MSTEN і SPIEN в регістрі управління SPI скидаються апаратно, переводячи інтерфейс в автономний стан. Таким чином, в системі з багатьма ведучими, ядро може визначити чи вільна шина

шляхом опиту прапора SLVSEL (SPIOCN.2) перед тим, як встановити MSTEN прапор (тобто визначити інтерфейсу режим ведучого) і ініціалізувати обмін.

Окрім первинного базового інтерфейсу в багатьох сучасних мікроконтролерах використовується також і розширений інтерфейс SPI.

У базовому варіанті інтерфейсу SPI регістр конфігурації SPIOCFG містить також біти:

BC2- BCO (SPI Bit Count) – що відображають номер поточного передаваного біта;

SPIFRS2- SPIFRSO (SPI Frame Size), що визначають розмір фрейма (довжини ередаваного слова).

У розширеному інтерфейсі ці біти відсутні, оскільки при сучасних високих швидкостях передачі визначення поточного передаваного біта і зміна формату фрейма втрачає всякий сенс. Замість цих бітів розширений інтерфейс SPI має наступні конфігуруючі біти, що дозволяють оптимізувати роботу систем з багатьма ведучими інтерфейсами:

MSTEN (Master Mode Enable) – біт дозволу режиму ведучого;

NSSIN (NSS Instantaneous Pin Input) – біт, що відображає стан входу NSS у момент читання;

RXBMT (Receive Buffer Empty) – біт, що відображає, що регістр читання порожній;

SLVSEL (Slave Selected Flag) – біт стану веденого;

SPIBSY (SPI Busy) – біт зайнятості інтерфейсу SPI, що в деякій мірі замінює біти SPI Bit Count базового інтерфейсу;

SRMT (Shift Register Empty) - біт, що відображає звільнення регістра зсуву даних.

Приведені біти дозволяють діагностувати інтерфейс при шинній організації системи з багатьма ведучими.

Четвертий регістр SPIOCN - регістр управління інтерфейсу SPI. Він має ряд бітів управління, загальних як для базової, так і для розширеної версій:

- MODF (Mode Fault Flag) - прапор помилки режиму, що відображає невідповідність призначеного режиму (біта ведучого MSTEN=1) і стану входу NSS=0;
- RXOVRN (Receive Overrun Flag) – біт, що відображає випадок, якщо розпочато прийом наступного байта при непрочитаному попередньому байті, тобто біт відображає розсинхронізацію читання;
- WCOL (Write Collision Flag) - прапор розсинхронізації запису, що встановлюється в разі, якщо здійснена спроба запису байта в регістр під час незавершеної передачі попереднього байта;
- SPIF (SPI Interrupt Flag) - прапор переривання інтерфейсу SPI, що встановлюється апаратно після завершення передачі;
- SPIEN (SPI Enable) - біт дозволу роботи інтерфейсу SPI.
- Окрім цього, базовий інтерфейс має ряд бітів, які в розширеному інтерфейсі були переведені в регістр конфігурації SPI0CFG:
- TXBSY (Transmit Busy Flag) - біт зайнятості, - еквівалентний новому біту SPIBSY (SPI Busy); а також біти SLVSEL і MSTEN.

Розширений інтерфейс SPI має наступні конфігуруючі біти:

NSSMD1- NSSMD 0 (Slave Mode Select) – біти вибору режиму веденого інтерфейсу: 3-хпровідний режим ведучого або веденого; 4-х провідний режим веденого або режим з багатьма ведучими (NSS – завжди в режимі входу); 4-х провідний режим одного ведучого.

Таким чином, очевидна основна відмінність розширеного інтерфейсу SPI від базового. У розширеному варіанті передбачена робота шини SPI в багатоконтролерному режимі, що виразилося в розширенні засобів діагностики помилок і можливості управління сигналом NSS.

Класична архітектура SPI шини для базового варіанту інтерфейсу приведена на рис.1.3. Це так звана класична 4х – провідна структура. Один ведучий в ній управляє декількома (N) веденими. Всі ведені підключені паралельно на лініях SCLK, MISO і MOSI шини SPI. Вибірка одного з

ведених відбувається за допомогою однієї з ліній портів введення /виведення, яка з'єднується з входом NSS відповідного веденого. Зазвичай така архітектура використовується для побудови мікроконтролерних систем з одним мікроконтролером, що виконує роль ведучого і рядом периферійних мікросхем, що виконують роль ведених. Як периферійні мікросхеми може бути використаний ряд сучасних мікросхем, оснащених інтерфейсом SPI: таймерів реального часу RTC, аналого-цифрових перетворювачів ADC, цифро-аналогових перетворювачів DAC, різних мікросхем пам'яті і тому подібне

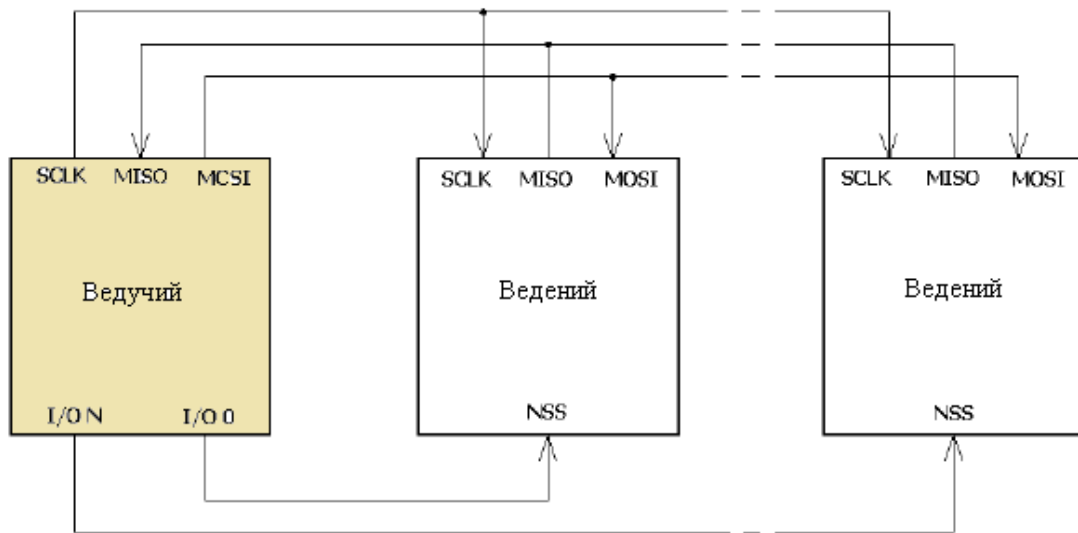


Рисунок 1.3 – Архітектура 4х-провідної SPI шини з одним ведучим і декількома веденими

Окремим випадком попередньої архітектури є так звана 3х-провідна структура, приведена на рис.1.4. Її відмінність полягає в тому, що ведена мікросхема постійно вибрана, і отже, необхідність в четвертій лінії інтерфейсу SPI відсутня.

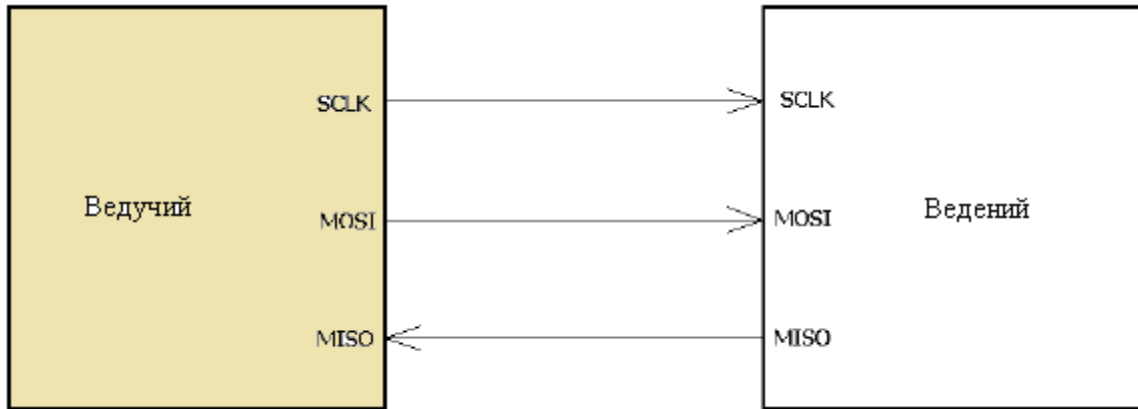


Рисунок 1.4 – Архітектура 3х-провідної шини SPI з ведучим і веденим.

Розширений інтерфейс дозволяє організувати двопроцесорну 4х- провідну архітектуру, приведена на рис.1.5. В цьому випадку кожен з двох вузлів може виконувати роль ведучого. На практиці така архітектура використовується досить рідко, зважаючи на алгоритмічну складність діагностики станів колізій (конфліктів, ведучих вузлів). Ширше використовується простіша в розумінні і діагностиці конфліктів, так звана архітектура з 4х-провідною «перехресною» вибіркою, приведена на рис.1.6. В цьому випадку для вибірки кожного з вузлів використовується лінія введення/виведення іншого вузла, що працює лише на вивід, а не на введення/виведення, як в попередній структурі.

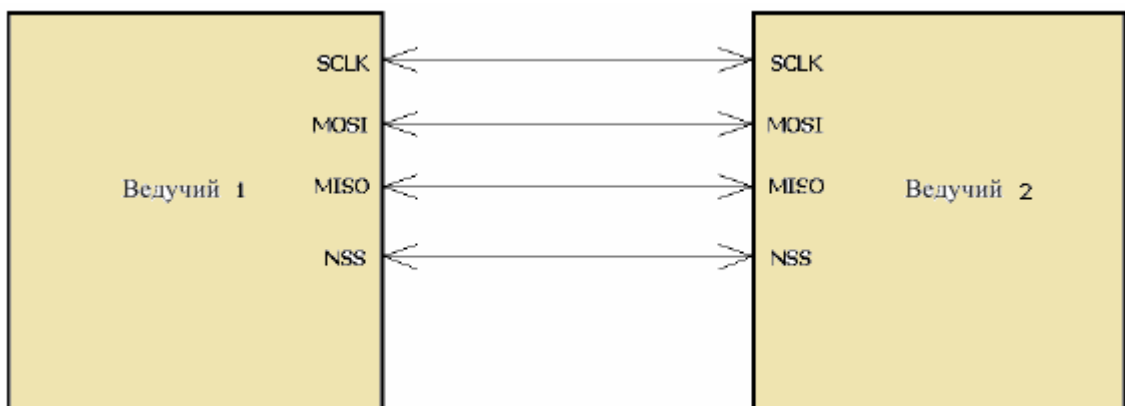


Рисунок 1.5 – Архітектура 4х-провідної SPI шини з двома ведучими

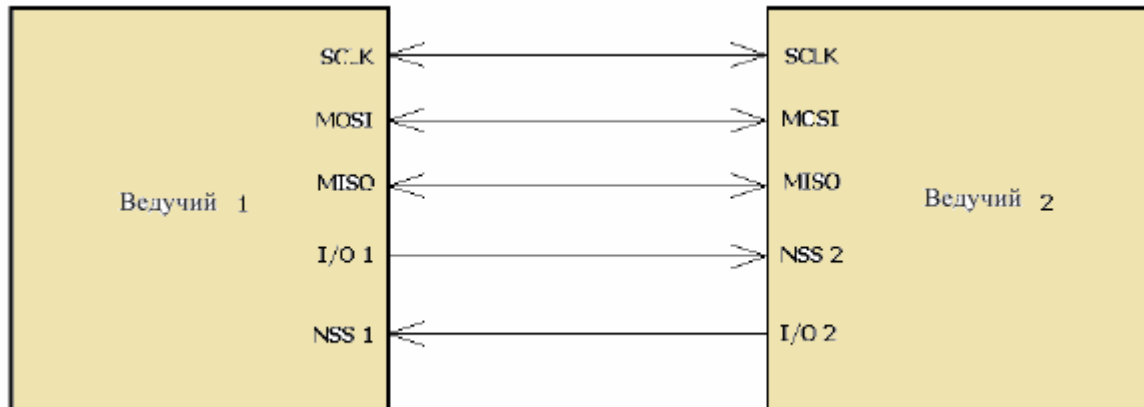


Рисунок 1.6 – Архітектура 4х-провідної SPI шини з двома ведучими і «перехресною» вибіркою

Відмітимо, що двопроцесорна архітектура використовується досить рідко, наприклад, в разі, якщо один з мікроконтролерів призначений для введення і обробки сигналів, а другий – для організації інтерфейсу з персональним комп'ютером або локальною мережею. В цьому випадку і один, і інший мікроконтролери повинні мати можливість ініціювати процедуру обміну даними, а отже, повинні мати можливість функціонування в режимі ведучого. Більш поширений випадок використання двопроцесорної структури з ресурсом, що розділяється, наприклад пам'яттю, наведений на рис.1.7.

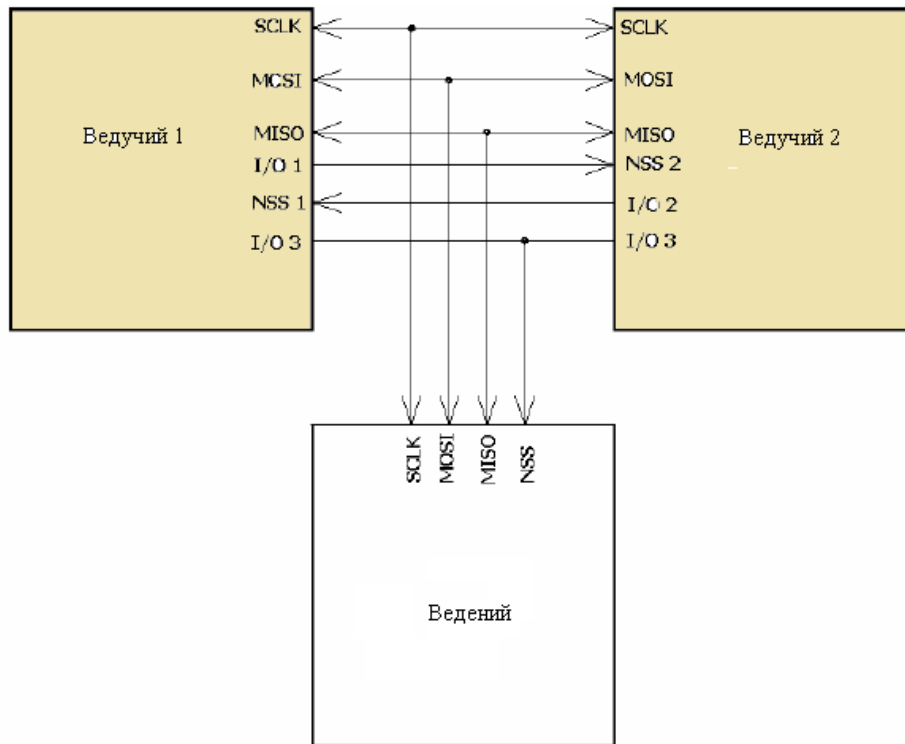


Рисунок 1.7 – Двопроцесорна архітектура з ресурсом, що розділяється

У приведеній на рис.1.7 архітектурі є два ведучих і один ведений. Зв'язки між ведучими мінімізовані і відповідають архітектурі 4x-проводної SPI шини з двома ведучими і «перехресною» вибіркою. Лінії основних сигналів SPI шини підключені також до загального веденого, а для вибірки загального веденого використовується по одній додатковій лінії введення/виведення від кожного з ведучих, об'єднаних за «монтажним АБО». Архітектура в принципі працездатна, проте аналіз зайнятості SPI шини і діагностики колізій в ній ще більш утруднений. Це пояснюється необхідністю послідовного опиту як мінімум двох ліній вибірки (індивідуальною контролера, що ініціює обмін, і загального ресурсу) перед заняттям шини, а отже і збільшенням вірогідності виникнення колізії. Для спрощення процедур обміну рекомендується ввести ще дві лінії прапорів запиту шини ID1 і ID2, як показано на рис.1.8.

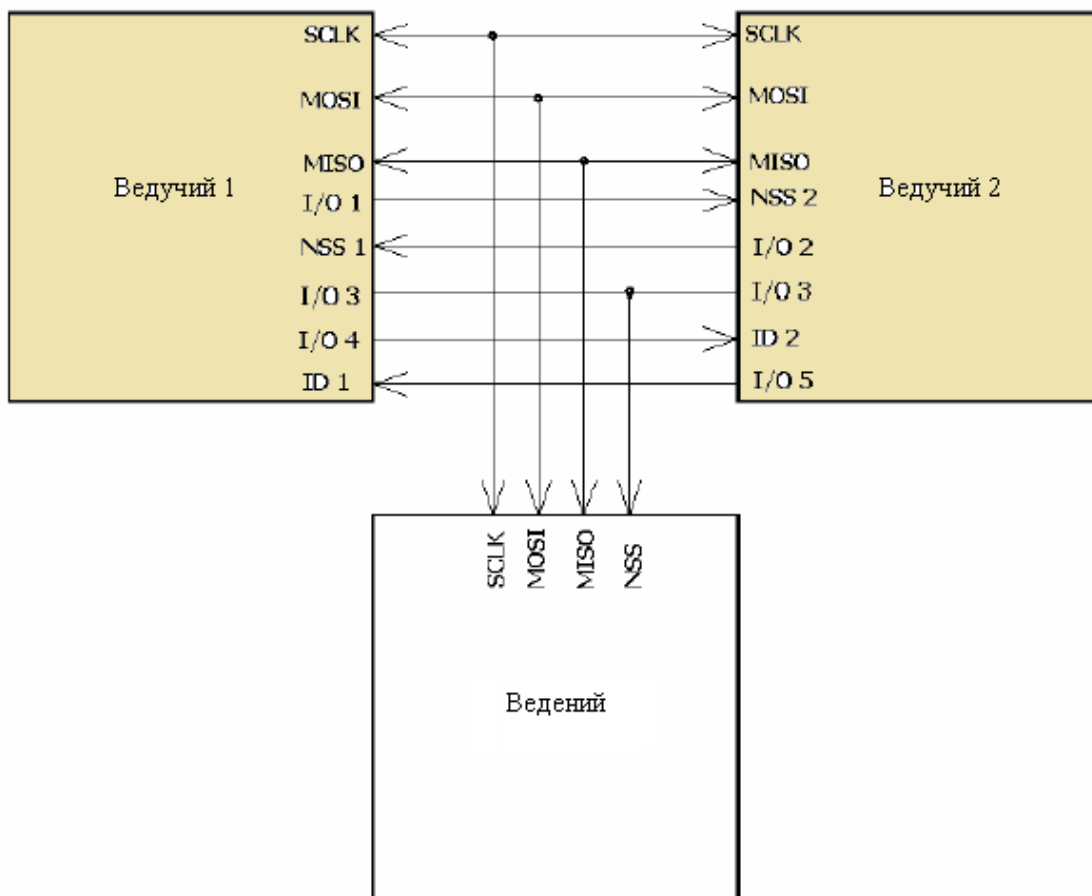


Рисунок 1.8 – Двопроцесорна архітектура з розділеним ресурсом і лініями запиту шини

В цьому випадку при необхідності заняття шини контролер виставляє свій прапор запиту шини, і чекає у відповідь прапора від другого контролера. Другий контролер, отримавши прапор запиту, або відразу виставляє свій прапор, підтверджуючи, що шина вільна, або закінчує поточні операції обміну і тільки тоді виставляє свій прапор. Такий механізм значно спрощує діагностику зайнятості шини.

Також відзначимо як мінімум два варіанти використання останніх двох архітектур. Варіант використання – двопроцесорна архітектура, що містить перший швидкодіючий мікроконтролер і другий мікроконтролер з інтерфейсом USB для зв'язку з персональним комп'ютером. Як загальний ресурс, що розділяється, використовується DataFlash пам'ять великого

об'єму для накопичення даних, наприклад AT45DB642. Перший мікроконтролер вимірює за допомогою вбудованих аналого-цифрових перетворювачів ADC дані з датчиків об'єкту спостереження або управління, обробляє їх і записує в DataFlash пам'ять, а другий контролер за певних умов передає накопичені дані в персональний комп'ютер по інтерфейсу USB.

Приведена архітектура необхідна лише тоді, поки перший мікроконтролер (надшвидкодійний або з високоточним ADC) не буде оснащений власним інтерфейсом USB. Такі випадки зустрічаються дуже рідко, та потреба в такій задачі все ж таки може виникнути.

Контрольні питання

1. Що таке SPI?
2. Який тип інтерфейсу SPI?
3. Які застосування може мати інтерфейс SPI?
4. Які існують варіанти підключення SPI?
5. Яка швидкість інтерфейсу SPI?
6. Скільки ліній використовується в інтерфейсі SPI? Перелічіть їх.
7. Що визначає параметр CPOL?
8. Що визначає параметр CPHA?
9. Чи може в одній системі бути лише два ведучих? Чому?
10. Чи може в одній системі бути лише два ведених? Чому?