

Лабораторна робота №5

Тема: Вивчення приймачів-передавачів асинхронного та синхронного обміну мікроконтролерів AVR

Асинхронна передача даних через приймач-передавач UART

Універсальний асинхронний приймач-передавач UART відсутній в мікроконтролері AT90S1200.

При синхронній послідовній передачі даних синхронізується передача окремих бітів даних за допомогою одночасно передаваного передавачем тактового сигналу. При цьому особливих вимог до такого тактового сигналу відносно синхронізації не пред'являється. Так, наприклад, при використанні декількох синхронних протоколів передачі такт для пристосування до повільніших пристроїв, підключених до шини, розтягується за часом.

Синхронна послідовна передача даних застосовується, головним чином, на рівні друкарської платні, зокрема — для обміну даними між різними інтегрованими блоками у складі схеми мікроконтролера і різними периферійними схемами.

При асинхронній передачі даних тактовий сигнал не передається. Це висуває строгі вимоги до розподілу інтервалів часу в каналах передачі і прийому. З цієї причини тимчасова розгортка для систем передачі даних, що працюють в асинхронному режимі, в більшості випадків має кварцеву стабілізацію.

Головною областю застосування асинхронної передачі даних, як правило, є не обмін даними у складі схеми, а комунікація між блоками, розділеними просторово і мають ознаки власного інтелекту. Наприклад, зв'язок між персональним комп'ютером і принтером, модемом, програмуємим пристроєм або реєстратором даних. Оскільки асинхронна передача даних може здійснюватися на великі відстані аж до декількох сотень метрів, її протоколи не передбачають рівнів TTL, дуже схильних до перешкод.

Стандарти асинхронної передачі даних

З'єднання двох пристроїв при асинхронній передачі даних здійснюється через замкнутий приймально-передавальний контур. На активній стороні в контурі протікає постійний струм силою 20 мА. Для передачі сигналів низького рівня підвід струму уривається, а при передачі сигналу високого рівня контур залишається замкнутим (швидкість 9600 бод).

У разі струмових інтерфейсів можна легко реалізувати розділення потенціалів між передавальною і приймаючою стороною через оптоелектронний пристрій, як це показано на рис.5.1, тому вони придатні для передачі даних на відстань до 1000 м і знаходять широке застосування в промисловості.

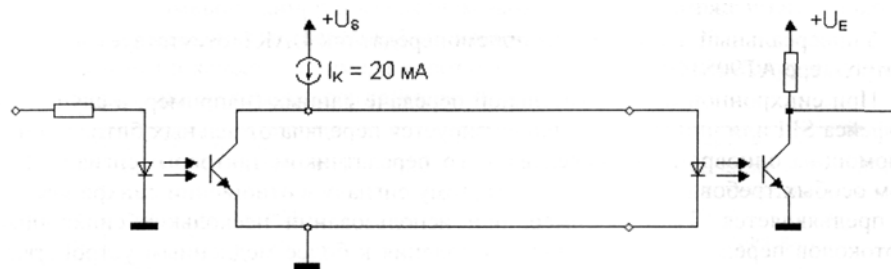


Рис. 5.1. Приклад передавального контуру для струмового інтерфейсу

Вже застарілі, але ще широко поширені інтерфейси RS232C або V.24 передають сигнали низького і високого рівня як значення напруги. При цьому на стороні приймача високому рівню TTL сигналу (+2...+5 В) виділений діапазон напруг -3...-15 В ("Відмітка"), а низькому рівню TTL сигналу (0...+0,8 В) — діапазон напруг +3...+15 В ("Пропуск"). На стороні передавача нижня межа для компенсації втрат напруги в лінії піднята +5 В і, відповідно -5 В. На рис.5.2 показано рівень сигналу, що допускається згідно нормам.

Для перетворення рівня від стандарту TTL в стандарт V.24 можна придбати цілий ряд інтегральних схем. Для формування з'єднання стандарт також передбачає деякі керуючі сигнали.

Проте вони також не повинні залишатися непідключеними в схемі, оскільки це може привести до помилкових інтерпретацій в протоколі.

У простому випадку для комунікації між двома пристроями досить мати просте з'єднання за допомогою трьох провідників: схрещені передавальний і приймаючий проводи TxD і RxD, а також провід заземлення. Не задіяні виводи можуть бути зайняті згідно рис.5.3.

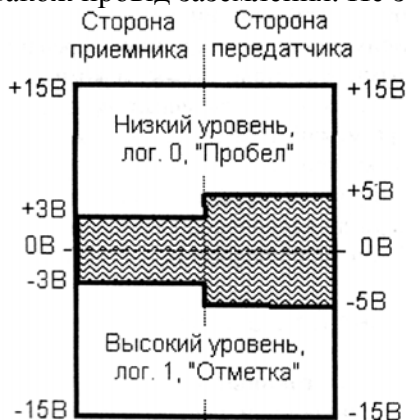


Рис.5.2. Рівень напруги відповідно до стандарту RS232C

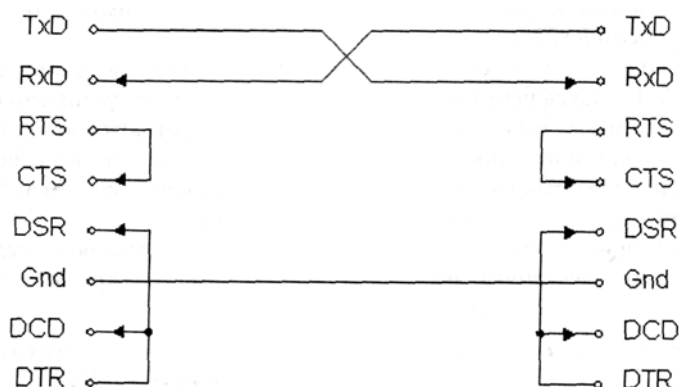


Рис. 5.3. Просте трипровідне з'єднання між двома приладами

На практиці, на інтерфейсі RS232C неможливо досягти швидкості передачі даних вище 19200 бод, а через гальванічне з'єднання і виникаючі порушень в роботі, обумовлених струмами перехідних процесів в дроті заземлення, передачу даних на відстані понад 15-20 метрів здійснити неможливо, хоча стандарт і визначає допустиму межу в 30,5 м.

Вимоги до дальності і швидкості передачі даних призвели до розробки нових стандартів асинхронної передачі даних. За допомогою інтерфейсів RS423A або V.11 можливі швидкості передачі даних до 100000 бод і дальності до 1200 метрів. У разі інтерфейсу RS423A, передавач і приймач сполучені коаксіальним кабелем опором 50 Ом з хвильовим опором Z (рис.5.4).

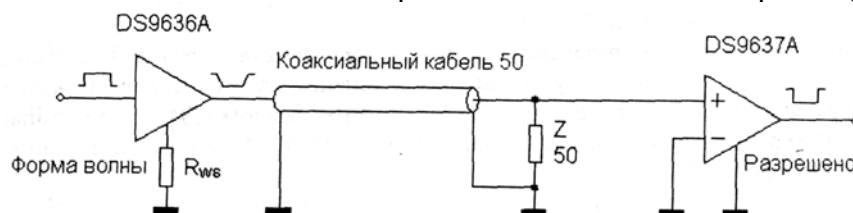


Рис.5.4. Передача через коаксіальний кабель відповідно до вимог стандарту RS423A

Інтерфейс RS423A, так само як і інтерфейс RS232C, працює асиметрично, тобто, провід заземлення служить як зворотний. З цієї причини в середовищі з високим рівнем перешкод застосовувати цей інтерфейс не доцільно, проте, якщо немає необхідності передачі даних на дуже великій відстані, то схема, показана на рис.5.4, може дати відчутні переваги в порівнянні з інтерфейсом RS232C.

Ще більше переваг має інтерфейс, відповідний стандартам RS422A або V.10. Тут передача даних відбувається симетрично, тобто, драйвер передавача має вихідний мостовий каскад, через який одночасно передається сигнал і інвертований сигнал. Інформація тут також представлена не у вигляді абсолютного значення напруги між виходом і "землею", а як різниця між обома вихідними напругами.

Завдяки застосуванню виті пари, досягається висока перешкодостійкість, оскільки розсіянні імпульси перешкод взаємно усувають один одного (рис. 5.5).

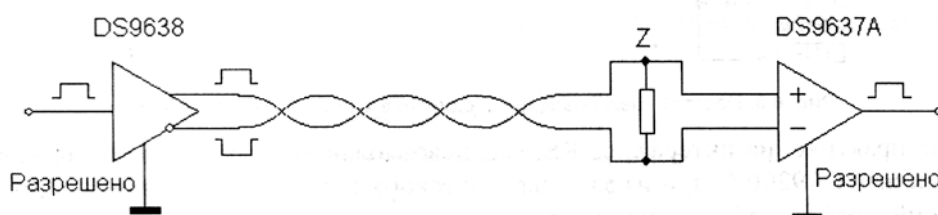


Рис.5.5. Передача сигналів за допомогою виті пари через інтерфейс RS422A

Наприклад, якщо на виході драйвера напруга +3 В, та напруга на його інвертуючому виході на другому проводі, відповідно, буде -3 В. В результаті різниця двох вихідних напруг складає 6 В. Коли обидві напруги із-за імпульсів перешкод підвищуються на +1 В, то перший дріт має потенціал +4 В по відношенню до "землі", а другий - 2 В, унаслідок чого різниця напруг як і раніше складає 6 В. Завдяки такому методу, можливі швидкості передачі даних аж до 10 Мбод при дальності до 1200 м.

Можливі швидкості передачі даних залежно від необхідних відстаней передачі для всіх розглянутих інтерфейсів показані на рис.5.6. Порівняння параметрів різних інтерфейсів показано у табл. 5.1.

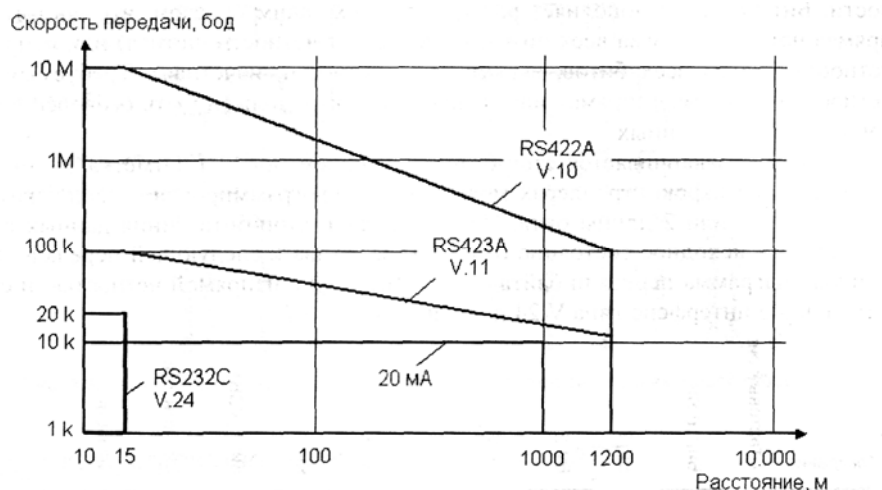


Рис.5.6. Допустимі значення швидкості передачі даних залежно від відстані для асинхронних інтерфейсів різних стандартів

Таблица 5.1

Порівняння основних параметрів інтерфейсів розглянутих стандартів

Параметр	20 мА	RS232C	RS423A	RS422A
Вид передачі	Симетрична	Асиметрична	Асиметрична	Симетрична
Вид проводу	Витий	Витий	Коаксіальний	Витий
Максимальна швидкість передачі даних, бод	10 К	20 К	100 К	10 М
Максимальна довжина проводу	1000 м	15м	1200 м	1200 м
Максимальний вихідний сигнал, без навантаження	20 мА	±25 В	±6 В	Різниця ± 6 В
Вихідний сигнал, з навантаженням	20 мА	±5 В	±3,6 В	Різниця ± 2 В
Чутливість входу	10 мА	±3 В	±0,2 В	Різниця ± 0,2 В

Формат передачі по асинхронному інтерфейсу

Відповідно до визначення стандарту V.24, в лінії передачі даних в стані очікування встановлена лог.1 ("Відмітка"). Передача може бути почата у будь-який момент часу. Для того, щоб передати приймачу повідомлення про початок передачі, посиляється стартовий біт з рівнем лог.0 ("Пропуск"). Після цього слідує розряди даних (в більшості випадків 7 або 8, їх число обмовляється в протоколі передачі даних), при цьому спочатку передається молодший значущий біт.

Для підвищення надійності передачі даних може бути доданий біт парності. Біт парності доповнює розряди даних таким чином, що досягається пряма парність (сума всіх бітів, включаючи біт парності, парна) або непряма парність (сума всіх бітів, включаючи біт парності, непарна). Необхідний вид парності (відсутність, пряма або непряма) також повинна бути обумовлена в протоколі передачі даних.

Передача закінчується стоп-бітом з рівнем лог.1 ("Відмітка"), який в деяких мікроконтролерах може бути запрограмований з тривалістю 1; 1,5 або 2 довжини біта. Після відправки стоп-біта лінія даних знову переходить в початковий стан очікування і готова до наступної передачі. Тимчасова діаграма передачі байта \$E5 (11100101) з непрямою парністю і стоп-бітом через інтерфейс типу V.24 показана на рис.5.7.

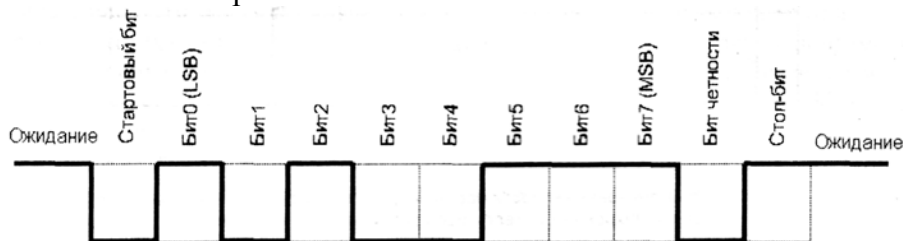


Рис.5.7. Часова діаграма передачі байта \$E5 (11100101) з непрямою парністю і стоп-бітом через інтерфейс типу V.24

У зв'язку з тим, що такт для синхронізації передачі не пересилається, початок розряду може бути розпізнаний тільки по вимірюванню часу, що пройшов з моменту появи спадаючого фронту стартового біта. Саме тому необхідно обумовлювати швидкості передачі даних, які є обов'язковими для передавальної і приймаючої сторони. Ці швидкості визначені в стандарті по інтерфейсу RS232C, і щоб уникнути помилок при розпізнаванні окремих розрядів, їх необхідно в точності дотримуватися.

На рис.5.8 показані наслідки дуже великої відмінності між тактами передачі і прийому, а також випадку, коли передача здійснюється по наростаючому фронту такту, а опит на стороні приймача — по спадаючому фронту такту, який може потрапляти в середину імпульсу даних.

На рис.5.8 показаний випадок, коли розряд 4 і розряд 7, а також біт парності розпізнані неправильно. Оскільки в даному прикладі обумовлена непарність (непряма парність), то помилка може бути відмічена на підставі біта парності, оскільки байт, розпізнаний як \$75, і встановлений біт парності дають в результаті пряму парність, але у разі виникнення двох помилок біт парності про це не просигналізував.

Відмінність в тактах на передавальній і приймаючій стороні в розглянутому прикладі складає 20%, що неприпустимо багато. Таке значення було вибрано з тією метою, щоб чітко показати ефект його впливу, проте щоб уникнути помилок і для забезпечення надійної передачі даних слід обов'язково уникати відхилень, що перевищують 3%. Точність передачі гарантована, якщо її фактична швидкість не відхиляється від вказаних в стандарті значень більш, чим на 2%.

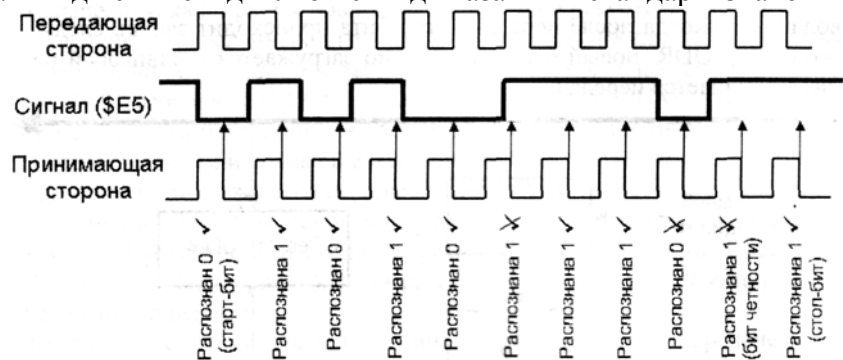


Рис.5.8. Вплив різних тактів передачі на передавальній і приймаючій стороні

Фізичний пристрій приймач-передавача UART

Асинхронну передачу даних можна реалізувати за допомогою програмного забезпечення, але це займатиме досить багато ресурсів центрального процесора, і тому всі представники базової серії сімейства AVR (за винятком моделі AT90S1200) оснащені апаратним приймач-передавачем UART. Він може працювати в дуплексному режимі і, завдяки цьому, в змозі одночасно передавати і приймати дані, тим самим розвантажуючи центральний процесор.

Для роботи UART виділені чотири регістри у області вводу/виводу. Разом з регістром управління UCR (UART Control Register), призначеного для управління функціями передавача приймача і для дозволу/заборони переривань UART, використовується також регістр стану USR

(UART Status Register); регістр даних UDR (UART Data Register), що фізично складається з двох регістрів (звернення до цих регістрів здійснюється окремо по тій же адресі, причому один з них використовується для передачі, а інший — для прийому даних), а також регістр UBRR (UART Baud Rate Register) для настройки необхідної швидкості передачі даних за допомогою вбудованого контролера, що дозволяє встановлювати найбільш поширені швидкості передачі за стандартом RS232C.

Універсальний асинхронний приймач-передавач UART мікроконтролерів AVR може автоматично розпізнавати два види помилок передачі даних і викликати три різні переривання.

Передавальний елемент UART (трансмідер)

Схема передавального елементу приймач-передавача UART показана на рис. 5.9. Передача даних починається із запису призначеного для передачі байта в регістр вводу/виводу UDR. Коли після передачі стоп-біта відбувається запис наступного байта в регістр UDR, новий байт негайно завантажується в регістр зсуву, після чого починається передача.

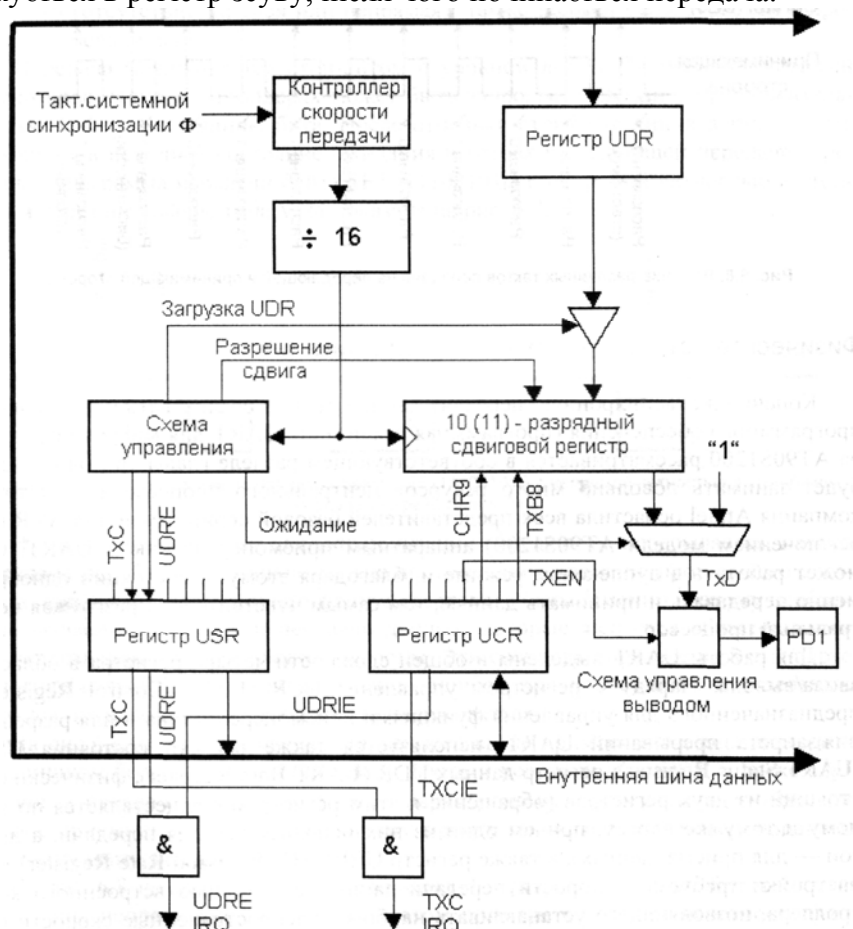


Рис. 5.9. Блок-схема передавального елементу (трансмідера) приймач-передавача UART

Якщо в регістр UDR в процесі передачі записується новий байт, то він може бути завантажений у регістр зсуву, а передача його може початися за допомогою стоп-біта тільки після закінчення поточної передачі даних. Як тільки байт з регістра UDR переноситься в регістр зсуву передачі, в регістрі стану USR встановлюється флаг UDRE (UART Data Register Empty — регістр даних передавача приймача порожній), що указує на готовність передавача приймача до прийому нового байта даних в регістр UDR.

Загальне число переданих бітів може складати 10 або 11, залежно від того, яка довжина слова даних була встановлена за допомогою розряду CHR9 регістра UCR: 8 розрядів ($\text{CHR9} = \text{лог. } 0$) або 9 розрядів ($\text{CHR9} = \text{лог. } 1$). Дев'ятий розряд може бути застосований для запису додаткових інформаційних даних, як біт парності або як ще один стоп-біт.

При прийомі в розряди 1-8 регістра зсуву біт 0 (старт-біт) цього регістра автоматичне стирається, і в тому випадку, якщо $\text{CHR9} = \text{лог. } 1$ і встановлений стоп-біт (розряд 9 при $\text{CHR9} =$

лог. 0 або розряд 10 при $\text{CHR9} = \text{лог. 1}$), розряд TXB8 з регістра управління UCR копіюється в розряд 9 регістра зсуву.

Разом з наступним тактовим імпульсом призначений для передачі байт з регістра UDR переноситься в регістр зсуву, біт 0 регістра зсуву як стартовий біт переноситься на вивід TxD, після чого слідує 8/9 розрядів даних (спочатку молодший розряд) і стоп-біт. Якщо під час поточної передачі даних в регістр UDR завантажується новий байт, то цей байт відразу ж після передачі стоп-біта передається в регістр зсуву, і починається нова передача. Одночасно з передачею цього байта в регістр зсуву в регістрі стану USR встановлюється флаг UDRE, який вказує на те, що передавальний елемент приймач-передавача UART знову готовий до прийому нового символу в регістр UDR.

Якщо після передачі стоп-біта в регістрі UDR не опиниться нового слова даних для передачі, то флаг UDRE, встановлений при передачі попереднього байта з регістра UDR в регістр зсуву, міститиме лог. 1 до тих пір, поки в регістр UDR не буде знову записаний байт. Після цього флаг UDRE скидається. Після пересилки стоп-біта в регістрі стану USR встановлюється флаг TXC (UART Transmit Complete — передача через UART завершена), який вказує на те, що слово даних було передано, і дані, які очікують передачі, відсутні.

За допомогою розряду TXEN регістра управління UCR робота трансмітера може бути заблокована ($\text{TXEN} = \text{лог. 0}$) або розблокована ($\text{TXEN} = \text{лог. 1}$). Якщо трансмітер заблокований, то вивід PD1 може бути використаний як загальний вхід/вихід. Якщо він розблокований, то вихід регістра зсуву буде сполучений з висновком PD1, незважаючи на настройку DDD1 у регістрі наряду передачі даних DDRD.

Приймальний елемент приймач-передавача UART (ресивер)

Блок-схема приймаючого елементу асинхронного приймач-передавача UART показана на рис. 5.10. Схема управління прийомом опитує вхідний сигнал 16 разів під час кожного періоду передачі даних. Якщо в чекаючому режимі на лінії прийому розпізнається рівень лог. 0, це розцінюється як спадаючий фронт стартового імпульсу, і починається послідовність дій для розпізнавання дійсного стартового біта.

До 8-го (а також до 9-го і до 10-го) моменту часу опиту (вважаючи від моменту, коли був розпізнаний спадаючий фронт) лінія прийому повинна містити низький рівень сигналу. Якщо при цих трьох скануваннях два або три рази буде розпізнаний високий рівень, то стартовий біт буде відхилений як імпульс перешкоди, і схема управління прийомом знову починає пошук спадаючого фронту стартового біта.

Якщо буде виявлений дійсний стартовий біт, то послідовно один за іншим розряди даних будуть записані в регістр зсуву. Як і у випадку стартового біта, тут діє рівень сигналу на лінії прийому аж до 8-го, а також 9-го і 10-го моменту опиту, вважаючи від початку розряду. Логічний рівень, який розпізнається, як мінімум, при двох опитах з трьох, буде сприйнятий як значення біта. На рис. 5.11 показаний процес опиту для прийому символу, що складається з 8 біт (без біта парності).

Для стоп-біта як мінімум два з трьох опитів повинні дати в результаті лог.1. Якщо цього не відбувається, то в регістрі стану USR встановлюється флаг помилки, який вказує на те, що стоп-біт символу був розпізнаний як некоректний. Програма перед читанням регістра UDR повинна постійно перевіряти флаг FE для того, щоб розпізнати потенційно недійсний символ в регістрі прийому.

Після закінчення циклу прийому даних в регістрі стану USR завжди встановлюється флаг RXC, і виконується завантаження прочитаного символу в регістр прийому UDR, незалежно від того, був розпізнаний коректний або недійсний стоп-біт. Якщо за допомогою розряду CHR9 у регістрі управління UCR було задано довжину слова даних 9 біт ($\text{CHR9} = \text{лог. 1}$), то цей дев'ятий біт переноситься в розряд RXB8 регістра UCR.

Якщо в регістр зсуву поступає нове слово даних ще до того, як буде лічений вже прийнятий і байт що знаходиться в регістрі UDR, то новий символ не може бути перенесений в регістр UDR і втрачається. В цьому випадку в регістрі USR для індикації переповнювання встановлюється флаг OR. Цей флаг буферизований. Це означає, що після прочитування дійсно вірного си-

мволу він оновлюється відповідно до вмісту регістра UDR. Саме тому програма користувача після читання регістра UDR повинна постійно перевіряти флаг OR, щоб розпізнати втрату одного символу, що поступив.

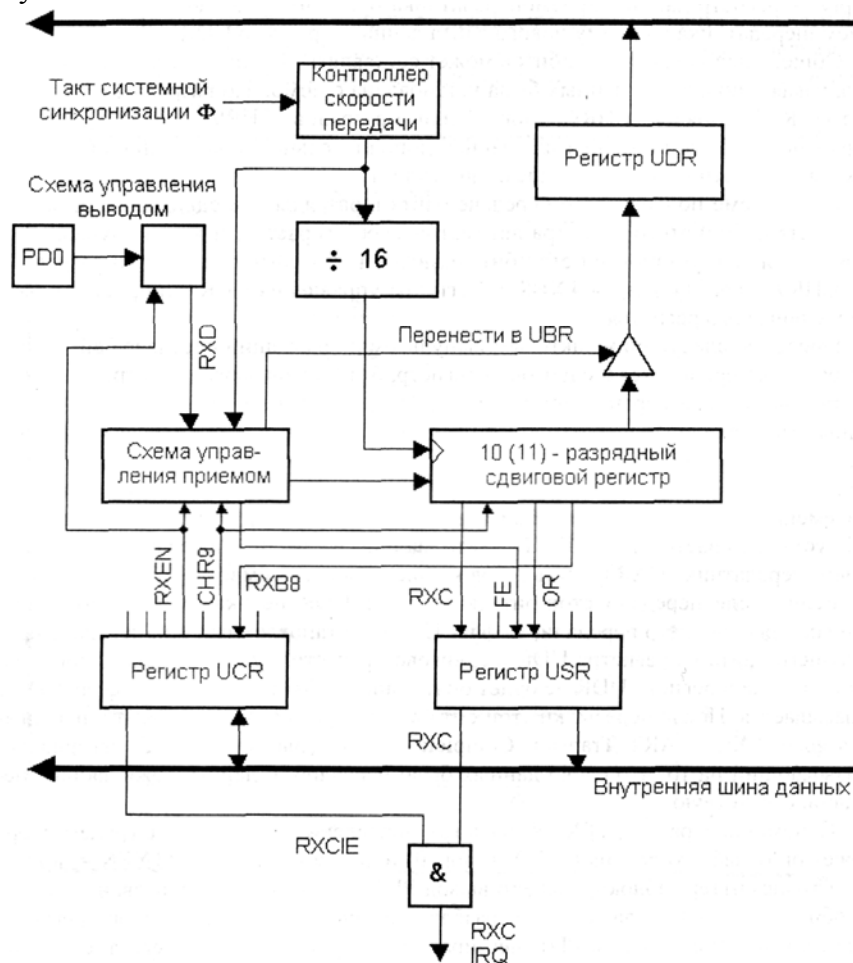


Рис. 5.10. Блок-схема приймаючого елементу приймач-передавача UART

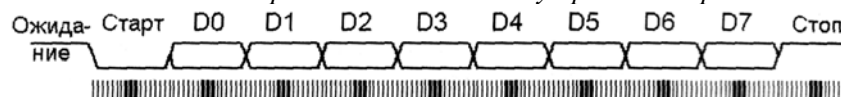


Рис. 5.11. Процес опиту для прийому символу, що складається з 8 біт

За допомогою розряду RXEN регістра управління UCR приймаючий елемент приймач-передавача UART може бути заблокований (RXEN = лог. 0) або розблокований (RXEN = лог. 1). Якщо приймальний елемент передавача приймача заблокований, то вивід PD0 може застосовуватися як загальний вхід/вихід. Якщо ж він розблокований, то вхід приймаючого елементу передавача приймача буде сполучений з виводом PD0, незважаючи на настройку DDD0 у регістрі напряму передачі даних DDRD.

Регістри UART

Регістр вводу/виводу даних UDR

Цей регістр приймач-передавача UART фізично складається з двох регістрів, звернення до яких відбувається поодиночці і тій же адресі, причому один з них використовується для передачі, а інший — для прийому даних. При читанні з UDR походить звернення до вхідного регістра, а при записі — до передавального регістра. Регістр UDR розташовується у області вводу/виводу за адресою \$0C. Він доступний для читання і запису.

Разряд	7	6	5	4	3	2	1	0	
\$0C (\$2C)	MSB							LSB	UDR

Регістр стану USR

Регістр стану USR приймач-передавача UART знаходиться у області вводу/виводу за адресою \$0B. Після надходження сигналу скидання всі розряди аж до розряду 5 (UDRE) ініціалі-

зуються з допомогою лог. 0. Розряд UDRE встановлюється в лог. 1 для того, щоб вказати, що передавач приймача готовий до передачі нового байта даних.

Розряд	7	6	5	4	3	2	1	0	
\$0B (\$2B)	RXC	TXC	UDRE	FE	OR	—	—	—	USR

Регістр стану USR інформує програму користувача про стан приймач-передавача UART.

У мікроконтролерах базової серії сімейства AVR використовуються тільки розряди 3-7 регістра USR. За винятком флага TXC, вони доступні тільки для читання. Розряди 0-2 зарезервовані і доступні тільки для читання (завжди містять лог. 0).

Флаг RXC (UART Receive Complete – прийом завершений) встановлюється в лог. 1 без урахування можливих помилок, які могли виникнути під час передачі даних, якщо прийняте слово даних було перенесено з регістра зсуву в регістр UDR.

В тому випадку, якщо в регістрі стану встановлений розряд I загального дозволу переривань, а в регістрі управління UCR встановлений розряд RXCIE, те виконання програми розгалужується за допомогою флага RXC мікроконтролера, встановленого за адресою UART Receive Complete (\$009 в AT90S8515 і AT90S4414, \$007 в AT90S2313). Флаг RXC встановлюється в початковий стан за допомогою читання регістра UDR.

Якщо призначена для користувача програма прочитуватиме свої дані через переривання UART Receive Complete, то протягом виконання підпрограми обслуговування переривання обов'язково повинен бути лічений регістр UDR. Інакше флаг RXC залишається встановленим, а підпрограма обробки переривання буде викликана повторно відразу ж після виходу.

Флаг TXC (UART Transmit Complete— передача передавачем приймача завершена) буде встановлений в лог. 1, якщо символ в сдвиговому регістрі був переданий повністю (тобто, включаючи стоп-біт), і з регістра UDR не очікується новий байт даних. Флаг дуже корисний в напівдуплексному режимі роботи, коли безпосередньо після передачі необхідно перемкнутися в режим прийому. При вході в підпрограму обробки переривання флаг завершення передачі TXC скидається апаратно. Альтернативно, флаг TXC також може бути скинутий записом лог. 1 в розряд 6 регістра USR.

Флаг UDRE (UART Data Register Empty — регістр даних передавача приймача порожній) встановлюється в лог. 1, якщо вміст регістра UDR було перенесено в регістр зсуву. З його допомогою користувач одержує повідомлення про те, що передавач приймача готовий до передачі нового символу. Під час скидання при включенні живлення флаг UDRE встановлюється в лог. 1, щоб показати, що передавач приймача готовий до передачі нового байта даних.

Флаг FE (Framing Error) встановлюється в лог. 1 при виявленні помилки. Це відбувається, якщо при трьох скануваннях стоп-біта була більше одного разу виявлена лог. 0, і тим самим стоп-біт був розпізнаний як сигнал низького рівня. Флаг FE скидається, коли стоп-біту відповідає сигнал високого рівня. Призначена для користувача програма повинна постійно перевіряти флаг FE перед читанням регістра UDR, щоб можна було розпізнати потенційно некоректний символ в регістрі прийому.

Флаг OR (Over Run — перевантаження) встановлюється в лог. 1, якщо один з символів, переданих в регістр UDR з регістра зсуву, не був прочитаний перед наступним символом. Цей флаг буферизований (тобто, він оновлюється після прочитування дійсного символу з регістра UDR), тому призначена для користувача програма повинна завжди перевіряти флаг OR після читання регістра UDR, щоб розпізнати втрату одного символу, що поступив. Флаг перевантаження скидається при перенесенні ліченого символу в регістр UDR.

Регістр управління UCR

Регістр управління UCR знаходиться у області вводу/виводу за адресою \$0A. Розряди 2-7 доступні для читання і запису, розряд 1 — тільки для читання, а розряд 0 — тільки для запису.

Розряд	7	6	5	4	3	2	1	0	
\$0A (\$2A)	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8	UCR

Якщо розряд RXCIE (RX Complete Interrupt Enable — дозвіл переривання по завершенню прийому) і розряд загального дозволу переривань I в регістрі стану SREG встановлені в лог. 1,

то дозволяється переривання по завершенню прийому UART Receive Complete. Цей процес починається після установки флага завершення прийому RXC в регістрі USR.

Якщо розряд TXCIE (TX Complete Interrupt Enable — дозвіл переривань по завершенню передачі) і розряд загального дозволу переривань I в регістрі стану SREG встановлені в лог. 1, то дозволяється переривання по завершенню передачі UART Transmit Complete. Цей процес починається після установки в регістрі USR флага завершення передачі TXC.

Якщо розряд UDRIE (UART Data Register Empty Interrupt Enable — дозвіл переривання по спустошенню регістра даних передавача приймача) і розряд загального дозволу переривань I в регістрі стану SREG встановлені в лог. 1, то вирішується переривання по спустошенню регістра даних приймач-передавача UART. Цей процес починається після установки в регістрі USR флага UDRE.

Якщо розряд RXEN (Receiver Enable — дозвіл прийому) встановлений в лог. 1, то відбувається розблокування приймача, і вивід PD0 стає входом UART. Якщо розряд RXEN містить низький рівень, то що приймальний елемент приймач-передавача UART блокується, і PD0 може використовуватися як звичайний вхід/вихід.

Якщо розряд RXEN містить лог. 0, то флаги OR і FE не можуть бути встановлені. Якщо ці флаги все ж таки встановлені, то вони за допомогою RXEN не скидаються.

Якщо розряд TXEN (Transmitter Enable — дозвіл передачі) встановлений в лог. 1, то відбувається розблокування передавача, а вивід PD1 стає виходом приймач-передавача UART. Якщо розряд TXEN містить низький рівень, то передавальний елемент UART блокується, і PD1 може використовуватися як звичайний вхід/вихід.

Якщо розряд TXEN під час процесу передачі встановлюється в лог. 0, то передавач не блокується до тих пір, поки поточний символ у регістрі зсуву, а також символ очікуючий на передачу в регістрі UDR не будуть повністю передані.

Якщо розряд CHR9 (9-Bit Characters — символ завдовжки 9 біт) встановлений в лог. 1, то слова даних, що підлягають передачі/прочитуванню мають довжину 11 біт (9 розрядів даних плюс стартовий і стоп-біт). Дев'ятий біт з розряду TXB8 при передачі потрапляє в UCR, а при прийомі — в розряд RXB8 регістра UCR. Дев'ятий біт може бути використаний для розміщення додаткових інформаційних даних, наприклад, як біт парності або другого стоп-біта.

Якщо розряд CHR9 встановлений у лог. 0, то слова даних, що підлягають передачі/зчитуванню мають довжину 10 біт (8 розрядів даних плюс стартовий і стоп-біт).

Розряд RXB8 — це дев'ятий біт даних ліченого символу, якщо розряд CHR9 у регістрі управління UCR містить лог. 1.

Розряд TXB8 — це дев'ятий біт символу, що підлягає зчитуванню, коли розряд CHR9 у регістрі управління UCR встановлений в лог. 1.

Регістр швидкості передачі даних UBRR

Регістр швидкості передачі даних UBRR приймач-передавача UART знаходиться у області вводу/виводу за адресою \$09. Він доступний для читання і запису.

Разряд	7	6	5	4	3	2	1	0	
\$09 (\$29)	MSB							LSB	UBRR

Вбудований контроллер швидкості передачі даних є дільником частоти, що визначає швидкість передачі даних безпосередньо на підставі такту системної синхронізації Φ .

Швидкість передачі може бути обчислена по наступному рівнянню:

$$f_{Baud} = \frac{\Phi}{16(UBRR + 1)}$$

де f_{Baud} — швидкість передачі в бодах, Φ — такт системної синхронізації; UBRR — вміст 8-розрядного регістра UBRR (0...255)

Синхронна передача даних через послідовний інтерфейс (SPI)

Послідовний інтерфейс периферійних пристроїв SPI (Serial Peripheral Interface) використовується тільки в мікроконтролерах AT90S4414 і AT90S8515.

У мікро контролерах AVR існує дві принципові можливості послідовного обміну даними:

- обмін даними за допомогою приймач-передавача UART, який підтримує режим асинхронної передачі;
- застосування синхронного послідовного інтерфейсу периферійних пристроїв SPI.

Інтерфейс SPI служить, в першу чергу, для комунікації мікроконтролера з периферійними блоками. Такими блоки можуть бути регістри зсуву або буквено-цифрові модулі індикації, а також складні мікропроцесорні системи або системи реєстрації даних.

Інтерфейс такого типу також застосовується при програмуванні в послідовному режимі центральних процесорів мікроконтролерів базової серії сімейства AVR. Через інтерфейс типу SPI можна дуже швидко і просто обмінюватися даними між провідним мікроконтролером (Master) і одним або декількома відомими блоками (Slave). Будова інтерфейсу SPI схематично показана на рис. 5.12.

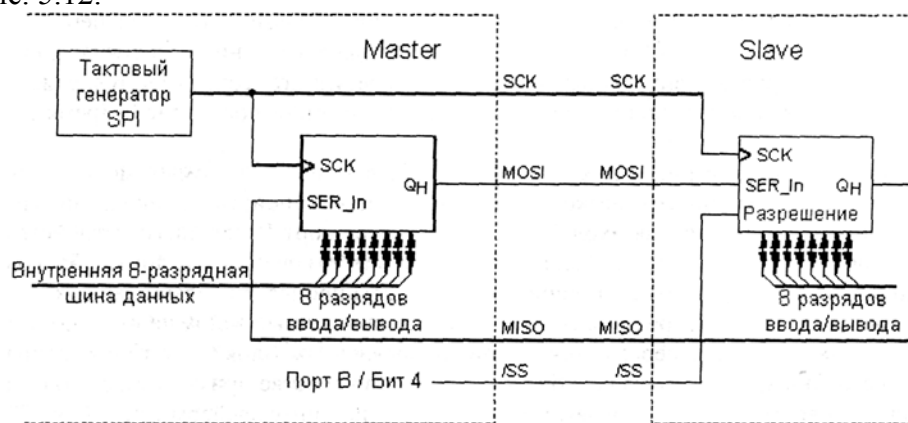


Рис. 5.12. Схематичне представлення послідовного інтерфейсу SPI

При передачі даних через інтерфейс SPI обмін даними завжди відбувається між двома пристроями: Master і Slave. Кожен мікроконтролер сімейства AVR, оснащений інтерфейсом SPI, може бути конфігурований на режими роботи як Master, так і Slave за допомогою установки/скидання розряду MSTR в регістрі управління інтерфейсу SPI. При цьому Master бере на себе активну частину обміну даними, викликаючи передачу і управляючи процесом. Пристрій (Slave) не може бути активним. Воно приймає і передає дані тільки тоді, коли відбувається його активізація із сторони провідного пристрою по лінії /SS (Slave Select — вибір відомого). Це можна порівняти з сигналом /CS (Chip Select — вибір кристала) при використанні блоків пам'яті RAM або EPROM. Ведучий пристрій також генерує такт для передачі по вихідній лінії SCK. Для відомого блоку вивід SCK є входом, через який він одержує від Master-пристрою такт.

Якщо Slave активізується ведучим пристроєм по лінії /SS, то починається обмін даними: Master записує належний передачі байт в свій регістр даних SPDR послідовного інтерфейсу SPI. За допомогою кожного виробленого тактового імпульсу Master переміщає один біт даних на вихід MOSI (Master Out Slave In — вихід ведучого, вхід відомого), а Slave одночасно у зворотному напрямі передає один біт на вхід MISO (Master In Slave Out — вхід ведучого, вихід відомого) провідного блоку. З цієї причини інтерфейс SPI можна порівнювати з 16-ступінчастим регістром циклічного зрушення, розділеним на два 8-розрядних регістри зсуву, перший з яких знаходиться у провідному пристрої, а другий — у відомому. В результаті, протягом циклу SPI, що складається з восьми тактових імпульсів, Master і Slave обмінюються байтами даних.

Після закінчення передачі даних в регістрі стану інтерфейсу SPI (як у конфігурації Master, так і в конфігурації Slave) встановлюється флаг переривання від інтерфейсу SPIF (SPI Interrupt Flag). Цей флаг указує на закінчення передачі і викликає запит на переривання після того, як в регістрі управління SPCR (SPI Control Register) буде встановлений розряд SPIE (SPI Interrupt Enable — дозвіл на переривання від інтерфейсу SPI), а в регістрі стану SREG — розряд I загального дозволу переривань. У режимі "Master" поточна передача даних може бути передчасно

завершена видачею в лінію /SS сигналу лог. 1. Лічильник розрядів і внутрішня логіка Slave в результаті скидаються, режим "Slave" стає неактивним і переходить у високоомний стан (з трьома станами).

Як показано на рис. 5.13, до інтерфейсу SPI як відомі можуть бути одночасно підключені декілька периферійних пристроїв, проте активним буде тільки той з них, на вхід /SS якого через порт В провідного пристрою буде поданий рівень лог. 0. Виходи MISO незадіяних відомих блоків знаходяться у високоомному стані і не впливають на процес передачі даних.

Блокам G1 і Gn на рис. 5.13 відповідають повноцінні відомі інтерфейси SPI, у яких дані передаються в обох напрямках. Блок G2, з погляду провідного блоку, є тільки блоком видачі (як приклад такого блоку можна привести цифро-аналоговий перетворювач з інтерфейсом SPI). Блок G3 по виконанню аналогічний блоку G2, але тільки як блок прийому (наприклад, цифро-аналоговий перетворювач).

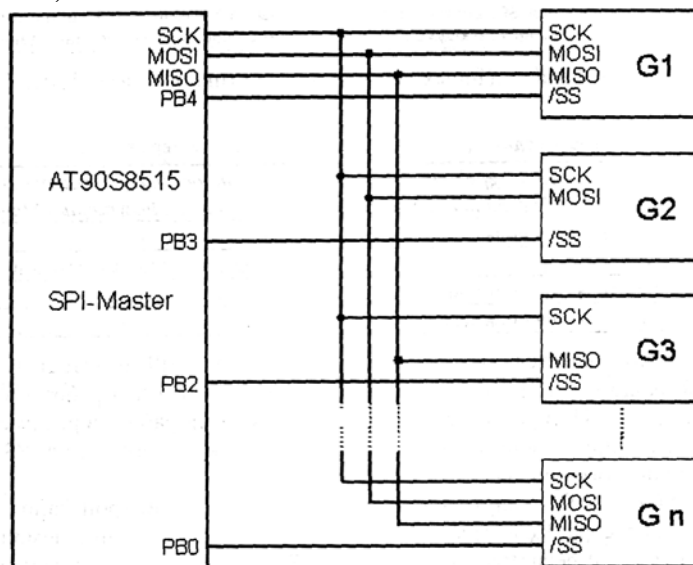


Рис. 5.13. Підключення декількох пристроїв до інтерфейсу SPI

Входи і виходи інтерфейсу SPI

Інтерфейс SPI складається з чотирьох ліній: такту зрушення SCK, ліній передачі даних MOSI і MISO, а також входу розблокування /SS, низького рівня, що активізується сигналом. Вони виконані у вигляді альтернативних функцій порту В (табл. 5.2).

Таблиця 5.2

Функції розрядів порту В для ліній інтерфейсу SPI

Лінія	Вивід	Функція
/SS	PortB / Розряд 4	Дозвіл (Slave Select)
MOSI	PortB / Розряд 5	Лінія даних (Master Out Slave In)
MISO	PortB / Розряд 6	Лінія даних (Master In Slave Out)
SCK	PortB / Розряд 7	Такт зрушення (Shift Clock)

Коли інтерфейс SPI відключений (розряд SPE в регістрі інтерфейсу SPCR містить лог.0), то ці чотири лінії функціонують як звичайні лінії вводу/виводу, стан яких визначається за станом регістра напряму передачі даних DDRB порту В. Проте, якщо інтерфейс SPI активний (розряд SPE в регістрі управління SPCR містить лог. 1), то в загальному випадку діє наступне правило.

Напрямок передачі даних через вивід активного інтерфейсу SPI показаний у табл. 5.3.

Таблиця 5.3.

Напрямок передачі даних через вивід активного інтерфейсу SPI

Вивід	Напрямок передачі при конфігурації "Master"	Напрямок передачі при конфігурації "Slave"
MOSI	Визначається користувачем	Вхід
MISO	Вхід	Визначається користувачем
SCK	Визначається користувачем	Вхід
/SS	Визначається користувачем	Вхід

Лінія SCK є виходом, коли інтерфейс SPI працює в режимі "Master", і входом, коли інтерфейс SPI конфігурований для роботи в режимі "Slave" (розряд MSTR в регістрі управління SPCR). При роботі в режимі "Slave" інтерфейс ігнорує свій вхід SCK до тих пір, поки на його лінії /SS не з'явиться сигнал низького рівня.

Лінія SCK ділиться внутрішнім тактом системної синхронізації Φ центрального процесора. Коефіцієнт ділення може бути вибраний за допомогою розрядів SPR1 і SPR0 регістра управління SPCR. Коли Master починає передачу даних, то генеруються вісім тактових імпульсів, які синхронізують передачу. Щоб гарантувати надійну постановку біта даних, він зрушується вперед за допомогою одного фронту тактового сигналу SCK, а за допомогою іншого фронту приймається (рис. 5.14). Вибір фронту сигналу залежить від протоколу передачі даних.

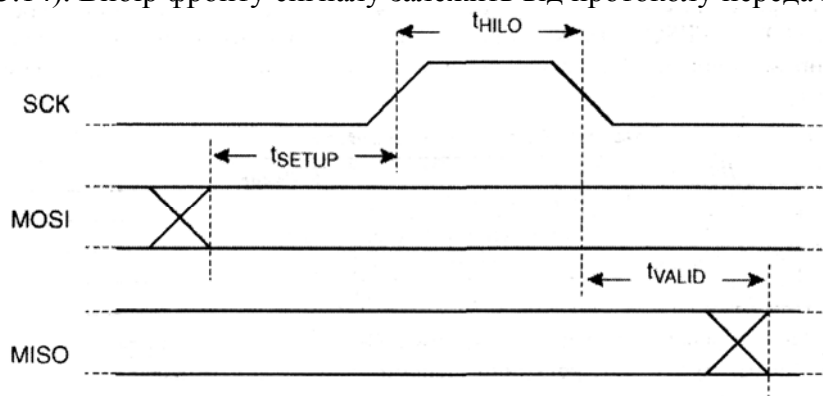


Рис. 5.14. Часова діаграма передачі по інтерфейсу SPI

На рис. 5.14 біти даних приймаються по наростаючому фронту тактового сигналу SCK, а подальше зрушення здійснюється за допомогою спадаючого фронту. Master передає біт даних на свій вихід MOSI, і лише після часу t_{SETUP} слідує наростаючий фронт тактового сигналу SCK, щоб гарантувати, що біт після прийому стабільно встановлений.

Після того, як з'явився спадаючий фронт тактового сигналу SCK, Master чекає період часу t_{VALID} , щоб переконатися в наявності біта даних від Slave, який можна було б прийняти за допомогою наступного фронту тактового сигналу SCK. Тривалість часу t_{HILO} імпульсу високого рівня тактового сигналу SCK, а також періодів часу t_{SETUP} і t_{VALID} залежить від вимог, периферійних пристроїв. Для того, щоб можна було підключати також і більш повільні периферійні пристрої, частота тактового сигналу SCK може бути поділена до $\Phi/128$.

Виводи MISO і MOSI служать для передачі і прийому послідовних бітів даних. На стороні Master MOSI – це вихід даних, а MISO – вхід. На стороні Slave ці два виводи міняються ролями. Як показано на рис. 5.13, до інтерфейсу типу SPI може бути підключено декілька відомих блоків. В цьому випадку всі лінії SCK сполучені між собою, так само як і всі лінії MOSI і MISO. Не дивлячись на те, що по інтерфейсу SPI, у принципі, можуть бути сполучені декілька пристроїв, активним може бути тільки одне з них. Цей активний Master вибирає по лінії /SS необхідний блок Slave і пересилає через його виходи SCK і MOSI тактові імпульси і біти даних на входи SCK і MOSI відомих блоків. Протягом того ж циклу Slave може пересилати біти даних через свій вихід MISO на вхід MISO провідного блоку. Завдяки автоматичному перемикачню напрямку передачі даних на цих виводах (табл. 5.3), не виникає конфлікт при зміні провідного пристрою в системах з декількома такими пристроями.

Призначення лінії /SS може мінятися, залежно від того, чи визначений інтерфейс SPI як Master або як Slave. Відомий блок деблокується ведучим для передачі даних з допомогою лог. 0 на виводі /SS, а його вивід MISO перетворюється на вихід, як тільки призначена для користувача програма встановить в лог. 1 розряд DDB6 у регістрі напряду передачі даних порту B. Тоді на виводі /SS знаходиться високий рівень, Slave ігнорує свої входи SCK і MOSI, а потім переводить вихід MISO у високоомний стан.

При роботі в режимі "Master" вивід /SS може бути конфігурований як звичайний вихід або як вхід для розпізнавання конфліктів в інтерфейсі SPI. Цей вибір визначається розрядом DDB4 у регістрі напряду передачі даних порту B. Якщо цей розряд містить лог. 1, то лінія функціонує як загальний вихід, який не впливає на інтерфейс SPI. Якщо розряд DDB4 у регістрі напряду

передачі даних встановлений в лог. 0 (вхід), то лінія /SS повинна утримуватися в лог. 1, якщо система SPI повинна коректно працювати як Master.

Щоб уникнути серйозних пошкоджень вихідного драйвера, апаратно виконуються наступні дії.

1. Розряд MSTR в регістрі управління SPCR встановлюється в лог. 0, і тим самим система інтерфейсу SPI конфігурується на роботу в режимі відомого блоку. Як наслідок, вивід MOSI і SCK перетворюються на входи.

2. Флаг SPIF в регістрі стану SPSR встановлюється в лог.0, і викликається переривання, як тільки в регістрі управління SPCR буде встановлений розряд SPIE, а в регістрі стану SREG – розряд загального дозволу переривань I.

Протокол передачі

Для передачі даних через інтерфейс SPI використовуються два принципово різних формати передачі. Відповідний формат вибирається за допомогою розряду CPHA (Clock Phase — фаза синхронізації) регістра управління SPCR.

Для кожного з двох форматів передачі (CPHA = лог. 0 і CPHA = лог. 1) активна фаза синхронізації може бути вибрана індивідуально за допомогою установки/скидання розряду CPOL в регістрі даних SPDR.

На представлених нижче рис. показані чотири можливі конфігурації CPHA і CPOL. У всіх чотирьох випадках передбачається, що лінія /SS провідного блоку або конфігурована як вихід, або знаходиться в лог. 1, а регістр управління SPCR ініціалізувався таким чином: SPE = лог. 1 (інтерфейс SPI розблокований); DORD = лог. 0 (спочатку виводиться старший розряд байта даних — MSB).

Щоб почати передачу даних (рис. 5.15), Master переводить лінію /SS, приєднану до одного з його портів, в стан лог. 0. Для відповідного відомого блоку передача починається по спадаючому фронту цього сигналу. Його вихід MISO переходить з високоомного в активний стан, і старший розряд байта, що знаходиться в його регістрі даних SPDR, з'являється на виході MISO. Власне передачу даних Master починає записом необхідного для передачі байта даних в свій регістр даних SPDR. Услід за цим на виході MOSI провідного блоку з'являється розряд MSB. Впродовж першої половини першого тактового імпульсу тактова лінія ще залишається в стані спокою для того, щоб забезпечити стабільну установку на відповідному вході біта даних від Master і Slave.



Рис. 5.15. Формат передачі через інтерфейс SPI, якщо CPHA = лог. 0, CPOL = лог. 0

По наростаючому фронту першого і кожного подальшого тактового імпульсу приймаються біти, розташовані на входах Master і Slave, а по спадаючому фронту наступний біт зрушується далі. Після восьмого тактового імпульсу передача даних завершена, флаги SPIF в регістрах стану провідного і відомого блоків встановлені, а вміст регістрів зсуву будуть перенесено у відповідні приймальні буфери. Вихід MOSI провідного блоку повертається в стан спокою (лог. 1), а на виході MISO відомого блоку, як правило, знаходиться старший розряд (MSB) байта, тільки що прийнятого провідним блоком. Одночасно з скиданням лінії /SS в початковий стан (лог. 1) Master завершує передачу, Slave стає неактивним, а його вихід MISO переходить у високоомний стан.

Для другого випадку, коли $CPHA = \text{лог. } 0$ і $CPOL = \text{лог. } 1$ (рис. 5.16), умови аналогічні першому випадку з тією різницею, що стан спокою тактової лінії тут встановлюється при лог. 1, біти даних переймаються першим і кожним подальшим тактовим імпульсом, а зрушення здійснюється по наростаючому фронту сигналу.



Рис. 5.16. Формат передачі через інтерфейс SPI, якщо для $CPHA = \text{лог. } 0$, $CPOL = \text{лог. } 1$
Для третього випадку $CPHA = \text{лог. } 1$ і $CPOL = \text{лог. } 0$ (рис. 5.17).



Рис. 5.17. Формат передачі інтерфейсу SPI, якщо $CPHA = \text{лог. } 1$, $CPOL = \text{лог. } 0$

Для того, щоб при цьому режимі почати передачу даних, Master, як і в першому випадку, переводить лінію /SS, приєднану до одного з його портів, в стан лог. 0. Блок Slave розблокований, і його вихід MISO переходить з високоомного в активний стан. Логічний рівень на MISO для цього випадку не визначений, але, як правило, на MISO знаходиться молодший розряд байта, переданого під час попередньої передачі від Slave до Master.

Власне передачу даних Master в цьому режимі так само починає за допомогою запису байта даних в регістр SPDR. Для відомого блоку передача починається по наростаючому фронту тактового сигналу. Старші розряди належних передачі байтів у провідному і відомому блоках за допомогою наростаючого фронту першого тактового імпульсу встановлюються на виході MOSI провідного блоку виході MISO відомого блоку. По спадаючому фронту першого і кожного подальшого тактового імпульсу вони переносяться в на входи Master і Slave, а по наростаючому фронту наступний розряд зрушується.

Після восьмого тактового імпульсу передача даних завершується, встановлюються флаги SPIF в регістрах стану інтерфейсів Master і Slave, а вміст їх регістрів зсуву переносяться у відповідні буфери прийому. Вихід MOSI провідного блоку повертається в стан спокою (лог. 1), на виході MISO відомого блоку залишається молодший розряд байта, тільки що переданого провідному блоку. Одночасно з поверненням в початковий стан лінії /SS (лог. 1) Master завершує передачу в цілому, Slave стає неактивним, а його вихід MISO переходить у високоомний стан.

Для четвертого випадку, коли $CPHA = \text{лог. } 1$, а $CPOL = \text{лог. } 1$ (рис. 5.18), умови аналогічні третьому випадку з тією різницею, що станом спокою тактової лінії тут є лог. 1, а біти даних зрушуються по спадаючому фронту першого і кожного подальшого тактового імпульсу, а приймаються по наростаючому фронту.



Рис. 5.18. Формат передачі інтерфейсу SPI, якщо CPHA = лог. 1, CPOL = лог. 1

Системні конфлікти SPI

У мікроконтролерах AVR система SPI буферизована одноразово у напрямі передачі, і двократно – у напрямі прийому. Це означає, що належний передачі байт може бути записаний в регістр даних SPI тільки тоді, коли буде завершена поточна передача. У протилежність цьому, тільки що прийнятий байт негайно переноситься в паралельний буфер прийому, унаслідок чого регістр зсуву відразу ж після передачі може здійснювати прийом другого послідовного байта. Поки перший байт буде зчитуватися з приймального буфера, другий байт не записується поки не буде прийнятий повністю, інакше прийнятий першим байт буде затертий наступним і загубиться.

Якщо під час передачі даних через інтерфейс SPI робиться спроба запису в регістр даних, то для індикації збою в регістрі стану SPCR встановлюється флаг WCOL (Write Collision – конфлікт запису). Оскільки система SPI у напрямі передачі не використовує подвійної буферизації, то запис в регістр даних слід поставити на один рівень з прямим записом у регістр зсуву SPI. Це привело б до знищення тільки що переданого байта даних. З цієї причини поточна передача даних буде доведена до кінця, і новий байт не буде записаний в регістр зсуву інтерфейсу SPI. Невдала спроба запису відображається за допомогою флага WCOL.

Схема інтерфейсу SPI розпізнає конфлікти запису як на стороні Master, так і на стороні Slave. Від цієї помилки постраждають тільки відомі блоки, оскільки провідний блок знає, коли почав передачу, а відомий блок не може зробити ніякого впливу на передачу, почату провідним блоком. Якщо потрібно, щоб Slave уникнув цієї помилки, рекомендується перевірити рівень сигналу на вхідному виводу /SS перед записом в регістр даних інтерфейсу SPI, і здійснювати звернення на запис тільки при високому рівні сигналу.

Регістр управління SPCR

Регістр управління SPCR інтерфейсу SPI знаходиться у області вводу/виводу за адресою \$0D і доступний для читання і запису.

Разряд	7	6	5	4	3	2	1	0	
\$0D (\$2D)	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR

Після закінчення передачі даних через інтерфейс SPI апаратна частина встановлює в регістрі стану SPCR розряд SPIF (флаг переривань від інтерфейсу SPI). Цей флаг указує на завершення передачі, і приводить до запиту на переривання як тільки в регістрі управління SPCR буде встановлений розряд SPIE (SPI Interrupt Enable – дозвіл переривання від SPI), а в регістрі стану SREG буде встановлений розряд загального дозволу переривань I.

Розряд SPE (SPI Enable – інтерфейс SPI доступний) включає систему інтерфейсу (SPE = лог. 1) або вимикає її (SPE = лог. 0). Після надходження сигналу скидання цей розряд повертається в початковий стан, і тим самим система SPI відключається.

Якщо розряд DORD (Data Order) містить лог. 0, то спочатку буде переданий старший розряд байта даних. При DORD = лог. 1 першим передається молодший розряд.

Коли розряд MSTR (Master/Slave Select – вибір режиму Master/Slave) містить лог. 0, то система SPI визначається як Slave, а при MSTR = лог. 1 вона буде визначена як Master. Коли лінія /SS в режимі Master конфігурована як вхід, то розряд MSTR при низькому рівні сигналу на виведенні /SS скидається в початковий стан, і тим самим інтерфейс SPI визначається як Slave. В

цьому випадку в регістрі стану встановлюється флаг SPIF. Для того, щоб знову конфігурувати інтерфейс SPI як Master, користувач повинен встановити розряд MSTR.

Коли розряд CPOL (Clock Polarity – полярність тактового імпульсу) містить лог. 0, то вихід SCK в неактивному стані містить сигнал низького рівня. Якщо CPOL = лог. 1, то SCK в неактивному стані містить сигнал високого рівня.

За допомогою розряду CPHA (Clock Phase Select — вибір фази синхронізації) встановлюється один з двох основних режимів передачі даних.

Розряди SPR1 і SPRO (SPI Clock Rate Select — вибір частоти тактових імпульсів) при роботі інтерфейсу SPI в режимі Master служать для вибору тактової частоти для лінії SCK. Якщо система SPI конфігурована як Slave, то ці розряди не мають ніякого значення. Взаємозв'язок між розрядами SPR1, SPR0 і частотою імпульсів в лінії SCK показана в табл. 5.4.

Таблиця 5.4

Частота імпульсів в SCK залежно від розрядів SPR1, SPR0 і такту системної синхронізації Ф

SPR1	SPR0	Частота імпульсів в лінії SCK
0	0	$\Phi/4$
0	1	$\Phi/16$
1	0	$\Phi/64$
1	1	$\Phi/128$

Регістр стану SPSR

Регістр стану SPSR інтерфейсу SPI розташований у області вводу/виводу за адресою \$0E і доступний тільки для читання. У мікроконтролерах AVR використовуються тільки розряди 6 і 7. Решта розрядів зарезервована і завжди містять лог. 0.

Разряд	7	6	5	4	3	2	1	0	
\$0E (\$2E)	SPIF	WCOL	–	–	–	–	–	–	SPSR

Після закінчення процесу передачі даних через SPI апаратна частина встановлює розряд SPIF (SPI Interrupt Flag — флаг переривань від інтерфейсу SPI). Цей флаг указує на завершення передачі і викликає запит на переривання, як тільки в регістрі управління SPCR буде встановлений розряд SPIE, а в регістрі стану SREG – розряд загального дозволу переривань I.

Коли лінія /SS в режимі Master конфігурована як вхід, то при низькому рівні сигналу на виведенні лінії /SS також буде встановлений флаг SPIF. Флаг SPIF скидається автоматично апаратною частиною при виконанні підпрограми обробки переривання від інтерфейсу SPI. Альтернативно, скидання може бути виконане уручну за допомогою зчитування регістра стану SPSR і подальшого звернення до регістра даних інтерфейсу SPI.

Флаг WCOL (Write Collision — конфлікт запису) в регістрі стану SPSR встановлюється у тому випадку, коли під час передачі даних через інтерфейс SPI робиться спроба запису в регістр даних SPI, що приводить до руйнування тільки що переданого байта даних. З цієї причини поточна передача даних доводиться до завершення, а новий байт не записується у регістр зсуву інтерфейсу SPI. Флаг WCOL повинен бути скинутий користувачем уручну за допомогою зчитування регістра стану і подальшого звернення до регістра даних інтерфейсу SPI.

Регістр даних SPDR

Регістр даних SPDR інтерфейсу SPI розташований у області вводу/виводу за адресою \$0F і доступний для читання і запису.

Разряд	7	6	5	4	3	2	1	0	
\$0F (\$2F)	MSB							LSB	SPDR

Запис байта в регістр даних починає передачу через інтерфейс SPI. При зчитуванні регістра даних повертається той байт, який знаходиться в буфері прийому інтерфейсу SPI.

Зміст звіту

1. Блок-схема трансмітера UART.
2. Блок-схема ресивера UART.
3. Схема послідовного інтерфейсу SPI.
4. Схема підключення декількох пристроїв до інтерфейсу SPI.

Контрольні запитання

1. Які стандарти асинхронної передачі даних Ви знаєте? Розкажіть про їх особливості.
2. Принципи передачі інформації по асинхронному інтерфейсу.
3. Склад та робота трансмітера UART (за блок-схемою рис.5.9).
4. Склад та робота ресивера UART (за блок-схемою рис.5.10).
5. Призначення регістрів вводу/виводу UDR та швидкості передачі даних UDDR інтерфейсу UART.
6. Призначення регістру стану USR інтерфейсу UART.
7. Призначення регістру управління UCR інтерфейсу UART.
8. Принципи синхронної передачі даних через послідовний інтерфейс SPI.
9. Яким чином здійснюється підключення декількох пристроїв до інтерфейсу SPI?
10. Поясніть зміст табл. 5.2 та 5.3.
11. Прокоментуйте часову діаграму передачі даних по інтерфейсу SPI.
12. Поясніть принципи передачі даних через інтерфейс SPI (чотири випадки рис.5.15-5.18).
13. Призначення регістру управління SPCR інтерфейсу SPI.
14. Призначення регістрів стану SPSR та даних SPDR інтерфейсу SPI.
15. Які системні конфлікти можливі при організації обміну за інтерфейсом SPI.