

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

М. В. Бреславець, Ю. В. Білоконська

**ІНТЕРФЕЙСИ ТА ЗАСОБИ ВЗАЄМОДІЇ ПРИСТРОЇВ
АВТОМАТИЗАЦІЇ**

Конспект лекцій

Харків «ХАІ» 2020

УДК 004.451.84(07)
Б87

Рецензенти: д-р техн. наук, проф. С. М. Фірсов,
д-р техн. наук, проф. О. Я. Ніконов

Бреславець, М. В.

Б87 Інтерфейси та засоби взаємодії пристроїв автоматизації [Електронний ресурс] : консп. лекцій / М. В. Бреславець, Ю. В. Білоконська. – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2020. – 146 с.

Розглянуто основи принципів функціонування інтерфейсних елементів: цифрових систем, протоколів інтерфейсів, поширених у системах автоматизації, та засобів конфігурування інтерфейсних вузлів. Наведено приклади інтерфейсних характеристик елементів цифрових систем та їх аналіз, протоколів передання даних, а також методи розроблення, конфігурування інтерфейсних блоків систем автоматизації.

Для студентів, які навчаються за спеціальностями 151 «Автоматизація та комп'ютерно-інтегровані технології» і 141 «Електротехніка, електроенергетика та електромеханіка».

Іл. 59. Табл. 15. Бібліогр.: 8 назв

УДК 004.451.84(07)

© Бреславець М. В., Білоконська Ю. В., 2020
© Національний аерокосмічний
університет ім. М. Є. Жуковського
«Харківський авіаційний інститут», 2020

Лекція 1

ВСТУП. КЛАСИФІКАЦІЯ ІНТЕРФЕЙСІВ

1.1 Загальні відомості про інтерфейси

Об'єктом вивчення дисципліни «Інтерфейси та засоби сполучення» є інтерфейси й засоби сполучення як функціональні елементи побудови комп'ютерно-інтегрованих технологічних систем.

Створення сучасних засобів обчислювальної техніки пов'язане із завданням об'єднання в один комплекс різних блоків обчислювальних машин, пристроїв зберігання й відображення інформації, апаратури даних і безпосередньо електронно-обчислювальних машин (ЕОМ). Це завдання покладається на уніфіковані системи сполучення – інтерфейси.

Під інтерфейсом розуміють сукупність схмотехнічних засобів, що забезпечують безпосередню взаємодію складових елементів обчислювальної системи. Інтерфейс забезпечує взаємозв'язок між складовими функціональними блоками або пристроями системи.

Основним призначенням інтерфейсу є уніфікація внутрішньосистемних і міжсистемних зв'язків і пристроїв сполучення з метою ефективної реалізації прогресивних методів проектування функціональних елементів обчислювальної системи.

Від характеристик інтерфейсів багато в чому залежать продуктивність і надійність обчислювальної машини. При розробленні систем уведення-виведення має бути вирішено такі проблеми:

1 Забезпечення можливості реалізації машин зі змінним складом устаткування (машин зі змінною конфігурацією), насамперед з різним набором периферійних пристроїв, щоб користувач міг вибрати склад обладнання (конфігурацію) машини відповідно до її призначення, легко доповнювати машину новими пристроями.

2 Паралельна реалізація в часі роботи процесора над програмою і виконання периферійними пристроями процедур уведення-виведення для ефективного й високопродуктивного використання обладнання ЕОМ.

3 Спрощення для користувача і стандартизація програмування операцій уведення-виведення, забезпечення незалежності програмування введення-виведення від особливостей того чи іншого периферійного пристрою (ПП).

4 Забезпечення автоматичного розпізнавання і реакції ядра ЕОМ на різноманітні ситуації, що виникають у ПП (готовність пристрою, відсутність носія, різні порушення нормальної роботи та ін.).

Термін «**інтерфейс**» зазвичай трактується як синонім слова «сполучення» і розуміється як сукупність схмотехнічних засобів, що забезпечують безпосередню взаємодію складових елементів пристрою, системи. Часто це означення використовується для позначення складових компонентів інтерфейсу. В одних випадках під інтерфейсом розуміють програмні засоби, що забезпечують взаємодію програм операційної системи, в інших – пристрої сполучення, що

забезпечують взаємозв'язок між складовими функціональними блоками або пристроями системи.

Під стандартним інтерфейсом розуміється сукупність уніфікованих апаратних, програмних і конструктивних засобів, необхідних для реалізації взаємодії різних функціональних елементів в автоматичних системах збору й оброблення інформації за умов, запропонованих стандартом і спрямованих на забезпечення інформаційної, електричної та конструктивної сумісності зазначених елементів.

Структурна схема інтерфейсу (рисунок 1.1) містить функціональний блок (ФБ), контролер (К), керувальний блок (КБ), інтерфейсний блок (ІБ).

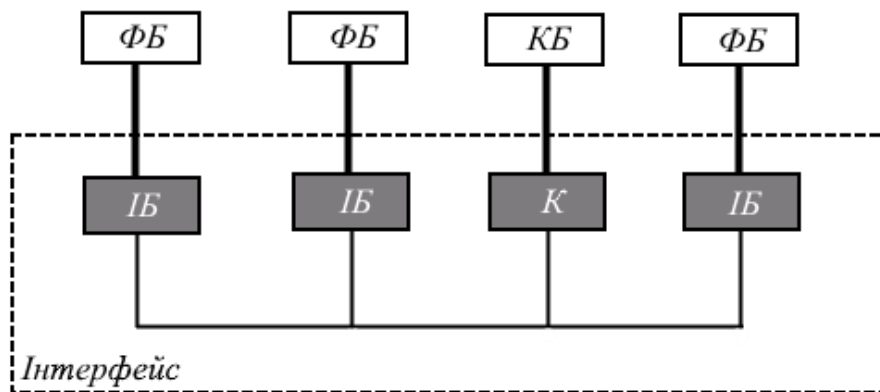


Рисунок 1.1 – Структурна схема інтерфейсу

Засоби інтерфейсу забезпечують спільну роботу незалежних різномірних функціональних блоків системи. Умовно ІБ можна поділити на частину, обернену до ФБ, у якій ураховано його специфіку, і частину, що підтримує взаємодію з іншими пристроями в межах вимог інтерфейсу.

Сучасні темпи розвитку мікроелектронної технології, а також тенденції і практика будівництва мікропроцесорних систем визначили такі напрями розвитку інтерфейсів:

1 Подальше підвищення рівня уніфікації інтерфейсного обладнання та стандартизації умов сумісності наявних найбільш поширених інтерфейсів на основі узагальнення досвіду їх широкого використання. Це вдосконалення спрямовано на створення нових стандартних інтерфейсів або підвищення рівня стандартизації наявних.

2 Модернізація й розширення функціональних можливостей наявних інтерфейсів без порушення умов сумісності завдяки новітнім досягненням у мікроелектронній технології й технології розроблення засобів передачі інформації. Основна мета цього напрямку – подовження термінів функціонального застарівання стандартних інтерфейсів і розширення області їх застосування.

3 Створення принципово нових інтерфейсів і розроблення вимог щодо їх уніфікації й стандартизації. Ця тенденція зумовлена насамперед розробленням систем з паралельним розподіленим обробленням інформації на основі якісно

нових принципів організації обчислювального процесу, а також інтегрованих розподілених систем.

1.2 Класифікація інтерфейсів

Наявні класифікації ґрунтуються зазвичай на одній класифікаційній ознаці або ж будуються для одного класу інтерфейсів. Класифікація інтерфейсів містить чотири основні ознаки класифікації:

1 Спосіб з'єднання компонентів системи (магістральний, радіальний, ланцюговий, змішаний).

Магістральний інтерфейс характеризується тим, що всі пристрої підімкнено до загальної уніфікованої шини зв'язку. Структурну схему інтерфейсу зображено на рисунку 1.2.

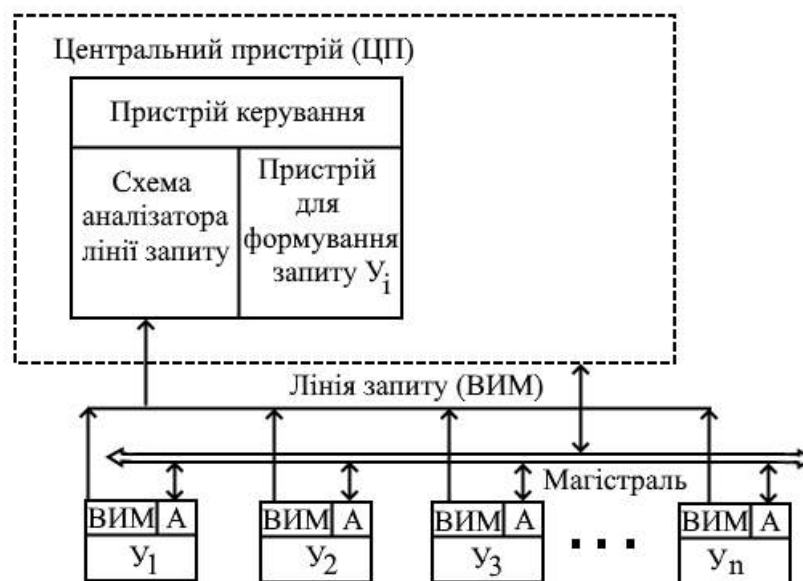


Рисунок 1.2 – Структурна схема магістрального інтерфейсу

Центральний пристрій (ЦП) пов'язаний з підпорядкованими пристроями U_1, \dots, U_n з допомогою єдиної магістралі, що базується на основі розділення часу.

Сигнал на будь-якій лінії магістралі є фізично доступним для кожного пристрою, тому для організації обміну між центральним пристроєм та одним з підпорядкованих пристроїв необхідно логічно від'єднати всі інші. Усім пристроям інформації, під'єднаними до магістралі, присвоєно адреси (номери), які фіксуються у вигляді своєї адреси на спеціальних регістрах, розміщених на всіх пристроях інформації (U_1, \dots, U_n). Адреси пристроїв однієї магістралі не повторюються, адреса в регістр пристрою записується вручну при під'єднанні його до магістралі.

Якщо обмін в інтерфейсі проводиться з ініціативи підпорядкованого пристрою інформації, то спочатку виключається можливість використання магістралі будь-яким іншим пристроєм. З цією метою в магістралі передбачено спеціальну лінію запитів (лінія ВИМ), на яку будь-який пристрій інформації

незалежно від інших може виставляти сигнал запиту (або вимоги ВИМ). Сигнал запиту означає для центрального пристрою, що на магістралі є один або кілька пристроїв інформації, що запитують обмін. Виявивши сигнал запиту (цю функцію виконує схема аналізу ВИМ), центральний пристрій має дати дозвіл на заняття магістралі тільки одному із запитувачів пристроїв інформації для виконання передачі даних.

У магістральних інтерфейсах для запам'ятовувальних пристроїв, контролерів переривань та інших пристроїв, що мають елементи пам'яті, зазвичай застосовують загальний адресний простір, діапазони адрес, що перемикаються заздалегідь запрограмованим дешифратором адреси.

Магістральний інтерфейс зазвичай використовує три шини: адреси, даних, керування.

За шиною адреси встановлюється адреса комірки пам'яті пристрою, з яким виконується обмін, а оскільки адресний простір є єдиним для всіх, то дешифратор адреси автоматично підмикає до потрібного пристрою. За шиною даних виконується обмін з вибраним пристроєм.

Шина керування використовується для визначення типу обміну (запис, читання). Пристрої не беруть участі в обміні, перебувають у відімкненому стані. На будь-який сигнал від активного пристрою має бути відгук від пасивного.

Радіальний інтерфейс характеризується тим, що є один керувальний модуль, до якого за радіальною схемою підмикаються керовані модулі. Структурну схему інтерфейсу зображено на рисунку 1.3.

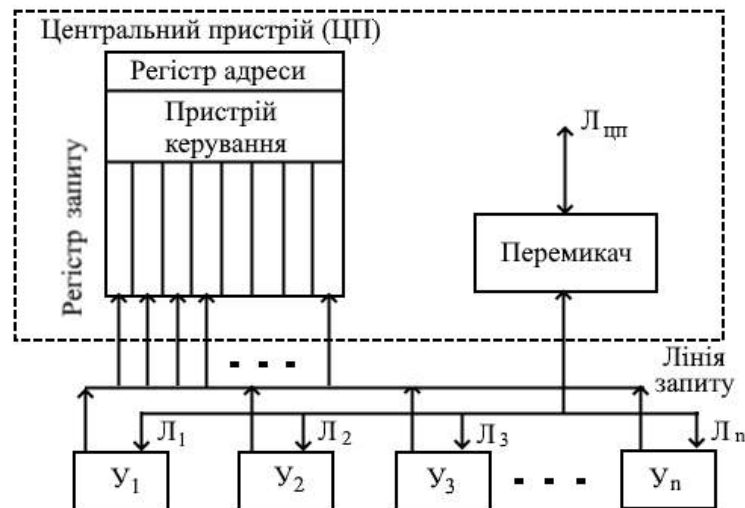


Рисунок 1.3 – Структурна схема радіального інтерфейсу

Центральний пристрій (ЦП) пов'язаний з підпорядкованими пристроями U_1, \dots, U_n з допомогою індивідуальних ліній, що належать кожному з них. Блок керування інтерфейсом знаходиться в пристрої (ЦП). Якщо необхідно передати або отримати блок інформації від пристрою інформації з ініціативи центрального пристрою, то до реєстра запиту заносяться дані адреси пристрою інформації і відповідно до нього перемикач з'єднує лінії запиту з лініями інформації. При цьому

пристрої лінії запиту й інформації з'єднуються між собою, а всі інші пристрої відмикаються і в обміні участі не беруть.

Якщо ініціатива обміну виходить від периферійного пристрою інформації, то він передає сигнал по своїй лінії запиту, який надходить в i -му розряді регістра запиту. Як тільки ЦП звільняється від попереднього обміну, його пристрій керування (ПК) інтерфейсом послідовно опитує розряди регістра запиту й з допомогою перемикача з'єднує лінії запиту з відповідними лініями інформації. Порядок опитування розрядів визначає пріоритет обслуговування пристроїв інформації.

У цьому типі інтерфейсу керовані модулі не можуть безпосередньо обмінюватися інформацією, весь обмін відбувається через керувальний модуль, при цьому зазвичай лінія обміну інформацією фізично складається з двох частин, для приймання й передачі інформації, і в цьому випадку реалізується двосторонній режим обміну інформації. Для радіального типу інтерфейсу адресний простір модуля не використовується. Звернення модуля відбувається шляхом зазначення його номера.

Ланцюговий інтерфейс. Усі пристрої мікропроцесорної системи через спеціальний пристрій зв'язку з'єднано в схему передачі кільцевої конфігурації сигналу, по якому інформація може передаватися тільки в одному напрямку (рисунок 1.4). Тут передача може здійснюватися всіма пристроями одночасно шляхом додавання до загального потоку інформації своїх даних. Пристрій, якому призначаються ці дані, має витягти їх з потоку. Якщо ж дані пройшли по кільцю в джерело, то приймач не працює.

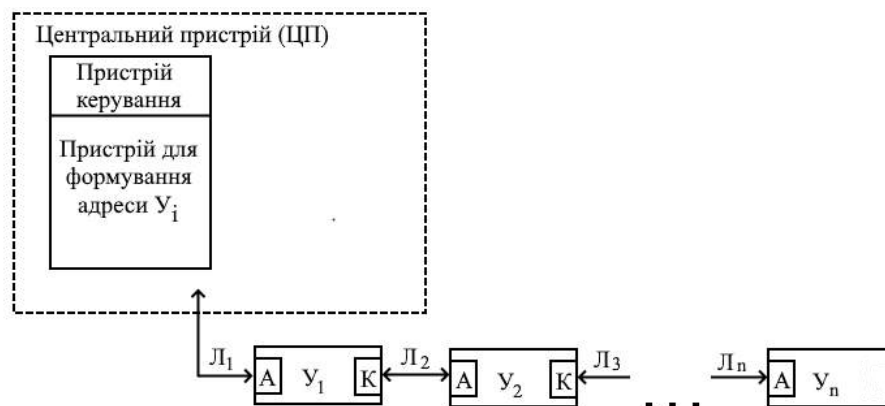


Рисунок 1.4 – Структурна схема ланцюгового інтерфейсу

При ланцюговому інтерфейсі підпорядковані пристрої ЦП, ..., Y_n підмикаються до центрального послідовно, утворюючи ланцюжок. У такому ланцюжку всім пристроям Y_1, \dots, Y_n присвоюються адреси, що не повторюються.

Якщо обмін ініціюється центральним пристроєм, то адреса запитуваного пристрою інформації передається на лінії L_1 і потрапляє в пристрій Y_1 . Запитувана адреса A в пристрої Y_1 порівнюється з власною адресою Y_1 . Якщо адреси не збіглися, то комутатор K з'єднує лінії L_1 з лініями L_2 . Таким чином, адреса запитуваного пристрою потрапляє в Y_2 і процедура повторюється. Якщо значення

адрес збіглися, то комутатор К залишається в розімкнутому стані, а пристрій, що розпізнав свою адресу, логічно підмикається до центрального пристрою. При ланцюговій схемі підмикання пристроїв процедура адресації виконується послідовно.

Цей тип інтерфейсу застосовується для мобільних систем керування, оскільки є дуже простим в реалізації і може замінити одні пристрої на інші без перепрограмування своєї системи.

Комбінований інтерфейс поєднує властивості магістрального й радіального інтерфейсів. Як пристрій керування використовується центральний процесор, а інші пристрої підімкнено до нього за двома або більше магістральними каналами. Така схема інтерфейсу застосовується в тому випадку, коли в системі є пристрої, що істотно різняться за швидкістю своєї роботи. У цьому випадку кожен магістральний канал містить тільки ті пристрої, швидкість яких є порівнянною.

2 Спосіб передачі інформації (паралельний, послідовний, паралельно-послідовний).

Паралельний спосіб – усі біти переданого слова виставляються й передаються по відповідних паралельних проводах одночасно. У портативному комп'ютері традиційно використовується паралельний інтерфейс, реалізований LPT-портом, шина SCSI і всі шини розширення.

Послідовний спосіб – біти передаються один за одним, зазвичай по одній (можливо, і двухпроводній) лінії. Ця лінія може бути як односторонньою (наприклад, у RS-232C, що реалізується COM-портом, шини SPI, JTAG), так і двосторонньою (USB, I²C).

3 Принцип обміну інформацією (асинхронний, синхронний).

Асинхронним режим (рисунок 1.5, а) називають тому, що кожен байт може бути трохи зміщеним у часі відносно побітових тактів попереднього байта. Така асинхронність передачі байтів не впливає на коректність прийнятих даних, тому що на початку кожного байта відбувається додаткова синхронізація приймача з джерелом завдяки бітам «старт». Більш «вільні» часові допуски визначають низьку вартість устаткування асинхронної системи.

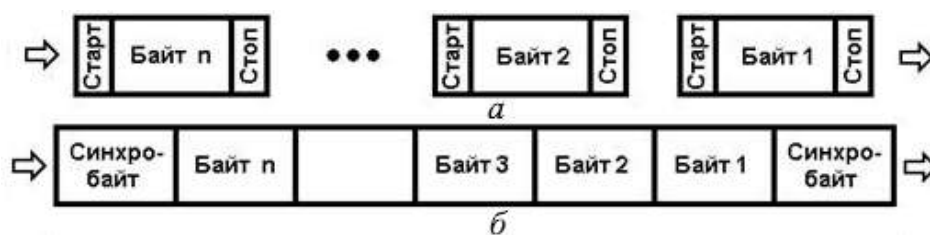


Рисунок 1.5 – Асинхронний (а) і синхронний (б) принципи обміну інформацією

В асинхронному режимі кожен байт даних супроводжується спеціальними сигналами «старт» і «стоп». Призначення цих сигналів полягає в тому, щоб, по-перше, сповістити приймач про надходження даних і, по-друге, дати приймачу досить часу для виконання деяких функцій, пов'язаних із синхронізацією, до надходження наступного байта. Сигнал «старт» має тривалість в один тактовий

інтервал, а сигнал «стоп» може тривати один, півтора або два такти, тому кажуть, що використовується один, півтора або два біти як стоповий сигнал, хоча ці сигнали не є користувацькими бітами.

При **синхронному** режимі передачі між кожною парою байтів старт-стоп-бітів немає. Користувацькі дані збираються в кадр, що випереджається байтами синхронізації (рисунок 1.5, б).

Байт синхронізації — це байт, що містить заздалегідь відомий код, наприклад 0111110, що оповіщає приймач про надходження кадру даних. При його одержанні приймач має входити в байтовий синхронізм із передавачем, тобто правильно розуміти початок чергового байта кадру. Іноді застосовується синхробайт для забезпечення більш надійної синхронізації приймача й передавача. Оскільки при передачі довгого кадру в приймачі можуть виникнути проблеми із синхронізацією бітів, використовуються коди, що самосинхронізуються.

4 Режим обміну інформацією (симплексний, напівдуплексний, дуплексний і мультиплексний).

При **симплексному** режимі приймач і передавач зв'язуються лінією зв'язку, по якій інформація передається тільки в одному напрямку. Передавальний вузол у симплексному режимі повністю займає канал. Приклади: радіомовлення, телемовлення.

При **напівдуплексному** режимі інформація передається в двох напрямках, але в різні моменти часу. Два вузли зв'язуються таким каналом зв'язку, що дає їм змогу по черзі (але не одночасно) передавати інформацію. Для змінення напрямку передачі зазвичай передається спеціальний сигнал і отримується підтвердження.

Дуплексний режим є найбільш швидкісним режимом роботи і дає змогу ефективно використовувати обчислювальні можливості швидкодійних ЕОМ у поєднанні з високою швидкістю передачі даних по каналах зв'язку. Приклад дуплексного режиму – телефонна розмова. Дуплексний режим може бути: *асиметричним*, якщо значення пропускної здатності в напрямках «туди» й «назад» істотно різняться; *симетричним*, якщо значення пропускної здатності в напрямках «туди» й «назад» істотно не різняться

Мультиплексний режим забезпечує зв'язок декількох абонентів у кожен момент часу. Зв'язок може здійснюватися між парою абонентів у будь-якому, але єдиному напрямку від одного до іншого.

1.3 Гальванічна розв'язка

Ефективним методом боротьби із завадами є гальванічна розв'язка джерел завад і пристроїв, на роботу яких завади шкідливо впливають. Для цього розділяють живлення силових і слабкострумових пристроїв, пристроїв вироблення керувальних сигналів і виконавчих пристроїв.

Крім боротьби із завадами гальванічна розв'язка забезпечує вирішення ще однієї проблеми – спільної роботи пристроїв, що знаходяться під різними потенціалами, оскільки просте електричне з'єднання ланцюгів таких пристроїв призведе до їх вигорання.

Гальванічна розв'язка – передання енергії або інформаційного сигналу між електричними каналами, що безпосереднього між собою не контактують.

Гальванічні розв'язки використовуються для передання сигналів з метою зниження завад, для безконтактного керування й захисту обладнання від пошкодження і людей від ураження електричним струмом. При гальванічній розв'язці електричні потенціали розділених ланцюгів можуть сильно відрізнятись, іноді кажуть, що ланцюги мають потенціали, які «плавають» один відносно одного.

За способом організації гальванічна розв'язка поділяється на трансформаторну, оптоелектронну (оптопари, оптоволоконні лінії зв'язку, фотогальванічні елементи), акустичну, радіоканальну, звукову, ємнісну, механічну (мотор-генератори, реле). Розглянемо три основні типи розв'язок.

Трансформаторна розв'язка – історично перший вид розв'язок, що застосовуються для передання аналогових сигналів. Використовується досі для передання як потужності, так і інформаційного сигналу. Через силові трансформатори можна передавати дуже великі потужності, аж до сотень мегават. Для передання інформації зазвичай використовують мініатюрні імпульсні й високочастотні трансформатори.

Недолік трансформаторної гальванічної розв'язки – принципова неможливість безпосереднього передання сигналів постійного струму й сигналів, що повільно змінюються через трансформатор. Тому в таких розв'язках удаються до будь-якого виду модуляції, наприклад частотної, і передання інформації при цьому відбувається шляхом передання високочастотного сигналу. На приймальному кінці високочастотний сигнал демодулюється з відновленням переданої інформації.

Оптоелектронна (оптична) розв'язка, основу якої становлять оптрони (оптопари), виконані із застосуванням тиристорів, діодів, транзисторів та інших компонентів, чутливих до світла. У пристроях цього типу сигнал передається з допомогою оптичного випромінювання й використовуються виключно для передання інформаційних сигналів, оскільки через такі розв'язки важко й технічно недоцільно передавати сигнали великої потужності.

Сьогодні оптичні розв'язки – найбільш використовуваний і популярний тип інформаційних розв'язок.

Принцип їх дії базується на випромінюванні світла будь-яким керованим електричним сигналом світловипромінювачів, переданні оптичного сигналу в гальванічно ізольовану частину й зворотне перетворення випромінювання на електричний сигнал.

Перевага оптоелектронної розв'язки порівняно з трансформаторною – менші габарити, дешевизна й можливість передавати повільнозмінні сигнали, у тому числі сигнали постійного струму.

Недолік оптичної розв'язки для передання аналогових сигналів низьких частот – суттєва нелінійність каналу при переданні, нерівномірність коефіцієнта передачі (10–30 %) у всьому діапазоні змінення сигналу. Тому для передання

повільнозмінних аналогових сигналів з достатньою точністю, як і для трансформаторної розв'язки застосовують модуляцію-демодуляцію.

Конденсаторна гальванічна розв'язка, яка застосовується винятково для передання інформаційних сигналів. Цю розв'язку можна назвати гальванічною тільки умовно, оскільки гальванічно розв'язувані ланцюги з'єднано електрично через ємнісний зв'язок, імпеданс якої є кінцевим і зменшується при збільшенні частоти різниці плавальних потенціалів «землі» розділюваних ланцюгів.

Якщо ємність конденсаторів є малою, то струми промислової частоти, що проходять через розв'язувальні конденсатори, є малими. Наприклад, звичайна ємність розв'язувальних конденсаторів становить близько 1 пФ, імпеданс гальванічної розв'язки для промислової частоти – близько 3 ГОм. Електрична міцність (пробійна напруга) розв'язувальних конденсаторів може становити декілька кіловольт, тому цей тип розв'язок можна застосовувати в електрофізичному устаткуванні для медичного обстеження й лікування пацієнтів, наприклад в електрокардіографах.

Перевага конденсаторного методу гальванічного поділу – простота, але недоліком є те, що потребується застосування модулятора-демодулятора.

1.4 Інтерфейси блоку живлення портативного комп'ютера

Блок живлення портативного комп'ютера (ПК) забезпечує напруженням постійного струму системний блок з усіма його складними пристроями. З найперших моделей ПК застосовується двотактна схема перетворювача з безтрансформаторним входом. Перетворювач є регульовальним елементом стабілізатора напруги основного джерела +5 В.

Решту напруги можна стабілізувати додатковими вихідними стабілізаторами, але частіше їх залишають нестабілізованими. При цьому чим більшим є навантаження блока по основному (стабілізованому) ланцюгу, тим вищими будуть напруги на інших шинах. Блоки живлення ПК можуть працювати навіть від мережі постійного струму і є критичними до частоти мережі 50 Гц.

Відносно застарілі блоки живлення мають перемикачі діапазону вхідної напруги. Перемикання діапазону вхідної напруги легко здійснюється перемикачем, який перетворює мостову схему випрямляча на схему випрямляча з подвоєнням для живлення від мережі 110...127 В.

Блок живлення ПК зазвичай має стандартний конструктив і набір джгутів з різними живлення системної плати й периферійних пристроїв.

Потужність блока живлення залежить від призначення корпусу системного блока й знаходиться в діапазоні 200...300 Вт для звичайних комп'ютерів і 400...600 Вт для потужних серверів. Традиційний блок живлення виробляє основну стабілізовану напругу:

- 1) +5 В при струмі 10...50 А, +12 В при струмі 3,5...15 А для живлення двигунів пристроїв та інтерфейсних ланцюгів;
- 2) –12 В при струмі 0,3...1 А для живлення інтерфейсних ланцюгів;

3) –5 В при струмі 0,3...0,5А (зазвичай не використовується, є тільки для дотримання стандарту ISA Bus).

Крім напруг живлення блок виробляє сигнал PG (Power Good) – живлення в нормі. Цей сигнал з рівнем 3-6 В виробляється через 0,1-0,5 с після вмикання живлення при нормальних вихідних напругах блока. Якщо цього сигналу немає, то на системній платі безперервно виробляється сигнал апаратного скидання процесора. Натискання кнопки Reset за дією є майже еквівалентним замиканню сигналу PG (Power Good) на «схемну землю».

Вихідні ланцюги блоків живлення виводяться гнучкими джгутами проводів зі стандартним набором рознімів. Розніми для живлення нагромаджувачів мають ключі для уникнення можливості неправильного з'єднання. Традиційні розніми живлення системної плати PS-8, PS-9 завжди встановлюються поруч так, щоб чотири чорних дроти GND було розташовано поспіль. Їх ключі є досить умовними, а помилка підімкнення може спричинити вигорання системної плати. Кольори проводів у джгутах є стандартизованими: GND – чорний, –12V – коричневий, +5V – червоний, –5V – голубий, +12V – жовтий, PG – білий (живлення в нормі).

1.5 Аналогова й цифрова земля

Системи промислової автоматизації є аналого-цифровими. У зв'язку з цим одним із джерел похибок цифрової частини є завада, що створюються аналоговою частиною системи. Для виключення проходження завад через ланцюг заземлення цифрову й аналогову землю виконують у вигляді незв'язаних провідників, з'єднаних разом тільки в одній загальній точці. Для цього модулі введення/виведення і промислові контролери мають окремі виводи аналогової (ANALOG A.GND) і цифрової (DIGITAL D.GND) землі.

Правило для схем, що містять аналогову й цифрову частини, – використання кожного вільного виводу для створення з'єднання із загальним проводом і відділення аналогових і цифрових сигналів. Слід уникати застосування з'єднань тонкими багатожильними монтажними проводами, включаючи з'єднання виводів живлення й загального виводу. Узагальнюючи наведене вище, можна сказати, що необхідно використовувати друковані плати щонайменше з двошаровою металізацією з максимальною площею загального виводу (найкращий варіант – полігон) і товстими, добре розташованими проводами, що підводять живлення і землю.

У всіх високошвидкісних схемах, що мають цифрові й аналогові сигнали, потрібно розносити їх один від одного якомога далі для запобігання можливому взаємовпливу між ними. Шини, що проводять цифровий сигнал, повинні мати мінімальну довжину. Чим коротшою є шина цифрового сигналу, тим меншою буде ймовірність виникнення зв'язку з аналоговою частиною схеми й аналоговими сигналами, які в більшості випадків більше піддаються зовнішньому впливу, ніж цифрові. Аналогові сигнали мають бути розташованими якнайдалі від цифрових сигналів, так далеко, як тільки розміри плати дають змогу це зробити. Аналогові й

цифрові сигнали в ідеальному випадку ніколи не повинні проходити паралельно один до одного на невеликій відстані. Якщо вони перетинаються, то перетин має бути виконано під прямим кутом (на різних шарах плати) для мінімізації взаємовпливу.

Для введення і виведення аналогових сигналів можуть знадобитися коаксіальні кабелі з їх механічним закріпленням. У разі коли на одній платі спільно використовуються схема вибірки-зберігання й аналогово-цифровий перетворювач (АЦП), необхідно розміщувати їх максимально близько один до одного. Усі загальні виводи необхідно під'єднати до єдиного низькоімпедансного полігону землі, і це під'єднання має бути виконано коректно (ще один аргумент для використання великого цільного полігону загального виводу, доступного на всій області друкованої плати).

Незважаючи на те, що локальні з'єднання аналогової і цифрової землі покращують якісні характеристики схеми, вони можуть спричинити проблеми при використанні АЦП і цифро-аналогового перетворювача (ЦАП). У схемах перетворювачі даних (АЦП і ЦАП) мають розглядатися як аналогові компоненти; схемотехнічний дизайн слід, розробляти таким способом, щоб захистити шини з міллівольтовими сигналами від завад на них. АЦП і ЦАП (також як і інші аналогові компоненти) мають бути розташованими поблизу від інших елементів аналогової частини схеми, з таких причин:

- відображення створюють труднощі під час передання аналогових сигналів на довгі відстані без втрати смуги й амплітуди;
- шум, що генерується в цифровій частині схеми, може проникати в аналогову частину через полігон землі або шини живлення, а також з допомогою випромінювання.

Шину зворотного струму живлення (загальний провід) кожної плати в комплексній схемі має бути з'єднано із загальним джерелом живлення товстим проводом. У випадках, коли аналогова й цифрова земля мають бути обов'язково розділеними, кожна з них необхідно окремо з'єднати з джерелом живлення, дві землі одним проводом з джерелом живлення з'єднувати не можна. Кожна шина живлення, що розташована на платі у високошвидкісної або високочутливої схеми, має бути ретельно шунтованою на шину зворотного струму живлення (шину загального проводу живлення) для запобігання проникненню завад по цих шинах. Шунтовані керамічні конденсатори (діапазон ємності від 0,01 до 0,1 мкФ) необхідно розміщувати так близько до схеми, як тільки можна.

Загальний провід – це провідник у системі, відносно якого відлічуються потенціали. Зазвичай він є загальним для джерела живлення і під'єднаних до нього електронних пристроїв. Прикладом може бути провід, загальний для всіх восьми входів 8-канального модуля аналогового вводу з окремими входами.

1.6 Джерела шумів і завад та методи боротьби з ними

Донедавна боротьба з шумами й завадами велася методом спроб і помилок при слабкому розумінні фізики процесів, що спричиняють ці шуми й завади. Такий

підхід поглинав масу часу, а при найменшому змінненні конфігурації апаратури все доводилося починати спочатку. Залишивши осторонь ефірні канали передачі інформації, стисло розглянемо джерела шумів і завад у дротяних лініях передавання аудіо- та відеоінформації.

Отже, потрібно передати сигнал (телевізійний, комп'ютерний, звуковий і т. д.) з одного пункту (Джерело) в інший пункт (Приймач). Якщо кабель короткий (наприклад, один метр), то більшості з розглянутих проблем, наймовірніше, не буде, хоча в деяких випадках навіть така відстань може виявитися згубною для сигналу.

Якщо ж реальний кабель має велику довжину, то сигнал у ньому неминуче зазнає впливу багатьох негативних чинників.

Основними джерелами шумів і завад є індустриальні заводи, наведення від сусідніх ланцюгів, різні низької якості, реактивний опір і низька якість кабелю, неточне узгодження кабелю з хвильовим опором передавача й приймача, живлення від різних фаз і наявність «петель заземлення».

Синфазна завада (рисунок 1.6) на двопровідній лінії передачі наводиться зазвичай зовнішнім джерелом перешкоди і через взаємний ємнісний зв'язок між провідниками лінії.

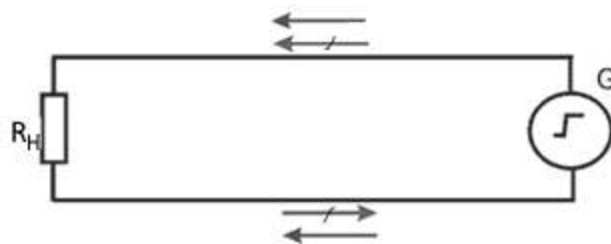


Рисунок 1.6 – Синфазна завада

На цьому рисунку стрілкою з рисою показано проходження корисного (інформаційного) сигналу по лінії, а стрілкою – синфазну заваду на обох провідниках лінії передачі, що поширюється до навантаження R_n , на якому сумуються обидва сигнали, що й призводить до спотворення сигналу.

Індустриальні, або промислові, завади виявляють себе в місцевостях, де працюють електростанції й різні електричні установки, апарати й прилади: електродвигуни, апарати електрозв'язку, медичні прилади, ЕОМ, електрозварювальні апарати, електричні дзвінки, системи електричного запалювання двигунів внутрішнього згоряння. Завади, що перешкоджають прийому сигналів іншими радіостанціями, також можна віднести до індустриальних.

Наведення від сусідніх ланцюгів виникають у тих випадках, коли сигнальний провід або кабель потрапляє в зону дії електромагнітного поля, створюваного іншим проводом або кабелем. Наприклад, якщо в квартирі поруч прокладено телефонний дріт і радіотрансляційна лінія, то, знявши слухавку, іноді можна почути музику або мову. Це і є наведення від сусідніх ланцюгів. Для вирішення цих проблем використовується коаксіальна лінія.

Розніми низької якості є зазвичай погано екранованими. Основним джерелом шумів в рознімах є так звані контактні шуми, які виникають унаслідок недосконалості контакту між матеріалами штиря й гнізда. Контактні шуми є прямо пропорційними до величини струму, що проходить через контактну пару, а щільність розподілу потужності шумів є оберненою до частоти. Якщо рознім має зовсім погану якість, то можуть виникнути «коливання» й навіть іскріння. Якщо матеріали в рознімі підібрано неправильно, без урахування їх взаємного положення в гальванічному ряду, то між ними може виникнути своєрідний електрохімічний елемент, який створює шуми й прискорює корозію.

Коаксіальна лінія використовується давно й успішно для передання як імпульсних, так і радіочастотних сигналів. Проходження сигналу по коаксіальній лінії показано на рисунку 1.7. Напрямок передання сигналу показано стрілкою, а перепад напруги – стрілкою з рисою.

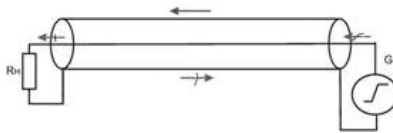


Рисунок 1.7 – Коаксіальна лінія

Вироблювані гнучкі коаксіальні лінії мають ступінь екранування за напруженістю електричного поля близько 30 Дб. Цієї величини ефекту екранування достатньо, щоб при наявному рівні завад у побутових, виробничих приміщеннях і всередині ПК можна було вважати, що синхронних завад у коаксіальних кабельних лініях передачі інформації немає.

Однак коаксіальні лінії мають свою особливість. Через кінцеве ослаблення обплетення сигнал із середини лінії просочується на зовнішній бік обплетення – створюється так звана поверхнева хвиля. Її особливість полягає в тому, що час проходження по кабелю відрізняється від часу проходження корисної сили і вони набувають певної величини R_n у різний момент часу.

Існують способи приглушення вихідних і вхідних високочастотних завад у лініях передачі інформації ПК. Один із них – з допомогою **феритових трубок**, які захищають зовнішні лінії під'єднання до комп'ютера від високочастотних завад, ефективно приглушують високочастотні завади, що поширюються по поверхні коаксіальних ліній, і синфазні завади у двопроводових лініях.

При використанні феритових трубок захист здійснюється шляхом під'єднання до лінії передачі послідовної індуктивності, яка істотно знижує синфазну заваду, але не впливає на сам сигнал. Цей спосіб застосовують аналогічно й до коаксіального кабелю, істотно знижуючи поверхневу хвилю. Ефективність феритового кільця є тим вищою, чим більше витків намотано на нього (тим вище індуктивність). Головне, щоб феритові кільце й трубка знаходилися якомога ближче до приймача сигналу. Якщо лінія є двонаправленою, то треба ставити дві феритові трубки на кінцях кабелю, безпосередньо біля рознімів.

Лекція 2 ПАРАЛЕЛЬНИЙ ІНТЕРФЕЙС LPT-ПОРТ

2.1 Загальні відомості про LPT-порт

Історично паралельний інтерфейс було введено в персональний комп'ютер для під'єднання принтера (звідси й аббревіатура LPT: Line printer – порядковий принтер). Однак згодом паралельний інтерфейс став використовуватися для під'єднання інших периферійних пристроїв – сканерів, дисководів типу Zip і деяких інших. Базовий різновид порту дає змогу передавати дані тільки в одному напрямку (від ПК до ПП), однак пізніше було розроблено кілька стандартів двобічного передавання даних.

Адаптер паралельного інтерфейсу являє собою набір регістрів, розташованих в адресному просторі пристроїв уведення-виведення. Кількість регістрів залежить від типу порту, проте три з них є стандартними й наявними завжди – регістр даних, регістр стану і регістр керування. Адреси регістрів відлічуються від базового, стандартні значення якого – 3BCh, 378h, 278h. Дізнатися кількість установлених портів у комп'ютері і їх базові адреси можна, просканувавши область даних BIOS за адресами 0: 408h для LPT1, 0: 40Ah для LPT2, 0: 40Ch для LPT3 і 0: 40Eh для LPT4. Якщо за цими адресами міститься слово (2 байти) з ненульовим значенням, то це і є базова адреса порту. Якщо слово містить нульове значення, то порт не встановлено. BIOS не підтримує більше чотирьох LPT-портів. Порт може використовувати апаратне переривання (IRQ7 або IRQ9). Більшість сучасних систем дають змогу змінювати режим роботи порту, його адресу й IRQ з налаштувань BIOS Setup. Наприклад, в AWARD BIOS є розділ Integrated Peripherals, що дає змогу налаштувати режим, адресу і IRQ порту.

LPT-порт має зовнішню 8-бітову шину даних, 5-бітову шину сигналів стану й 4-бітову шину керувальних сигналів. При початковому завантаженні BIOS намагається виявити паралельний порт, причому робить це примітивним і не завжди коректним чином – за можливими базовими адресами портів передається тестовий байт, який складається з набору нулів та одиниць, що чергуються (55h або AAh), потім проводиться читання за тією ж адресою, і якщо прочитаний байт збігся з записаним, то вважається, що за цією адресою знайдено LPT-порт. Визначити адресу порту LPT4 BIOS не може. Для роботи з ПУ в BIOS передбачено переривання INT 17h, що дає змогу передавати дані побайтово, формувати ПУ й отримувати інформацію про його стан.

2.2 Інтерфейс Centronics

Поняття Centronics належить до набору сигналів, протоколу обміну й розніму, що встановлюється в принтерах. Більшість сучасних принтерів є сумісними з інтерфейсом Centronics. Призначення сигналів і контакти розніму ПП, на який їх виведено, наведено в таблиці 2.1, а часові діаграми обміну з принтером зображено на рисунку 2.1.

Таблиця 2.1 – Призначення сигналів і контакти розніму інтерфейсу

Сигнал	Активний рівень сигналу	Контакт	Напрямок передачі відносно принтера	Призначення
Strobe	0	1	I	Строб даних. Передається ЕОМ, дані фіксуються за низьким рівнем сигналу
D0...D7	1	2...9	I	Лінії даних. D0 – молодший біт
Ack	0	10	O	Acknowledge – імпульс підтвердження приймання байта (запит на приймання наступного). Може використовуватися для формування переривання
Busy	1	11	o	Зайнятий. Приймання даних можливе тільки при низькому рівні сигналу
PE	1	12	o	«1» – сигналізує про закінчення паперу
Select	1	13	o	«1» – сигналізує про ввімкнення принтера (ГП – готовність приймача), зазвичай +5 В через резистор від джерела живлення ПК)
Auto LF	0	14	i	Автоматичне переведення рядка. Якщо «0», то ПУ при отриманні символу CR (переведення каретки) виконує функцію LF – новий рядок
Error	0	32	o	Помилка ПК (off-line, немає паперу, немає тонера, внутрішня помилка)
Init	0	31	i	Ініціалізація (перехід до початку рядка, скидання всіх параметрів на значення за замовчуванням)
Slct In	0	36	i	Вибір принтера. При «1» принтер не сприймає інші сигнали інтерфейсу
GND		19...30, 33	–	Загальний провід

Примітка. Активний рівень сигналу: «1» – високий, «0» – низький. Напрямок передавання сигналу відносно принтера: «i» – вхід, «o» – вихід.

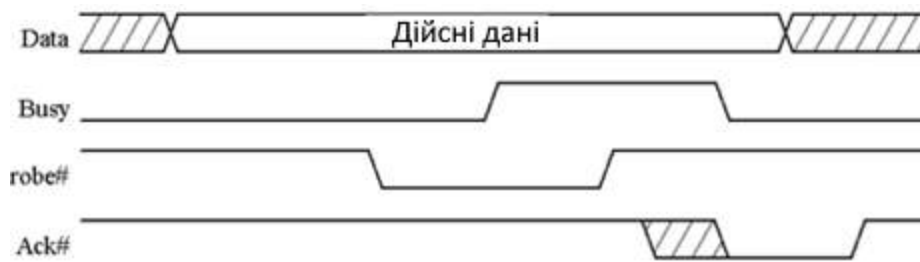


Рисунок 2.1 – Передання сигналу інтерфейсу

Сигнал Auto LF майже не застосовується, але його неправильне значення призводить до того, що принтер або пропускає рядки, або друкує їх один поверх іншого, або дублює.

Передання сигналу (рисунок 2.1) починається з перевірки джерелом сигналу «Error». Якщо його встановлено, то обмін не проводиться. Потім перевіряється стан сигналу «Busy». Якщо він дорівнює нулю, то джерело починає передавати байти даних. Для передання байта даних джерело виставляє його на лінії D0-D7 і видає сигнал Strobe. Приймач за сигналом Strobe# (тут і далі по тексту значок «#» після назви сигналу є ознакою того, що сигнал має низький активний рівень) читає дані з шини даних і виставляє сигнал Busy на час його оброблення. Після закінчення оброблення приймач видає сигнал АСК # і знімає сигнал Busy. Якщо протягом тривалого часу (6–12 с) джерело не отримує АСК #, то він приймає рішення про помилку – «тайм-аут» (time-out) пристрою. Якщо після приймання байта приймач з якої-небудь причини не готовий приймати дані, то він не знімає сигнал Busy. При програмній реалізації обміну за вказаним протоколом бажано обмежити час очікування зняття Busy (зазвичай 30–45 с), інакше можливе зависання програми.

2.3 Стандартний паралельний порт ПК

Стандартний паралельний порт має назву **SPP** (Standard Parallel Port) є односпрямованим, на його базі програмно реалізується протокол обміну Centronics. Порт забезпечує можливість генерації IRQ за імпульсом АСК# на вході. Сигнали порту виводяться на стандартний рознім DB-25S (розетка), який розміщено безпосередньо на платі адаптера або з'єднується з ним плоским шлейфом (у разі, якщо адаптер є інтегрованим з материнською платою). Назви сигналів збігаються з назвами сигналів інтерфейсу Centronics (таблиця 2.2).

У цій таблиці I/O – напрямком передання: I – вхід; O – вихід; O(I) – вихід, стан якого можна визначити за певних умов, O/I – вихідні лінії, стан яких читається при читанні з відповідних регістрів порту, I* – вхід АСК#, з'єднаний з живленням +5 В через резистор 10 кОм, який зроблено для запобігання хибним перериванням, тому що переривання генеруються за негативним перепадом сигналу на вході АСК#.

Таблиця 2.2 – Призначення сигналів інтерфейсу SPP

Контакт DB-25S	Провід шлейфа	I/O	Сигнал	Контакт DB-25S	Провід шлейфа	I/O	Сигнал
1	1	O/I	Strobe#	10	19	I*	ACK#
2	3	O(I)	D0	11	21	I	Busy
3	5	O(I)	D1	12	23	I	PE
4	7	O(I)	D2	13	25	I	Select
5	9	O(I)	D3	14	2	O/I	Auto LF#
6	11	O(I)	D4	15	4	I	Error#
7	13	O(I)	D5	16	6	O/I	Init#
8	15	O(I)	D6	17	8	O/I	Select In#
9	17	O(I)	D7	18-25	10,22,14, 16,18,20 22,24,26	–	GND

Як недоліки стандартного LPT-порту (SPP) слід зазначити невисоку швидкість передання даних (100–150 КБ/с), завантаження процесора під час передання даних, неможливість двонаправленого побайтового обміну. Існує «радіоаматорська» методика двонаправленого обміну, яка полягає в тому, що для введення даних на лінії D0-D7 виставляють «1», а як передавач використовують мікросхеми з відкритим колектором, які при відкритому транзисторі можуть «підсаджувати» напругу логічної одиниці до рівня порядку 1,5–1,7 В. Струм є обмеженим на рівні 30 мА. Як видно з рівнів сигналів, вони не відповідають рівням ТТЛ, тому більшість портів не працюють у такому режимі або працюють нестабільно. Крім того, такий спосіб може бути небезпечним для адаптера порту, який буде працювати з граничними для нього струмами.

2.4 Стандарт IEEE 1284

У стандарті для паралельного інтерфейсу IEEE 1284, прийнятий 1994 року, визначено терміни SPP, EPP і ECP, а також п'ять режимів обміну даними, метод узгодження режиму, фізичний та електричний інтерфейси. Згідно з IEEE 1284, можливими є кілька режимів обміну даними через паралельний порт.

Compatibility Mode – односпрямований (вивід) за протоколом Centronics. Цей режим відповідає стандартному (традиційному) порту SPP.

Nibble Mode – уведення байта за два цикли (по чотири біти) з використанням для введення лінії стану. Цей режим обміну може застосовуватися на будь-яких адаптерах.

Byte Mode – уведення байта цілком, використовуючи для приймання лінії даних. Цей режим працює тільки на портах для читання вихідних даних (Bi – Directional або PS/2 Type1).

EPP (Enhanced Parallel Port) Mode – двонаправлений обмін даними, при якому сигнали інтерфейсу генеруються апаратно під час циклу звернення до порту (читання або запису в порт), є ефективним при роботі з пристроями зовнішньої пам'яті, адаптерами локальних мереж.

ECP (Extended Capability Port) Mode – двонаправлений обмін з можливістю апаратного стиснення даних за методом RLE (Run Length Encoding), використання FIFO-буферів і DMA. Керувальні сигнали інтерфейсу генеруються апаратно. Є ефективним для принтерів і сканерів.

У сучасних автоматичних машинах з LPT-портом на системній платі режими порту SPP, EPP, ECP або їх комбінація задається в BIOS Setup. Режим Compatibility Mode повністю відповідає SPP і часто встановлюється за замовчуванням. Усі інші режими розширюють функціональні можливості інтерфейсу й підвищують його продуктивність. Крім того, по стандарту регламентується спосіб узгодження режиму, доступного як для ПК, так і для периферійного пристрою.

З допомогою стандарту IEEE 1284 можна визначити фізичні характеристики приймачів і передавачів сигналів. Налаштування стандартного порту не встановлюють типи вихідних схем, граничних значень величин навантажувальних резисторів і конденсаторів, що вносяться ланцюгами й провідниками. На відносно невисоких швидкостях обміну розкид цих параметрів не викликав проблем сумісності. Однак розширені (функціонально й за швидкістю передання) режими потребують чіпких налаштувань. Стандарт IEEE 1284 визначає два рівні інтерфейсної сумісності. Перший рівень (Level I) визначено для пристроїв повільних, але у яких використовується змінення напрямку передання даних. Другий рівень (Level II) призначено для пристроїв, що працюють у розширених режимах з високими швидкостями й довгими кабелями. До передавачів ставляться такі вимоги:

1 Рівні сигналів без навантаження мають становити $-0,5...+5,5$ В.

2 Рівні сигналів при струмі навантаження 14 мА мають бути не нижчими від +2,4 В для високого рівня (V_{oh}) і не вищими від +0,4 В для низького рівня (V_{ol}) на постійному струмі.

3 Вихідний імпеданс R_o , виміряний на рознімі, має становити (50 ± 5) Ом на рівні $V_{oh} \sim V_{ol}$. Для забезпечення заданого імпедансу використовують послідовні резистори у вихідних колах передавача. Узгодження імпедансу передавача й кабелю знижує рівень імпульсних завад.

4 Швидкість наростання (зниження) імпульсу має становити $0,05...-0,4$ В/нс.
Вимоги до приймачів:

1 Допустимі пікові значення сигналів: $-2,0...+7,0$ В.

2 Пороги спрацювання мають бути не вищими від 2,0 В (V_h) для високого рівня і не нижчими від 0,8 В (V_l) для низького.

3 Приймач повинен мати гістерезис у межах 0,2...1,2 В (гістерезис мають спеціальні мікросхеми – тригери Шмітта).

4 Вхідний струм мікросхеми не повинен перевищувати 20 мкА, вхідні лінії з'єднуються з шиною живлення +5 В резистором 1,2 кОм.

5 Вхідна ємність не повинна перевищувати 50 пФ.

Коли виникла специфікація ECP, фірма Microsoft рекомендувала застосування динамічних термінаторів на кожен ліній інтерфейсу. Однак сьогодні використовують специфікацію IEEE 1284, у якій динамічні термінатори не застосовуються.

Термінатор — поглинач енергії (зазвичай резистор) на кінці довгої лінії, опір якого дорівнює хвильовому опору цієї лінії. Слово «термінатор» застосовується в комп'ютерному жаргоні, термінологічним синонімом йому є «узгоджене навантаження».

Імпеданс — аналогія поняттю «опір» при змінному струмі. Це — міра нездатності матеріалу проводити змінний електричний струм. Імпеданс матеріалів може змінюватися при зміні частоти прикладеного електричного потенціалу залежно від властивостей твердого тіла або рідини.

2.5 Функції BIOS для LPT-порту

BIOS забезпечує підтримку LPT-порту, необхідну для організації виведення по інтерфейсу Centronics.

Під час початкового тестування POST BIOS перевіряє наявність паралельних портів за адресами 378h і 278h, 3BCh, 2BCh і поміщає базові адреси виявлених портів у комірки BIOS DATA AREA 0: 0408h, 040Ah, 040Ch, 040Eh. Ці комірки зберігають адреси портів з логічними іменами LPT1-LPT4. До комірок 0: 0478, 0479, 047A, 047B заносяться константи, що задають витримування тайм-ауту для цих портів.

Пошук портів зазвичай виконується за базовою адресою. Якщо зчитаний байт збігся із записаним, то вважається, що знайдено LPT-порт, і його адресу поміщають у комірку BIOS DATA AREA. Адресу порту LPT4 BIOS самостійно встановити не може, оскільки в списку стандартних адрес пошуку є лише три наведені вище.

Виявлені порти не ініціалізуються — шляхом запису в регістр керування формується і знімається сигнал Initff, після чого записується значення 00h, що відповідає вихідному стану сигналів інтерфейсу.

Програмне переривання BIOS INT 17h забезпечує такі функції підтримки LPT-порту:

- 00h — виведення символу з регістра AL за протоколом Centronics. Дані поміщаються у вихідний регістр, і після готовності принтера формується строб;
- 01h — ініціалізація інтерфейсу й принтера;
- 02h — опитування стану принтера.

Під час виклику INT 17h номер функції задається в регістрі AH, номер порту — в регістрі DX (0 — LPT1, 1 — LPT2 ...). При поверненні після будь-якої функції регістр AH містить код стану — біта регістру стану SR [7: 3] (біти 6 і 3 є

інвертованими) і прапор тайм-ауту біта 0. Прапор тайм-ауту встановлюється при невдалій спробі виведення символу.

2.6 Режими передавання даних

У стандарті IEEE 1284 наведено п'ять режимів обміну, один з яких повністю відповідає стандартному виводу за протоколом Centronics, а також спосіб, за яким програмне забезпечення може визначити режим, доступний і хосту (PC), і ПП (або під'єднаному другому комп'ютеру). Режими нестандартних портів, що реалізують протокол обміну Centronics апаратно (Fast Centronics, Parallel Port FIFO Mode), можуть і не бути режимами IEEE 1284, незважаючи на наявність в них ознак EPP і ECP.

Для опису режимів обміну застосовують такі поняття:

- Хост – комп'ютер, що має паралельний порт;
- ПП – периферійний пристрій, що під'єднується до цього порту;
- Ptr – у передавальний сигнал ПП;
- Прямий канал – канал виведення даних з хоста в ПП;
- Зворотній канал – канал уведення даних у хост з ПП.

Полубайтний режим введення – Nibble Mode

Цей режим призначено для двонаправленого обміну. Може працювати на всіх стандартних портах. Порти мають п'ять ліній уведення стану, використовуючи які ПП може посилати в хост байт тетрадами (nibble – напівбайт, 4 біти) за два прийоми. Сигнал Ask#, що викликає переривання, яке може використовуватися в цьому режимі, відповідає біту 6-го регістра стану, що ускладнює програмні маніпуляції з бітами при складанні байта. Сигнали порту наведено в таблиці 2.3, тимчасові діаграми - на рисунку 2.2.

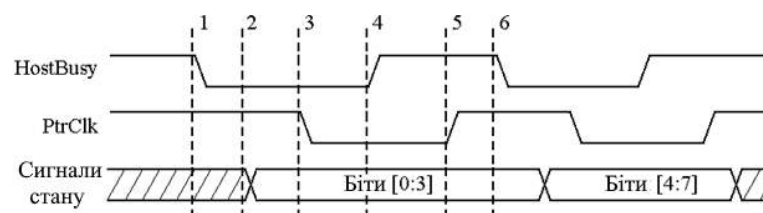


Рисунок 2.2 – Півбайтовий режим Nibble Mode

Приймання байта даних у півбайтовому режимі складається з таких фаз:

- 1 Хост сигналізує про готовність приймання даних установленням низького рівня на лінії HostBusy.
- 2 ПП у відповідь поміщає тетраду на входні лінії стану.
- 3 ПП сигналізує про готовність тетради встановленням низького рівня на лінії PtrClk.
- 4 Хост установлює високий рівень на лінії HostBusy, указуючи на зайнятість прийоманням й обробленням тетради.

5 ПП відповідає встановленням високого рівня на лінії PtrClk.

6 Кроки 1–5 повторюються для другої тетради, після чого ПУ може сигналізувати про наявність даних для хоста (Select) і зайнятості прямого каналу (Busy), а також викликати переривання (ACK).

Таблиця 2.3 – Сигнали порту півбайтового режиму

Контакт	Сигнал SPP	i/o	Біт	Опис
14	AutoFeed#	O	CR.1\	HostBusy - сигнал квитювання. Низький рівень означає готовність до приймання тетради, високий підтверджує приймання тетради
17	SelectIn#	O	CR.3\	Високий рівень свідчить про обмін у режимі IEEE 1284 (у режимі SPP рівень є низьким)
10	Ack#	I	SR.6	PtrClk. Низький рівень означає готовність тетради, високий – відповідь на сигнал HostBusy
11	Busy	I	SR.7	Приймання біта даних 3, потім – біта 7
12	PE	I	SR.5	Приймання біта даних 2, потім – біта 6
13	Sdect	I	SR.4	Приймання біта даних 1, потім – біта 5
15	Error#	I	SR.3	Приймання біта даних 0, потім – біта 4

Півбайтний режим сильно навантажує процесор, оскільки сигнали CR і SR формуються програмно, і тому збільшити швидкість обміну понад 50 кбайт/с не вдається. Безумовно, його перевага полягає в тому, що він працює на всіх портах. Його застосовують у тих випадках, коли потік даних є невеликим (наприклад, для зв'язку з принтерами). Однак при зв'язку з адаптерами локальних мереж, зовнішніми дисковими нагромаджувачами і CD-ROM приймання великих обсягів даних потребує неабиякого терпіння з боку користувача.

Двонаправлений байтовий режим – Byte Mode

У цьому режимі дані приймаються з використанням двонаправленого порту, вихідний буфер даних якого може відмикатися установленням біта. Як і попередні, цей режим є програмно-керованим – усі сигнали квитювання аналізує й установлює драйвер. Сигнали порту наведено в таблиці 2.4, часові діаграми зображено на рисунку 2.3.

Фази приймання байта даних:

1 Хост сигналізує про готовність приймання даних установленням низького рівня на лінії HostBusy.

2 ПП у відповідь поміщає байт даних на лінії Data [0:7].

3 ПУ сигналізує про дійсність байта установленням низького рівня на лінії PtrClk.

4 Хост установлює високий рівень на лінії HostBusy, указуючи на зайнятість прийманням і обробленням байта.

5 ПП відповідає встановленням високого рівня на лінії PtrClk.

6 Хост підтверджує приймання байта імпульсом HostClk.

7 Кроки 1–6 повторюються для кожного наступного байта.

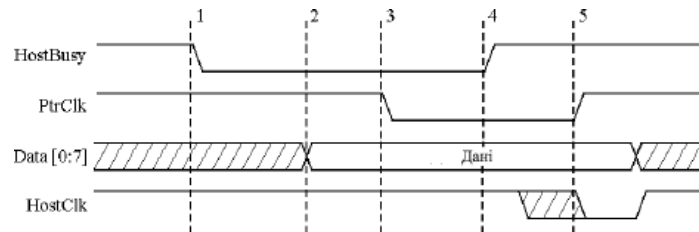


Рисунок 2.3 – Двонаправлений режим Byte Mode

Таблиця 2.4 – Сигнали порту двонаправленого режиму

Контакт	Сигнал SPP	Ім'я в байтовому режимі	i/o	Біт	Опис
1	Strobe#	HostClk	o	CR.0\	Імпульс (низького рівня) підтверджує приймання байта наприкінці кожного циклу
14	AutoFeed#	HostBusy	o	CR.1\	Сигнал квитювання. Низький рівень означає готовність хоста прийняти байт; високий рівень встановлюється після приймання байта
17	SelectIn#	1284Active	o	CR.3\	Високий рівень свідчить про обмін у режимі IEEE 1284 (у режимі SPP рівень є низьким)
16	Init#	Init#	o	CR.2	Не використовується; встановлено високий рівень
10	Ack#	PtrClk	i	SR.6	Установлюється низький рівень для індикації дійсності даних на лініях Data [0:7]. Низький рівень встановлюється у відповідь на сигнал HostBusy
11	Busy	PtrBusy	i	SR.7\	Стан зайнятості прямого каналу

Продовження таблиці 2.4

Контакт	Сигнал SPP	Ім'я в байтовому режимі	i/o	Біт	Опис
12	PE	AckDataReq	I	SR.5	Установлюється ПП для вказання про наявність зворотного каналу передання
13	Select	Xflag	i	SR.4	Прапорець розширюваності
15	Error#	DataAvail#	i	SR.2	Установлюється ПП для вказання про наявність зворотного каналу передання
2-9	Data [0:7]	Data [0:7]	i/o	DR[0:7]	Двоспрямований (прямий і зворотний) канал даних

Сигнали AckDataReq, Xflag, DataAvail# діють у послідовності узгодження.

Побайтовий режим дає змогу збільшити швидкість зворотного каналу до швидкості прямого каналу в стандартному режимі. Однак він може працювати тільки у двоспрямованих портах.

Режим ECP

Протокол ECP (Extended Capability Port – порт з розширеними можливостями) запропонували Hewlett Packard і Microsoft для зв'язку з ПП принтерів і сканерів. Як і EPP, цей протокол забезпечує високопродуктивний двонаправлений обмін даними хоста з ПП.

Протокол ECP в обох напрямках забезпечує цикли запису й читання даних, а також командні цикли запису й читання.

Командні цикли поділяються на два типи: передання каналних адрес і лічильника RLC (Run-Length Count).

На відміну від EPP разом з протоколом ECP відразу виникає стандарт на програмну (реєстрову) модель його адаптера, викладений у документі «The IEEE 1284 Extended Capabilities Port Protocol and ISA Interface Standard» компанії Microsoft. У цьому документі визначено властивості протоколу, не задані в стандарті IEEE 1284: компресію даних хост-адаптером за методом RLE, буферизацію FIFO для прямого й зворотного каналів, застосування DMA й програмного введення/виведення.

Компресія в реальному часі за методом RLE (Run-Length Encoding) дає змогу досягти значення коефіцієнта стиснення 64 : 1 під час передання растрових зображень, які мають довгі рядки байтів, що повторюються. Компресію можна використовувати, тільки якщо її підтримують і хост, і ПП.

Канальна адресація ECP застосовується для адресації багатьох логічних пристроїв, що належать до одного фізичного. Наприклад, у комбінованому пристрої факс/принтер/модем, що під'єднується тільки до одного паралельного

порту, можливим є одночасне приймання факсу і друкування на принтері. У режимі SPP, якщо принтер установить сигнал зайнятості, канал буде зайнято даними, поки принтер їх не прийме. У режимі ECP програмний драйвер просто адресується до іншого логічного каналу того ж порту.

Обмін драйвера з FIFO-буфером може здійснюватися з використанням як DMA, так і програмного введення-виведення. Обмін ПУ з буфером апаратно виконує адаптер ECP. Драйвер у режимі ECP не має інформації про точний стан процесу обміну, але тут зазвичай важливим є тільки те, чи завершено його.

Пряме передання даних на зовнішньому інтерфейсі складається з таких кроків:

1 Хост поміщає дані на шину каналу і встановлює ознаку циклу даних (високий рівень) або команди (низький рівень) на лінії HostAck.

2 Хост встановлює низький рівень на лінії HostClk, указуючи на дійсність даних.

3 ПП відповідає встановленням високого рівня на лінії PeriphAck.

4 Хост встановлює високий рівень лінії HostClk, і цей перепад може використовуватися для фіксації даних у ПП.

5 ПП встановлює низький рівень на лінії PeriphAck для указання про готовність до приймання наступного байта.

Оскільки передання в ECP відбувається через FIFO-буфери, які можуть бути наявним з обох боків інтерфейсу, важливо розуміти, на якому етапі дані можна вважати переданими. Дані вважаються переданими на кроці 4, коли лінія HostClk (рисунок 2.4) переходить на високий рівень. У цей момент модифікуються лічильники переданих і прийнятих байтів. У протоколі ECP є умови, що спричиняють припинення обміну між кроками 3 і 4. Тоді ці дані не мають розглядатися як передані.

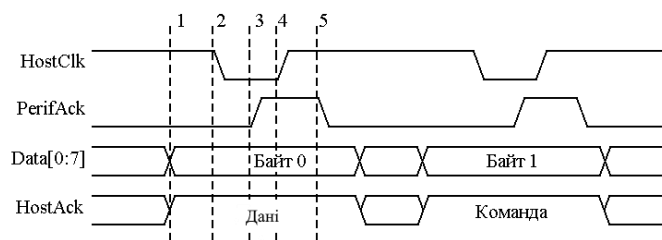


Рисунок 2.4 – Передання даних у режимі ECP

Режим EPP

Протокол EPP (Enhanced Parallel Port – покращений паралельний порт) розробили компанії Intel, Xircom і Zenith Data Systems задовго до прийняття IEEE 1284. Його призначено для підвищення продуктивності обміну з паралельного порту. EPP реалізовано в чіпсеті Intel 386SL (мікросхема 82360) і згодом прийнято багатьма компаніями як додатковий протокол паралельного порту. Версії протоколу, реалізовані до прийняття IEEE 1284, відрізняються від нинішнього стандарту.

Протокол EPP забезпечує чотири типи циклів обміну: запис і читання даних, запис і читання адреси.

Призначення циклів запису і читання даних є очевидним. Адресні цикли використовуються для передання адресної, каналної й керувальної інформації. Цикли обміну даними відрізняються від адресних циклів застосовуваними стробувальними сигналами.

На відміну від програмно-керованих режимів, описаних раніше, зовнішні сигнали EPP-порту для кожного циклу обміну формуються апаратно по одній операції запису або читання в регістрі порту. На рисунку 2.5 зображено діаграму циклу запису даних, що ілюструє зовнішній цикл обміну, вкладений в цикл запису системної шини процесора (іноді ці цикли називають пов'язаними). Адресний цикл запису відрізняється від циклу даних тільки стробом зовнішнього інтерфейсу.

Цикл запису даних складається з таких фаз:

- програма виконує цикл виведення (IOWR#) у порт 4 (EPP Data Port);
- адаптер установлює сигнал Write* (низький рівень), і дані записуються на вихідну шину LPT-порту;
- при низькому рівні Wait# установлюється строб даних;
- порт очікує підтвердження від ПП (переведення Wait# на високий рівень);
- знімається строб даних – зовнішній EPP-цикл завершується;
- завершується процесорний цикл виведення;
- ПП встановлює низький рівень Wait#, указуючи на можливість початку наступного циклу.

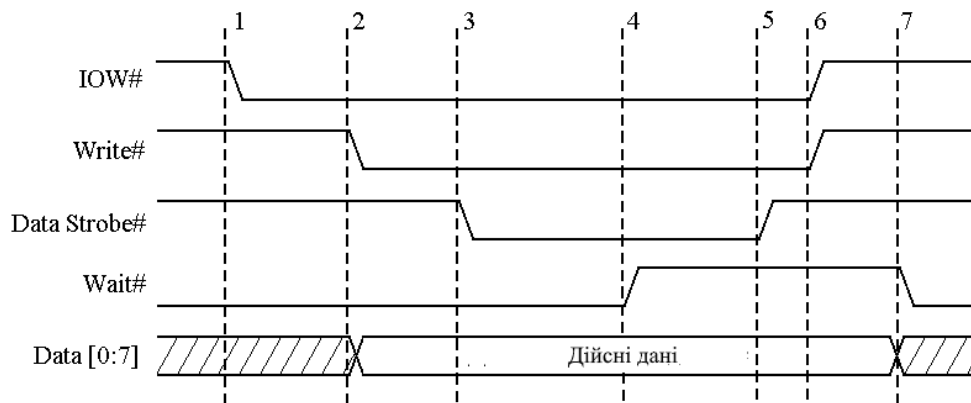


Рисунок 2.5 – Передання даних у режимі EPP

Головною відмінною рисою EPP є виконання зовнішнього передання під час одного процесорного циклу ведення-виведення. Це дає змогу досягати високих швидкостей обміну (0,5...2 Мбайт/с). ПП, під'єднаний до паралельного порту EPP, може працювати зі швидкістю пристрою, що під'єднується через слот ISA. Протокол блокованого квитиювання (interlocked handshakes) дає змогу автоматично налаштуватися на швидкість обміну, що є доступною і хосту, і ПП. ПП може регулювати тривалість усіх фаз обміну з допомогою всього лише одного сигналу Wait#. Протокол автоматично підлаштовується під довжину кабелю,

затримки, що вносяться, призведуть тільки до подовження циклу. Оскільки кабелі, що відповідають IEEE 1284, мають однакові хвильові властивості для різних ліній, порушення передання, пов'язаного зі «змаганнями» сигналів, не повинно бути. При під'єднанні мережних адаптерів або зовнішніх дисків до JPP-порту можна спостерігати незвичне явище: зниження продуктивності зі збільшенням довжини кабелю.

Використання реєстра даних EPP дає змогу здійснювати передання блока даних з допомогою однієї інструкції REP INSB або REP OUTSB. Деякі адаптери допускають 16- або 32-бітове звернення до реєстра даних EPP. При цьому адаптер просто дешифрує адресу зі зміщенням у діапазоні 4–7 як адресу реєстра даних EPP, але процесору повідомляє про розрядність восьми бітів. Тоді 16- або 32-бітове звернення за адресою реєстра даних EPP приведе до автоматичної генерації двох або чотирьох шинних циклів за зростаючими адресами, починаючи зі зсуву 4. Ці цикли будуть виконуватися швидше, ніж така сама кількість одиничних циклів. Таким чином забезпечується продуктивність до 2 Мбайт/с, достатня для адаптерів локальних мереж, зовнішніх дисків, стримерів і CD-ROM. Адресні цикли EPP завжди виконуються тільки в однобайтовому режимі.

Важливою ознакою EPP є те, що звернення процесора до ПП здійснюється в реальному часі, тобто немає буферизації. Драйвер може відстежувати стан і подавати команди в точно відомі моменти часу. Цикли читання й запису можуть чергуватися в довільному порядку або йти блоками. Такий тип обміну є зручним для реєстроорієнтованих ПК або ПП, що працюють у реальному часі: мережних адаптерів, пристроїв збору інформації й керування і т. ін.

Лекція 3

ПОСЛІДОВНИЙ ІНТЕРФЕЙС СОМ-ПОРТ

3.1 Загальні відомості про СОМ-порт

Послідовний інтерфейс призначено для передання даних по одній сигнальній лінії в одному напрямку. Інформаційні біти передаються послідовно один за одним, звідси й назва інтерфейсу.

Послідовне передання даних може здійснюватися в синхронному й асинхронному режимах. При асинхронному переданні кожному байту даних передуює старт-біт, за ним передаються біти даних, після них може передаватися біт паритету (парності) і стоп-біт, що гарантує певну витримку часу між сусідніми пакетами даних (рисунок 3.1). Старт-біт наступного пакета даних може передаватися в будь-який момент часу, починаючи з моменту закінчення стоп-біта попереднього пакета даних, таким чином, між передачами можуть бути паузи довільної (і необмеженої) тривалості.



Рисунок 3.1 – Структура послідовного передавання даних

Старт-біт дає змогу організувати просту синхронізацію приймача за сигналом від передавача. При цьому припускається, що приймач і передавач працюють на одній і тій же швидкості обміну (тобто мають одну й ту ж саму тактову частоту), яка вимірюється кількістю переданих бітів за секунду. Внутрішній генератор синхронізації приймача містить лічильник-подільник частоти, який обнуляється в момент початку старт-біта від передавача. Цей лічильник формує внутрішні строби F_c , за якими приймач фіксує прийняті біти. В ідеальному випадку ці внутрішні строби мають припадати на середину бітових інтервалів, однак через різницю швидкостей приймача і передавача виникає неузгодженість, що нагромаджується. Для безпомилкового прийняття восьми бітів даних, одного біта паритету і стоп-біта гранична неузгодженість не може перевищувати 5 % на біт (в ідеалі строби припадають на середину біта, при відхиленні 50 % беремо біт з помилкою, $50\% / 10 \text{ біт} = 5\%$). Очевидно, що чим більшою є швидкість передавання, тим суворішими будуть вимоги до синхронізації приймача й передавача, крім того, зі збільшенням швидкості обміну посилюється вплив фронтів імпульсів.

Для асинхронного режиму прийнято кілька стандартних швидкостей обміну: 50, 75, 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 і 115200 біт/с.

Кількість бітів даних встановлюється програмно і може бути 5, 6, 7 і 8 бітів. Кількість стоп-бітів може бути 1, 1.5, 2 (за довжиною). Біта паритету може не бути.

Таким чином, для забезпечення зв'язку при асинхронному режимі необхідно встановити однаковими такі параметри приймача й передавача: швидкість обміну, кількість бітів даних, наявність біта паритету, тип паритету (парність – непарність), довжина стоп-біта.

Асинхронний обмін у ПК реалізується з допомогою СОМ-порту з використанням протоколу RS-232C.

При синхронному режимі передавання даних передбачається постійна активність каналу зв'язку. Пакет даних починається з передавання синхробайта, за яким послідовно передаються біти. Якщо передавач не має даних для передавання, то він заповнює паузу безперервним передаванням старт-бітів. Очевидно, що при синхронному обміні можна добитися великих швидкостей, тому що не потрібно змінювати старт-біт, стоп-біт, біт паритету (в асинхронному на кожні вісім бітів даних припадає по 3–4 службових біти). При синхронному обміні необхідна зовнішня синхронізація, оскільки передається великий масив даних. При

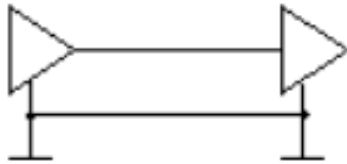

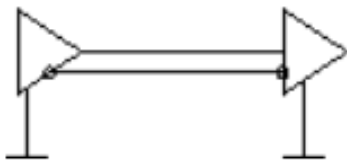
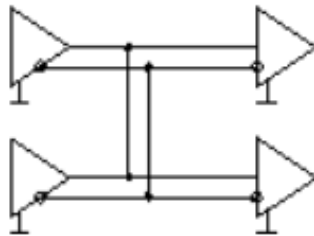
синхронізації тільки на початку передання блоку даних навіть невелика різниця частот призведе до виникнення помилок, що швидко нагромаджуються.

Зовнішня синхронізація може забезпечуватися шляхом як використання окремої лінії для передання сигналу синхронізації, так і застосування самосинхронізаційних методів кодування даних (манчестерський код, NRZ-код), при використанні яких синхросигнал виокремлюється приймачем з потоку прийнятих даних.

3.2 Фізичні реалізації послідовного інтерфейсу

Існує кілька споріднених послідовних інтерфейсів: RS – 232C, RS – 423A, RS – 422A, RS – 455A. Характеристики цих інтерфейсів зведено в таблицю 3.1.

Таблиця 3.1 – Характеристики інтерфейсів

Тип	Схема	Швидкість V Довжина L
RS-232C	Дуплекс 	L = 15 м, V = 20 кбит/с
RS-423A	Дуплекс 	L = 9 м, V = 100 кбит/с L = 91 м, V = 10 кбит/с L = 1200 м, V = 1 кбит/с
RS-422A	Дуплекс 	L = 12 м, V = 10 Мбит/с L = 120 м, V = 1 Мбит/с L = 1200 м, V = 100 кбит/с
RS-485A	Напівдуплекс (до 32 паралельно під'єднаних приймачів) 	L = 12 м, V = 10 Мбит/с L = 120 м, V = 1 Мбит/с L = 1200 м, V = 100 кбит/с

Лінії інтерфейсів RS-232C і RS-423A є несиметричними, мають найнижчу захищеність від синфазної завади. RS-423A має приймач з диференціальним входом, що трохи підвищує його завадозахищеність. Кращі параметри мають симетричний двобічний інтерфейс RS-422A і його напівдуплексний магістральний аналог RS-485A. Приймач і передавач RS-422A і RS-485A мають диференціальні входи і, отже, високу захищеність від синфазних завад. Інтерфейс RS-485A має передавач підвищеної потужності із захистом від короткого замикання лінії, перегріву при тривалому перевантаженні й колізій (одночасної роботи кількох паралельно ввімкнених передавачів).

Одним з варіантів послідовного інтерфейсу є інтерфейс типу «струмова петля», у якому сигналом є не рівень напруги, а струм у двопровідній лінії. Зазвичай за одиницю беруть струм 20 мА, за нуль – відсутність струму. При такому варіанті інтерфейсу приймач може розпізнавати обрив лінії. При обриві приймаються одні нулі й обрив розпізнається за відсутністю стоп-бітів. Зазвичай при «струмовій петлі» передбачається наявність гальванічної розв'язки приймача й передавача. Цей інтерфейс виконується з допомогою оптронів.

Струмова петля може живитися від передавача (варіант з активним передавачем) або від приймача (активний приймач). «Струмова петля» з гальванічною розв'язкою дає змогу передавати дані на відстань до кількох кілометрів, що визначається рівнем завад і опором пари проводів. Оскільки інтерфейс потребує пари проводів для кожного сигналу, то зазвичай застосовують дві пари – прийняті дані й дані, що передаються, а керування потоком проводиться за протоколом XON/XOFF. Одним з класичних прикладів інтерфейсу «струмова петля» є інтерфейс MIDI, що застосовується у звукових картах.

Усі перелічені вище варіанти реалізації інтерфейсу є основними, однак існують і бездротові варіанти, найбільшого поширення серед яких набув інфрачервоний (ІЧ) інтерфейс. Більшість ІЧ-інтерфейсів працюють на відстані від одного до кількох метрів на низькій (до 115,2 кбіт/с), середній (до 1.152 Мбіт/с) або високій (до 4 Мбіт/с) швидкості.

3.3 Інтерфейс RS-232C

Інтерфейс RS-232C призначено для під'єднання апаратури, що передає або приймає дані (КОД – кінцеве обладнання даних, або АПД – апаратура передання даних; DTE – Data Terminal Equipment), до кінцевої апаратури каналів даних (АКД; DCE – Data Communication Equipment). Як АПД може бути комп'ютер, принтер, плотер та інше периферійне устаткування. Як АКД зазвичай є модем. Кінцевою метою підімкнення є з'єднання двох пристроїв АПД. Повну схему з'єднання показано на рисунку 3.2, інтерфейс дає змогу вимкнути канал віддаленого зв'язку разом з парою пристроїв АКД, з'єднавши пристрої безпосередньо з допомогою нуль-модемного кабелю (рисунок 3.3).

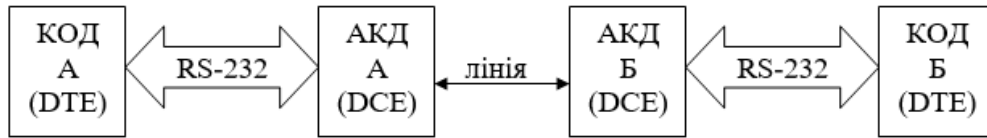


Рисунок 3.2 – Повна схема з'єднання з допомогою інтерфейсу RS-232C



Рисунок 3.3 – Скорочена схема з'єднання з допомогою інтерфейсу RS-232C

У стандарті описуються керувальні сигнали інтерфейсу, пересилання даних, електричний інтерфейс і типи рознімів, а також у ньому передбачено асинхронний і синхронний режими обміну, але СОМ-порти підтримують лише асинхронний режим. Функціонально RS-232C є еквівалентним стандарту МККТТ V.24/V.28 і стику С2, але вони мають різні назви сигналів.

У стандарті RS-232C описуються несиметричні передавачі і приймачі – сигнал передається відносно загального проводу – «схемної землі» (симетричні диференціальні сигнали використовуються в інших інтерфейсах, наприклад в RS-422). Інтерфейс не забезпечує гальванічної розв'язки пристроїв. Логічній одиниці відповідає напруга на вході приймача в діапазоні від -12 до -3 В. Для ліній керувальних сигналів цей стан позначається ON (увімкнено), для ліній послідовних даних – MARK. Логічному нулю відповідає діапазон від $+3$ до $+12$ В. Для ліній керувальних сигналів стан позначається OFF (вимкнено), а для ліній послідовних даних – SPACE. Діапазон від -3 до $+3$ В – це зона нечутливості, яка зумовлює гістерезис приймача: стан лінії буде вважатися зміненим тільки після перетину порогу (рисунок 3.4). Рівні сигналів на виходах передавачів мають знаходитися в діапазонах від -12 до -5 В і від $+5$ до $+12$ В для подання одиниці й нуля відповідно. Різниця потенціалів між «схемними землями» (SG) з'єднаних пристроїв має бути меншою від 2 В, при більш високій різниці потенціалів можливе неправильне прийняття сигналів.

При застосуванні інтерфейсу передбачається наявність захисного заземлення для пристроїв, що з'єднуються, якщо вони обидва живляться від мережі змінного струму й мають мережні фільтри.

Підімкнення та відімкнення телекомунікаційних пристроїв з автономним живленням має проводитися при відімкненому живленні. Інакше різниця невіривняних потенціалів пристроїв у момент комутації може виявитися прикладеною до вихідних або входних (що небезпечніше) ланцюгів інтерфейсу й вивести з ладу мікросхеми.

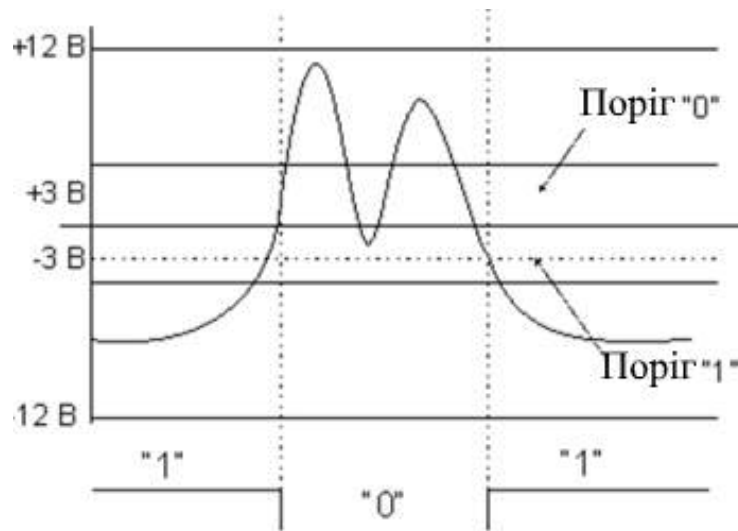


Рисунок 3.4 – Характеристика сигналу RS-232C

У стандарті RS-232C регламентуються типи вживаних рознімів.

На апаратурі АПД (у тому числі на СОМ-портах) прийнято встановлювати вилки DB-25P або більш компактний варіант – DB-9P. Дев'ятиштиркові розніми не мають контактів для додаткових сигналів, необхідних для синхронного режиму (у більшості для 25-штиркових рознімів ці контакти не використовуються).

На апаратурі АКД (модемах) установлюють розетку DB-25S або DB-9S.

За цим правилом передбачається, що розніми АКД можуть під'єднуватися до рознімів АПД безпосередньо або через перехідні «прямі» кабелі з розеткою й вилкою, у яких контакти з'єднано «один в один». Перехідні кабелі можуть бути й перехідниками з 9- на 25-штиркові розніми.

Якщо апаратура АПД з'єднується без модемів, то розніми пристроїв (вилки) з'єднуються між собою нуль-модемним кабелем (Zero-modem, або Z-modem) з розетками на обох кінцях, контакти яких з'єднано перехресно.

Якщо на якомусь пристрої АПД встановлено розетку – це майже 100 % того, що до іншого пристрою його необхідно під'єднувати прямим кабелем, аналогічним кабелю під'єднання модема. Розетка встановлюється зазвичай на тих пристроях, у яких віддалене під'єднання через модем не передбачено.

Підмножину сигналів RS-232C, що належать до асинхронного режиму, розглянемо з позиції СОМ-порту ПК. Для зручності будемо користуватися мнемонікою назв, прийнятою в описах СОМ-портів і більшості пристроїв. Нагадаємо, що активному стану сигналу «ввімкнено» і логічній одиниці переданих даних відповідає негативний потенціал (нижче від -3 В) сигналу інтерфейсу, а стану «вимкнено» і логічному нулю – позитивний (вище від $+3$ В). Призначення сигналів інтерфейсу наведено в таблиці 3.2. Нормальну послідовність керувальних сигналів при під'єднанні модема до СОМ-порту зображено рисунку 3.5.

Таблиця 3.2 – Характеристика сигналів RS-232C

Сигнал	Призначення
PG	Protected Ground – захисна земля, з'єднується з корпусом пристрою й екраном кабелю
SG	Signal Ground – сигнальна (схемна) земля, відносно якої діють сигнали
TD	Transmit Data – послідовні дані – вихід передавача
RD	Receive Data – послідовні дані – вхід приймача
RTS	Request To Send – вихід запиту передання даних: стан «увімкнено» повідомляє модем про наявність у терміналу даних для передання. У напівдуплексному режимі використовується для керування напрямком – стан «увімкнено» є сигналом модему на перемикання в режим передання
CTS	Clear To Send – вхід сигналу дозволу терміналу передавати дані. Стан «вимкнено» забороняє передання даних. Сигнал використовується для апаратного керування потоками даних
DSR	Data Set Ready – вхід сигналу готовності від апаратури передання даних (модем у робочому режимі є під'єднаним до каналу і таким, що завершив дії за узгодженням з апаратурою на протилежному кінці каналу)
DTR	Data Terminal Ready – вихід сигналу готовності терміналу до обміну даними. Стан «увімкнено» підтримує комутований канал у стані з'єднання
DCD	Data Carrier Detected – вхід сигналу виявлення головного віддаленого модема
RI	Ring Indicator – вхід індикатора виклику (дзвінка). У комутованому каналі цим дзвінком модем сигналізує про прийняття виклику

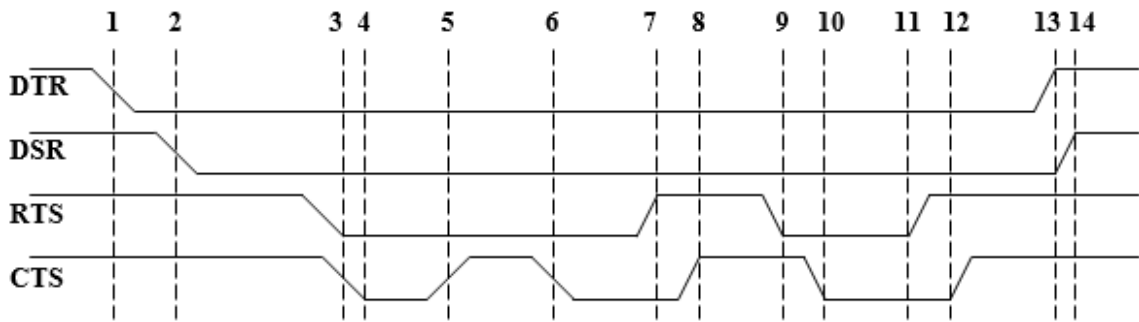


Рисунок 3.5 – Призначення сигналів інтерфейсу

- 1 Установленням DTR комп'ютер повідомляє про використання модему.
 - 2 Установленням DSR модем сигналізує про свою готовність і встановлення з'єднання.
 - 3 Сигналом RTS комп'ютер запитує дозвіл на передання й інформує про свою готовність брати дані від модема.
 - 4 Сигналом CTS модем повідомляє про свою готовність до приймання даних від комп'ютера й передання їх у лінію.
 - 5 Зняттям CTS модем сигналізує про неможливість подальшого приймання (наприклад, буфер заповнено) – комп'ютер має призупинити передання даних.
 - 6 Сигналом CTS модем дозволяє комп'ютеру продовжити передання (у буфері з'явилося місце).
 - 7 Зняття RTS може означати як заповнення буфера комп'ютера (модем має призупинити передання даних у комп'ютер), так і брак даних для передання в модем. У цьому випадку модем зазвичай припиняє пересилання даних у комп'ютер.
 - 8 Модем підтверджує зняття RTS скиданням CTS.
 - 9 Комп'ютер повторно встановлює RTS для поновлення передання.
 - 10 Модем підтверджує готовність до цих дій.
 - 11 Комп'ютер сигналізує про завершення обміну.
 - 12 Модем відповідає підтвердженням.
 - 13 Комп'ютер знімає DTR, що зазвичай є сигналом на розірвання з'єднання «повісити трубку».
 - 14 Модем скиданням DSR сигналізує про розірвання з'єднання.
- Розглянувши цю посліовність, можна зрозуміти з'єднання DTR – DSR і RTS – CTS в нуль-модемних кабелях.

3.4 Керування потоком даних

Для керування потоком даних можуть використовуватися два види протоколів – апаратний і програмний. Не слід плутати термін «керування потоком» з терміном «квитиювання». Квитиювання – пакет повідомлення про отримання

елемента, тоді як при керуванні потоком передбачається пакет повідомлення про тимчасову неможливість подальшого приймання даних.

Апаратний протокол керування RTS/CTS використовує сигнал CTS, який дає змогу зупинити передання даних, якщо приймач не готовий до роботи. Передання даних за цим протоколом показано на рисунку 3.6, *а*. Байт, який передається на момент надходження CTS, буде передано, проте з моменту закінчення його передання передавач переходить до очікування готовності приймача (тобто зняття CTS), забезпечує найшвидшу реакцію передавача на стан приймача, дає змогу організувати обмін, не застосовуючи буферизацію. Часто використовується в принтерах і для з'єднання комп'ютерів. Лінія CTS ПК має з'єднуватися з лінією RTS принтера, при з'єднанні двох ПК необхідно перехресне з'єднання CTS – RTS. Якщо апаратний протокол обміну не використовується, то на лінію CTS ПК необхідно подати сигнал «увімкнено», що зазвичай виконується з'єднанням CTS ПК з його ж RTS перемичкою на рознімі. Апаратний обмін є неможливим через мінімальний нуль-модемний кабель.

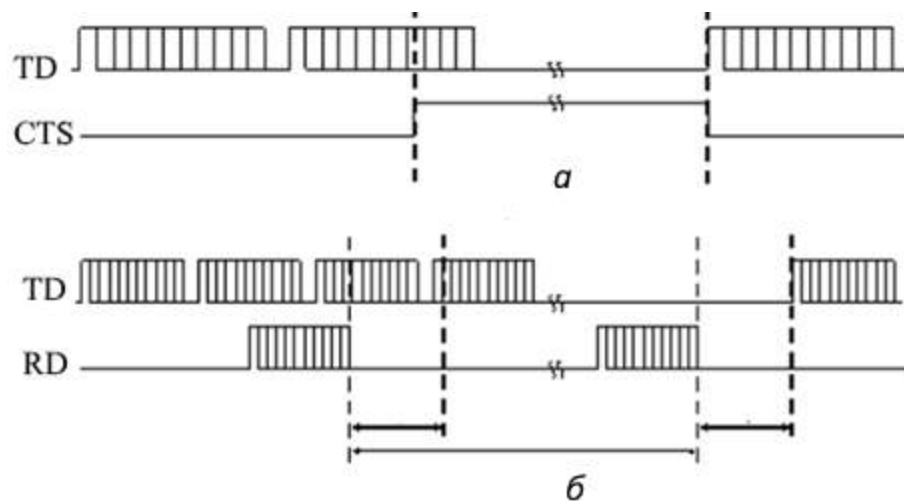


Рисунок 3.6 – Апаратний протокол керування (*а*), часові діаграми (*б*)

Апаратний протокол DTR/DSR є аналогічним RTS/CTS, але використовує іншу пару сигналів (у цьому випадку – «готовність приймача»).

У програмному протоколі **XON/XOFF** передбачається наявність двонаправленого каналу обміну, а також наявність у приймачі буфера, оскільки час реакції передавача t_p може виявитися досить великим, часові діаграми обміну показано на рисунку 3.6, *б*. Коли буфер приймача заповнюється до певного рівня (зазвичай 80–90 %), він передає на приймач команду XOFF (байт з кодом 13h). Прийнявши цю команду, передавач припиняє передання і переходить у стан очікування до приходу команди XON (байт з кодом 11h), за яким передавач відновлює передання.

Програмний протокол АСК. При обміні для отримання чергового байта приймач з цього протоколу посилає передавачу команду АСК (байт з кодом 6h). У

відповідь передавач посилає приймачу один байт (або пакет байтів певного розміру).

3.5 Робота з СОМ-портом на низькому рівні

Перетворення паралельного коду на послідовний під час передання й послідовного на паралельний під час контролю помилок, формування запитів переривання і деяких інших сервісних функцій здійснюється спеціалізованою мікросхемою, яка має назву UART (Universal Asynchronous Receiver – Transmitter), або УАП (універсальний асинхронний приймач). СОМ-порти ІВР РС базуються на мікросхемах, сумісних на рівні регістрів з UART і8250 – 8250/16450/16450А. Кожна з перелічених мікросхем є сумісною з попередньою, але не навпаки. Це слід урахувати при складанні програми керування: якщо програма орієнтована на і8250, то вона буде працювати з усіма подальшими модифікаціями УАП. У деяких вітчизняних ЕОМ застосовувалися мікросхеми КР580ВВ51 – аналог і8251, так званий УСАП (USART) – універсальний синхронно-асинхронний приймач. Ця мікросхема не є сумісною на рівні регістрів з і8250, однак у сучасних ПК її не застосовують.

Порівняльні характеристики УАП, сумісних з і8250, наведено в таблиці 3.3. Усі мікросхеми серії 8250 мають низьку швидкодію і не допускають звернення до своїх регістрів у суміжних циклах процесора, для коректної роботи між зверненнями організують програмні затримки (команда типу JMP\$ + 2).

Таблиця 3.3 – Порівняльні характеристики універсальних приймачів

УАП	Характеристика
8250	Уважається за базову, є найпримітивнішою за можливостями. Має помилки, що призводять до можливості виникнення помилкових переривань
8250А	Модернізована 8050, помилку усунено, непридатний в АТ при швидкості 9600 біт/с, є несумісним з ХТ BIOS
8250В	Виправлено помилки 8050 і 8050А, є сумісним з ХТ BIOS, працює у FN (крім 9600 Кбіт/с)
16450	Швидкодійна версія 8250 для АТ. Відповідає мінімуму при роботі з OS/2. Не має помилок, властивих для 8250
16550	Має буфер FIFO (з помилками, що не дають змоги його коректно використовувати), Може використовувати DMA для обміну
16550А	Нормально працює з 16-байтовим FIFO-буфером і DMA, які повинні застосовуватися для роботи на швидкостях 9600 біт/с і більше без утрати даних

З 16550А є сумісною переважна більшість мікросхем – контролерів СОМ-портів сучасних ПК. Особливістю 16550 і 16550А є те, що вони мають додаткові (порівняно з 8250) регістри. Багато бітів, що вважаються в 8250 резервом, у 16550

задіяно для керування його новими функціями. Однак усі регістри 8250 збігаються з відповідними регістрами 16550, що забезпечує сумісність.

Лекція 4

УНІВЕРСАЛЬНА ПОСЛІДОВНА ШИНА

4.1 Загальні відомості про USB

USB (Universal Serial Bus – універсальна послідовна шина) є промисловим стандартом розширення архітектури ПК, орієнтованим на інтеграцію з телефонією й пристроями побутової електроніки. Версію 1.0 було опубліковано в січні 1996 року. Архітектура USB визначається такими критеріями:

- легко реалізоване розширення периферії PC;
- дешеве рішення, що підтримує швидкість передання до 12 Мбіт/с;
- повна підтримка в реальному часі передання аудіо- та відеоданих;
- гнучкість протоколу змішаного передання ізохронних даних і асинхронних повідомлень;
- інтеграція з пристроями, що випускаються;
- доступність у PC усіх конфігурацій і розмірів;
- забезпечення стандартного інтерфейсу, що зможе швидко завоювати ринок;
- створення нових класів пристроїв, що розширюють PC.

Кінцевого користувача зазвичай приваблюють такі риси USB: простота кабельної системи й під'єднань, ПП, що самоідентифікується, автоматичний зв'язок пристроїв з драйверами й конфігураціями, можливість динамічного під'єднання і конфігурування ПП.

Із середини 1996 року випускаються ПК з убудованим контроллером USB, реалізованим чіпсетом. Уже з'явилися модеми, клавіатури, сканери, динаміки та інші пристрої введення-виведення з підтримкою USB, а також моніторів з USB-адаптерами, які виконують функцію концентраторів для під'єднання інших пристроїв.

4.2 Структура USB

USB забезпечує одночасний обмін даними між хост-комп'ютером і багатьма периферійними пристроями (ПП). Розподіл пропускної здатності шини між ПП планується хостом і реалізується ним шляхом посилення маркерів. Шина дає змогу підмикати, конфігурувати, використовувати й відмикати пристрої під час роботи хоста і самих пристроїв.

Пристрої (Device) USB можуть бути хабами, функціями або їх комбінацією. Хаб (Hub) забезпечує додаткові точки під'єднання пристроїв до шини. Функції (Function) USB надають системі додаткових можливостей, наприклад під'єднання до ISDN, цифрового джойстика, акустичних колонок з цифровим інтерфейсом і т. ін. Пристрій USB повинен мати інтерфейс USB, що забезпечує

повну підтримку протоколу USB, виконання стандартних операцій (конфігурація і скидання) і надання інформації, яка описує пристрій. Багато пристроїв, що під'єднуються до USB, мають у своєму складі і хаб, і функції. Роботою всієї системи USB керує хост-контролер (Host Controller), який є програмно-апаратною підсистемою хост-комп'ютера.

Фізичне з'єднання пристроїв здійснюється за топологією багаторівневої зірки. Центром кожної зірки є хаб, кожен кабельний сегмент з'єднує дві точки – хаб з іншим хабом або функцією. У системі є один (і тільки один) хост-контролер, розташований у вершині піраміди пристроїв і хабів. Хост-контролер інтегрується з кореневим хабом (Root Hub), що забезпечує одну або кілька точок під'єднання – портів. Контролер USB, що входить до складу чіпсетів, зазвичай має вбудований двопортовий хаб.

Функції являють собою пристрої, здатні передавати або приймати дані або керувальну інформацію по шині. Зазвичай функції являють собою окремі ПП з кабелем, що під'єднується до порту хаба. Фізично в одному корпусі може бути кілька функцій з убудованим хабом, що забезпечує їх під'єднання до одного порту. Ці комбіновані пристрої для хоста є хабами з постійно під'єднаними пристроями-функціями.

Кожна функція надає конфігураційну інформацію, що описує можливості ПП і вимоги до ресурсів. Перед використанням функція має бути налаштованою хостом – для неї має бути виділено смугу в каналі й вибрано опції конфігурації. Приклади функцій: вказівники – миша, планшет; пристрій введення – клавіатура або сканер; пристрій виведення – принтер, звукові колонки (цифрові); телефонний адаптер ISDN.

Хаб – ключовий елемент системи в архітектурі USB, є кабельним концентратором. Точки під'єднання називають портами хаба. Кожен хаб перетворює одну точку під'єднання на їх безліч. В архітектурі допускається з'єднання кількох хабів (рисунок 4.1).



Рисунок 4.1 – Архітектура під'єднання

Кожний хаб має один висхідний порт (Upstream Port), призначений для під'єднання до хоста або хаба верхнього рівня. Решта портів є спадними (Downstream Ports), призначеними для під'єднання функцій або хабів нижнього рівня. Хаб може розпізнати під'єднання пристроїв до портів або від'єднання від

них і керувати поданням живлення на їх сегменти. Кожен з портів може бути дозволеним або забороненим і налаштованим на повну або обмежену швидкість обміну. Хаб забезпечує ізоляцію сегментів з низькою швидкістю від високошвидкісних. Хаби можуть керувати поданням живлення на спадні порти, передбачається установка обмеження на струм, який споживає кожний порт.

Система USB поділяється на три рівні з певними правилами взаємодії. Пристрій USB містить інтерфейсну частину, частину пристрою й функціональну частину. Хост теж поділяється на три частини – інтерфейсну, системну й ПО пристрою. Кожна частина відповідає тільки за певне коло завдань.

4.3 Фізичний інтерфейс USB

У стандарті USB визначено електричні й механічні специфікації шини. Інформаційні сигнали й напруга живлення 5 В передаються по чотирипровідному кабелю. Використовується диференційний спосіб передавання сигналів D+ й D– по двох проводах. Рівні сигналів передавачів у статичному режимі мають бути нижчими від 0,3 В (низький рівень) або вищими від 2,8 В (високий рівень). Приймачі витримують вхідну напругу в межах $-0,5...+3,8$ В. Передавачі повинні вміти набирати високоімпедансного стану для двобічного напівдуплексного передавання по одній парі проводів.

Передавання по двох проводах в USB не обмежується диференціальними сигналами. Крім диференціального приймача кожен пристрій має лінійні приймачі сигналів D + й D–, а передавачі цих ліній керуються індивідуально. Це дає змогу розрізнити більше двох станів лінії, що використовуються для організації апаратного інтерфейсу. Стани Diff0 і Diff1 визначають за різницею потенціалів на лініях D+ й D– понад 200 мВ за умови, що на одній з них потенціал є вищим від порогу спрацювання VSE. Стан, при якому на обох входах (D+ й D–) є низький рівень, називають лінійним нулем (SEO – Single-Ended Zero). Інтерфейс визначає такі стани:

- 1) Data J State й Data Do State – стани переданого біта (або просто J й K), визначаються через стани Diff0 і Diff1;
- 2) Idle State – пауза на шині;
- 3) Resume State – сигнал «пробудження» для виведення пристрою зі «сплячого» режиму;
- 4) Start of Packet (SOP) – початок пакета (перехід з Idle State в K) ;
- 5) End of Packet (EOP) – закінчення пакета;
- 6) Disconnect – пристрій від'єднано від порту;
- 7) Connect – пристрій під'єднано до порту;
- 8) Reset – скидання пристрою.

Стани визначаються з'єднанням диференціальних і лінійних сигналів, для повної й низької швидкості стани Diff0 і Diff1 мають протилежне призначення. У декодуванні станів Disconnect, Connect й Reset урахується час перебування ліній (понад 2,5 мс) у певних станах.

Шина має два режими передавання. Повна швидкість передавання сигналів USB становить 12 Мбіт/с, низька – 1,5 Мбіт/с. Для повної швидкості використовується екранована кручена пара з імпедансом 90 Ом і довжиною сегмента до 5 м, для низької – некручений неекранований кабель до 3 м. Низькошвидкісні кабелі й пристрої є дешевшими за високошвидкісні. Одна й та сама система може одночасно використовувати обидва режими, перемикання для пристроїв здійснюється прозоро.

Для роботи з невеликою кількістю ПП не потребується висока швидкість. Швидкість, яку використовує пристрій, під'єднаний до конкретного порту, визначається хабом за рівнями сигналів на лініях D+ і D–, змінюють навантажувальними резисторами.

Переріз провідників вибирається відповідно до довжини сегмента для забезпечення гарантованого рівня сигналу й живильної напруги. У стандарті визначено два типи рознімів (рисунок 4.2, таблиця 4.1)

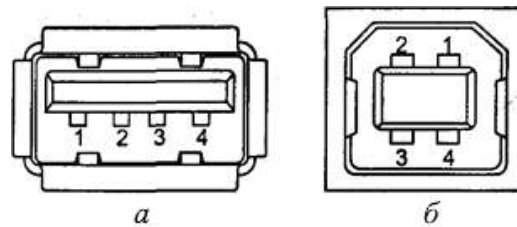


Рисунок 4.2 – Розніми Upstream Connector (а), Downstream Connector (б)

Розніми, зображені на рисунку 4.2, а (Upstream Connector), застосовуються для під'єднання до хабів. Вилки встановлюються на кабелях, що не від'єднуються від пристроїв (наприклад, клавіатура, миша тощо). Гнізда встановлюються на низхідних портах (Downstream Port) хабів. Розніми зображені на рисунку 4.2, б (Downstream Connector), встановлюються на пристроях, від яких з'єднувальний кабель може від'єднуватися (принтери й сканери). Відповідна частина (вилка) встановлюється на сполучному кабелі, протилежний кінець якого має вилку типу Upstream Connector.

Таблиця 4.1 – Стандарт рознімів

Контакт	Коло	Контакт	Коло
1	VBus	3	D+
2	D–	4	GND

Розніми зображені на рисунку 4.2, розрізняються механічно, що виключає неприпустимі петльові з'єднання портів хабів. Чотириконтактні розніми мають ключі, що виключають неправильне під'єднання. Конструкція рознімів забезпечує згодом з'єднання й раннє від'єднання сигнальних ланцюгів порівняно з живильними. Для розпізнавання розніму USB на корпусі пристрою ставлять стандартне символічне позначення.

Пристрої USB можуть живитися від кабелю (Bus-Powered Devices) або власного блока живлення (Self-Powered Devices). Хост забезпечує живленням безпосередньо під'єднані до нього ПП. Кожен хаб, зі свого боку, забезпечує живлення пристроїв, під'єднаних до його низхідних портів. При деяких обмеженнях топології допускається застосування хабів, що живляться від шини.

USB підтримує як одно-, так й двоспрямовані режими зв'язку. Дані передаються між програмним забезпеченням хоста й кінцевою точкою пристрою. Пристрій може мати кілька кінцевих точок, зв'язок з кожною з них (канал) встановлюється незалежно.

В архітектурі USB допускається чотири базових типи передання даних.

1 Керувальні послання (Control Transfers), які використовуються під час під'єднання для конфігурації й під час роботи керування пристроями. Протокол забезпечує гарантоване доставляння даних. Довжина поля даних керувального послання не перевищує 64 байти на повній швидкості й 8 байтів на низькій.

2 Суцільне передання (Bulk Data Transfers) порівняно великих пакетів без жорстких вимог до часу доставляння. Передання займають усю вільну смугу пропускання шини. Пакети мають поле даних розміром 8, 16, 32 або 64 байти. Пріоритет цього передання є найнижчим, цей процес може припинятися при великому завантаженні шини. Передання допускається тільки на повній швидкості.

3 Переривання (Interrupt) – короткі (до 64 байтів – на повній швидкості, до 8 байтів – на низькій) передання типу символів, що вводять, або координат. Переривання мають спонтанний характер і повинні обслуговуватися не повільніше, ніж того потребує пристрій. Межа часу обслуговування встановлюється в діапазоні 1...255 мс для повної швидкості й 10...255 мс – для низької.

4 Ізохронне передання (Isochronous Transfers) – безперервне передання в реальному часі, що займає попередньо узгоджену частину пропускну здатності шини й мають задану затримку доставляння. У разі виявлення помилки ізохронні дані передаються без повторення – недійсні пакети ігноруються. Приклад – цифрове передання голосу. Пропускна здатність визначається вимогами до якості передання, а затримка доставляння може бути критичною, наприклад, при реалізації телеконференцій.

Смуга пропускання шини поділяється між усіма встановленими каналами. Виділена смуга закріплюється за каналом, і якщо встановлення нового каналу потребує такої смуги, яка не «вписується» в розподіл, що вже існує, то запит на виділення каналу відкидається.

В архітектурі USB передбачається внутрішня буферизація всіх пристроїв, причому чим більшої смуги пропускання потребує пристрій, тим більшим має бути його буфер. USB має забезпечувати обмін з такою швидкістю, щоб затримка даних у пристрої, спричинена буферизацією, не перевищувала кількох мілісекунд.

Ізохронне передання класифікується за способом синхронізації кінцевих точок – джерел або одержувачів даних – із системою: розрізняють асинхронний,

синхронний й адаптивний класи пристроїв, кожному з яких відповідає свій тип каналу USB.

4.4 Протокол

Усі обміни (транзакції) по USB складаються з трьох пакетів. Кожна транзакція планується й починається з ініціативи контролера, який посилає пакет-аркер (Token Packet). У ньому описується тип і напрям передання, адреса пристрою USB і номер кінцевої точки. У кожній транзакції обмін є можливим тільки між пристроєм, що адресується (його кінцевою точкою), і хостом. Пристрій, що адресується маркером, розпізнає свою адресу й готується до обміну. Джерело даних (визначене маркером) передає пакет даних (або повідомлення про відсутність даних, призначених для передання). Після успішного приймання пакета приймач даних посилає пакет-підтвердження (Handshake Packet).

Планування транзакцій забезпечує керування поточними каналами. На апаратному рівні використання відмови від транзакції (NAck) при неприпустимій інтенсивності передання запобігає переповненню буферів зверху й знизу. Маркери відкинутих транзакцій повторно передаються у вільний для шини час. Керування потоками дає змогу гнучко планувати обслуговування одночасних різнорідних потоків даних.

Стійкість до помилок забезпечують такі властивості USB:

- висока якість сигналів, що досягається завдяки застосуванню диференціальних приймачів/передавачів та екранованих кабелів;
- захист полів керування й даних CRC-кодами;
- виявлення під'єднання й від'єднання пристроїв і конфігурація ресурсів на системному рівні;
- самовідновлення протоколу з тайм-аутом під час утрати пакетів;
- керування потоком для забезпечення ізохронності й керування апаратними буферами;
- незалежність функцій від невдалих обмінів з іншими функціями.

Для виявлення помилок передання кожен пакет має контрольні поля CRC-кодів, що дають змогу виявляти всі одиничні й подвійні бітові помилки. Апаратні засоби виявляють помилки передання, а контролер автоматично виробляє трикратну спробу передання. Якщо повтори є невдалими, то повідомлення про помилку передається клієнтському ПЗ.

4.5 Функції та хаби

Можливості шини USB дають змогу використовувати її для під'єднання різноманіпних пристроїв. Усі пристрої мають підтримувати набір загальних операцій, перелічених нижче, динамічне під'єднання й від'єднання. Ці події відстежуються хабом, який повідомляє про них хост-контролер і скидає під'єднаний пристрій. Пристрій після сигналу скидання має відгукуватися на

нульову адресу, при цьому він не є налаштованим і не припиняє роботу. Після призначення адреси, за яку відповідає хост-контролер, пристрій має відгукуватися тільки на свою унікальну адресу.

Конфігурація пристроїв, що виконується хостом, є необхідною для їх використання. Для конфігурації зазвичай використовується інформація, зчитана із самого пристрою. Пристрій може мати безліч інтерфейсів, кожному з яких відповідає власна кінцева точка, що надає хосту функції пристрою. Інтерфейс у конфігурації може мати альтернативні набори характеристик; змінення наборів підтримується протоколом. Для підтримки адаптивних драйверів дескриптори пристроїв та інтерфейсів мають поля класу, підкласу й протоколу.

Передача даних можлива з допомогою одного з чотирьох типів передач. Для кінцевих точок, для яких дозволяються різні типи передачі, після конфігурації доступним є тільки один з них.

Керування енергоспоживанням є дуже розвинутою функцією USB. Для пристроїв, що живляться від шини, потужність є обмеженою. Будь-який пристрій при ввімкненні не повинен споживати від шини струм, що перевищує 100 мА. Робочий струм (не більше 500 мА) заявляється в конфігурації, і якщо хаб не зможе забезпечити пристрою заявлений струм, пристрій не конфігурується і, отже, не може використовуватися.

Пристрій USB має підтримувати припинення (Suspended Mode), у якому його споживаний струм не перевищує 500 мкА. Пристрій повинен автоматично зупинятися при припиненні активності шини.

Можливість віддаленого пробудження (Remote Wakeup) дає змогу призупиненому пристрою подати сигнал хост-комп'ютеру, який теж може перебувати в призупиненому стані. Можливість віддаленого пробудження описується в конфігурації пристрою. При конфігуруванні ця функція може бути забороненою.

Хаб в USB виконує комутацію сигналів і видачу напруги живлення, а також відстежує стан під'єднаних до нього пристроїв, повідомляючи хост про зміни. Хаб складається з двох частин – контролера (Hub Controller) і повторювача (Hub Repeater). Повторювач являє собою керований ключ, що з'єднує вихідний порт зі вхідним. Він має засоби підтримки скидання й припинення передачі сигналів. Контролер містить регістри для взаємодії з хостом. Доступ до регістрів здійснюється за специфічними командами звернення до хаба. Команди дають змогу конфігурувати хаб, керувати спадними портами й спостерігати їх стан.

Спадні (Downstream) порти хабів можуть перебувати в таких станах:

1 Powered (живлення відімкнено) – на порт не подається живлення (є можливим тільки для хабів, що комутують живлення). Вихідні буфери переводяться у високоімпедансний стан, вхідні сигнали ігноруються.

2 Disconnected (від'єднаний) – порт не передає сигнали ні в одному напрямку, але може виявити під'єднання пристрою (якщо немає стану SEO протягом 2,5 мкс). Тоді порт переходить у стан Disabled, а за рівнями вхідних сигналів (Diff0 або Diff1 у стані Idle) визначає швидкість під'єданого пристрою.

3 Disabled (заборонений) – порт передає тільки сигнал скидання (за командою від контролера), сигнали від порту (крім виявлення відімкнення) не сприймаються. При виявленні відімкнення (2,5 мкс стану SEO) порт переходить у стан Disconnect, а якщо відімкнення виявлено «сплячим» хабом, то контролеру буде надіслано сигнал Resume.

4 Enabled (дозволений) – порт передає сигнали в обох напрямках. За командою контролера або після виявлення помилки кадру порт переходить у стан Disabled, а після виявлення відімкнення – у стан Disconnect.

5 Suspended (припинений) – порт передає сигнал переведення в стан припинення роботи («сплячий» режим). Якщо хаб перебуває в активному стані, то сигнали через порт не пропускаються ні в одному напрямку. Однак «сплячий» хаб сприймає сигнали змінення стану незаборонених портів, подаючи «пробуджувальні» сигнали від пристрою, що активізувався навіть через ланцюжок «сплячих» хабів. Стан кожного порту ідентифікується контролером хаба з допомогою окремих регістрів. Є загальний регістр, біти якого відображають факт змінення стану кожного порту (що фіксується під час EOF). Це дає змогу хост-контролеру швидко дізнатися про стан хаба, а в разі виявлення змін спеціальними транзакціями уточнити стан.

4.6 USB-пакети

Дані по USB-шині завжди передаються спочатку молодшими бітами. USB-пакет складається з таких полів.

1 **Поле синхронізації (Sync Field)**. Усі пакети мають починатися з поля синхронізації. Поле синхронізації має розмір 8 бітів для низькошвидкісних і повношвидкісних пристроїв або 32 біти для високошвидкісних пристроїв і використовується для підсинхронізації тактового генератора, убудованого в USB-контролер. Останні два біти поля синхронізації є маркерами, які використовуються для ідентифікації кінця області синхронізації й початку PID-поля.

2 **Поле ідентифікатора пакета (PID Field)**. Ідентифікатор пакета знаходиться безпосередньо після поля синхронізації в кожному переданому USB-пакеті. PID складається з чотирибітових типів пакета, який розташовано за чотирибітовим перевірочним полем, як показано на рисунку 4.3.

(LSB)							(MSB)
PID0	PID1	PID2	PID3	<i>PID0</i>	<i>PID1</i>	<i>PID2</i>	<i>PID3</i>

Рисунок 4.3 – Поле ідентифікатора пакета

Контрольна область PID генерується як інверсія чотирибітових типів пакета і є необхідною для усунення помилкового декодування даних, що йдуть за цим полем. Помилка PID детектується в тому випадку, коли інвертована контрольна область не збігається з відповідними бітами ідентифікатора пакета. Якщо PID отримано з помилкою або його не зазначено в таблиці 4.2, то одержувач пакета його ігнорує.

Таблиця 4.2 – Характеристика PID

Тип PID	Ім'я PID	PID[3...0]	Характеристика
Ознака (Token)	Out	0001	Адреса + номер кінцевої точки для транзакції ХОСТ-> Функція
	In	1001	Адреса + номер кінцевої точки для транзакції ХОСТ<- Функція
	SOF	0101	Маркер початку пакета й його номер
	SETUP	1101	Адреса + номер кінцевої точки для SETUP-транзакції
Дані	DATA0	0011	Парний PID пакета даних
	DATA1	1011	Непарний PID пакета даних
Статус	ACK	0010	Приймач прийняв пакет даних без помилок
	NAK	1010	Приймальний пристрій не може прийняти дані або передавальний не може їх відправити
	STALL	1110	Кінцева точка призупинена (HALT) або запит по керувальному каналу не підтримується
Спеціальний	PRE	1100	ХОСТ ініціює преамбулу, дозволяючи трафік для низькошвидкісних пристроїв

PID-біти наведено в таблиці 4.2 в MSB-порядку. Коли вони передаються по шині, правий біт (біт 0) видається першим.

3 Поле адреси пристрою (Addr Field). Це поле використовується для ідентифікації пристрою, до якого спрямовано поточний пакет. Розмірність поля – сім бітів, що дає змогу адресувати 127 унікальних USB-пристроїв. Після скидання або ввімкнення живлення адреса пристрою встановлюється за замовчуванням у «0» і має бути запрограмованою хостом під час процесу енумерації. Адресу «0» (задану за замовчуванням) зарезервовано для пристроїв, що знову під'єднуються, і не може бути призначено для нормальної роботи.

4 Поле адреси кінцевої точки (Endpoint Field) має розмірність чотири біти й дає змогу розміщувати в пристрої до 16 кінцевих точок. Усі функції мають підтримувати одну керувальну нульову кінцеву точку. Низькошвидкісні пристрої підтримують тільки три канали передання: керувальний канал, зв'язаний з нульовою кінцевою точкою, і два додаткових канали. Повношвидкісні пристрої підтримують максимум 16 кінцевих точок.

5 Поле номера кадру (Frame Number Field) є 11-бітове поле, яке інкрементується хостом при ініціюванні нового кадру, і починається заново з нуля при набутті максимального значення 0x7FF та використовується тільки для SOF- маркерів в кожному початку кадру.

6 Поле циклічного контролю за надмірністю (CRC). Циклічний контроль при надмірності (CRC) використовуються для захисту всіх полів, крім PID у маркерах і пакетах даних. Захист маркера й пакета даних забезпечує 100 % виявлення всіх одиничних й подвійних бітових помилок. Для маркерів передбачено п'яти бітове поле CRC, яке використовується для захисту полів ADDR і ENDP пакетів IN, SETUP, OUT або поля позначки часу маркера SOF. Для пакета даних використовується 16-бітовий поліном, що кодує все поле даних пакета.

7 Поле кінця пакета (EOP) являє собою сигнал закінчення пакета й установлюється на шині шляхом виставлення сигналу «Single Ended Zero» SE0 протягом двох бітових інтервалів, що йдуть за станом J, установленим на час одного бітового інтервалу.

4.7 Пріоритети передання даних по USB-шині

Усі операції з передання даних ініціюються хост-системою незалежно від того, чи приймає вона дані або пересилає в периферійний пристрій. Усі невиконані операції зберігаються у вигляді чотирьох списків за типами передання: ізохронне передання, передання переривань, передання керувальних команд, передання даних великих обсягів.

Списки постійно оновлюються новими запитами. Планування операцій з передання інформації відповідно до впорядкованих у вигляді списків запитів виконується хостом з мілісекундним інтервалом. На початку кожного такого інтервалу хост посилає по шині пакет SOF (Start Of Frame –

початок кадру), після чого починається обслуговування запитів зі списку ізохронних передач (тому що вони мають найвищий пріоритет).

Після того як всі запити з цього списку буде обслужено, хост-система переходить до списку операцій з передання переривань, потім до списку запитів на передання даних великого обсягу.

Після закінчення 90 % зазначеного мілісекундного інтервалу хост автоматично переходить до обслуговування запитів на передання керувальних команд незалежно від того, чи встиг він повністю обслужити інші три списки. Тим самим гарантується, що керувальному переданню завжди буде виділено не менше 10 % пропускної здатності шини. Якщо передання всіх керувальних пакетів буде завершено до закінчення виділеної для них частини інтервалу планування, то те, що залишилося, буде використано хостом для передання даних великого обсягу (до кінця зазначеного мілісекундного інтервалу).

Таким чином:

- ізохронне передання гарантовано отримує 90 % пропускної здатності шини;
- передання переривань займають решту ізохронних операцій частини цієї 90 % частки;
- для передання даних великого обсягу виділяється весь час, що залишився після ізохронного передання й передання переривань (як й раніше, у межах 90 % частки пропускної здатності);
- керувальним передавачам гарантується 10 % пропускної здатності.

Лекція 5 **ПОСЛІДОВНИЙ ІНТЕРФЕЙС – «ШИНА FIREWIRE»**

5.1 Загальні відомості про стандарт IEEE 1394

Стандарт для високопродуктивної послідовної шини (High Performance Serial Bus), що отримав офіційну назву IEEE 1394, було прийнято 1995 року. Метою було створення шини, яка не поступається аналогічним у вартості й підвищенні зручності під'єднання (завдяки переходу на послідовний інтерфейс). Стандарт базується на шині FireWire («вогнений провід»), яку використовує Apple Computer як дешеву альтернативу SCSI у комп'ютерах Macintosh і PowerMac. Іншу назва цього інтерфейсу – iLink (іноді – Digital Link) – використовує фірма Sony для пристроїв побутової електроніки. MultiMedia Connection – ім'я, яке використовується в логотипі 1394 High Performance Serial Bus Trade Association (1394ТА).

Стандарт підтримує пропускну здатність шини на рівнях 100, 200, 400, 800 і 1600 Мбіт/с. Залежно від можливостей під'єднаних пристроїв одна пара з них може обмінюватися сигналами на швидкості 100 Мбіт/с, тоді як інша на тій же шині – на швидкості 400 Мбіт/с. Такі високі показники пропускної здатності послідовної шини майже виключають необхідність використання паралельних шин.

Основні властивості шини FireWire:

1 Багатофункціональність. Шина забезпечує цифровий зв'язок до 63 пристроїв без застосування додаткової апаратури (хабів). Пристрої побутової електроніки – цифрові камкордери (записувальні відеокамери), камери для відеоконференцій, фотокамери, приймачі кабельного й супутникового телебачення, цифрові відеоплеєри (CD і DVD), акустичні системи, цифрові музичні інструменти, а також периферійні пристрої комп'ютерів (принтери, сканери, пристрої дискової пам'яті) і самі комп'ютери – можуть об'єднуватися в єдину мережу.

2 Висока швидкість обміну й зхронного передання. Шина дає змогу навіть на початковому рівні (S100) передавати одночасно два канали відео (30 кадрів за одну секунду) стереозвук з якістю CD.

3 Низька ціна компонентів і кабелю.

4 Легкість установлення й використання. FireWire розширює технологію PnP. При застосуванні системи допускається динамічне (гаряче) під'єднання й від'єднання пристроїв.

5 16-розрядна адреса дає змогу адресувати до 64 К вузлів на шині.

6 Гранична теоретична довжина шини становить 224 метри.

Пристрої автоматично розпізнаються й конфігуруються при ввімкненні/вимкненні. Живлення від шини (струм до 1, 5А) дає змогу під'єднаним пристроям спілкуватись із системою навіть при відімкненні їх від живлення. Керувати шиною та іншими пристроями можуть не тільки комп'ютери, але й інші «інтелектуальні» пристрої побутової електроніки.

FireWire з ініціативи VESA позиціонується як шина «домашньої мережі», що об'єднує всю побутову й комп'ютерну техніку в єдиний комплекс. Ця мережа є тимчасовою (peer-to-peer), чим істотно відрізняється від USB.

Основні переваги шини FireWire:

1 Цифровий інтерфейс дає змогу передавати дані між цифровими пристроями без втрати інформації.

2 Використання кабелів малого діаметра й мініатюрних рознімів (4- або 6-шпиркові). Цікаво, що рознім було запозичено з комп'ютерної гри Nintendo Gameboy, оскільки показав високу зносостійкість в умовах нещадної експлуатації.

3 Простота конфігурування й широта можливостей. Шина дає змогу під'єднувати до 63 пристроїв без застосування концентраторів. На одному пристрої може бути до 27 рознімів для під'єднання до комп'ютера та інших пристроїв. Шина підтримує конфігурацію Plug & Play.

4 Підтримка «гарячого» (Fire) підімкнення й відімкнення. Автоматичне розпізнавання під'єднання й від'єднання апаратури й можливість робити це при працюючому комп'ютері, тобто навіть тоді, коли шина працює в повному режимі.

5 Невелика вартість для кінцевих користувачів.

6 Висока швидкість (100, 200 або 400 Мб/с для IEEE 1394a; 800 Мб/с для IEEE 1394b, у перспективі 1,6 і 3,2 Гб/с). Можливість оброблення мультимедіа-сигналу в реальному часі.

7 Пакетне передання даних. Мультимедійні дані, наприклад відеофільм, розбиваються на пакети з інтервалами між ними. Кількість пакетів визначається

довжиною фільму, що надсилається, а в інтервалах надсилається службова інформація, наприклад «Стоп» або «Пуск».

8 Підтримка асинхронного й ізохронного передання даних. При асинхронному переданні отримання кожного пакета даних перевіряється, і якщо його не отримано або прийнято з пошкодженням, передання повторюється й помилки виправляються.

9 Живлення зовнішніх пристроїв через кабель IEEE 1394.

10 Гнучка топологія – рівноправність пристроїв, при якій допускаються різні конфігурації.

11 Відкрита архітектура – немає необхідності використовувати спеціальне програмне забезпечення.

5.2 Компоненти FireWire

Функціональну схему інтерфейсу Firewire показано на рисунку 5.1. Тут знизу знаходиться фізичний рівень, на якому відбувається переведення мультимедійних сигналів, що стікуються в комп'ютерні формати, або, навпаки, формування, кодування/декодування й арбітраж, що визначає, у якому порядку пристрої FireWire однієї мережі можуть працювати.

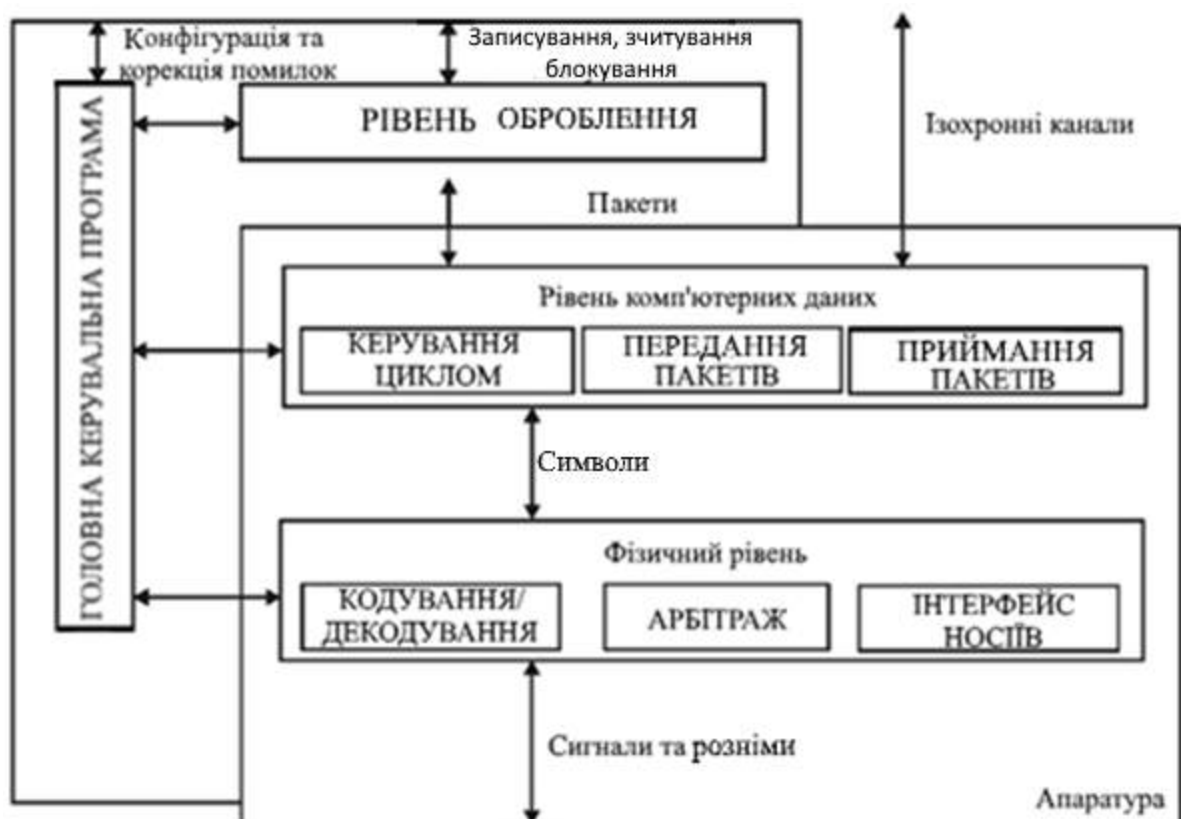


Рисунок 5.1 – Функціональна схема інтерфейсу Firewire

На рівні компонування даних обробляються й формуються пакети даних, організовується їх приймання й передання. Цих рівнів достатньо для ізохронного

передання даних, коли контроль за переданою й одержуваною інформацією не ведеться. При асинхронному переданні даних такий контроль проводиться на програмному рівні оброблення, де дані перевіряються й відправляються споживачеві, якщо помилок не виявлено. В іншому випадку процедури на нижньому рівні повторюються до усунення помилок.

Фізичний рівень може містити кілька рознімів FireWire, причому два будь-яких пристрої IEEE 1394 можуть з'єднуватися між собою за схемою «точка – точка» (point-to-point).

Кабельна мережа збирається за простими правилами – усі пристрої з'єднуються один з одним кабелями за будь-якою топологією (деревоподібною, ланцюговою, зіркоподібною). Кожний «повнорозмірний» пристрій (вузол мережі) зазвичай має три рівноправних сполучних розніми. Деякі малогабаритні пристрої можуть мати тільки один рознім, що обмежує можливі варіанти їх розташування. У стандарті допускається й до 27 рознімів на одному пристрої, що буде виконувати функцію кабельного концентратора.

Допускається безліч варіантів під'єднання пристроїв, але з такими обмеженнями:

- між будь-якою парою вузлів може бути не більше 16 кабельних сегментів;
- довжина сегмента стандартного кабелю не повинна перевищувати 4,5 м;
- сумарна довжина кабелю не повинна перевищувати 72 м (застосування більш якісного кабелю дає змогу послабити вплив цього обмеження);
- топологія не повинна мати петель.

Стандартний кабель 1394 містить шість проводів, укладених у загальний екран, і має однотипні 6-контактні розніми на кінцях (рисунки 5.2, а). Дві кручені пари використовуються для передання сигналів (ТРА і ТРВ) окремо для приймача й передавача, два дроти задіяно для живлення пристроїв (8–40 В, до 1,5 А). У стандарті передбачено гальванічну розв'язку пристроїв, для чого використовуються трансформатори (напруга ізоляції розв'язки до 500 В) або конденсатори (у дешевих пристроях з напругою розв'язки до 60 В відносно загального проводу). Деякі побутові пристрої мають тільки один 4-контактний рознім меншого розміру (рисунки 5.2, б), у якому реалізовано тільки сигнальні кола.

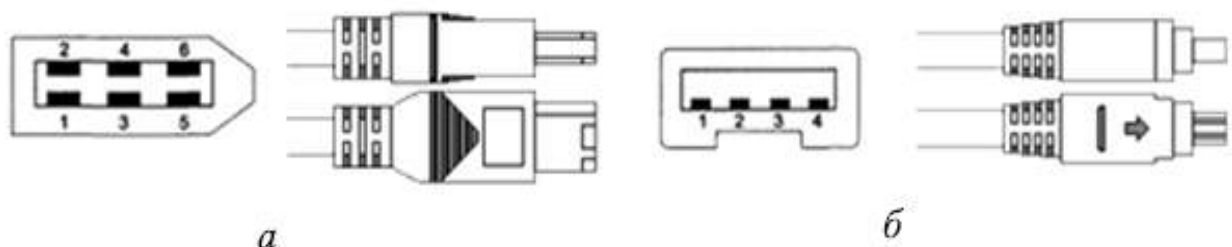


Рисунок 5.2 – Типи кабелів: 6-контактні (а) і 4-контактні (б)

Ці пристрої під'єднуються до шини через спеціальний перехідний кабель тільки як кінцеві (хоча можливе застосування спеціальних адаптерів-розгалужувачів). У кабелях FireWire сигнальні пари з'єднуються перехресно (таблиця 5.1), оскільки всі порти є рівноправними.

Таблиця 5.1 – Характеристики сигнальних пар

Рознім А			Провід	Характеристика	Рознім Б		
4-кон-тактний	6-кон-тактний	Коло			Коло	6-кон-тактний	4-кон-тактний
–	1	Power	Білий	Нерегульований DC; 30 В без навантаження	Power	1	
–	2	GND	Чорний	Земля живлення й внутрішній екран кабелю	GND	2	
1	3	TRB–	Помаранчевий	Скручена пара В, диференціальні сигнали	TPA–	5	3
2	4	TPB+	Синій	Скручена пара В, диференціальні сигнали	TPA+	6	4
3	5	TPA–	Червоний	Скручена пара А, диференціальні сигнали	TPB–	3	1
4	6	TPA+	Зелений	Скручена пара А, диференціальні сигнали	TPB+	4	2
Обплетення			Зовнішній	Екран кабелю	Обплетення		

У стандарті 1394 визначено дві категорії шин: кабельні шини та крос-шини (Backplane).

Шину backplane призначено для забезпечення паралельного передання даних, що є альтернативою послідовного передання даних між пристроями, під'єднаними до backplane. Кабельна шина являє собою деревоподібну мережу, що складається з шинних бриджив і вузлів (кабельні пристрої). Шестибітовий ідентифікатор імені вузла дає змогу мати до 63 вузлів, під'єднаних до одного шинного бриджа; 10-бітовий шинний ідентифікатор дає змогу мати до 1,023 бриджа в системі. Це означає, наприклад, що до 63 пристроїв може бути під'єднано до однієї карти адаптера 1394 в ПК.

Кожен вузол зазвичай має три розніми, хоча в стандарті передбачено від 1 до 27 рознімів на один пристрій рівня РНУ. До неї може бути під'єднано до 16 вузлів за схемою типу «ромашка» з допомогою кабелів завдовжки 4,5 м. При цьому

сумарна довжина кабелів дорівнює 72 м. Шина 1394 може розглядатися як шина plug-and-play.

Мережа може складатися з багатьох шин, з'єднаних мостами – спеціальними пристроями, що здійснюють передання пакетів між шинами, фільтрацію трафіка, а для з'єднання різнорідних шин ще й необхідне перетворення інтерфейсів. Інтерфейсна карта шини FireWire для PC являє собою міст PCI-1394. Мостами є також з'єднання кабельної шини 1394 з крос-шинами периферійних пристроїв. Мости можуть з'єднувати й кабельні шини, що розширює топологічні можливості з'єднання пристроїв.

Можливими є ізохронний й асинхронний обмін даними. Ізохронний режим передання шини 1394 забезпечує гарантовану смугу й визначає необхідний час затримки при високошвидкісному переданні через кілька каналів. При перезапуску шини або при ввімкненні ізохронного режиму вузол запитує смугу. Якщо потрібна смуга є недоступною, то пристрій буде виконувати періодично повторювані запити.

5.3 Протокол IEEE 1394

Протокол IEEE 1394 реалізується на трьох рівнях (рисунок 5.3).

1 Рівень транзакцій (Transaction Layer) перетворює пакети на дані, що надаються додаткам, і навпаки. Він реалізує протокол запитів-відповідей, що відповідає стандарту ISO/IEC 13213:1994 (ANSI/IEEE 1212, редакції 1994 р.) архітектури реєстрів керування й стану CSR (Control and Status Register) для мікрокомп'ютерних шин (читання, запис, блокування). Це полегшує зв'язок шини 1394 зі стандартними паралельними шинами.

2 Рівень зв'язку (Link Layer) з даних фізичного рівня формує пакети й виконує зворотні перетворення. Він забезпечує обмін вузлів датаграми з підтвердженнями. Рівень відповідає за передання пакетів і керування ізохронним переданням.

3 Фізичний рівень (Physical Layer) виробляє й приймає сигнали шини. Він забезпечує ініціалізацію й арбітраж. Припускається, що в будь-який момент часу працює тільки один передавач. Рівень передає потоки даних і рівні сигналів послідовної шини рівню, що знаходиться вище. Між цими рівнями можлива гальванічна розв'язка, при якій мікросхеми фізичного рівня живляться від шини. Гальванічна розв'язка необхідна для запобігання паразитним контурам загального проводу, які можуть виникати через захисний провід блоків живлення.

Апаратна частина FireWire зазвичай складається з двох спеціалізованих мікросхем – трансиверів фізичного рівня PHY Transceiver і моста зв'язку з шиною LINK Chip. Зв'язок між ними можливий, наприклад, по інтерфейсу IBM – Apple LINK-PHY. Мікросхеми рівня зв'язку виконують усі функції свого рівня й частину функцій рівня транзакцій, інша частина рівня транзакцій виконується програмно.

Для передання асинхронних повідомлень використовується 64-бітова адресація реєстрів пристроїв 1394. В адресі виділяється 16 бітів для адресації вузлів мережі: при 6-бітовому полі ідентифікатора вузла можна під'єднати до 63

пристроїв у кожній шині; при 10-бітовому полі ідентифікатора шини допускається використання в системі до 1023 шин різного типу (включаючи внутрішні), з'єднаних мостами. Протокол шини дає змогу звертатися до пам'яті (регістрів) пристроїв у режимі DMA. В адресному просторі кожного пристрою є конфігураційні регістри, у яких міститься вся інформація, необхідна для взаємодії з ним інших пристроїв. Дані передаються пакетами, на початку кожного пакета передаються біти стану арбітражу. Пристрій може передавати дані тільки після успішного проходження арбітражу. Є два основних типи передання даних – ізохронний, для якого й будувалася шина, та асинхронний. Ізохронне передання забезпечує гарантовану смугу пропускання й час затримки, асинхронне передання забезпечує гарантовану доставку.

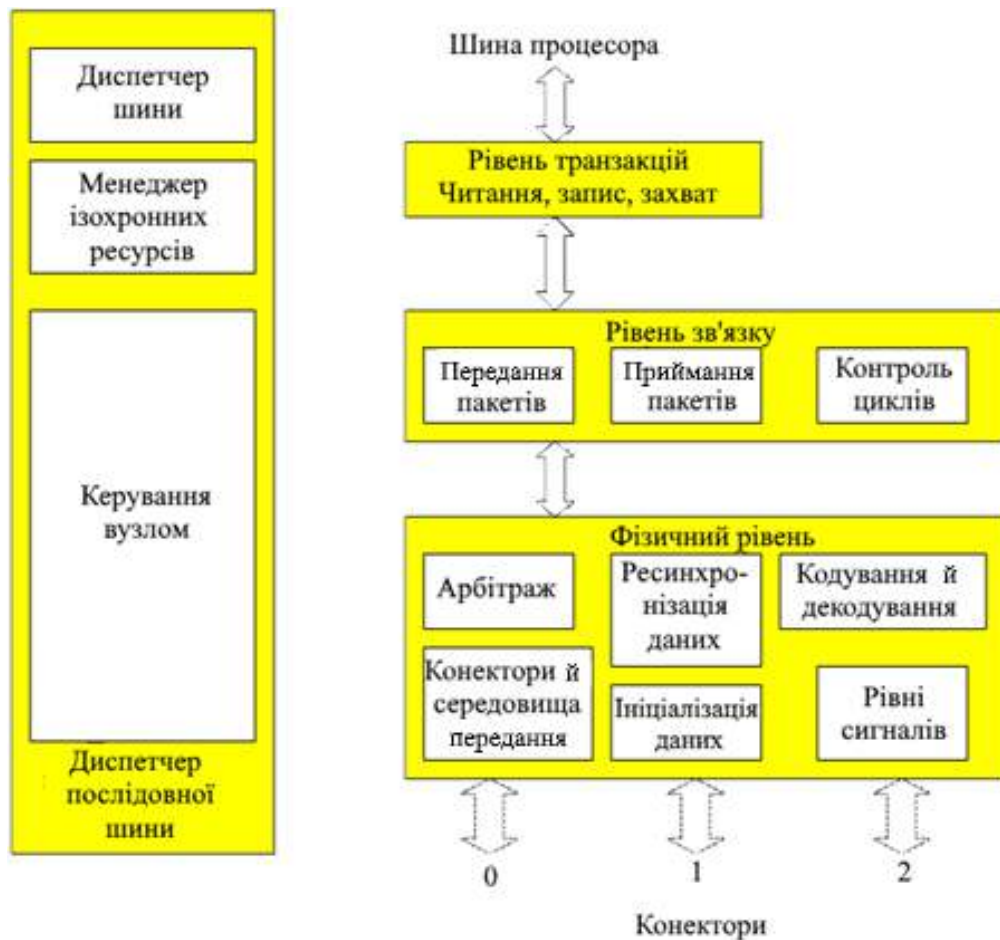


Рисунок 5.3 – Рівні реалізації протоколу

Асинхронні повідомлення передаються між двома пристроями. Ініціатор посилає запит необхідному пристрою, на який він відразу (через короткий проміжок часу, у якому шина знаходиться в спокої) відповідає підтвердженням приймання, позитивним (ACK) або негативним (NACK), якщо виявлено помилку даних. Змістовну відповідь на запит (якщо потрібно) буде передано назад аналогічним способом (одержувач має надіслати підтвердження). Якщо

підтвердження АСК ніхто не почув, передання буде повторюватися кілька разів до досягнення успіху або фіксації помилки.

Ізохронне передання ведеться широкомовно. У мережі може бути організовано до 64 ізохронних каналів, і кожен пакет ізохронного передання, крім власне даних, несе номер каналу. Цілісність даних контролюється CRC-кодом. Ізохронне передання всіх каналів «чують» усі пристрої шини, але з усіх пакетів приймають тільки дані, що цікавлять їх канали. Пристрій, що є джерелом ізохронних даних (камера, приймач, програвач) на етапі конфігурації отримує номер і параметри виділеного йому каналу.

Шина підтримує динамічну реконфігурацію – можливість «гарячого» під'єднання й від'єднання пристроїв. Коли пристрій умикається в мережу, він широкомовно передає короткий асинхронний пакет самоідентифікації. Усі вже під'єднані пристрої, прийнявши такий пакет, фіксують появу новачка й виконують процедуру скидання шини. При скиданні визначається структура шини, кожному вузлу призначається фізична адреса й проводиться арбітраж майстра циклів, диспетчера ізохронних ресурсів і контролера шини. Через секунду після скидання всі ресурси стають доступними для подальшого використання і кожен пристрій має повне уявлення про всі під'єднані пристрої і їх можливості. Від'єднання пристрою від шини також виявляється всіма пристроями. Завдяки наявності ліній живлення інтерфейсна частина пристрою може залишатися під'єднаною до шини навіть при відімкненні живлення функціональної частини пристрою.

Майстер циклів – це пристрій, що посилає кожні 125 мкс короткі широкомовні пакети початку циклів. У кожному такому пакеті майстер циклів передає значення 32-бітового лічильника часу, що інкрементує з частотою 24,576 МГц, для кожного вузла, що підтримує ізохронний обмін. У кожному циклі спочатку передається по одному пакету кожного активного ізохронного каналу, потім на деякий проміжок часу шина знаходиться в стані спокою. Після цього починається частина циклу, яку відводять для передання асинхронних пакетів. Кожен пристрій, що потребує асинхронного передання, у цій частині циклу може передати по одному пакету. Пристрій, що не має пакета для передання, не займає шину. Після того як всі пристрої, що потребують передання по одному пакету, за час, що залишився до кінця циклу пристрою, можуть передати й додаткові пакети.

Диспетчер ізохронних ресурсів – пристрій, який керує розподілом номерів каналів і смуги шини для ізохронного передання. Диспетчер потрібен, коли на шині є хоч один пристрій, здатний до ізохронного передання. Диспетчер можна вибрати з допомогою арбітражу з пристроїв, що підтримують ізохронний обмін. Після перезавантаження пристрої, що потребують ізохронного передання, запитують необхідну смугу. Смуга вимірюється в спеціальних одиницях розподілу, кількість яких у 125-мікросекундному циклі становить 6144. Одиниця займає близько 20, що відповідає часу передання одного квадліта (quadlet, 32-бітове слово) на частоті 1600 Мбіт/с. При такому способі вимірювання смуги враховується можливість спільної роботи пристроїв з різними швидкостями – в одному циклі сусідні пакети можуть передаватися на різних швидкостях. Щонайменше 25 мкс

циклу резервується під асинхронний трафік, тому сумарна смуга ізохронного трафіка становить 4915 одиниць.

Контролер шини (Bus Master) – необов'язковий елемент мережі 1394, який керує пристроями. Ним може бути комп'ютер або спеціальний інтелектуальний пульт керування. Контролер шини, який реалізує карту топології й швидкостей (TopologyMap й Speed_Map), може використовувати кілька частот в одній шині відповідно до можливостей конкретної пари пристроїв, що беруть участь в обміні. Інакше при під'єднанні пристроїв, розрахованих на різні швидкості, усі данні будуть передаватися на швидкості, доступній для всіх активних пристроїв.

5.4 Специфікація FireWire

IEEE 1394. Наприкінці 1995 року IEEE прийняв стандарт під порядковим номером 1394. У цифрових камерах Sony інтерфейс IEEE 1394 було встановлено раніше від прийняття стандарту й під назвою iLink. Інтерфейс спочатку позиціонувався для передання відеопотоків, але зацікавив і виробників зовнішніх накопичувачів, забезпечуючи високу пропускну здатність для сучасних високошвидкісних дисків. Сьогодні більшість системних плат, а також майже всі сучасні моделі ноутбуків підтримують цей інтерфейс.

Швидкість передання даних – 100, 200 й 400 Мбіт/с, довжина кабелю – до 4,5 м.

IEEE 1394a. 2000 року було затверджено стандарт IEEE 1394a і проведено кілька вдосконалень, що підвищило сумісність пристроїв. Було введено час очікування 1/3 секунди на скидання шини, поки не закінчиться перехідний процес установлення надійного під'єднання або від'єднання пристрою.

IEEE 1394b. 2002 року розроблено стандарт IEEE 1394b з новими швидкостями: S800 – 800 Мбіт/с й S1600 – 1600 Мбіт/с. Відповідні пристрої позначаються FireWire 800 або FireWire 1600 залежно від максимальної швидкості. Змінилися використовувані кабелі й розніми. Для набуття максимальних швидкостей на максимальних відстанях передбачено використання оптики, пластмасової – для довжин до 50 метрів, й скляної – для довжин до 100 метрів. Незважаючи на змінення рознімів, стандарти залишилися сумісними, чого можна добитися, використовуючи перехідники. 12 грудня 2007 року було представлено специфікацію S3200 з максимальною швидкістю – 3,2 Гбіт/с. Для позначення цього режиму використовується також назва «beta mode» (схема кодування 8B10B). Максимальна довжина кабелю може становити 100 метрів.

IEEE 1394.1. 2004 року розроблено стандарт IEEE 1394.1. Завдяки цьому стандарту можна значно збільшити кількість пристроїв, що під'єднуються до них.

IEEE 1394c, який було розроблено 2006 року, дає змогу використовувати кабель Cat 5e від Ethernet. Його можна застосовувати паралельно з Gigabit Ethernet, тобто використовувати дві логічні мережі на одному кабелі, що не залежать одна від одної. Максимальна заявлена довжина – 100 м, максимальна швидкість відповідає S800 – 800 Мбіт/с.

Лекція 6 ШИНА PCI. ПОРТ AGP

6.1 Основні відомості про шину PCI

Стандарт PCI розроблено й розповсюджується спеціальною групою взаємодії периферійних компонентів (Peripheral Component Interconnect Special Interest Group – PCI SIG), що є некопоративною асоціацією представників мікрокомп'ютерної промисловості.

Група PCI SIG поставила завдання розробити стандарт для апаратного розширення, який можна застосовувати на різноманітних платформах.

PCI (Peripheral Component Interconnect bus) – шина з'єднання периферійних компонентів, має особливе місце в ПК-архітектурі, будучи мостом між локальною шиною процесора й шиною введення-виведення ISA/EISA або MCA. Розроблено для Pentium-систем, добре поєднується й з 486 процесорами. Частота шини – 20...30 МГц, теоретично максимальна швидкість – 132/264 Мбайт/с для 32/64 бітів. Має версії з живленням 5 В, 3,3 В та універсальну. Ключами є пропущені ряди контактів 12, 13 і 50, 51. Для 5В-слота ключ розташований на місці контактів 50, 51; для 3,3 В – 12, 13; для універсального – два ключа: 12, 13 й 50, 51; 32-бітовий слот закінчується контактами A62/B62, 64-бітовий – A94/У94.

Технологія PCI дає змогу відтворювати кольорові відеофільми з високою роздільною здатністю в багатьох вікнах екрана, використовувати додаткові порти, автоматично встановлювати свою конфігурацію, а також гарантує найвищу швидкість обміну з жорсткими дисками.

Строго кажучи, PCI зовсім не є локальною шиною, а являє собою додаткову, або проміжну. PCI займає проміжний рівень між процесорною шиною системи й такими стандартними шинами розширення, як ISA, EISA або MCA, причому з'єднується з ними з допомогою електронних мостів (рисунок 6.1) Завдяки ізоляції від локальної шини центрального процесора шина PCI під'єднує більше пристроїв, ніж VL-BUS, оскільки ці пристрої не будуть мати електричного навантаження на шини центрального процесора. У специфікації PCI передбачено до десяти одиничних навантажень.

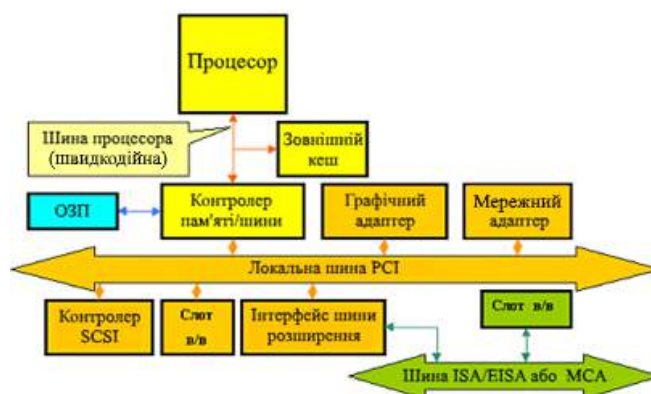


Рисунок 6.1 – Структура PCI

Якщо розв'язок процесора від шини PCI реалізовано не досить удамо, це може спричинити затримки під час виконання будь-яких операцій обміну між ЦП і пристроями, під'єднаними до шини PCI. Компанія Intel, яка розробила шину PCI, створила й набори мікросхем PCI, що дають змогу вирішити цю проблему завдяки раціонально організованій буферизації в мостовому інтерфейсі при виконанні операцій читання й запису. Набір IC компанії Intel дає змогу центральному процесору комп'ютера зробити запис (видачу) даних на периферійний пристрій, щоб контролер PCI негайно запам'ятав ці дані у своєму буфері, а ЦП міг швидко перейти до наступної операції, не чекаючи завершення виведення. Потім буфер сам передасть дані пристрою PCI в ефективному режимі групового обміну. Ізоляція шини PCI – це свого роду крок у майбутнє архітектури PC, оскільки тим самим забезпечується можливість роботи з більш швидкодіючими мікропроцесорами. Теоретично можна під'єднати шину PCI до 100 МГц МП Pentium або 200 МГц RISK-процесора Alpha, тоді як про пряме з'єднання істинної локальної шини з ЦП фактично не може йтися. Така незалежність від процесора є дуже важливою перевагою з позиції виробника системи, оскільки дає змогу зменшити витрати на розроблення завдяки тому, що одні й ті самі периферійні пристрої шини можна при мінімальних доробках використовувати з центральними процесорами кількох різних поколінь. Оскільки шина PCI не залежить від процесора, периферійна плата або IC цієї шини буде однаково добре працювати в машині з процесором 486, Pentium, Alpha, а згодом й в комп'ютері Macintosh з процесором PowerPC.

Автоконфігурування пристроїв (вибір адрес, запитів переривання, каналів DMA) підтримується засобами BIOS і його орієнтовано на технологію Plug and Play.

У стандарті PCI визначено для кожного слота конфігураційний простір розміром до 256 восьмибітових регістрів, не приписаних а ні до простору пам'яті, ані до простору введення-виведення. Доступ до них здійснюється за спеціальними циклами шини Configuration Read і Configuration Write, що виробляє контролер при звертанні процесора до регістрів контролера шини PCI, розташованого в його просторі введення-виведення.

6.2 Сигнали шини PCI

AD [31: 0] – мультиплексована шина адреси/даних. Адреса передається за сигналом FRAME, у наступних тактах передаються дані.

C/BE [3: 0] – команда/дозвіл звернення до байта. Команда, яка визначає тип чергового циклу шини (читання-запис пам'яті, введення-виведення або конфігураційне читання-запис, підтвердження переривання та інші) задається кодом у фазі адреси (за сигналом FRAME).

FRAME – індикатор фази адреси (інакше – передання даних).

DEVSEL – спроба ініціатора звернутися до основної пам'яті.

IRDY – готовність ініціатора до обміну даними.

LOCK – використовується для встановлення, обслуговування та звільнення захоплення ресурсу на PCI.

REQ [3: 0] – запит від PCI-майстра на захоплення шини (для слотів 3: 0).

GNT [3: 0] – дозвіл майстру на використання шини.

PAR – загальний біт паритету для ліній AD [31: 0] і C / BE [3: 0].

ParityER – сигнал про помилку паритету (від пристрою, що її виявив).

RST – скидання всіх регістрів у початковий стан.

IDSEL – вибір пристрою в циклах конфігураційного зчитування й запису.

SERR – системна помилка, що активізується будь-яким пристроєм PCI і викликає NMI.

REQ64 – запит на 64-бітовий обмін.

ACK64 – підтвердження 64-бітового обміну.

INTR A, B, C, D – лінії запитів переривання, циклічно зсуваються в слотах й спрямовуються на доступні лінії IRQ з допомогою конфігураційних регістрів. Запит відносно низького рівня дає змогу використовувати одну лінію кількома джерелами.

Clock – тактова частота шини.

Test Clock, TSTRES, TestDO, TestDI – сигнали для тестування адаптерів за інтерфейсом JTAG (на системній платі зазвичай не задіяно).

TSTMSLCT – переведення в режим тестування.

TRDY – готовність цільового пристрою до обміну даними.

STOP – запит цільового пристрою до ініціатора на припинення поточної транзакції. Сигнали шини зображено на рисунку 6.2.

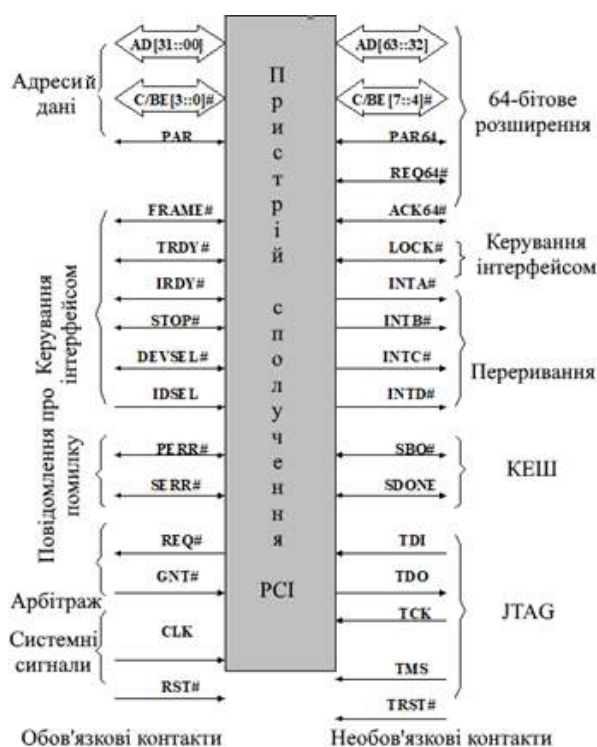


Рисунок 6.2 – Сигнали шини

Визначено два типи пристроїв стандарту PCI – цільовий і провідний. Цільовий пристрій сприймає команди й реагує на запити ведучого. Провідний пристрій являє собою більш «інтелектуальний» пристрій, який може проводити оброблення незалежно від шини або інших пристроїв. Провідний пристрій розділяє шину з основним процесором і цільовими пристроями. Крім того, він може бути цільовим пристроєм для інших провідних пристроїв. Визначення стандарту PCI потребує 47 контактів тільки для цільового й 49 контактів для ведучого пристрою. Ця кількість здається неймовірно малою, якщо врахувати потенційні можливості шини і той факт, що сюди включено функції передання даних та адресації, керування інтерфейсом, арбітражу, а також системні функції. Однак у специфікації передбачено до 120 з'єднань для стандартної 32-бітової і 184 для 64-бітової плат. Основою стандарту є мультиплексування, при якому через одні й ті самі контакти передаються різні типи сигналів. Адреси й дані мультиплексуються на одні й ті самі контакти, тому одиничне передання по шині PCI складається з фази адресації, що супроводжується однією або кількома фазами даних. Провідний пристрій видає адресу й звертається до конкуруючого пристрою на шині. Вибраний пристрій переходить у відповідний режим для приймання даних або інструкцій, а потім ведучий пристрій посилає пакет даних тим же контактам, які використовувалися для виклику. Після визначення адреси ведучий пристрій може посылати дані без повторення адресації, оскільки цільовий пристрій вже вибрано. Зазначимо, що передання даних може містити і читання, і запис інформації.

Для PCI визначаються три фізичних адресних простори: пам'яті, уведення-виведення й конфігурації. Адресація пам'яті й уведення-виведення є аналогічною застосовуваній у всіх шинах. Адресний простір конфігурації PCI призначено для вхідного визначення стандарту засобу автоматичної апаратної конфігурації.

Ще однією цікавою особливістю шини, що сприяє її спрощенню, є розподілене дешифрування адреси, коли кожний під'єднаний до локальної шини PCI пристрій виробляє дешифрування адреси. Завдяки цьому стають непотрібними схеми централізованого дешифрування адреси й сигнали вибору пристроїв, за винятком одного сигналу, призначеного для конфігурації.

6.3 Порт AGP

Фреймбуфер – це область пам'яті на відеокарті, і є тимчасовим сховищем для короткочасного зберігання одного або кількох відеокадрів.

Текстурними даними називають «матеріали», які відображаються на 3D-об'єктах (колір, фактура). Для зберігання текстур використовується спеціальна текстурна пам'ять, розташована на відеокарті.

Інформацію про «видимість» тривимірних поверхонь зберігає z-буфер, також розташований на відеокарті.

Отже, наприклад, для того щоб у тривимірній грі віртуальний вояк не розмахував даремно бластером, чекаючи змінення інтер'єру, необхідно якомога швидше «прокачати» інформацію про ці зміни між графічним акселерометром і відповідними областями пам'яті. У багатьох графічних підсистемах для цього

використовуються дорогі швидкодійні чіпи пам'яті, що збільшують і вартість комп'ютера. По-перше, для підвищення якості відображення 3D-графіки необхідно все більше й більше текстур, відповідно й текстурної пам'яті, тому розміщення їх у відеопам'яті продовжувало б збільшувати вартість графічної підсистеми. І, найголовніше, ця пам'ять свідомо втрачена для системи в цілому, оскільки має тільки цільове призначення (зберігання текстур). По-друге, текстурні дані є достатньо статичними за своєю природою, і їх все одно треба кудись помістити на період роботи конкретного 3D-додатка. Альтернативним вирішенням цієї проблеми стало би переміщення текстур з пам'яті відеокарти в основну пам'ять. Таке переміщення дало би змогу звільнити ресурси, які могли би використовуватися іншими елементами системи. Однак швидке накладення текстур на об'єкти в масштабі реального часу потребує значно великої пропускну здатності, що може надати шина PCI з портом AGP.

AGP (Accelerated Graphics Port) – спеціалізована надбудова над шиною PCI, що дає змогу створити швидкісний канал обміну даними між графічним акселератором і системною логікою ПК (рисунок 6.3). Синім кольором зображено переміщення текстурних даних з кадрового буфера в основну пам'ять. Це дає змогу оптимізувати потоки даних і зменшити вартість графічної підсистеми.

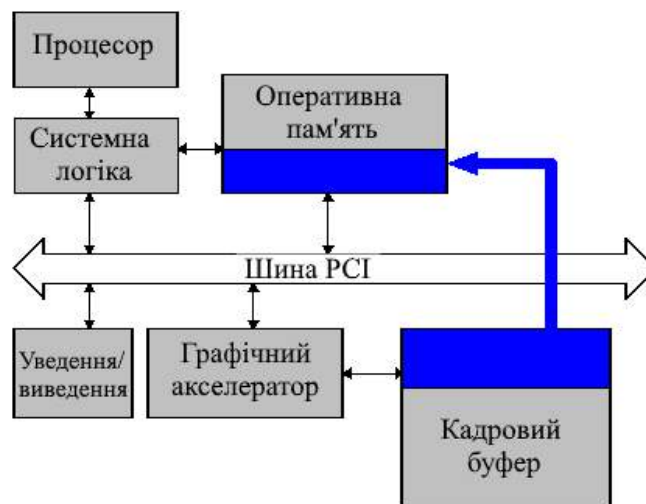


Рисунок 6.3 – Характеристика швидкісного каналу обміну

AGP – розширення основної PCI-архітектури, що ґрунтується на трьох китах: багатоканальній шині адреси, конвеєрному обробленні даних і робочій частоті шини 133 МГц. Для того щоб досягти високої швидкості передання, AGP визначено як безпосереднє або пряме з'єднання (point-to-point), а не через загальну шину (рисунок 6.4). Як видно, нова підсистема зв'язана з пам'яттю й процесором через системний логічний пристрій й призначена виключно для графіки, причому ця система підтримує графічний контролер як на материнській платі, так і на зовнішній графічній карті. Безпосереднє з'єднання двох кінцевих пристроїв, безсумнівно, спростить завдання синхронізації, послабить AGP-протоколи й усуне арбітражні втрати, що є характерними для шини PCI.

Як видно з рисунка 6.4, системна архітектура пропонованого порту, додаючи високошвидкісний маршрут між графічною підсистемою й системною пам'яттю, не впливає на будь-який інший периферійний порт. При цьому Intel намагається в трансляційному режимі зберегти значною мірою частину PCI-специфікації, особливо уникаючи використання будь-яких із зарезервованих областей і штирків у PCI-специфікації. Більшість PCI-сигналів використовується в AGP-командах, але порівняно з PCI додано ще 16 сигналів, що відповідають за розширені функції AGP. Специфічні AGP-протоколи, подібно до конвеєра, перекриваються на PCI таким чином, що для стандартного пристрою в шині PCI «надбудова» AGP є «прозорою». Проте всім AGP-пристроєм буде потрібно відповідати на PCI-запити, вони повинні оперувати єдиними сигналами, як визначено в специфікації PCI.



Рисунок 6.4 – Системна архітектура порту PCI

Незважаючи на сумісність, AGP і PCI призначено для різних пристроїв. Головна причина в тому, що AGP – не є шиною в широкому розумінні й підтримує єдиний графічний пристрій. Якби Intel намагалася сформувати AGP як розширений набір PCI, установлюючи AGP-пристрій у цій шині, то рішення мало би або обмежити AGP частотою 33 МГц, або зобов'язати всі PCI-пристрої підтримувати частоту 66 МГц, здорожуючи більшість периферійних пристроїв. Крім того, AGP визначає власний слот, що відрізняється від PCI, і багатоканальна адресація сигналів не повинна пристосовуватися до наявного PCI-слота. Значна сумісність з PCI-шиною дала змогу Intel розробити специфікацію на AGP дуже швидко.

Конвеєр даних – основне розширення протоколу, що передбачається в AGP. Однак тільки операції читання й запису, ініційовані відеоприскорювачем і спрямовані в основну пам'ять, поставлено на конвеєр. Системна логіка може мати доступ до графічного чіпа тільки за стандартами PCI-протоколу. Усі операції шини, включаючи читання й запис, звернення до графічного контролера, виконано

як стандартні PCI-запити. Конвеєрне оброблення даних, що визначається AGP, працює за методом поділу транзакцій (групові операції читання/запису).

Розглянемо це на прикладі читання із системної пам'яті. Графічний чіп видає запит на конвеєрне оброблення (запит доступу) у чіп системної логіки. Системна логіка відповідає на запит і забезпечує потім передання даних з основної пам'яті. Графічний чіп, не чекаючи отримання попередньої порції даних, видає наступний запит. Це перекриття результатів і забезпечує конвеєр різних запитів (читання або запису), причому черга запитів є завжди активною. Що особливо важливо, AGP-специфікація не нав'язує межі довжини черги запитів, ця межа виокремлюється можливостями системної реалізації. Для конвеєрного AGP-оброблення характерною є незв'язаність фаз адреси й даних (рисунок 6.5).

Інша суттєва відмінність між AGP і PCI – техніка подвійний синхронізації, що використовується для досягнення швидкості передання даних з тактовою частотою 133 МГц. Безпосереднє з'єднання графічного акселератора й чіпа системної логіки спростить не тільки електричну середу їх взаємодії, а й забезпечить гарне навантажувальне узгодження, таким чином, відкривається можливість здійснювати синхронізацію від імпульсів одного й того ж тактового генератора частотою 66 МГц як по передньому фронту імпульсу, так і по задньому.



Рисунок 6.5 – Системна реалізація

Завдяки цьому й виникає можливість синхронізації за стробами 66 і 133 МГц. При цьому передання даних на частотах 66 МГц є певною розрахунковою специфікацією 66 МГц шини PCI, а режим AGP-133 потребує додаткового інтерфейсу – Intel додає 8-розрядну адресну шину (SBA Bus), причому адреси й дані можуть демультимплексувати, а адресацію за шиною SBA призначено виключно для передання AGP-запитів доступу до системної пам'яті. На частоті 133 МГц AGP набуває максимальної пропускної здатності 533 Мбайт/с. Очевидно, що принцип подвійної синхронізації ставить особливі вимоги до якості імпульсів.

Текстурні дані, які планується завантажувати в основну пам'ять, мають бути безперервними з огляду на тривимірний додаток й графічний акселератора. Типова

текстура розмірністю 256 x 256 пікселів і 16 бітів на колір займає 128 Кбайтів. Intel: треба розташувати текстурні дані в системній пам'яті посторінково по 4 КБ, а інформацію про їх місцезнаходження помістити в якусь область пам'яті – карту текстури, а потім розподілити ці ділянки у фізичній пам'яті (тут простежується явна аналогія з формуванням FAT – file allocation table на вінчестераі). Таблиця такого переведення ВА у фізичну Intel назвав GART (Graphics Address Remapping Table). Intel в AGP установив чіп, у якому й реалізовано GART. Цей чіп названо 440LX, його розробляли для взаємодії з процесором Klamath. GART потенційно є досить великим. Опис чотирьох мегабайтів текстурних даних потребує 1024 записів по чотири байти кожна. Логічна специфікація AGP не описує, як GART буде формуватися цим чіпом. Натомість постачальник такого чіпа має забезпечити його драйвером, який керує GART згідно з викликами API (Application Programming Interface), що визначено в AGP-специфікації. Такий підхід має допомогти іншим постачальникам у розробленні AGP-пристроїв, сумісних з Intel на системному рівні.

Отже, AGP – не універсальне рішення, а засіб оптимізації відеографіки шляхом більш повної реалізації можливостей графічних акселераторів. Це нововведення ніяк не позначиться на швидкодії шини PCI.

Лекція 7

ШИНИ РОЗШИРЕННЯ ISA, EISA ТА PC/104

7.1 Основні властивості шини ISA

Шина ISA (Industrial Standard Architecture – промислова стандартна архітектура) використовувалася в першому комп'ютері IBM PC, виготовленому 1981 року, а 1984 року – у розширеному 16-розрядному варіанті в IBM PC/AT. Шина ISA – це основоположний базис архітектури персональних комп'ютерів. Її використовували аж до кінця 1990-х років. Здається дивним, що шина з такою архітектурою використовувалась у високопродуктивних комп'ютерах, що виготовлялися до кінця 1990-х років, але це пояснюється її надійністю, широкими можливостями й сумісністю. До того ж ця шина досі працює швидше від більшості периферійних пристроїв, що до неї під'єднуються.

Існує два варіанти шини ISA, що розрізняються кількістю розрядів даних: попередня 8-розрядна версія й нова 16-розрядна. Попередня версія працювала на тактовій частоті 4,77 МГц у комп'ютерах класів PC й XT. Нова версія використовувалася в комп'ютерах класу AT з тактовою частотою 6 і 8 МГц. Пізніше було досягнуто згоди про стандартну максимальну тактову частоту 8,33 МГц для обох версій шин, що забезпечило їх сумісність. У деяких системах допускається використання шин при роботі з більшою частотою, але не всі плати адаптерів витримують таку швидкість. Для передання даних по шині потрібно від двох до восьми тактів, тому максимальна швидкість ISA становить 8,33 Мбайт/с: $8,33 \text{ МГц} \times 16 \text{ біт} : 2 \text{ такти} = 66,64 \text{ Мбіт/с}$ (або 8,33 Мбайт/с). Смуга пропускання

8-розрядної шини є вдвічі меншою (4,17 Мбайт/с) від 16-розрядної. Однак – це теоретичні максимуми, оскільки через складний протокол обміну даними реальна пропускна здатність шини є набагато нижчою (зазвичай удвічі). Однак навіть у цьому випадку шина ISA працює швидше, ніж більшість під'єднаних до неї периферійних пристроїв.

8-розрядна шина ISA. Ця шина використовувалася в першому комп'ютері IBM PC. У нових системах її не застосовують, але вона досі експлуатується в сотнях тисяч комп'ютерів, у тому числі в системах на базі процесорів 286 і 386. У рознім вставляється плата адаптера з 62 контактами. На рознім подаються вісім ліній даних і 20 ліній адреси, що дає змогу адресувати до одного мегабайта пам'яті. Призначення й розташування контактів розніму 8-розрядної шини ISA показано на рисунку 7.1.

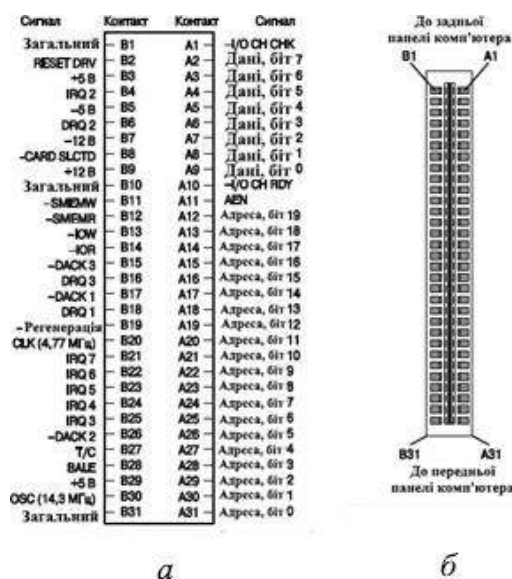


Рисунок 7.1 – Призначення (а – призначення сигналів) й розташування (б – загальний вигляд) контактів

Хоча ця шина є дуже простою, компанія IBM до 1987 року не публікувала її повного опису й часових діаграм сигналів на лініях даних та адреси. Тому при створенні плат адаптерів для перших IBM-сумісних комп'ютерів розробникам доводилося самим розбиратися в її роботі. У міру поширення IBM-сумісних комп'ютерів і їх перетворення на промисловий стандарт процес розроблення істотно спростився. Плата адаптера для 8-розрядної шини ISA має такі розміри: висота – 4,2 дюйма (106,68 мм); довжина – 13,13 дюйма (333,5 мм); товщина – 0,5 дюйма (12,7 мм).

16-розрядна шина ISA. Компанія IBM буквально «підірвала» світ ПК, представивши 1984 року модель AT, оснащену процесором 286. Цей процесор підтримував 16-розрядну шину даних, що давало змогу забезпечити взаємодію між процесором, системною платою й пам'яттю з використанням 16-розрядних, а не 8-розрядних даних. Хоча процесор і можна було встановити на системній платі з 8-розрядною шиною введення-виведення, все одно забезпечувалася підвищена

швидкодія при обміні даними з різними платами, що під'єднуються до шини. Замість того щоб створювати нову шину введення-виведення, IBM вирішила забезпечити сумісність системи з 8- й 16-розрядними адаптерами, залишивши той же 8-розрядний рознім, але додавши до нього ще один додатковий. Унаслідок цього було отримано рознім для установаження 16-розрядних адаптерів. Уперше представлена в комп'ютерах PC/AT у серпні 1984 року 16-розрядна шина ISA також мала назву шини AT. Додатковий рознім у кожному 16-розрядному рознімі розширення додає 36 контактів (загальна кількість контактів для передання даних при цьому збільшується до 98), необхідних для передання даних більшої розрядності. Крім того, було змінено призначення двох контактів 8-розрядної частини розніму. Однак подібні зміни ніяк не позначилися на роботоздатності 8-розрядних плат.

Звичайна плата адаптера класу AT має такі розміри: висота – 4,8 дюйма (121,92 мм); довжина – 13,13 дюйма (333,5 мм); товщина – 0,5 дюйма (12,7 мм). У комп'ютерах класу AT можуть стояти плати висотою як 4,8, так й 4,2 дюйма (відповідають попереднім платам для комп'ютерів класу PC/XT). Плати зі зменшеною висотою встановлювалися в комп'ютері класу XT моделі 286. У цій моделі із системною платою, призначеною для комп'ютера класу AT, використовувався корпус від XT, тому висоту плат адаптерів довелося зменшити до 4,2 дюйма. Після цього більшість виробників стали випускати тільки адаптери зі зменшеною висотою, які можна встановити в будь-який корпус.

32-розрядна шина ISA. Через деякий час після випуску 32-розрядного процесора було розроблено перші стандарти на відповідну шину. Ще до появи перших проектів архітектур MCA і EISA деякі компанії почали розробляти власні конструкції, що являють собою розширення архітектури ISA. Хоча їх було випущено порівняно небагато, деякі з них установажують навіть зараз. Додаткові лінії цих шин зазвичай використовувалися тільки при роботі з платами розширення пам'яті й відеоадаптерами, що випускаються компаніями, які створили цей стандарт. Їх параметри та розводки рознімів істотно відрізняються від стандартних, до того ж їх специфікації й схеми контактів не поширювалися у використанні.

7.2 Основні властивості шини EISA

З появою 32-розрядних мікропроцесорів 80386 (версія DX) фірми Compaq, NEC і деякі інші створили 32-розрядну шину EISA, повністю сумісну з ISA.

Шина EISA (Extended ISA) – жорстко стандартизоване розширення ISA до 32 бітів. Конструктивне виконання забезпечує сумісність з нею і звичайних ISA-адаптерів. Вузькі додаткові контакти розширення розташовано між ламелями розніму ISA і нижче таким чином, що адаптер ISA, який не має додаткових ключових прорізів у крайньому рознімі, не дістає до них. Установаження карт EISA в слоти ISA є неприпустимим, оскільки її специфічні кола потраплять на контакти кіл ISA, унаслідок чого системна плата виявиться нероботоздатною.

Розширення шини стосується не тільки збільшення розрядності даних та адреси: для режимів EISA використовуються додаткові керувальні сигнали, що забезпечують можливість застосування більш ефективних режимів передання. У звичайному (не пакетному) режимі передання за кожен пару тактів може бути передано до 32 бітів даних (один такт – на фазу адреси, один – на фазу даних). Максимальну продуктивність шини реалізує **пакетний режим (Burst Mode)** – швидкісний режим пересилання пакетів даних без вказання поточної адреси всередині пакета. У середині пакета чергові дані можуть передаватися в кожному такті шини, довжина пакета може становити 1024 байти. Для шини передбачено й більш продуктивні режими DMA, при яких швидкість обміну може становити 33 Мбайт/с. У лінії запитів переривань допускається розділене використання, причому зберігається й сумісність з ISA-картами: кожна лінія запиту може програмуватися на чутливість за перепадом, як в ISA, і за низьким рівнем. У шині допускається споживання кожною картою розширення потужності до 45 Вт, але повну потужність зазвичай не споживає жоден адаптер.

Кожен слот (максимум 8) і системна плата можуть мати селективний дозвіл адресації введення-виведення й окремі лінії запиту й підтвердження керування шиною. Арбітраж запитів виконує пристрій ISP (Integrated System Peripheral). Обов'язковою належністю системної плати до шини EISA є незалежна пам'ять конфігурації NVRAM, у якій зберігається інформація про пристрої EISA для кожного слота. Формат записів стандартизовано, для модифікації конфігураційної інформації застосовується спеціальна утиліта ECU (EISA Configuration Utility). Архітектура дає змогу при використанні програмної конфігурації адаптерів автоматично вирішувати конфлікти використання системних ресурсів програмним шляхом, але на відміну від специфікації PnP в EISA не допускається динамічного реконфігурування. Усі зміни конфігурації можливі тільки в режимі конфігурації, після виходу з якої необхідно перезавантажити комп'ютер. Ізольований доступ до портів введення-виведення кожною картою під час конфігурації забезпечується просто: сигнал AEN, що дає змогу декодувати адресу в циклі введення-виведення, на кожен слот надходить по окремій лінії AENx, у певний час програмно-керованою. Таким чином, можна окремо звертатись і до звичайних карт ISA, але ж це є даремним, оскільки карти ISA не підтримують обмін конфігураційною інформацією, передбачений у шині EISA. На деяких ідеях конфігурації EISA збільшилася специфікація PnP для шини ISA (формат конфігураційних записів ESCD багато в чому схожий на NVRAM EISA).

EISA – дорога архітектура, але вона виправдовує себе, оскільки застосовується в багатозадачних системах, на файл-серверах і скрізь, де потрібно високоефективне розширення шини введення-виведення.

7.3 Сумісність шин EISA та PCI

При вирішенні завдань, пов'язаних з обробленням високочастотних сигналів, як у лабораторних, так й в промислових умовах, правильніше зупинити свій вибір на платах збору даних у стандарті PCI. Сучасні пасивні шини з установленими активними мостами

PCI-PCI дають змогу встановлювати в комп'ютер до 17 PCI-плат розширення. У сукупності з процесорними платами на базі Pentium, Pentium II/III такий контролер зможе вводити й обробляти сигнали частотою понад 100 кГц, наприклад проводити аналіз спектра високочастотних сигналів у реальному часі. Така платформа є найбільш прийнятною для створення промислових серверів різного рівня й наукових комп'ютерних стендів.

Завдяки вдалому технічному рішенню сьогодні набула поширення гібридна шина ISA і PCI PCISA, яка дає змогу використовувати в половинному розмірі материнські плати з високопродуктивними процесорами. Таким чином, можна без проблем модернізувати вже наявні системи з шиною ISA, отримуючи можливість використовувати конструктивні новітні процесорні й периферійні плати з шинами ISA і PCI.

Найбільш характерним представником плат збору даних для шини PCI є плата аналого-цифрового перетворювача (АЦП) PCI-1800 з частотою 330 кГц. Крім основних вона має функцію plug-and-play, а також убудований FIFO-буфер, що дає змогу працювати з платою операційної системи Windows без утрати даних, різними режимами запуску перетворення й режимом сканування «Magic scan».

Режим сканування Magic Scan дає змогу сканувати канали з різними коефіцієнтами підсилення й у довільному порядку, одночасно вимірювати низько- й високовольтні сигнали. Це реалізовано шляхом додавання спеціальної пам'яті (буфера) розміром 48 комірок, у якій зберігається черга номера інформаційних пар каналу-коефіцієнта підсилення, що спрощує приймання й оброблення сигналів від датчиків з різним рівнем вихідного сигналу.

Плата має вбудований цифровий фільтр, принцип роботи якого базується на простому усередненні сигналу, що допомагає позбутися від шумів і підвищити точність вимірювання. Таким чином, разом з фільтрацією сигналу можна задавати різні швидкості відцифрування для кожного каналу. Ця можливість позбавляє від необхідності забивати пам'ять зайвою інформацією про величину вхідних сигналів, що повільно змінюються.

На вході плати розташовано програмований компаратор, що дає змогу задавати два граничних коефіцієнти (високий і низький) і вибирати один з чотирьох режимів порівняння: перевищення верхньої межі або вихід за нижню межу, знаходження вхідного значення всередині або поза заданим цими двома межами інтервалом.

7.4 Характеристики шини PC-104

Специфікацію PC-104 було розроблено як метод розширення додатковими функціональними можливостями великих комп'ютерних плат. Пізніше було розроблено перші процесорні плати. Сьогодні понад 100 фірм у всьому світі пропонують процесорні плати, плати дискретного й аналогового введення-виведення, комунікаційні плати, джерела живлення, корпуси додаткові аксесуари. Оскільки так багато компаній пропонують сумісне обладнання, то стандарт PC-104 є надійною основою для вбудування високонадійних систем з

тривалим терміном доступності. Інженери можуть використовувати плати від різних виробників, а також, замінюючи процесорні плати й плати периферії, отримувати високопродуктивні системи, які відповідають усім вимогам щодо експлуатації.

Сам форм-фактор PC-104 було прийнято 1992 року у відповідь на вимоги про зменшення габаритних розмірів та енергоспоживання для комп'ютерних систем. Кожну з цих цілей було досягнуто без зниження апаратної й програмної сумісності з популярними комп'ютерними стандартами. У специфікації PC-104 пропонується повна архітектурна, апаратна й програмна сумісність з комп'ютерними стандартами в компактних розмірах плат 3,6 x 3,8 (91,44 x 96,52 мм). Назву стандарт отримав через застосування 104-контактної шини ISA, розташованої в нижній частині плати.

Форм-фактор PC-104-Plus було прийнято 1997 року шляхом додавання шини PCI з допомогою 120-контактного роз'єму у верхній частині плати. Застосування шини PCI дало змогу розширити функціональні можливості систем з допомогою PCI-сумісних функцій: відео, Ethernet та інших комунікаційних можливостей. При цьому плати, виконані в стандарті PC-104-Plus, є повністю сумісними з усіма платами, виконаними в стандарті PC-104.

PCI-104 було прийнято 2004 року. Цей стандарт склався при видаленні шини ISA з плати стандарту PC-104-Plus. На платі залишається тільки 120-контактна шина PCI на верхній частині плати. Видалення великого роз'єму шини ISA дає змогу вивільнити додаткові 10 % місця на платі і збільшити функціональність плат додатковими функціями введення-виведення. При такому рішенні плата функціонує тільки в стандарті PCI, тоді як шина ISA використовується як шина розширення x86-комп'ютерних систем. Шина PCI може використовуватися з іншими процесорними архітектурами, такими як XScale, PowerPC, ARM і FPGA.

EBX (Embedded Board eXpandable) включено до складу консорціуму 2005 року, хоча розроблено його було 1997 року. У цьому стандарті визначено одноплатні комп'ютери більшого розміру 5,75 x 8,00 (146 x 203 мм) з шиною розширення PC-104-Plus. Більший розмір дає змогу додати до процесорної плати більшу функціональність і зменшити загальну кількість плат у системі. Дуже багато постачальників пропонують плати в стандарті EBX на додаток до сотень різних плат стандартів PC-104-Plus і PCI-104.

EPIC (Embedded Platform for Industrial Computing) було включено до складу консорціуму 2005 року, оскільки була необхідна плата проміжного розміру між PC-104 й EBX. EPIC має розміри 6,5 x 4,5 (165 x 115 мм) і дає змогу використовувати сучасні процесори. У стандарті визначено зони введення-виведення й наявність таких функцій, як Ethernet, послідовні порти, цифровий та аналоговий вхід-вихід, відео і т. д.

2008 року відповідно до сучасних тенденцій розвитку комп'ютерної індустрії до стандарту PC-104 було введено сучасні послідовні інтерконекти PCI Express і USB, і прийнято нові форм-фактори PCI-104-Express, PCIe-104, EPIC-Express і EBX-Express. Їх основна особливість полягає в тому, що було прийнято рішення відмовитися від шини ISA й замінити її новою, що містить 4 PCIe x 1, 1 PCIe x 16

і USB. Для цього було розроблено спеціалізовані розніми, що зберігають механічну сумісність з попередніми специфікаціями й можуть пропускати високошвидкісні сигнали. Отже, нові специфікації дають змогу зберігати такі основні переваги стандарту, як компактні розміри й висока механічна міцність, а також додавати продуктивність і розширювати можливості готових убудованих систем.

7.5 Стандарт на промислові PC-комп'ютери для вбудованих додатків (розширення стандарту IEEE-P996 (ISA))

За останні десятиліття архітектура PC-комп'ютера стала визнаною платформою й зробила крок далеко за межі настільного застосування. Спеціалізовані PC-комп'ютери використовуються в бортовій апаратурі супутникових, морських і промислових систем з дослідними цілями, у вимірвальній апаратурі, пристроях зв'язку, медичних приладах і як універсальні контролери. При введенні стандарту на апаратну й програмну частини поширеної PC-архітектури вбудовані системи можуть значно зменшити вартість, ризик й час розроблення, оскільки для PC-комп'ютерів створено велику кількість прикладного програмного забезпечення, а архітектуру таких комп'ютерів добре знають розробники. Швидке розроблення програмного забезпечення й використання готових PC-модулів дає змогу економити кошти при розробленні систем, швидше виходити на ринок з новими виробами, підвищувати універсальність електронних виробів і забезпечувати їх швидку модернізацію.

Інша важлива особливість використання PC-архітектури – низька вартість програмної й апаратної підтримки, яка є набагато економічнішою, ніж для таких платформ, як STB, VME, MULTIBUS тощо, а це – додаткове зменшення витрат і збільшення надійності роботи. Для забезпечення сумісності промислових PC-комп'ютерів деякі компанії об'єдналися для випуску продукції, що задовольняє їх вимоги. Розміри конструктиву материнської плати (800 x 299 мм) за стандартом IEEE-P996 (ISA) і пов'язаними з ним корпусу й задньої панелі є дуже громіздкими й незручними для вбудованих систем. Виникла необхідність створення PC-комп'ютера, що мав би менші розміри, низьке споживання й додаткові функції (підтримка дисків, Watchdog-таймерів, LCD-дисплеїв, інтерфейсів PCMCIA і т. д.), які необхідні для вбудованих систем керування. Усі ці переваги необхідно було отримати при повній сумісності зі стандартною PC-машиною як за апаратною, так і за програмною частинами.

Стандарт PC-104 (IEEE-996.1) було розроблено згідно з цими вимогами. У ньому пропонується повна сумісність з PC-архітектурою, апаратною та програмною частинами, але у виключно компактному (90 x 96 мм) варіанті стикування модулів. Головна особливість стандарту – це те, що він дає змогу різноманітного компонування PC-комп'ютера. Наприклад, використання стекової архітектури забезпечує мінімальні габарити комп'ютера, а базової плати – його мінімальну висоту. Застосування нових рознімів для PC-шини забезпечує надійну роботу комп'ютерів у жорстких умовах експлуатації (підвищена вібрація, сольовий

туман, широкий діапазон температур і т. д.). Новітні технології виробництва електронних компонентів (субмікронна технологія), застосування поверхневого монтажу високої щільності дали змогу різко зменшити габарити й споживання енергії модулів PC-104, завдяки чому їх можна використовувати в закритих об'ємах без додаткового охолодження. Наприклад, типовий комп'ютер PC-104 серії i486 споживає 2,5 Вт. Малі габарити дають змогу легко вирівняти температуру комп'ютерів при використанні в умовах наднизьких температур (-60°C і нижче).

Стандарт PC-104, таким чином, ідеально задовольняє вимоги щодо вбудованих систем, що зробило його надзвичайно популярним не тільки в США, але й у країнах Європи й Азії.

Сьогодні збільшення популярності стандарту PC-104 додатково сприяє світова тенденція переходу до розподілених систем керування, які мають велику гнучкість, легкість в обслуговуванні й високі показники надійності. При будівництві систем широко використовуються мережні інтерфейси: Ethernet, Arcnet, ProfiBus, InterBus-S, CAN, LON і т. д., а апаратна й програмна підтримка для операційних систем QNX, RTXC, AMX, MS-DOS тощо широко представлена фірмами, що працюють у стандарті PC-104. Вдала конструкція модулів PC-104 дає змогу використовувати як готові конструктиви DIN-rail і євростандарти, які найбільш широко використовуються в промисловості, так і спеціальні, які призначено для жорстких умов експлуатації.

Бурхливий розвиток стандарту в 1994–1995-х роках привернуло підвищену увагу компаній, що випускають обладнання для військового застосування. Сьогодні стандарт PC-104 випускають основні інтерфейси MIL-STD, конструктиви з вимогами MIL-STD, інтерфейси для авіаційної промисловості (ARINC-419/429/561/575/615).

Ключові відмінності PC-104 і PC ISA (IEEE-P996):

1 Компактність конструкції, розмір зменшено до 90 x 96 мм.
2 Зручна конструкція шини, завдяки чому припускаються різні варіанти компонування P-модулів, зменшує вартість, громіздкість задніх панелей і виключає використання кошика.

3 Розніми PC-104 (сквозні штиркові). Стандартні розніми PC-шини, розташовані на бічній стороні плат для зменшення габаритів і забезпечення роботи в жорстких умовах експлуатації, було замінено компактними 64-і 40-контактними штирковими рознімами.

4 Стандарт IEEE-P996.1 (PC-104) дає змогу зменшити струм шинних комп'ютерів до 4 мА, це дає змогу знизити споживання енергії до 1–2 Вт на модуль, що є важливим чинником при роботі без примусового охолодження в закритому об'ємі. Типові модулі процесорів серії 486/586 зазвичай містять шинні формувачі з вихідним струмом не менше 24 мА, що рівносильно електричному навантаженню на звичайній ISA-шині. При цьому на шину PC-104 можливе під'єднання до 12 модулів.

Використовуючи стандарт PC-104, фірми виграють від застосування стандартних виробів з широкою номенклатурою, що містить великий вибір виробів промислового призначення, отримуючи при цьому підтримку від фірм-

виробників. Застосування стандартних виробів дає змогу уникнути залежності від конкретного виробника й модернізувати продукцію, що буде випускатися в подальшому.

7.6 Види використання модулів РС-104

Різне компоновання модулів РС-104 майже не обмежує їх використання у вбудованих системах. Наприклад, стекова архітектура забезпечує мінімальні габарити комп'ютера, а використання базової плати – його мінімальну висоту.

Компоновання окремих модулів (стекова архітектура). Як буде показано далі, модулі РС-104 стикуються один з одним, утворюючи стекову конструкцію. При такій компоновці відпадає необхідність у сполучній крос-платі й кошику, забезпечуючи надійну конструкцію РС-машини й мінімальні габарити. Зістиковані модулі розташовано на відстані 15 мм один від одного. Три модулі, зістиковані разом, займають обсяг 90 x 96 x 51 мм (рисунок 7.2).

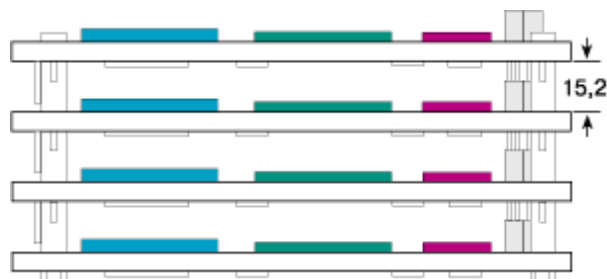


Рисунок 7.2 – Стекова архітектура

При такій компоновці модулі поміщають на призначену для користувача несну плату, що забезпечує кріплення модулів РС-104 і з'єднання між ними. Нарощення РС-комп'ютера в такій конфігурації забезпечується шляхом вибору розмірів несної плати, якщо необхідна плоска конструкція, або додаткове установлення модулів (стековий варіант). Застосування комбінованого компоновання дає великі можливості при проектуванні й модернізації великих систем.

Лекція 8 ІНТЕРФЕЙСИ 1-WIRE, SPI, JTAG

8.1 Однопровідний інтерфейс 1-Wire

Стандарт на цей інтерфейс створено фірмою Dallas Semiconductor. Розроблення інтерфейсу для виробів у двовивідному корпусі знадобилося для реалізації мікросхеми пам'яті.

Орієнтованість 1-Wire на двовивідні компоненти дала змогу втілити в життя ідею «паразитного живлення». Для живлення портів і пов'язаних з ними цільових пристроїв по лінії передання даних подаються імпульси, що заряджають убудовані в порт конденсатори. Розряджаючись, вони забезпечують живлення. При

ввімкненні системи подається довгий імпульс, що дає змогу зарядити повністю розряджені ємності, під час роботи подаються короткі імпульси, що підтримують заряд (рисунок 8.1).



Рисунок 8.1 – Характеристика імпульсів системи

Для роботи портів у режимі «паразитного» живлення на лінії передання даних постійно мають бути імпульси напругою не менше 2,8 В. Після сигналу «Скидання» іде довгий імпульс, що дає змогу зарядити розряджені живильні конденсатори портів. Під час обміну даними на початку кожного біта обов'язково є короткий імпульс, що підтримує заряд конденсаторів. Dallas Semiconductor прийняв рішення записувати унікальну адресу свого пристрою при виготовленні, що призвело до введення дуже довгих адрес.

Загальна довжина адреси – 8 байтів. Код типу виробів визначає найменування типу виробу. Серійний номер є унікальним для виробів цього типу, записується під час виробництва і його не можна змінити. Контрольна сума формується шляхом циклічного підсумовування, захищає адресу від помилок передання. Настільки довга адреса пристрою знижує корисну продуктивність інтерфейсу.

У кожній 1-Wire-системі обов'язково є тільки один контролер, що керує обміном даних та іншими операціями, усі інші порти є веденими кінцевими пристроями. Вони можуть тільки відпрацьовувати команди контролера й виставляти переривання.

Функції контролера різко вирізняють його від кінцевих пристроїв:

- контролер не має адреси, оскільки до нього немає звернень з боку кінцевих пристроїв;
- контролер має забезпечити паразитне живлення кінцевих пристроїв;
- контролер має керувати всіма операціями на шині й усіма кінцевими пристроями.

Обмін даними відбувається в рівнях мікросхем КМОП/ТТЛ (рівень логічного «0» – не більше 0,8 В, рівень логічної «1» – не менше 2,2 В). На початку кожного біта обміну даними контролер виставляє імпульс підзарядження. Усі вихідні кола контролера й кінцевих пристроїв побудовано за схемою із загальним затвором. До лінії живлення лінію обміну даними під'єднано через «підтягувальний» резистор. Тому верхню «полицю» імпульсу формує «підтягувальний» резистор, а негативний перепад на його кінці – вихідний транзистор лінійного передавача контролера. Цей перепад запускає таймер очікування достовірних даних.

Через заданий проміжок часу біт даних на лінії вважається достовірним і може бути прийнятим приймачем контролера або кінцевого пристрою. Під час передання рівня логічного «0» відкритий вихідний транзистор передавача утримує напругу на лінії на низькому логічному рівні. Під час передання рівня логічної «1» при закритих вихідних транзисторах усіх портів «підтягувальний» резистор повертає лінію на високий логічний рівень. Позитивний перепад є довшим від негативного, оскільки опір кола в цьому випадку є більшим. Форма й тривалість перепаду залежать від параметрів лінії передання даних і «підтягувального» резистора.

Під час передання даних від контролера до кінцевого пристрою нижню «полицю» логічного «0» формує контролер, під час передання від кінцевого пристрою до контролера – термінал.

Таймери очікування достовірних даних у кожному крайовому пристрої працюють від власних тактових генераторів. Через надзвичайну обмеженість енергетичних ресурсів при «паразитному живленні» стабільність їх частот є дуже низькою. Час проходження й тривалість фронтів сильно залежать від параметрів лінії передання даних. З цих причин у стандарті закладено більш ніж 100-відсотковий розкид часу спрацювання таймера. Необхідність урахувувати цей розкид обмежує продуктивність інтерфейсу 1-Wire (рисунок 8.2).

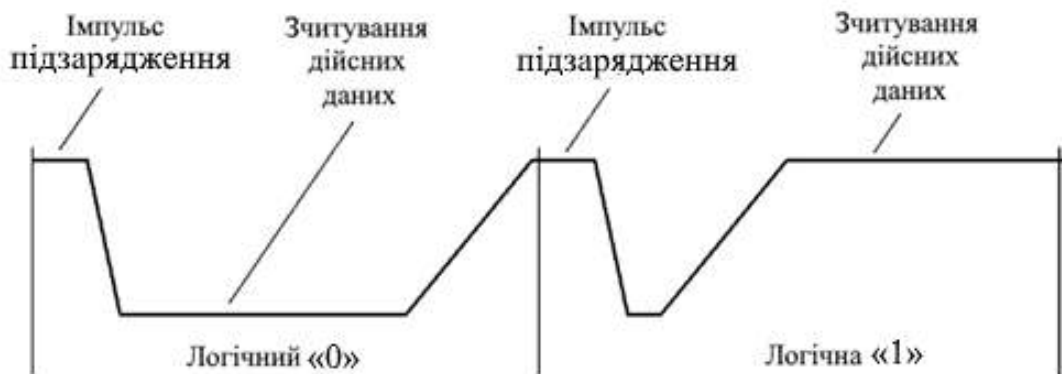


Рисунок 8.2 – Характеристика сигналу

Контролер керує кінцевими пристроями з допомогою закладених у стандарт команд. Код команди має довжину 1 байт й передається на початку кожного обміну даними. Після команди відбувається передання адреси або обмін даними. Кожен порт інтерфейсу 1-Wire має закладену у нього при виробництві унікальну адресу.

Після ввімкнення системи, або там, де передбачається «гаряче» під'єднання-від'єднання портів, контролер має визначити склад під'єднаних портів. Є алгоритм з безліччю повторюваних циклів. Програмування не викликає особливих проблем, але виконання операції займає чимало часу.

Ініціалізувавши пошук портів і визначення їх адрес, контролер видає команду пошуку. Після приймання цієї команди протягом двох наступних циклів обміну всі порти передають спочатку пряме, а потім інверсне значення першого біта адреси.

Якщо значення бітів у всіх портах дорівнює нулю, то на лінії буде кодова комбінація 01. Якщо у всіх портах значення бітів дорівнює одиниці, то – 10. Якщо в деяких портах одиниця, а в деяких нуль, то, використовуючи схему «монтажне I», на лінії буде кодова комбінація 00. Потім контролер виставляє значення «1» або «0», порти з відповідним значенням біта адреси залишаються активними, а з інверсним бітом адреси переходять у відімкнений стан. Інші активні порти протягом наступних двох циклів передають пряме й інверсне значення наступного біта адреси. Весь цикл повторюється доти, доки не буде визначено адресу одного з портів.

Для активізації раніше відімкнених портів контролер видає імпульс скидання. Далі перебиранням усіх можливих значень шукаються інші порти.

Фірма Dallas Semiconductor випускає широкий спектр виробів з портами інтерфейсу 1-Wire. Це прилади двохвивідної топології корпусу серії iButton, що вмикають датчики й реєстратори температури, пам'ять, таймери реального часу й «цифрові мітки», – найпростіші пристрої, що «уміють» тільки повертати значення своєї адреси. Такі пристрої широко застосовують як ключі електронних замків, систем обмеження доступу й т. ін.

Відсутність гальванічної розв'язки й шинна структура інтерфейсу обмежують розмір системи, для її сегментування фірма Dallas Semiconductor випускає відгалужувачі (DS2409) – спеціальні порти, до яких під'єднано перемикачі гілок інтерфейсу.

Застосування інтерфейсу 1-Wire є виправданим для систем збору даних і керування з невеликою продуктивністю.

8.2 Лінії зв'язку й топологія інтерфейсу 1-Wire

Велике значення при будівництві 1-Wire-мереж має виконання однопровідної лінії зв'язку. Зазвичай такі лінії мають структуру, що складається з трьох основних провідників: DATA – шина даних, RET – поворотний або земляний провід, EXT_POWER – зовнішнє живлення не тільки обслуговує ведені пристрої, але й зовнішні відносно них ланцюги датчиків та органів керування. Залежно від способу прокладення, сполучення з відомими пристроями й використання при прокладенні матеріалів відповідно до таблиці 8.1 розрізняють три основні варіанти якості організації 1-Wire-мереж, кожен з яких використовує особливу технологію та аксесуари при реалізації лінії .

Часто при організації складних однопровідних мереж для зручності проведення лінії зв'язку, зменшення її протяжності або зниження електричного навантаження на лінії завдяки зменшенню кількості пристроїв, що одночасно працюють на ній, необхідно забезпечити деревоподібну або променеву структуру магістралі, що значно відрізняється від структури загальної шини. Для цього використовують розгалуження 1-Wire-мереж одного або кількох рівнів.

Таблиця 8.1 – Характеристика мережі

Класифікація лінії	Довжина лінії, м	Кількість ведених пристроїв, шт.	Тип використовуваного кабелю	Топологія	Майстер лінії
Короткі	До 30	До 50	4-провідний телефонний	Вільна	Пасивне підтягування (резистор)
Середні	До 100	До 200	Вита пара 5-ї категорії	Загальна шина	Активне підтягування (DS2480, DS2490 або спеціальне схемне рішення)
Довгі	До 300	До 300	IEEE1394 (Firewire)	Загальна шина з єдиним стовбуром	Активне підтягування з урахуванням струму в лінії

Основним елементом при будівництві таких гілок є або звичайний ключ, що адресується типу DS2406, який забезпечує розгалуження завдяки комутації поворотного дроту однопровідної лінії, або спеціалізований розгалужувач DS2409, що комутує безпосередньо шину даних 1-Wire-лінії. Останній варіант є кращим, тому компоненти на відімкненій гілці, до якої належить розгалужувач, залишаються завжди в активному стані. Почергове обслуговування майстром мережі кожної з гілок при відімкнених інших гілках дає змогу значно збільшити загальну довжину лінії й кількість ведених пристроїв на ній.

Якщо ж організація 1-Wire-системи на базі персонального комп'ютера пов'язана з особливими труднощами, то найбільш оптимальним є використання інтелектуального адаптера для COM-порту типу LINK, реалізованого на базі мікропроцесора. При цьому пристрій повністю відтворює з боку послідовного порту роботу популярного адаптера DS9097U виробництва Dallas Semiconductor Corp., підтримуючи таким чином усе розроблене раніше для персональних комп'ютерів програмне забезпечення, завдяки вбудованим власним інтелектуальним ресурсам реалізує пільговий режим роботи однопровідних приладів на проблемних 1-Wire-лініях в умовах перешкод. LINK багаторазово покращує механізм активного підтягування на лінії, що дає змогу дійсно отримувати ідеальні сигнали обміну при довжині кабелю понад 300 метрів і кількості супроводжуваних однопровідних компонентів понад 100 шт., а

використання процесором приладу алгоритмів цифрової фільтрації багаторазово поліпшує стійкість обслуговування однопровідної лінії до електромагнітних завад.

8.3 Послідовний периферійний інтерфейс SPI

Послідовний периферійний інтерфейс SPI (Serial Peripheral Interface) являє собою повнодуплексний чотирипровідний інтерфейс з шинною конфігурацією під'єднання вузлів (пристроїв). SPI-інтерфейс дає змогу під'єднати до одного ведучого вузла кілька ведених вузлів через загальну шину. Окремий сигнал SSS (Slave-Select signal) вибору веденого пристрою використовується для вибору відомого пристрою при здійсненні з ним обміну даними. Крім того, можливо також побудувати систему з багатьма ведучими вузлами. Виявлення конфліктів при одночасному переданні кількох ведучих вузлів також передбачено.

Інтерфейс SPI можна запрограмувати для роботи як ведучого або як веденого. Якщо інтерфейс запрограмовано як ведучий, то він може працювати на максимальній швидкості передання даних, біт/с, що дорівнює половині тактової частоти. Якщо інтерфейс запрограмовано для роботи як ведений, то його максимальна швидкість у повнодуплексному режимі дорівнює одній десятій тактової частоти. Мається на увазі, що джерелом синхронізації в обох випадках є системний генератор тактової частоти. Якщо ведучий інтерфейс виробляє SCK, SSS і послідовні вхідні дані асинхронно, то максимальна швидкість передання має бути меншою від однієї десятої тактової частоти.

Існує ще один особливий режим, коли ведучий вузол має тільки передавати дані веденому (напівдуплексний режим) і не повинен приймати дані від нього. У цьому випадку максимальна швидкість передання становить чверть від системної тактової частоти при синхронному режимі роботи.

На рисунку 8.3 показано функціональну схему роботи SPI-інтерфейсу, а на рисунку 8.4 – типову структурну схему SPI-мережі.

Інтерфейс SPI має чотири сигнальні лінії: MOSI, MISO, SCK і SSS.

Лінія MOSI (Master-Out, Slave-In) – вихідна лінія даних ведучого інтерфейсу й вхідні лінії даних веденого. З назви випливає, що лінію призначено для передання даних від ведучого (Master) інтерфейсу (або вузла мережі) до веденого (Slave) інтерфейсу (або вузла мережі).

Лінія MISO (Master-In, Slave-Out) – вхідна лінія даних ведучого інтерфейсу й вихідна лінія даних веденого. Лінію призначено для передання даних від веденого інтерфейсу до ведучого. Дані передаються байтами побітово, починаючи зі старшого біта. Слід пам'ятати, що вивід MISO веденого інтерфейсу перебуває у високоімпедансному стані, якщо ведений інтерфейс не вибрано по лінії SSS.

Лінія SSS (Slave Select) – лінія вибірки веденого вузла, її призначено для вибірки низьким логічним потенціалом веденого інтерфейсу ведучим.

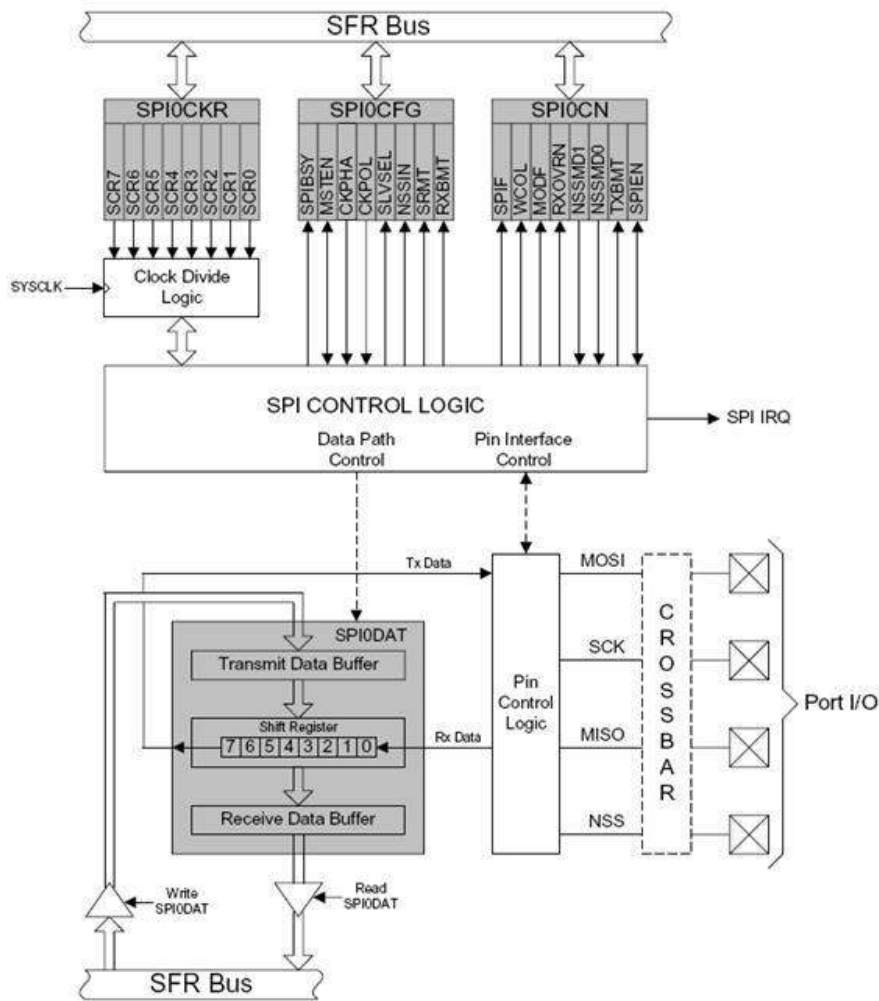


Рисунок 8.3 – Функціональна схема роботи SPI

Лінія SCK (Serial Clock) – вихідна лінія тактових імпульсів ведучого вузла й вхідна лінія тактових імпульсів веденого вузла. Лінія SCK використовується для синхронізації передання даних між ведучим й веденим інтерфейсами по лініях MOSI і MISO.

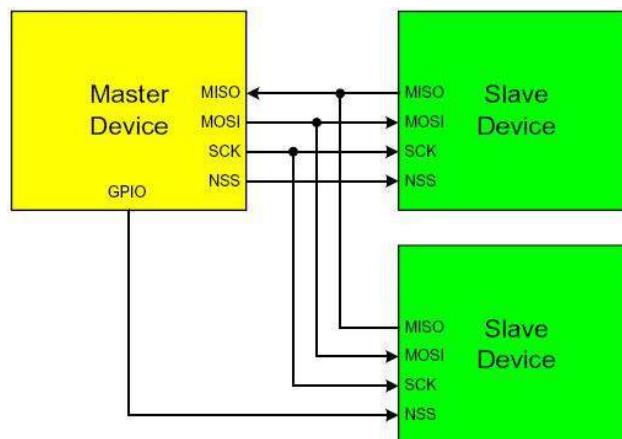


Рисунок 8.4 – Структурна схема SPI

У мережі на базі SPI-інтерфейсів тільки один інтерфейс може бути ведучим. Інтерфейс встановлюється в режим ведучого шляхом встановлення прапорця MSTEN (Master Enable flag) – біта SPI0CN.1. Якщо інтерфейс встановлено в режим ведучого, то запис байта даних у регістр даних SPIODAT призводить до початку передання. Ведучий інтерфейс негайно побітно зсуває дані й видає їх на лінію MOSI в супроводженні тактових імпульсів на лінії SCK. Після завершення передання встановлюється прапорець SPIF (SPI0CN.7). Якщо дозволено переривання, то видається відповідне переривання. Крім того, інтерфейс може бути запрограмовано на видачу від одного до восьми бітів для здійснення зв'язку з SPI-приладами, що мають різну довжину слова. Довжину передання (кількість переданих бітів) може бути задано бітами SPIFRS у регістрі конфігурації SPI0CFG.[2: 0] (SPI Configuration Register). З'єднання двох інтерфейсів SPI (ведучого й веденого) показано на рисунку 8.5.

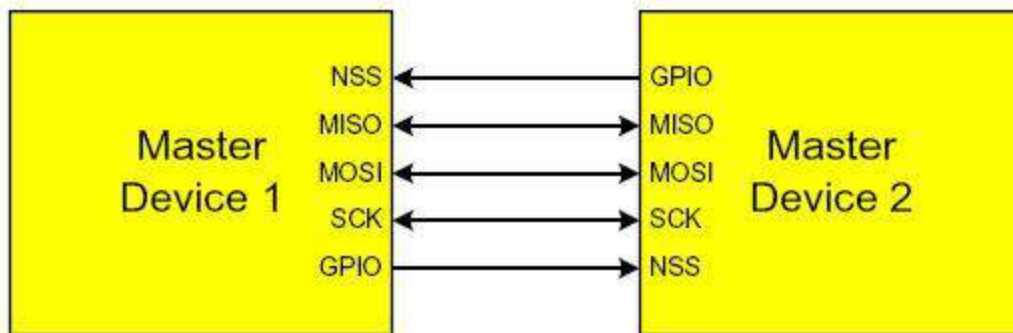


Рисунок 8.5 – З'єднання двох інтерфейсів SPI

Інтерфейс може працювати в повнодуплексному режимі. Це означає, що можливе одночасне передання даних по лініях MOSI від ведучого інтерфейсу до веденого й MISO від веденого до ведучого. Дані, отримані від веденого інтерфейсу, замінюють дані в регістрі даних ведучого інтерфейсу. Цей регістр двічі буферізовано на введення, але не на виведення. Таким чином, якщо в регістрі даних SPIODAT проводиться спроба запису даних під час передання попереднього байта, то встановлюється прапорець WCOL (SPI0CN.6) і спроба запису ігнорується. Отже, поточне передання даних триває безперервно. Читання з регістра даних SPIODAT приводить до читання приймального буфера. Якщо приймання не закінчено, то встановлюється прапорець RXOVRN (SPI0CN.4). Нові дані не передаються в регістр читання, поки попередній прийнятий байт не буде прочитано. Очевидно, що при затримці читання прийнятих байтів може відбутися втрата даних. Якщо SPI-інтерфейс не налаштовано як ведучий, то він буде працювати в режимі веденого.

Крім того, підтримується режим мережі з багатьма ведучими інтерфейсами. Прапорець MODF SPI0CN.5 (Mode Fault flag) встановлюється в логічну 1, якщо інтерфейс визначено як ведучий (MSTEN = 1) і вивід SSS переведено в низький логічний рівень, тобто якщо SPI інтерфейс намагаються використовувати як ведений. Якщо при цьому встановлено прапорець MODF, то біти MSTEN і SPIEN

в регістрі керування SPI стираються автоматично апаратно, переводячи інтерфейс в автономний стан. Таким чином, у системі з багатьма ведучими режимами ядро може визначити, чи є вільною шина шляхом опитування прапорця SLVSEL (SPI0CN.2) перед тим, як установити MSTEN-прапорець (тобто призначити інтерфейсу режим ведучого) і форматувати обмін.

На рисунку 8.6 показано часові діаграми роботи SPI інтерфейсу. Можливі чотири комбінації фаз тактових імпульсів і їх полярності залежно від комбінації керувальних бітів у регістрі конфігурації SPIOCFG (SPI Configuration Register).

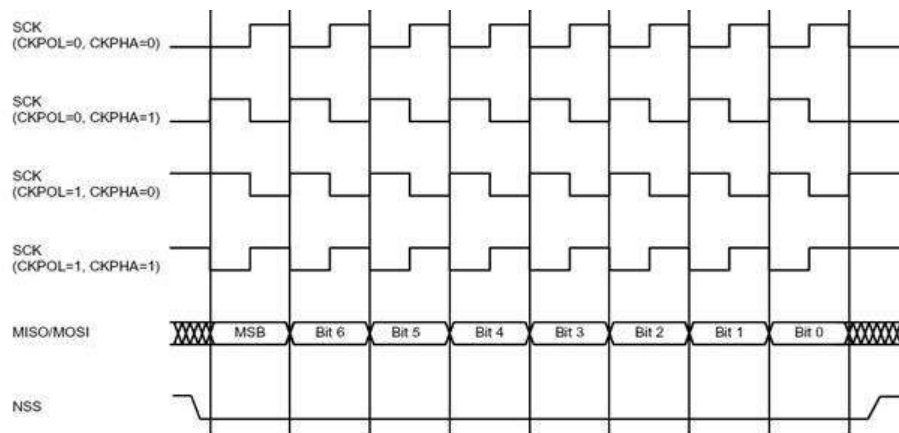


Рисунок 8.6 – Часові діаграми роботи SPI-інтерфейсу

Біт СКРНА (SPIOCFG.7) вибирає одну з двох фаз тактових імпульсів, тобто Фронт, по якому здійснюється запис даних. Інший біт СКPOL (SPIOCFG.6) визначає активну полярність (високий або низький рівень). Очевидно, що й ведучий, і ведений вузли повинні мати однакові налаштування фази й полярності. Ще одна важлива особливість налаштування полягає в тому, що інтерфейс SPI має бути забороненим шляхом обнулення біта SPIEN (SPI0CN.0) на час налаштування фази й полярності тактових імпульсів.

Крім описаних SFR-регістрів при налаштуванні SPI-інтерфейсу використовується регістр налаштування швидкості передання (SPIOCKR).

8.4 Послідовний периферійний інтерфейс JTAG

Інтерфейс JTAG (Joint Test Action Group) призначено для тестування складних логічних схем, установлених у цільовий пристрій. Цей інтерфейс описано в стандарті IEEE 1149.1 Boundary Scan Architecture, він є послідовно-синхронним, але у зв'язку зі специфікою застосування значно відрізняється від описаних вище інтерфейсів, орієнтованих на передання даних. Інтерфейсом JTAG керує один пристрій-контролер (найчастіше це ПК з відповідним інтерфейсним адаптером), до якого може бути під'єднано кілька тестованих пристроїв. Існують такі сигнали інтерфейсу JTAG.

TCK (Test Clock) – сигнал синхронізації послідовних даних; генерується контролером. Частота синхронізації може становити 16 МГц.

IMS (Test Mode Select) – сигнал вибору тестового режиму; генерується контролером.

TDI (Test Data Input) – вхідні дані, що приймає пристрій у послідовному двійковому коді (молодшим бітом уперед).

TOO (Test Data Output) – вихідні дані, що передаються пристроєм у послідовному двійковому коді.

TRST (Test Logic Reset) – необов'язковий сигнал скидання логіки інтерфейсного порту, що генерується контролером.

Ці сигнали (усі односпрямовані) мають звичайні логічні рівні й утворюють тестовий порт TAP (Test Access Port), через який тестований пристрій під'єднується до тестувального обладнання (контролера). До завдання тестувального обладнання належить формування тестових сигналів за програмою тестування, визначеною розробником тестованого пристрою, і порівняння отриманих результатів зі зразками. Одні й ті самі контролер і порт можуть використовуватися для тестування будь-якої кількості пристроїв, що підтримують JTAG. Для цього пристрої своїми портами TAP з'єднуються в ланцюжок (рисунок 8.7). Стандартизований логічний формат дає змогу контролеру незалежно спілкуватися з кожним із пристроїв ланцюжка (для цього, звичайно, вони повинні мати справні комірки JTAG)

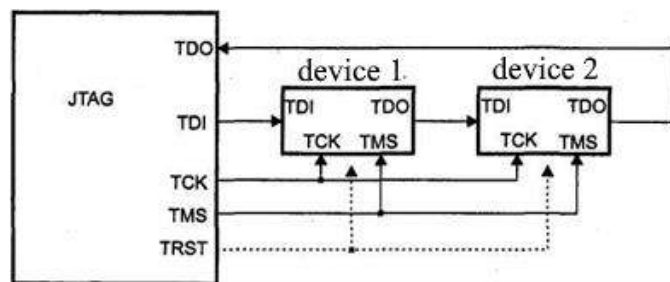


Рисунок 8.7 – Під'єднання кількох пристроїв

Ідею тестування будь-якої цифрової схеми ілюструє рисунок 8.8, на якому показано умовну цифрову систему, що має вхідні, вихідні (можливо, з третім станом) і двоспрямовані сигнали. Комірки тестування V/S врізаються між реальними зовнішніми виводами пристрою і власне логічним пристроєм, тобто розташовуються на логічній межі (boundary) пристрою. TAP-контролер може сканувати комірки – керувати ними і зчитувати з них інформацію. Звідси й пішла назва Boundary Scan, яку можна перекласти як «периферійне сканування». При ввімкненому тестовому режимі TAP-контролер може логічно від'єднати сигнали від зовнішніх виводів, задавати вхідні впливи і зчитувати результати – власне, це все, що необхідно для тестування послідовних схем (автоматів з пам'яттю). Привабливість JTAG полягає в тому, що незалежно від складності пристрою його можна тестувати з допомогою всього лише чотирьох сигналів – усі складнощі знаходяться в досить простих комірках, що «огортають» його сигнальні виводи.

Тестова логіка, убудована в пристрій, що підтримує JTAG, складається з таких елементів: тестовий порт TAP (чотири інтерфейсних сигнали), TAP-контролер, що керує тестовими регістрами, регістр інструкцій IR (Instruction Register), який приймає послідовний код зі входу TDI (код інструкції використовується для вибору виконувальної тестової операції або регістра тестових даних, до яких виконується звернення), регістри тестових даних, з яких будь-який пристрій повинен мати три обов'язкових регістри: у PR (Bypass Register), DID (Device Identification Register) і BSR (Boundary Scan Register).

Регістри інструкцій і даних являють собою незалежні зсувні регістри, з'єднані паралельно. На їх входи (старші біти) надходить сигнал TDI, з виходів (молодші біти) знімається сигнал TDO. По кожному позитивному перепаду дані просуюються на один біт.

Регістр BPR має довжину один біт. Його використовують як найкоротший обхідний шлях для послідовних даних, коли інші регістри не беруть участі в обміні.

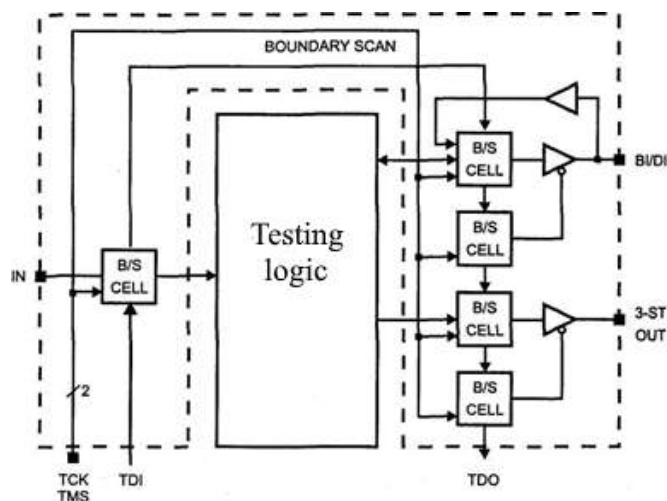


Рисунок 8.8 – Тестування цифрової схеми

Регістр BSR є довгим зсувним регістром, кожним бітом якого є примежові комірки, установлені на всіх входних і вихідних сигналах процесора. Для двоспрямованих сигналів (або їх груп) крім власне інформаційних комірок регістра, що відповідають зовнішнім, є й керувальні комірки, які задають режим роботи інформаційних комірок. Наприклад, у процесорів P6 довжина BSR становить 159 бітів.

Регістр DID довжиною 32 біти містить ідентифікатор виробника, код пристрою й номер версії, за якими TAP-контролер може розпізнати, з яким пристроєм він буде працювати.

Регістр IR призначено для зберігання виконуваної тестової інструкції. Його довжина залежить від тестованого пристрою. Для всіх пристроїв обов'язковими є інструкції BYPASS, IDCODE, SAMPLE і EXTEST.

Інструкцію BYPASS (усі біти коду – одиничні) призначено для під'єднання однобітового обхідного регістра із забезпеченням швидкого проходження даних через пристрій; при цьому інструкція ніяк не реагує на вузол, через який проходить потік. Вхід TDI зазвичай «підтягують» резистором до високого рівня, при цьому розрив ланцюжка JTAG приведе до під'єднання обхідних регістрів у всіх пристроях після точки обриву. Це виключає можливі непередбачувані дії пристрою разі обриву.

Інструкція ідентифікації IDCODE (молодші біти коду – 10) під'єднує до інтерфейсу регістр DID, даючи змогу зчитувати його вміст (вхідні дані, не можуть змінити його значення).

Інструкція SAMPLE/PRELOAD (молодші біти коду – 01) має два призначення. Коли TAP-контроллер знаходиться в стані Capture-DR, ця інструкція дає змогу виконати моментальний «знімок» стану всіх зовнішніх сигналів без впливу на роботу пристрою. Значення сигналів фіксується за позитивним перепадом TCK. У стані Update-DR за цією інструкцією дані завантажуються у вихідні комірки тестового порту (але ще не на виходи пристрою), звідки згодом вони будуть виводитися (подаватися на висновки процесора) за інструкцією EXTEST. Дані завантажуються за спадом сигналу TCK.

Інструкцію EXTEST (молодші біти коду – 00) призначено для перевірки зовнішніх ланцюгів (відносно тестованого пристрою). При цьому на вихідні виводи подаються сигнали, попередньо записані в регістр BSR, а стан вхідних сигналів фіксується в цих регістрах. Двоспрямовані сигнали попередньо конфігуруються відповідними керувальними бітами комірок BSR.

У стандарті 1149.1 передбачено й інструкцію тестування внутрішньої логіки пристрою INTEST, але її підтримують не всі пристрої.

Контролер тестового порту (TAP-controller) являє собою синхронний кінцевий автомат, що змінює стан за фронтом сигналу TCK і за ввімкненням живлення. Змінення станів керує сигнал TMS (Test Mode State), що сприймається за позитивним перепадом TCK. Граф станів і переходів керувального автомата показано на рисунку 8.9. Біля стрілок переходів указано значення сигналу TMS під час фронту TCK.

У початковий стан Test-Logic-Reset контролер автоматично переходить після ввімкнення живлення й з будь-якого іншого стану може бути переведено високим рівнем TMS, утримуваним не менше п'яти тактів TCK. Для переведення в стан Test-Logic-Reset іноді використовують і додатковий сигнал TRST. У цьому стані тестова логіка є забороненою і пристрій працює в нормальному режимі.

Стан Run-Test/Idle є проміжним між виконанням тестових операцій. У цьому стані регістри не змінюють свого значення.

У стані Capture-DR під час виконання інструкцій EXTEST і SAMPLE/PRELOAD сканувальний регістр фіксує тільки дані на вхідних лініях.

У стані Shift-DR дані з TDI просуваються через під'єднаний зсувний реєстр на вихід TDO.

У стані Pause-DR контроллер тимчасово забороняє просування даних через зсувний регістр.

У стані Update-DR за спадом ТСК сигнали із зсувного регістра фіксуються на виходах тестових комірок.

У стані Capture-IR контроллер завантажує у зсувний регістр інструкцій код інструкції SAMPLE.

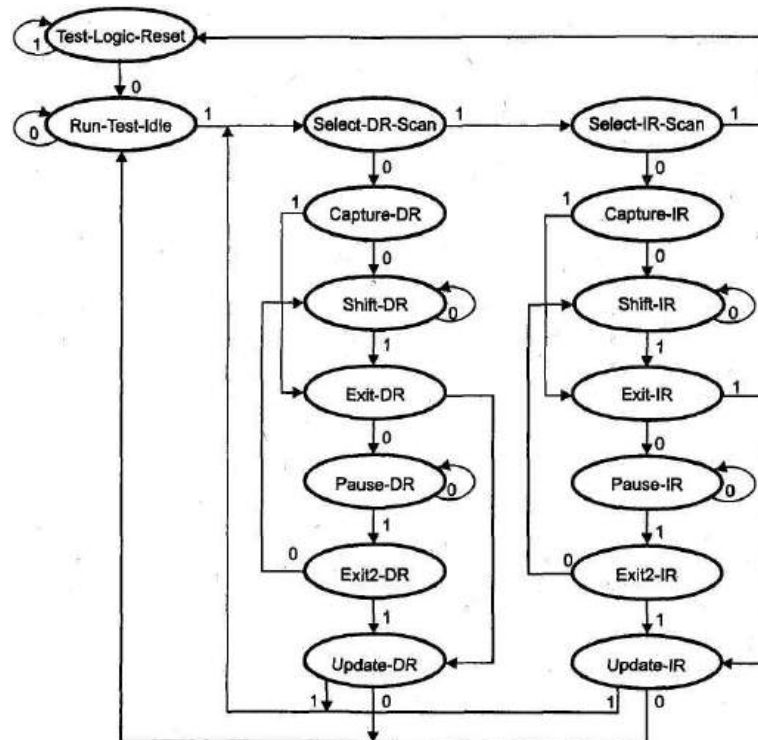


Рисунок 8.9 – Граф станів і переходів

У стані Shift-IR до ланцюга між TDI і TDO під'єднується зсувний регістр інструкцій, але ще виконується попередня інструкція.

У стані Pause-IR контроллер тимчасово забороняє просування даних через зсувний регістр інструкцій.

У стані Update-IR за спадом ТСК фіксується нова інструкція, що виконується, і до кола TDI–TDO під'єднується відповідний регістр.

Крім цих основних станів контролера, що визначають дії тестового обладнання, є й тимчасові проміжні стани, необхідні для реалізації переходів автомата. До них належать Select-DR-Scan, Exit1-DR, Exit2-DR, Select-DR-Scan, Exit1-IR й Exit2-IR.

Для інтерфейсу JTAG існує спеціальна мова опису пристроїв BSDL (Boundary Scan Description Language). Склад і порядок проходження інформаційних і керувальних комірок у зсувному регістрі даних є специфічним для кожного пристрою (для чого й існує ідентифікаційний регістр).

Інтерфейс JTAG використовується не тільки для тестування, але й для програмування різних пристроїв, у тому числі й незалежної пам'яті мікроконтролерів. Контакти для сигналів JTAG є на шині PCI, однак при їх використанні одноманітності не спостерігається (або залишаються непід'єднаними, або з'єднуються для організації ланцюжка). Інтерфейс JTAG є в сучасних процесорах:

тут він дає змогу не тільки тестувати сам процесор (це не є особливим прикладним інтересом), але й організувати зондовий режим налаштування (probe mode), який є потужним середовищем налаштування системного програмного забезпечення. Звичайний процесор, зв'язаний з тестовим контроллером інтерфейсом JTAG, перетворюється на внутрішньосхемний емулятор – мрію розробника системного забезпечення.

Лекція 9 БЕЗДРОТОВІ ІНТЕРФЕЙСИ ПЕРИФЕРІЙНИХ ПРИСТРОЇВ

9.1 Інфрачервоний інтерфейс IrDA

Застосування випромінювачів і приймачів інфрачервоного (ІЧ) діапазону дає змогу здійснювати бездротовий зв'язок між парою пристроїв, віддалених на відстань до кількох метрів. **Інфрачервоний зв'язок – IR (Infra Red) Connection** – є безпечним для здоров'я, не створює завад у радіочастотному діапазоні й забезпечує конфіденційність передання. ІЧ-промені не проходять через стіни, тому зона приймання обмежується невеликим, легкоконтрольованим простором. Інфрачервону технологію застосовують для зв'язку портативних комп'ютерів зі стаціонарними комп'ютерами або док-станціями. Інфрачервоний інтерфейс мають деякі моделі принтерів, ними оснащено багато сучасних малогабаритних пристроїв: кишенькові комп'ютери (PDA), мобільні телефони, цифрові фотокамери й т. ін.

Розрізняють інфрачервоні системи низької (до 115,2 кбіт/с), середньої (1,152 Мбіт/с) і високої (4 Мбіт/с) швидкості. Низькошвидкісні системи призначено для обміну короткими повідомленнями, високошвидкісні – для обміну файлами між комп'ютерами, під'єднання до комп'ютерної мережі, виведення на принтер, проєкційний апарат тощо. 1993 року було створено асоціацію розробників систем інфрачервоного передання даних IrDA (Infrared Data Association), призначену забезпечити сумісність обладнання різних виробників. Сьогодні діє стандарт IrDA 1.1, нарівні з яким існують і власні системи фірм Hewlett Packard-HP-SIR (Hewlett Packard Slow Infra Red) і Sharp-ASK IR (Amplitude Shifted Keyed IR). Ці інтерфейси забезпечують такі швидкості передання: IrDA SIR (Serial Infra Red), HP-SIR-9,6...115,2 Кбіт/с, IrDA HDLC, відомий як IrDAMIR (Middle InfraRed) – 0,576 і 1,152 Мбіт/с, IrDA FIR (Fast Infra Red) – 4 Мбіт/с, ASKIR – 9,6...57,6 Кбіт/с.

Випромінювачем для ІЧ-зв'язку є світлодіод, який має пік спектральної характеристики потужності 880 нм. Світлодіод дає конус ефективного випромінювання з кутом близько 30°. Як приймач використовують PIN-діоди, що ефективно приймають ІЧ-промені в конусі 15°. У специфікації IrDA визначено вимоги до потужності передавача й чутливості приймача, причому для приймача задається як мінімальна, так і максимальна потужність ІЧ-променів. Імпульси занадто малої потужності приймач не «побачить», а надто велика потужність «засліплює» приймач – прийняті імпульси зіллються в нерозрізнений сигнал. Крім корисного сигналу на приймач впливають завади: засвічування сонячним світлом

і лампами розжарювання, що дає постійну складову оптичної потужності, і завади від люмінесцентних ламп, що дають змінну (але низькочастотну) складову. Ці завади доводиться фільтрувати. Специфікація IrDA забезпечує рівень бітових помилок (BitErrorRatio, BER) не більше 10 при дальності до 1 м й денному світлі (освітленість до 10 люкс). Оскільки передавач майже неминуче спричиняє засвічування свого ж приймача, уводячи його в насичення, доводиться задіювати напівдуплексний зв'язок з певними часовими проміжками при змінненні напрямку обміну. Для передання сигналів використовують двійкову модуляцію (є світло – немає світла) і різні схеми кодування.

У специфікації IrDA визначено багаторівневу систему протоколів, яку розглянемо знизу вгору.

Наведено варіанти, що є можливими на фізичному рівні IrDA.

1 **IrDA SIR** – для швидкостей 2,4...115,2 кбіт/с використовується стандартний асинхронний режим передання (як в COM-портах): старт-біт (нульовий), 8 бітів даних і стоп-біт (одиничний). Нульове значення біта кодується імпульсом тривалістю 3/16 бітового інтервалу (1,63 мкс на швидкості 115,2 кбіт/с), одиничний – відсутністю імпульсів (режим IrDA SIR-A). Таким чином, у паузі між посиленнями передавач не світиться, а кожне посилення починається з імпульсу старт-біта. У специфікації 1.1 передбачено й інший режим – IrDASIR-B з фіксованою тривалістю імпульсу 1,63 мкс для всіх цих швидкостей.

2 **ASKIR** – для швидкостей 9,6...57,6 кбіт/с також використовується асинхронний режим, але кодування є іншим: нульовий біт кодується шляхом посилення імпульсів з частотою 500 кГц, на одиничному біті імпульсів немає.

3 **IrDAHDLC** – для швидкостей 0,576 і 1,152 Мбіт/с використовується синхронний режим передання й кодування, аналогічний протоколу SIR, але з тривалістю імпульсу ¼-бітового інтервалу. Формат кадру відповідає протоколу HDLC, початок і кінець кадру позначають прапорцями 01111110, усередині кадру ця бітова послідовність виключається шляхом вставлення бітів (bitstuffing). Для контролю достовірності кадр містить 16 бітів CRC-коду.

4 **IrDA FIR (IrDA4PPM)** – для швидкості 4 Мбіт/с також застосовується синхронний режим, але кодування є дещо складнішим. Тут кожна пара суміжних бітів кодується позиційно-імпульсним кодом: 00 → 1000, 01 → 0100, 10 → 0010, 11 → 0001 (у четвірках символів «І» означає посилення імпульсу у відповідній чверті двобітового інтервалу). Такий спосіб кодування дав змогу вдвічі зменшити частоту ввімкнення світлодіода порівняно з попереднім. Сталість середньої частоти прийнятих імпульсів полегшує адаптацію до рівня зовнішнього засвічування. Для підвищення достовірності застосовується 32-бітовий CRC-код.

Над фізичним рівнем розташовано протокол доступу. **IrLAP (IrDAInfrared Link Access Protocol)** – модифікація протоколу HDLC, що відображає потреби ІЧ-зв'язку. Цей протокол інкапсулює дані в кадри й запобігає конфліктам пристроїв: при наявності не більше двох пасивних пристроїв, що «бачать» один одного, один з них призначається первинним, а другий – вторинними. Зв'язок завжди є напівдуплексним. В IrLAP описано процедуру встановлення, нумерації й закриття з'єднань. З'єднання встановлюється на швидкості 9600 біт/с, після чого

узгоджується швидкість обміну за максимумом сигналів, що є доступними обом (9,6, 19,2, 38,4, 57,6 або 115,2 Кбіт/с) і встановлюються логічні канали (кожен канал керується одним ведучим пристроєм).

Над IrLAP розташовано протокол керування з'єднанням **IrLMP (IrDA Infrared Link Management Protocol)**. З його допомогою пристрій повідомляє іншим про свою наявність у зоні охоплення (конфігурація пристроїв IrDA може змінюватися динамічно: для її змінення достатньо піднести новий пристрій або віднести його трохи далі). Протокол IrLMP дає змогу виявляти сервіси, що надаються пристроєм, перевіряти потоки даних і відігравати роль мультиплексора для конфігурацій з безліччю доступних пристроїв. Додатки з допомогою IrLMP можуть дізнатися, чи є необхідний їм пристрій у межах досяжності. Однак гарантованої доставки даних цей протокол не забезпечує.

Транспортний рівень забезпечується протоколом типу **TP (IrDA Transport Protocols)**, де обслуговуються віртуальні канали між пристроями, обробляються помилки (утрачені пакети, помилки даних і т. ін.), упаковуються дані в пакети і складаються вихідні дані з пакетів (протокол нагадує TCP). На транспортному рівні може працювати й протокол IrTP.

Протокол IrCOMM дає змогу через ІЧ-зв'язок емулювати звичайне дротове під'єднання:

- Три-дротове за RS-232C (TXD, RXD і GND);
- дев'яти-дротове за RS-232C (весь набір сигналів COM-порту);
- Centronics (емуляція паралельного інтерфейсу).

Протокол IrLAN забезпечує доступ до локальних мереж, даючи змогу передавати кадри мереж Ethernet і Token Ring. Для ІЧ-під'єднання до локальної мережі потрібен пристрій-провайдер з інтерфейсом IrDA, під'єднаний звичайним (проводовим) способом до локальної мережі, й відповідне програмне забезпечення в клієнтському пристрої (яке повинно увійти в мережу). Протокол об'єктного обміну **IrOBEX (ObjectExchangeProtocol)** – простий протокол, що визначає команди PUT і GET для обміну «корисними» двійковими даними між пристроями. Цей протокол розташовано над протоколом типу TP. У протоколі IrOBEX є розширення для мобільних комунікацій, яке визначає передання інформації, і належить до мереж GSM (записна книжка, календар, керування викликом, цифрове передання голосу й т. ін.).

Цими протоколами не вичерпується весь список протоколів, що належать до ІЧ-зв'язку. Зауважимо, що для дистанційного керування побутовою технікою (телевізори, відеомагнітофони та ін.) використовується той же діапазон 880 нм, але з іншими частотами й методами фізичного кодування. Приймач IrDA можна під'єднати до комп'ютера різними способами. Відносно системного блока він може бути як внутрішнім (розміщеним на лицьовій панелі), так й зовнішнім (розміщеним у довільному місці). Розміщувати приймач слід з урахуванням кута «зору» (30° – у передавача й 15° – у приймача) і відстані до необхідного пристрою (до 1 м).

Внутрішні передавачі, що працюють на швидкостях до 115,2 кбіт/с (IrDASIR, HP-SIR, ASKIR), під'єднуються через звичайні мікросхеми UART, які є сумісними

з 16450/16550 через порівняно нескладні схеми модуляторів-демодуляторів. У деяких сучасних системних платах на використання інфрачервоного зв'язку (до 115,2 кбіт/с) може конфігуруватися порт COM2. Для цього крім UART чіпсет містить схеми модулятора й демодулятора, що забезпечують один або кілька протоколів інфрачервоного зв'язку. Щоб порт COM2 використовувався для інфрачервоного зв'язку, у CMOSSetup необхідно вибрати відповідний режим (заборона інфрачервоного зв'язку означає звичайне використання COM2). Існують внутрішні адаптери й у вигляді карт розширення (для шин ISA, PCI, PC Card), вони мають вигляд додаткових COM-портів.

На середніх і високих швидкостях обміну застосовуються спеціалізовані мікросхеми контролерів IrDA, орієнтовані на інтенсивний програмно-керований обмін або DMA, з можливістю прямого керування шиною. Тут звичайний приймач UART є непридатним, оскільки він не підтримує синхронний режим і високу швидкість. Контролер IrDAFIR виконується у вигляді карти розширення або інтегрується в системну плату, зазвичай такий контролер підтримує й режими SIR.

Приймач під'єднується до розніму IR-Connector системної плати безпосередньо (якщо його встановлено на лицьову панель комп'ютера) або через проміжний рознім (mini-DIN), розташований на скобі-заглушці задньої стінки корпусу. На жаль, єдиної розкладки ланцюгів на внутрішньому конекторі немає, і для більшої гнучкості приймач (або проміжний рознім) обладнують кабелем з окремими контактами розніму. Скласти їх в належному порядку має користувач. Варіанти призначення контактів конектора інфрачервоного приймача наведено в таблиці 9.1. Деякі приймачі, які підтримують режими FIR і SIR, мають роздільні виходи – IRRX (для SIR) та iFIRRX (для FIR). Якщо контролер підтримує тільки один режим, то один з контактів залишиться непід'єднаним.

Таблиця 9.1 – Характеристика сигналів

Коло	Призначення	Контакт/варіант			
		1	2	3	4
IRRX (RX)	Вхід з приймача	1	3	3	3
FIRRX (RXH)	Вхід з приймача FIR	5	–	–	4
IRTX (TX)	Вихід на приймач	3	5	1	1
GND	Загальний	2, 7	4	2	2
Vcc (+5B)	Живлення	4, 6	1	5	5
NC	Вільний	–	2	4	–

Зовнішні ПК-адаптери випускають з інтерфейсом RS-232C для під'єднання до COM-порту або ж із шиною USB. Пропускної здатності USB достатньо навіть для FIR, COM-порт придатний тільки для SIR. Зовнішній ІЧ-адаптер IrDASIR для COM-порту не такий простий, як здається: для роботи модулятора-демодулятора потрібен сигнал синхронізації з частотою, що дорівнює 16-кратній частоті передання даних (цей сигнал надходить на синхровхід мікросхеми UARTCOM-порту). Такого сигналу на виході COM-порту немає, і його доводиться

відновлювати з асинхронного бітового потоку. Адаптер ASKIR у цьому плані є простішим – передавач має передавати високочастотні імпульси весь час, поки вихід TXD перебуває у високому стані, а приймач має формувати обвідну прийнятих імпульсів.

Для прикладного використання IrDA крім фізичного під'єднання адаптера й трансивера потрібно установити й налаштувати відповідні драйвери. В ОС Windows9x/ME/2000 контролер IrDA потрапляє до групи «мережне відімкнення». Сконфігуроване програмне забезпечення дає змогу встановлювати з'єднання з локальною мережею (для виходу в Інтернет, використання мережних ресурсів). Передавати файли між парою комп'ютерів, виводити дані на друк, синхронізувати дані PDA, мобільного телефону й настільного комп'ютера, завантажувати зняті зображення з фотокамери в комп'ютер та виконувати інші корисні дії без використання кабелю.

9.2 Радіоінтерфейс Bluetooth

Bluetooth – це фактичний стандарт на мініатюрні недорогі засоби передавання інформації з допомогою радіозв'язку між мобільними (і настільними) комп'ютерами, мобільними телефонами та будь-якими іншими портативними пристроями на невеликі відстані. Специфікацію розробляє група провідних фірм в областях телекомунікацій, комп'ютерів і мереж – 3Com, Agere Systems, Ericsson, IBM, Intel, Microsoft, Motorola, Nokia, Toshiba. Ця група, що створила Bluetooth Special Interest Group, і вивела цю технологію на ринок. Специфікація Bluetooth є у вільному доступі в Мережі (www.bluetooth.com) (близько 15 Мбайт PDF-файлів). Назва походить від прізвища датського короля, який об'єднав Данію й Норвегію.

Кожен пристрій Bluetooth має радіопередавач і приймач, що працюють в діапазоні частот 2,4 ГГц. Цей діапазон у більшості країн відведено для промислової, наукової й медичної апаратури і не потребує ліцензування, що забезпечує повсюдне застосування пристроїв. Для Bluetooth використовуються радіоканали з дискретною (двійковою) частотною модуляцією, несна частота, МГц, каналів $F^{\circ} = 2402 + k$, де $k = 0, \dots, 78$. Для кількох країн (наприклад, Франції, де в цьому діапазоні працюють військові) може бути застосовано скорочений варіант $cf^{\circ} = 2454 + k$ ($k = 0, \dots, 22$). Кодування є простим – логічній одиниці відповідає позитивна девіація частоти, нулю – негативна. Передавачі можуть бути трьох класів потужності (з максимальною потужністю 1, 2,5 і 100 МВт) причому має бути можливість зниження потужності з метою економії енергії.

Передавання ведеться з перескоком несної частоти з одного радіоканалу на інший, що допомагає в боротьбі з інтерференцією й завмираннями сигналу. Фізичний канал зв'язку є певною псевдовипадковою послідовністю використовуваних радіоканалів (79 або 23 можливі частоти). Група пристроїв, які знаходяться на одному каналі (тобто «знають» одну й ту саму послідовність перескоків), створює так звану пікомережу (piconet), до якої може належати від двох до восьми пристроїв. У кожній пікомережі є один ведучий пристрій і до семи активних ведених. Крім того, у зоні охоплення ведучого пристрою в його ж

пікомережі можуть перебувати «припарковані» ведені пристрої: вони теж «знають» послідовність перескоків і синхронізуються (за перескоками) з ведучим пристроєм, але не можуть обмінюватися даними доти, доки ведучий пристрій не дозволить їх активність. Кожен активний ведучий пристрій пікомережі має свій тимчасовий номер від 1 до 7. Коли ведучий пристрій деактивується (паркується), він віддає свій номер для використання іншим. При подальшій активізації пристрій вже може отримати інший номер (тому він і є тимчасовим). Пікомережі можуть перекриватися зонами охоплення, утворюючи «розкидану» мережу (scatternet). При цьому в кожній пікомережі ведучий пристрій є тільки один, але ведучі пристрої можуть входити в кілька пікомереж, використовуючи поділ часу (частину часу він працює в одній пікомережі, частину – в іншій). Більш того, ведучий пристрій однієї пікомережі може бути веденим пристроєм іншої пікомережі. Ці пікомережі ніяк не синхронізовано, кожна з них має свій канал (послідовність перескоків).

Канал поділяється на тайм-слоти тривалістю 625 мкс, слоти послідовно нумеруються з циклічністю 227. Кожен тайм-слот відповідає одній частоті в послідовності перескоків (1600 перескоків за секунду). Послідовність частот визначається адресою ведучого пристрою пікомережі. Передання ведеться пакетами, кожен пакет може займати від 1 до 5 тайм-слотів. Якщо пакет довгий, то він весь передається на одній несній частоті, але відлік слотів по 625 мкс триває, і після довгого пакета наступна частота буде відповідати черговому номеру слота (тобто кілька перескоків буде пропущено). Ведучий й ведений пристрої здійснюють передання по черзі: у парних слотах передання здійснює ведучий пристрій, а в непарних – адресований їм ведений пристрій (якщо йому є що «передати»).

Між ведучим і веденим пристроями можуть установлюватися фізичні зв'язки двох типів: синхронні й асинхронні.

Синхронні зв'язки (або ізохронні) з установленням з'єднання (SCOLink – Synchronous Connection-Oriented) використовуються для передання ізохронного трафіка (наприклад, оцифрованого звуку). Ці зв'язки типу точка – точка попередньо встановлює ведучий пристрій з вибраними веденими пристроями, і для кожного зв'язку визначається період (у слотах), через який для неї резервуються слоти. Зв'язки будуть симетричними двосторонніми. Повторне передання пакетів у разі помилок приймання не використовуються. Ведучий пристрій може встановити до трьох зв'язків SCO з одним або різними веденими пристроями. Ведений пристрій може мати до трьох зв'язків з одним ведучим пристроєм або мати по одному зв'язку SCO з двома різними ведучими пристроями. За мережною класифікацією зв'язку до SCO належать комутації ланцюгів.

Асинхронні зв'язки без установлення з'єднання (ACLlink – Asynchronous Connection-Less) реалізують комутацію пакетів за схемою точка – безліч точок між ведучим пристроєм і всіма веденими пристроями пікомережі. Ведучий пристрій може зв'язуватися з будь-яким з ведених пристроїв пікомережі в слотах, не зайнятих під SCO, пославши йому пакет і очікуючи відповіді. Ведений пристрій може розпочати передання, тільки отримавши звернений до нього запит ведучого

пристрою (безпомилково декодувати свою адресу). Для більшості типів пакетів передбачається повторне передання в разі виявлення помилки приймання. Ведучий пристрій може посилати й безадресні ширококомовні пакети для всіх ведених пристроїв своєї пікомережі. З кожним зі своїх ведених пристроїв ведучий пристрій може встановити лише один зв'язок ACL.

Інформація передається пакетами, у яких поле даних може мати довжину 0...2745 біт. Для зв'язків в ACL передбачено кілька типів пакетів із захистом CRC-кодом (у разі виявлення помилки передбачається повторне передання) та один без захисту (без повторного передання). Для зв'язків SCO дані не захищено CRC-кодом і, отже, повторне передання помилкового приймання не передбачено.

Захист даних від спотворення і контроль достовірності проводиться кількома способами. Дані деяких типів пакетів захищено CRC-кодом, і приймач інформації має підтверджувати приймання правильного пакета або повідомити про помилку приймання. Для зменшення кількості повторів застосовується надмірне кодування FEC (Forward Error Correction code). У схемі FEC1/3 кожен корисний біт передається тричі, що дає змогу вибрати найбільш прийнятний варіант. Схema FEC 2/3 є дещо складнішою, тут використовується код Хеммінга, що дає змогу виправляти всі одноразові й виявляти всі дворазові помилки в кожному 10-бітовому блоці.

Кожен голосовий канал забезпечує швидкість по 64 кбіт/с в обох напрямках. У каналі може використовуватися кодування у форматі PCM (імпульсно-кодова модуляція) або CVSD (Continuous Variable Slope Delta Modulation – варіант адаптивної дельта-імпульсно-кодової модуляції). При кодуванні PCM допускається компресія за G.711, який забезпечує лише суто «телефонну» якість сигналу (мається на увазі цифрова телефонія, 8-бітові вибірки з частотою 8 кбіт/с). Кодер CVSD забезпечує більш високу якість – він упакує вхідний PCM-сигнал з частотою вибірок 64 кбіт/с, однак і при цьому спектральна щільність сигналу в смузі частот 4...032 кГц має бути незначною. Для передання високоякісного аудіосигналу голосові (мовні) канали Bluetooth є непридатними, проте стиснений сигнал (наприклад, потік MP3) цілком можна передавати по асинхронному каналу передання даних.

Асинхронний канал може забезпечувати максимальну швидкість 723,2 кбіт/с з асиметричною конфігурацією (залишаючи для зворотного каналу смугу 57,6 кбіт/с) або 433,9 кбіт/с у кожний бік у симетричній конфігурації.

Для забезпечення безпеки в Bluetooth застосовується автентифікація та шифрування даних на рівні зв'язку (linklayer), які, звичайно ж, можуть доповнюватися й засобами верхніх протокольних рівнів.

Важливою частиною Bluetooth є протокол виявлення сервісів SDP (Service Discovery Protocol), що дає пристрою змогу знайти «цікавого співрозмовника». Надалі, установивши з ним з'єднання, пристрій зможе скористатися необхідними сервісами (наприклад, виводити документи на друк, під'єднуватися до Мережі тощо).

Протокол RFCOMM забезпечує емуляцію послідовного порту (9-провідного RS-232) через L2CAP. З його допомогою традиційні кабельні з'єднання пристроїв (у тому числі й нуль-модемні) можна легко замінити на радіозв'язок без будь-яких модифікацій верхніх рівнів. Протокол дає змогу встановлювати й множинні зв'язки (одного пристрою з декількома), і радіозв'язок замінити громіздкі й дорогі мультиплексори й кабелі. Через протокол RFCOMM може працювати протокол OBEX, який використовується в інфрачервоних бездротових з'єднаннях (в ієрархії протоколів IrDA). Через RFCOMM може працювати й протокол PPP, над яким знаходяться протоколи стеку TCP/IP, – це відкриває шлях до всіх програм для інтернету. Через RFCOMM працюють і AT-команди, що керують телефонними з'єднаннями й сервісами передання факсів (ці ж команди використовують у модемах для комутованих ліній).

Спеціальний біт-орієнтований телефонний протокол TCSBIN (Telephony Control protocol-Binary), що визначає сигналізацію виклику для зв'язку пристроїв Bluetooth (мовного зв'язку та обміну даними), теж працює через L2CAP. У протоколі є й засоби керування групами пристроїв TCS.

Інтерфейс хост-контролера HCI (Host Controller Interface) – це метод доступу до апаратно-програмних засобів нижніх рівнів. Він надає набір команд для керування радіозв'язком, отримання інформації про стан і власне передання даних. Через цей інтерфейс взаємодіють протокол L2CAP та апаратура. Фізично апаратура Bluetooth може під'єднуватися до різних інтерфейсів: шини розширення (наприклад, PCCard), шини USB, COM-порту. Для кожного з цих під'єднань є відповідний протокол транспортного рівня HCI-прошарку, що забезпечує незалежність HCI від способу під'єднання.

Лекція 10 **ПРОМИСЛОВІ МЕРЕЖІ**

10.1 Загальні характеристики шини CAN

Уперше ідею CAN було запропоновано в середині 80-х років німецькою компанією Robert Bosch, яка задумувала її як економічний пристрій для об'єднання контролерів, розташованих усередині автомобіля. Традиційний спосіб зв'язку розподілених по об'єкту контролерів джгутами проводів за своєю технічною складністю, цінovими й ваговими параметрами для такого масового виробу, яким є автомобіль, виявився непридатним. Потребувалося альтернативне рішення, що зменшить кількість проводів, тому було запропоновано протокол CAN, для якого достатньо будь-якої провідної пари.

Ідея полягала в тому, щоб створити мережне рішення для розподілених систем, які працюють у реальному часі. Спочатку CAN застосовувалася в автомобілях, але потім область її застосування поширилась і на проблеми автоматизації технологічних процесів.

CAN забезпечує високий рівень захисту даних від пошкодження навіть при роботі в складних умовах (сильні завади), при цьому досягається досить велика

швидкість передавання даних (до 1 Мбіт/с). Важливою перевагою CAN є також те, що розробник системи може впливати на пріоритет повідомлень, щоб найважливіші з них не очікували в черзі на відправлення. Ця здатність CAN дає змогу будувати мережі, що підтримують реальний масштаб часу.

Високий ступінь і надійність мережі завдяки розвиненим механізмам виявлення й виправлення помилок, самоізоляції несправних вузлів, нечутливості до високого рівня електромагнітних завад забезпечує мережі найширшу сферу застосування.

Серед численних факторів, що забезпечили зліт популярності CAN останніми роками, слід зазначити різноманітність елементної бази CAN й її дешевизну.

Чимале значення має і можливість підтримки різнотипних фізичних середовищ передавання даних: від дешевої крученої пари до оптоволокна й радіоканалу. Кілька оригінальних механізмів мережної взаємодії (широкомовлення, побіговий арбітраж) у поєднанні з високою швидкістю передавання даних (до 1 Мбіт/с) сприяють ефективній реалізації режиму реального часу в системах розподіленого керування.

10.2 Топологія мережі CAN

У будь-якій реалізації CAN-носій (фізичне середовище передавання даних) інтерпретується як ефір, у якому контролери працюють як приймачі й передавачі. При цьому, почавши передавання, контролер не перериває слухання ефіру, зокрема він відстежує й контролює процес передавання поточних, переданих ним же, даних. Це означає, що всі вузли мережі одночасно приймають сигнали й передають їх по шині. Неможливо послати повідомлення будь-якого конкретного вузла. Усі вузли мережі приймають весь трафік, що передається по шині. Однак CAN-контролери надають апаратну можливість фільтрації CAN-повідомлень.

CAN-мережу призначено для комунікації так званих вузлів. Кожен вузол має дві складові. Це власне CAN-контролер, який забезпечує взаємодію з мережею й реалізує протокол, і мікропроцесор (CPU).

CAN-контролери з'єднуються з допомогою шини, яка має щонайменше два дроти CAN_H і CAN_L, по яких передаються сигнали з допомогою спеціалізованих ІМС приймачів-передавачів. Крім того, ІМС приймачів-передавачів реалізують додаткові сервісні функції:

- регулювання швидкості наростання вхідного сигналу шляхом змінення струму на вході;
- убудована схема обмеження струму, що захищає виходи передавачів від пошкодження при можливих замиканнях ліній CAN_H і CAN_L з ланцюгами живлення, а також від короткочасного підвищення напруги в цих лініях;
- внутрішній тепловий захист;
- режим зниженого енергоспоживання, у якому приймачі продовжують повідомляти контролеру про стан шини, для того щоб при виявленні на шині інформаційних сигналів він міг вивести приймачі в нормальний режим роботи.

Найбільшого поширення набули два типи приймачів-передавачів (трансиверів): High-Speed (ISO 11898-2) і Fault-Tolerant.

Приймачі-передавачі, виконані відповідно до стандарту High-Speed (ISO 11898-2), є найбільш простими, дешевими й дають змогу передавати дані зі швидкістю до 1 Мбіт/с. Приймачі-передавачі Fault-Tolerant (нечутливі до пошкоджень на шині) дають змогу побудувати високонадійну мережу зі швидкістю передавання даних не вище за 125 кбіт/с.

10.3 Фізичний рівень каналу CAN

Фізичний рівень (Physical Layer) протоколу CAN визначає опір кабелю, рівень електричних сигналів у мережі й т. ін. Існує кілька фізичних рівнів протоколу CAN (ISO 11898, ISO 11519, SAE J2411). У переважній більшості випадків використовується фізичний рівень CAN, визначений у стандарті ISO 11898.

ISO 11898 як середовище передавання визначає двопровідну диференціальну лінію з опором (термінатори) 120 Ом (допускається коливання імпедансу в межах від 108 до 132 Ом).

Максимальна швидкість мережі CAN відповідно до протоколу становить 1 Мбіт/с. При швидкості 1 Мбіт/с максимальна довжина кабелю дорівнює приблизно 40 м. Обмеження на довжину кабелю пов'язане з кінцевою швидкістю поширення сигналу й механізмом побігового арбітражу (під час арбітражу всі вузли мережі мають отримувати поточний біт передавання одночасно, а сигнал має встигнути поширитися по всьому кабелю за одиничний відлік часу в мережі).

Співвідношення між швидкістю передавання й максимальною довжиною кабелю: швидкість передавання 1000 кбіт/с – максимальна довжина мережі 40 м; швидкість передавання 500 кбіт/с – максимальна довжина мережі 100 м; швидкість передавання 250 кбіт/с – максимальна довжина мережі 200 м; швидкість передавання 125 кбіт/с – максимальна довжина мережі 500 м; швидкість передавання 10 кбіт/с – максимальна довжина мережі 6 км.

Розніми для мережі CAN досі не стандартизовано. Кожен протокол високого рівня зазвичай визначає свій тип рознімів для CAN-мережі.

Логічний нуль реєструється, коли на лінії CAN_H сигнал є вищим, ніж на лінії CAN_L, логічна одиниця – коли сигнали CAN_{HI} і CAN_{LO} є однаковими (відрізняються менш ніж на 0,5 В).

Використання такої диференціальної схеми передавання робить можливою роботу CAN-мережі в дуже складних зовнішніх умовах.

Логічний нуль має назву доміантного біта, а логічна одиниця – рецесивного. Ці назви відображають пріоритет логічних одиниць й нуля на шині CAN.

При одночасному передаванні в шину логічних нуля й одиниці на шині буде зареєстровано тільки логічний нуль (доміантний сигнал), а логічну одиницю буде заглушено (рецесивний сигнал).

10.4 Арбітраж шини CAN

Швидкодія CAN-мережі (до 1 Мбіт/с) досягається завдяки механізму недеструктивного арбітражу шини за шляхом порівняння біт конкуруючих повідомлень, тобто якщо станеться так, що одночасно почнуть передання кілька контролерів, то кожен з них буде порівнювати біт, який збирається передати на шину, з бітом, який намагається передати на шину конкурентний контролер. Якщо значення цих бітів є однаковим, то обидва контролери будуть намагатися передати наступний біт. І так відбувається доти, доки значення переданих бітів не стануть різними. Тепер контролер, який передавав логічний нуль (більш пріоритетний сигнал), буде продовжувати передання, а другий контролер перерве своє передання, поки шина знову не звільниться. Звісно, якщо шина в певний момент є зайнятою, то контролер не почне передання до моменту її звільнення.

У цій специфікації CAN припускається, що всі CAN-контролери отримують сигнали з шини одночасно, тобто в один й той же час один й той же біт приймається всіма контролерами в мережі. З одного боку, такий стан речей робить можливим побітовий арбітраж, а з іншого – обмежує довжину CAN-bus. Сигнал поширюється по CAN-bus з величезною, але кінцевою швидкістю, і для правильної роботи CAN потрібно, щоб всі контролери «почули» його майже одночасно, тому що кожен контролер приймає біт протягом певного проміжку часу, що відлічується системним годинником. Таким чином, чим більшою є швидкість передання даних, тим меншою буде довжина CAN-bus.

10.5 Структура формату передання даних

Дані по CAN-мережі пересилаються у вигляді окремих кадрів стандартного формату. Найбільш важливими полями є поле ідентифікатора (identifier) і власне дані (data).

Ідентифікатор є унікальним ім'ям для типу повідомлення й визначає те, хто прийме і як буде інтерпретовано наступне поле даних. Чому саме (арифметично) дорівнює це число, у загальному випадку не має значення. Така контекстна адресація відрізняється кількома перевагами для мереж невеликого масштабу і забезпечує максимально можливу простоту модернізації. Оскільки децентралізовані контролери не пов'язані між собою логічно, додавання нового елемента в систему не вплине на поведінку всіх інших.

Більш цікавим є використання ідентифікаторів як основного інструмента, який використовується в процедурі вирішення колізій. У CAN як основний критерій для розглянення колізій, прийняття рішення щодо черги в ефірі використовується пріоритет повідомлень. Якщо одночасно кілька станцій почали передання і при цьому сталася колізія, то відбувається суперпозиція переданих ідентифікаторів. Ідентифікатори послідовно, побітово (bitwise), починаючи зі старшого, накладаються один на одного і в їх «протиборстві» виграє той, у кого менше арифметичне значення ідентифікатора, а отже, вищий пріоритет. Домінантний «нуль» заглушить одиницю й у будь-якому випадку до кінця

передання поля ідентифікатора стане більшим за пріоритетним значенням. Отже, система дає змогу на рівні проектування (і визначення ідентифікатора) для будь-якого повідомлення заздалегідь визначити його пріоритетність в обслуговуванні.

Пріоритетність повідомлення, таким чином, зумовлюється значенням ідентифікатора. Пріоритет є тим більшим, чим меншим є ідентифікатор. Зазвичай контролер дає змогу задавати лише ці два поля. Інші поля використовуються для передання специфічних даних, необхідних для функціонування CAN.

Дані в CAN передаються короткими повідомленнями – кадрами стандартного формату. У CAN існує чотири типи повідомлень: Data Frame, Remote Frame, Error Frame, Overload Frame.

Data Frame – це найбільш часто використовуваний тип повідомлення. Він складається з таких основних частин: поле арбітражу (arbitration field) та поле даних (data field). Поле арбітражу визначає пріоритет повідомлення, коли два або більше вузлів одночасно намагаються передати дані в мережу.

Поле арбітражу для стандарту CAN-2.0A складається з 11-бітового ідентифікатора й 1 біта RTR (retransmit), для стандарту CAN-2.0B – з 29-бітового ідентифікатора й 1 біта RTR (retransmit).

Слід ще раз зазначити, що поле ідентифікатора, незважаючи на свою назву, ніяк не ідентифікує ні вузол в мережі, ні вміст поля даних.

Біт RTR для Data завжди виставлено в логічний нуль (домінантний сигнал). Поле даних (data field) містить від 0 до 8 байтів даних. Поле CRC (CRC field) містить 15-бітову контрольну суму повідомлення, яка використовується для виявлення помилок слот-підтвердження (Acknowledgement Slot) (1 біт), кожен CAN-контролер, який правильно прийняв повідомлення, посилає біт-підтвердження в мережу. Вузол, який послав повідомлення, слухає цей біт, і в разі якщо підтвердження не надійшло, то повторює передання. У разі прийняття слота-підтвердження передавальний вузол може бути «впевнений» лише в тому, що хоча б один з вузлів у мережі правильно прийняв його повідомлення.

Remote Frame – це Data Frame без поля даних і з виставленим бітом RTR (однорецесивні біти). Основне призначення Remote кадру – ініціація одним з вузлів передання в мережу даних іншим вузлом. Така схема дає змогу зменшити сумарний трафік мережі. Однак на практиці Remote Frame зараз використовується нечасто (наприклад, у DeviceNet Remote Frame зовсім не використовується).

Error Frame – це повідомлення, яке явно порушує формат повідомлення CAN. Передання такого повідомлення призводить до того, що всі вузли мережі реєструють помилку формату CAN-кадру, і автоматично передають у мережу Error Frame. Результатом цього процесу є автоматичне повторне передання даних у мережу передавальним вузлом. Error Frame складається з поля Error Flag, що має шість бітів однакового значення (і, таким чином, Error frame порушує перевірку Bit Stuffing), і поля Error Delimiter, що має вісім рецесивних бітів. Error Delimiter дає змогу іншим вузлам при виявленні Error Frame послати свій Error Flag.

Overload Frame – повторює структуру й логіку роботи Error-кадру, з тією різницею, що його використовує перевантажений вузол, який в цей момент не може обробити повідомлення, яке надходить, і тому просить з допомогою

Overload-кадру про повторне передання даних. Сьогодні Overload-кадр майже не використовується.

10.6 Механізм оброблення помилок

Надійність CAN-мережі визначається також механізмами виявлення помилок. У стандарті CAN визначено такі методи виявлення помилок у мережі CAN:

- Check Bit monitoring;
- Bit stuffing;
- Frame check;
- ACKnowledgement Check;
- Check CRC.

Check Bit monitoring – кожен вузол під час передання бітів у мережу порівнює значення біта, що передається, зі значенням біта, яке виникає на шині. Якщо ці значення не збігаються, то вузол генерує помилку Bit Error. Істотно, що під час арбітражу на шині (передання поля арбітражу в шину) цей механізм перевірки помилок відмикається.

Bit stuffing – коли вузол передає послідовно в шину п'ять бітів з однаковим значенням, то він додає шостий біт з протилежним значенням. Приймальні вузли цей додатковий біт видаляють. Якщо вузол виявляє на шині більше п'яти послідовних бітів з однаковим значенням, то він генерує помилку Stuff Error.

Frame Check – деякі частини CAN-повідомлення мають однакове значення у всіх типах повідомлень, тобто протокол CAN точно визначає, які рівні напруги й коли мають появлятися на шині. Якщо формат повідомлень порушується, то вузли генерують помилку Form Error.

ACKnowledgement Check – кожен вузол отримавши правильне повідомлення, посилає в мережу домінуючий нульовий біт. Якщо ж цього не відбувається, то передавальний вузол реєструє помилку Acknowledgement Error.

CRC Check – кожне повідомлення CAN містить CRC-суму, і кожен приймальний вузол підраховує значення CRC для кожного отриманого повідомлення. Якщо підраховане значення CRC-суми не збігається зі значенням CRC у тексті листа, який приймає, то вузол генерує помилку CRC Error.

Кожен вузол мережі CAN під час роботи намагається виявити одну з п'яти можливих помилок. Якщо помилку виявлено, то вузол передає у мережу Error Frame, руйнуючи тим самим весь поточний трафік мережі (передання й приймання поточного повідомлення). Усі інші вузли виявляють Error Frame і скидають прийняте повідомлення.

Крім того, кожен вузол має два лічильника помилок: Transmit Error Counter (лічильник помилок передання) й Receive Error Counter (лічильник помилок приймання).

Ці лічильники збільшуються або зменшуються відповідно до кількох правил. Самі правила керування лічильниками помилок є досить складними, але зводяться до простого принципу: помилка передання збільшує лічильник Transmit Error на 8, помилка приймання – до збільшення лічильника Receive Error на 1, будь-яке

коректне передання/приймання повідомлення – до зменшення відповідного лічильника на 1. Ці правила призводять до того, що лічильник помилок передання передавального вузлів збільшує швидше, ніж лічильник помилок приймання приймальних вузлів. Це правило відповідає припущенням щодо великої ймовірності того, що джерелом помилок є передавальний вузол.

Кожен вузол CAN-мережі може перебувати в одному з трьох станів. Під час старту вузол перебуває в стані Error Active. Коли значення хоча б одного з двох лічильників помилок перевищує 127, вузол переходить у стан Error Passive. Коли значення хоча б одного з двох лічильників перевищує 255, вузол переходить у стан Bus Off.

Вузол перебуває в стані Error Active і в разі виявлення помилки на шині передає в мережу Active Error Flags. Active Error Flags складається з шести доміантних бітів, тому його реєструють усі вузли.

Вузол у стані Passive Error передає в мережу Passive Error Flags при виявленні помилки. Passive Error Flags складається з шести рецесивних бітів, тому інші вузли мережі його не помічають і Passive Error Flags лише призводить до збільшення Error-лічильника вузла.

Вузол у стані Bus Off нічого не передає в мережу (не тільки Error-кадри, але й узагалі ніякі інші).

10.7 Адресація і протоколи високого рівня

Мережних сервісів специфікації Robert Bosch CAN Specification 2.0A/B і міжнародного стандарту ISO 11898 найчастіше явно недостатньо для ефективного розроблення CAN-мереж. Справа в тому, що в цих документах описано лише два найнижчих рівня еталонної моделі взаємозв'язку відкритих систем OSI/ISO – фізичний й каналний, а також формати повідомлень, процеси передання даних завдовжки до восьми байтів, механізми виявлення помилок, деякі фізичні параметри середовища передання даних (тільки в ISO 11898) та ін.

Однак «за кадром» залишаються такі важливі на етапі розроблення моменти, як адресація вузлів, розподіл між ними CAN-ідентифікаторів, інтерпретація вмісту фрейму даних, передання даних довжиною понад вісім байтів та ін.

У CAN не існує явної адресації повідомлень і вузлів, повідомлення не мають явної адресації приймача. Джерело виставляє на шину свої ідентифікатор і дані, а приймач самостійно, виходячи з розв'язуваних завдань, обробляє прийняті дані від цього джерела або ігнорує їх.

У протоколі CAN не зазначено, що поле арбітражу (Identification field + RTR) має використовуватися як ідентифікатор повідомлення або вузла. Таким чином, ідентифікатори повідомлень та адреси вузлів можуть знаходитися в будь-якому полі повідомлення (у полі арбітражу або полі даних, або і там, і там).

З іншого боку, у стандарті протоколу передбачено можливість віддаленого запиту даних (RTR). На відміну від попереднього опису приймач не очікує появи

необхідних даних, а запитує дані в необхідного вузла. Так само протокол не забороняє використовувати поле арбітражу для передання даних.

Стандарт CAN не регламентує, яким чином конкретні програми будуть передавати специфічні для себе дані по мережі CAN. Виникає потреба у використанні якого-небудь протоколу верхнього рівня. Можна придумати свій протокол, який давав би додаткам змогу працювати з CAN-мережею просто й зручно, але навряд чи варто витратити на це сили, якщо вже існує безліч високорівневих протоколів на основі CAN-технології, причому це – відкриті протоколи, тобто можна отримати вже готові специфікації і навіть брати участь у подальшому розвитку цих систем. Тому з початком масового випуску CAN компонентів і поширення CAN-додатків кілька незалежних компаній і некомерційних асоціацій в області систем промислової автоматизації, транспорту тощо провадять роботу зі створення і стандартизації специфікацій протоколів верхнього рівня HLP (Higher Level Protocol) для CAN-мереж.

Утилізація поля арбітражу й поля даних, розподіл адрес вузлів, ідентифікаторів повідомлень і пріоритетів у мережі є предметом розгляду так званих протоколів високого рівня (HLP – Higher Layer Protocols).

У назві HLP відображено той факт, що протокол CAN описує тільки два нижніх рівні сталонної мережної моделі ISO/OSI, а інші рівні описують протоколи HLP.

До теперішнього часу відомо вже понад чотири десятки CAN HLP. Серед такого різноманіття CAN HLP найбільшого поширення, особливо в системах промислової автоматизації, набули чотири, які підтримує асоціація CiA: CAL/CANopen, CAN Kingdom, DeviceNet і SDS (Smart Distributed System) та CAL/CANopen.

Розроблення й підтримка відкритого протоколу прикладного рівня для мереж промислової автоматизації були одними з пріоритетних цілей створення організації CiA 1992 року. Основою такого протоколу став HLP, розроблений фірмою Philips, після доопрацювання й удосконалення якого робочою групою CiA 1993 року було опубліковано специфікацію CAL CAN Application Level (CiA DS 20x).

Мережні CAN-додатки, що базуються на прикладному рівні CAL, сьогодні успішно працюють у медичній електроніці, системах контролю дорожнього руху, на транспорті, у промисловому обладнанні. Результатом доповнення CAL (точніше, деякої його підмножини) системою профілів (пристроїв, інтерфейсів, додатків і т. д.) і специфікаціями фізичного рівня (типи з'єднувачів, правила битового квантування і т. д.) стала поява більш «конкретного» стандарту протоколу CANopen. По суті CANopen є додатком прикладного рівня CAL. Спочатку CANopen призначався для мереж керування рухомими механізмами в системах промислової автоматики. Однак згодом протокол став застосовуватись у медицині, морській електроніці, на транспорті й у системах автоматизації будівель. CANopen базується на двох рівнях стандарту CAN (ISO 11898 й Bosch CAN Specification 2.0 A/B).

На додаток до специфікацій фізичного рівня ISO 11898 (середовище передання даних двопровідна диференціальна лінія) CANopen містить власні

правила бітового квантування, а також визначає три рекомендованих типи з'єднувачів. При розведенні контактів для всіх типів з'єднувачів передбачено можливість подання живлення на трансивери вузлів, що мають гальванічну розв'язку. У мережі CANopen визначено вісім градацій швидкостей передання даних: 1 Мбіт/с, 800, 500, 250, 125, 50, 20 і 10 кбіт/с. Підтримка швидкості 20 кбіт/с є обов'язковою для всіх модулів.

CAN Kingdom

Протокол шведської компанії KVASER-AB (www.kvaser.se) займає особливе місце серед CAN HLP завдяки оригінальній концепції мережної взаємодії та ефективності CAN-додатків на його основі.

Роботи над першою версією (поточна є третьою) протоколу CAN Kingdom 1990 року розпочалися завдяки багаторічному досвіду компанії в області створення систем розподіленого керування. Протокол було спеціально розроблено для керування рухомими машинами й механізмами, промисловими роботами, текстильними верстатами, мобільними гідравлічними пристроями, що дає змогу досягти високої продуктивності в режимі реального часу при задоволенні жорстких вимог безпеки.

CAN Kingdom є також основою американського військового стандарту CDA 101 і широко використовується у військовій техніці від надувних човнів і систем наведення на цілі до надзвукових винищувачів і ракет. Основною метою створення протоколу було надання системному розробнику максимальної свободи в реалізації своїх ідей при будівництві мережі, зберігши при цьому можливість використання стандартних модулів від незалежних виробників. CAN Kingdom не є «готовим» протоколом, наприклад, відносно стандартів типу CANopen або DeviceNet. Це, скоріше, набір примітивів метапротоколу, з допомогою яких можна «скласти» протокол під конкретну мережу модулів. Цим досягається унікальне поєднання простоти інтеграції готових модулів з високим ступенем «закритості» оригінального протоколу. Наріжним каменем концепції мережної взаємодії CAN Kingdom є принцип «Модулі обслуговують мережу» (MSN Modules Serves the Network) на відміну від принципу «Мережа обслуговує користувачів» (NSM Network Serves the Modules), що є властивим для комп'ютерних мереж.

У мережі CAN Kingdom не існує якихось рекомендованих швидкостей передання даних, але за перші 200 мс після подання живлення вузол зобов'язаний налаштуватися на прослуховування шини на швидкості 125 кбіт/с.

DeviceNet

DeviceNet – протокол, розроблений та опублікований 1994 року компанією Allen-Bradley (www.ab.com) корпорації Rockwell і згодом переданий до спеціально організованої для його підтримки асоціації ODVA (Open DeviceNet Vendor Association Inc., www.odva.org).

DeviceNet – недороге, просте й ефективне рішення для об'єднання різноманітних пристроїв промислової автоматизації незалежних виробників у єдину систему: фото- і термодатчики, стартери, зчитувачі штрихових кодів, елементи людино-машинного інтерфейсу клавіатури, дисплейні панелі нарівні з керувальними пристроями PLC, комп'ютерами й т. д. При розробленні протоколу

крім зниження вартості також було завдання спрощення та уніфікації діагностики подібних пристроїв. Перші пристрої, що задовольняють специфікації DeviceNet, появилися на ринку на початку 1995 року. DeviceNet також побудовано на двох нижніх рівнях стандарту CAN, доповнених більш детальними, ніж в інших HLP, специфікаціями фізичного середовища.

Мережа DeviceNet має шинну топологію з відводами. Фізичним середовищем передавання є чотирипроводовий кабель (CAN_H, CAN_L, Vcc, Ground), причому можливі два його різновиди: товстий (зовнішній діаметр 12,2 мм) і тонкий (6,9 мм). Передбачено лише три значення швидкості передавання даних: 125, 250 і 500 кбіт/с.

Важливою особливістю мережі DeviceNet є можливість живлення модулів безпосередньо від мережного кабелю (24 В, до 8 А на товстому кабелі), а також допускається застосування кількох джерел живлення в будь-якій точці шини. Усе це дає змогу побудувати автономну мережу, що не залежить від наявності або якості зовнішнього живлення, а при необхідності дасть змогу легко демонтувати її знову розгорнути систему на новому місці.

Мережа DeviceNet допускає «гаряче» (без знеструмлення мережі) під'єднання й від'єднання модулів. Стандарт DeviceNet містить також докладний опис численних типів перехідників, розгалужувачів (одиночних і багатопортових), з'єднувачів (Mini, Micro), мережних відводів і т. ін. При описі організації типів даних, мережного поводження модулів у DeviceNet використовується об'єктно-орієнтована модель.

Максимальна кількість вузлів у мережі DeviceNet-64.

SDS (Smart Distributed System)

SDS – розроблення компанії Honeywell Inc. (Micro Switch Division, www.honeywell.sensing.com). Нарівні зі стандартом DeviceNet, SDS є ще одне недороге й закінчене рішення для мережного керування інтелектуальними датчиками й актуаторами від центрального контролера (PLC, комп'ютера) у системах промислової автоматизації. За ступенем завершеності від специфікацій фізичного середовища до прикладного рівня, орієнтуванням на зниження вартості SDS-стандарт нагадує DeviceNet. Шинна топологія являє собою лінійну шину (магістраль або транк) з короткими відводами.

Визначено два базові типи кабельного розводу: Mini (застосовуваний при складанні транка мережі) – чотирипроводовий кабель з максимальним струмовим навантаженням 8А, п'ятиконтактний рознім і Micro (для під'єднання фізичних пристроїв до мережі) – чотирипроводовий кабель, 3 А, чотириконтактний рознім без окремого контакту для екрана кабелю.

У мережі SDS допускається й звичайна дротова розводка з використанням відкритих клемних з'єднувачів. Для всіх типів кабельної розводки й з'єднувачів, як й в мережі DeviceNet, передбачено підведення напруги живлення до вузлів.

Мережа SDS завжди потребує наявності єдиного майстра–менеджера мережі щонайменше на етапі ввімкнення для виконання автоналаштування швидкості передавання модулів. Під час роботи мережі допускається наявність кількох майстрів на шині, але вони мають функціонувати у межах своїх адрес доменів, а

при ввімкненні мережі тільки один з них може брати на себе функцію мережного менеджера для автоналаштування швидкості пристроїв.

10.8 Інтерфейс PROFIBUS

Назва інтерфейсу PROFIBUS походить від скороченого PROcess FIEld BUS, що приблизно перекладається як «промислова шина для технологічних процесів». Стандарт Profibus було спочатку прийнято в Німеччині 1987 року, потім, 1996 року, він став міжнародним (EN 50170 й EN 50254). Мережа Profibus використовує тільки перший і другий рівні моделі OSI (таблиця 10.1).

Profibus має три модифікації: Profibus DP, Profibus FMS і Profibus PA.

Profibus DP (Profibus for Decentralized Peripherals – «Profibus для децентралізованої периферії») використовує рівні 1 і 2 моделі OSI, його також призначено для користувачів інтерфейсу, який до моделі OSI не входить. Безпосередній доступ з призначеного для користувача додатку до каналного рівня здійснюється з допомогою DDLM (Direct Data Link Mapper – «прямий перетворювач для каналного рівня»). Інтерфейс забезпечує функції, необхідні для зв'язку з пристроями введення-виведення й контролерами. Profibus DP на відміну від FMS і PA побудовано таким чином, щоб забезпечити найбільш швидкий обмін даними з пристроями, під'єднаними до мережі.

Таблиця 10.1 – Характеристика мережі Profibus

№ п/п	Призначення рівня	Profibus DP	Profibus FMS	Profibus PA
7	Прикладний	Немає	Fieldbus Message Specification (FMS)	Немає
6	Подання	Немає		
5	Сеансовий			
4	Транспортний			
3	Мережний			
2	Канальний (передання даних)	FDL	FDL	IEC 1158–2
1	Фізичний	RS-485, оптоволоконний інтерфейс	RS-485, оптоволоконний інтерфейс	Інтерфейс IEC 1158–2

Profibus FMS (Profibus з FMS протоколом) використовує рівень 7 моделі OSI і застосовується для обміну даними з контролерами й комп'ютерами на реєстровому рівні. Profibus FMS надає великої гнучкості

при переданні великих обсягів даних, але програє протоколу DP у популярності через свою складність.

Profibus FMS і DP використовують один й той же фізичний рівень, що базується на інтерфейсі RS-485, і можуть працювати в загальній мережі.

Profibus PA (Profibus for Process Automation – для автоматизації технологічних процесів) використовує фізичний рівень на основі стандарту IEC 1158–2, який забезпечує живлення мережних пристроїв через шину й не є сумісним з RS-485. Особливістю Profibus PA є можливість роботи у вибухонебезпечній зоні.

В останні роки появився стандарт PROFINet, який базується на Industrial Ethernet і технологіях COM, DCOM. Він легко забезпечує зв'язок промислової мережі Profibus з офісною мережею Ethernet.

Profibus є багатомайстерною мережею (з кількома ведучими пристроями). Веденими пристроями є зазвичай пристрої введення-виведення, клапани, вимірювальні перетворювачі. Вони не можуть самостійно отримати доступ до шини й тільки відповідають на запити ведучого пристрою.

10.9 Фізичний рівень інтерфейсу PROFIBUS

На фізичному рівні Profibus DP і FMS використовують стандарт RS-485 при швидкості передання до 12 Мбіт/с і з розмірами сегментів мережі до 32 пристроїв. Кількість пристроїв можна збільшити з допомогою повторювачів інтерфейсу.

Особливі вимоги встановлено до мережного кабелю. Він повинен мати хвильовий опір від 135 до 165 Ом при пороговій ємності не більше 35 пФ/м, площа поперечного перерізу провідників має становити понад 0,34 мм², а пороговий опір – не більше 110 Ом/км. Кабель має складатися з однієї або двох кручених пар з мідним екраном у вигляді сітки або фольги.

У стандарті для шини Profibus рекомендується рознім D-sub (DB–9) з дев'ятьма контактами, цоколюнку розніму наведено в таблиці 10.2. На пристроях встановлюється рознім з гніздами, на кабелі – зі штирями. Якщо необхідності мати ступінь захисту IP65/67, то рекомендується використовувати циліндричний рознім типу M12 (IEC 947-5-2), HAN-BRID або гібридний рознім фірми Siemens.

По обидва боки від лінії передання з допомогою мікроперемикачів під'єднуються погоджувальні резистори, які конструктивно встановлено в усі мережні розніми. При швидкостях передання понад 1,5 Мбіт/с для узгодження лінії додатково використовуються плоскі (друковані) котушки індуктивності.

Для збільшення дальності передання в Profibus передбачено можливість роботи з оптоволоконним кабелем. При використанні скляного оптоволокна дальність зв'язку може бути збільшено до 15 км. Оптоволоконні інтерфейси виконують у вигляді змінних модулів для контролерів.

Таблиця 10.2 – Характеристика розніму

Контакт	Сигнал	Примітка
1	Shield	Екран
2	M24	-24 В
3	Rx/Tx-DP	Приймання/передання даних (позитивний висновок, провід В)
4	CNTR-P	Сигнал для керування напрямком передання, позитивний висновок
5	DGND	Загальний провід даних
6	VP	Напруга живлення, «+»
7	P24	+24 В
8	Rx/Tx-N	Приймання/передання даних (негативний висновок, провід А)
9	CNTR-N	Сигнал для керування напрямком передання, негативний висновок

Profibus PA використовує фізичний рівень, що відповідає стандарту IEC 1158-2. Дані передаються з допомогою рівнів струму +9 мА і -9 мА («струмова петля»). Використовується манчестерський код (логічний нуль відповідає зміненню негативного струму на позитивний, а логічна одиниця – позитивного на негативний). Швидкість передання становить 31,25 кбіт/с, як лінія передання використовується кручена пара в екрані або без нього. Один сегмент мережі може містити до 32 пристроїв. Максимальна довжина кабелю становить 1,9 км. У кожному сегменті по обидва боки від кабелю під'єднано RC-ланцюжки, що складаються з послідовно з'єднаних конденсатора ємністю 1 мкФ і резистора опором 100 Ом. Завдяки низькій енергії сигналу, що передається Profibus PA є іскробезпечним електричним колом і його можна використовувати у вибухонебезпечних зонах.

Для передання даних використовується NRZ-кодування й 11-бітовий формат, ідентичний формату HART-протоколу (рисунок 10.1), що містить стартовий біт «0», 8 бітів даних молодшими розрядами вперед, біт паритету (парний) і стоп-біт «1». Біт паритету дорівнює нулю, якщо кількість бітів у слові є парною, і дорівнює одиниці в іншому випадку.

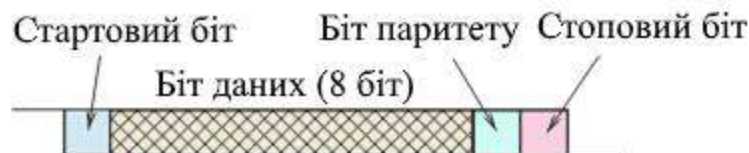


Рисунок 10.1 – Характеристика сигналу

У слові «11000011» кількість одиниць є парною, тому біт паритету встановлюється таким, що дорівнює 0. Такий формат використовується для всіх

без винятку слів, включаючи заголовки повідомлень. При передаванні слів, довгих за 8 бітів, старший байт відправляється першим, за ним інші в порядку старшинства.

Кілька років тому для Profibus було розроблено модифікацію інтерфейсу RS-485 для вибухонебезпечних зон, яка отримала назву RS-485-IS (Intrinsically Safe) – внутрішньо безпечний. Істотною її відмінністю є наявність резисторів, що обмежують струм у лінії до значень, установлених у стандарті на іскробезпечні електричні ланцюги.

10.10 Канальний рівень Profibus DP

Далі розглянемо тільки Profibus DP, оскільки він набув незрівнянно більшого поширення, ніж FMS і PA.

Канальний рівень моделі OSI у Profibus має назву FDL-рівня (Fieldbus Data Link – промисловий канал зв'язку). Об'єкт MAC (Medium Access Control – керування доступом до каналу) на каналному рівні визначає процедуру передавання даних пристроями, включаючи керування правами на передавання даних через мережу. Протокол каналного рівня забезпечує виконання таких важливих вимог: під час комунікації між ведучими пристроями необхідно забезпечити виконання кожним з них свого завдання протягом заздалегідь визначеного проміжку часу, взаємодія ведучих пристроїв (контролерів) з веденими має відбуватися максимально швидко.

Для доступу ведучих пристроїв до мережі Profibus використовується метод передавання маркера (рисунок 10.2). У цьому методі мережа має логічну топологію кільця (кільця на рівні адрес пристроїв) і кожен ведучий пристрій отримує доступ до мережі тільки при отриманні маркера. Маркер відіграє роль арбітра, який надає пристрою права доступу. Після закінчення певного часу цей пристрій має передати маркер наступному ведучому пристрою, який отримує доступ також на час, поки маркер знаходиться у нього. Таким чином, кожному ведучому пристрою виділяється точно заданий проміжок часу, який можна встановити під час конфігурування системи.

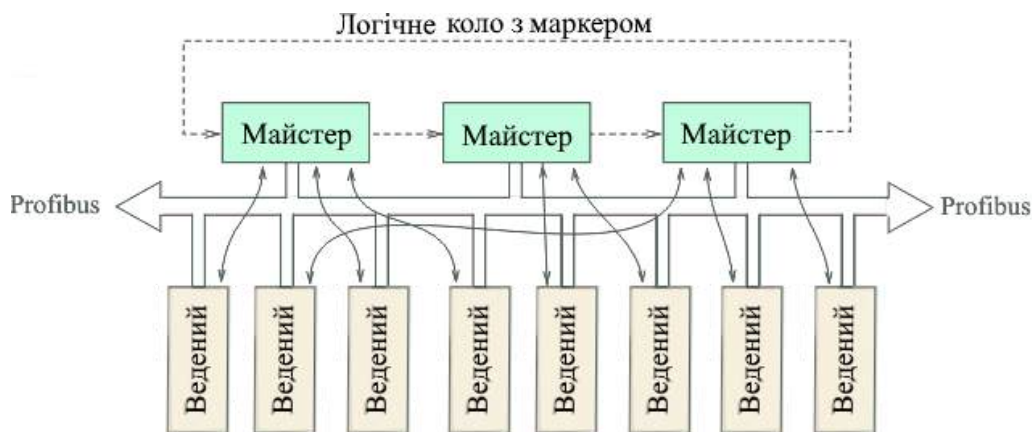


Рисунок 10.2 – Метод передавання маркера

Кожному майстру в мережі призначаються свої ведені пристрої. У методі «ведучий/ведений» процедуру комунікації з веденими пристроями виконує майстер, який має маркер. На час володіння маркером майстер стає ведучим також відносно інших майстрів, тобто може виконувати з ними комунікацію типу «майстер – майстер».

Profibus має також ширококомовний режим роботи, коли ведучий пристрій посилає повідомлення «всім», не чекаючи повідомлення про отримання, і багатоабонентський режим, коли ведучий пристрій посилає одне й те саме повідомлення відразу декільком учасникам мережі.

До завдань об'єкта MAC активного пристрою (який отримав маркер) належать виявлення наявності або відсутності маркера відразу після початку роботи мережі, передання маркера наступному пристрою в порядку збільшення адрес, видалення адрес поламаних або вимкнених пристроїв і додавання нових, відновлення втраченого маркера, усунення дублікатів маркерів, дублювання мережних адрес і забезпечення заданого періоду звернення маркера по мережі.

Основна функція комунікаційного профілю DP полягає у ефективному обміні даними ПЛК або комп'ютера з датчиками й виконавчими пристроями. Обмін даними з цими пристроями зазвичай виконується періодично, але комунікаційний профіль DP надає також додатковий сервіс аперіодичного обміну для установлення параметрів, контролю режимів роботи й оброблення сигналів тривоги (алармів).

У звичайному режимі центральний контролер (ведучий) періодично зчитує інформацію, що надходить на входи ведених пристроїв і записує інформацію для їх виходів. Додатково до цього періодичного обміну даними DP забезпечує інформацією потужні засоби для діагностики та інсталяції системи, а також стійкість до зовнішніх дестабілізуювальних факторів.

Засоби діагностики DP забезпечують швидке виявлення місця виникнення помилки й пересилання відповідного повідомлення ведучому пристрою. Діагностичні повідомлення мають три рівні:

- пристрою (що стосується загальних ознак роботоздатності пристрою, таких як перегрів, вихід напруги за допустимі межі та ін.);
- модуля введення-виведення, що входить до складу пристрою модульної конструкції;
- каналу модуля.

До однієї мережі може бути під'єднано до 128 пристроїв (але не більше 32 в одному сегменті). Специфікація для конфігурації системи містить кількість вузлів мережі, розподіл адрес пристроїв, формат діагностичних повідомлень, параметри шини.

У мережі можуть використовуватися пристрої трьох типів.

1 DP-майстер класу 1 (DPM1) – центральний контролер, який циклічно обмінюється інформацією з веденими пристроями із заздалегідь визначеним періодом.

2 DP-майстер класу 2 (DPM2) – пристрій, призначений для конфігурації, налагодження, обслуговування або діагностики системи.

3 Ведений пристрій – такий, що збирає інформацію або видає її виконавчим пристроям.

Ці пристрої можуть надсилати або приймати не більше 256 байтів інформації за один цикл обміну.

Комунікаційний профіль DP дає змогу конфігурувати як одномайстерну, так й багатомайстерну мережу. В одномайстерній мережі ведучий пристрій (майстер) може посилати запити й отримувати відповіді тільки від ведених пристроїв.

У багатомайстерній мережі є кілька ведучих пристроїв, які мають свої одномайстерні підмережі й у межах підмережі є пристроями класу DPM1. Ведучі пристрої у багатомайстерній мережі можуть бути також пристроями класу DPM2. Вхідні й вихідні дані підпорядкованих пристроїв можуть бути прочитаними будь-яким майстром мережі. Однак записувати дані у пристрої може тільки один майстер, який при конфігуруванні системи було позначено як DPM1.

Ведучий контролер (DPM1) може перебувати в одному з трьох станів: Stop – коли не відбувається обміну даними, Clear – коли DPM1 може зчитувати дані, але не може записувати їх, і виходи всіх пристроїв переводяться у безпечні стани, Operate – звичайний робочий стан.

Якщо у системі виникає повідомлення про помилку, то DPM1 устанавлює виходи всіх пристроїв виведення у безпечний стан, а сам переходить у стан «Clear». Безпечним вважається такий стан, при якому виконавчі пристрої перебувають у безпечному (для людини або системи) стані. Такий стан самоконтролю системи може бути встановлено (або не встановлено) під час її конфігурації. При відімкненому стані самоконтролю система продовжує працювати, незважаючи на виникнення помилок.

При конфігуруванні системи користувач призначає кожному ведучому свої ведені пристрої й черговість їх опитування, а також указує пристрої, які не треба опитувати періодично.

Передача даних між майстром DPM1 і веденими пристроями поділяється на три фази: параметризація, конфігурація й передача даних. У фазах параметризації й конфігурації перевіряється, чи відповідає конфігурація й параметри веденого пристрою запланованим у DPM1 установам. Перевіряється тип пристрою, формат і довжина переданих повідомлень, кількість входів або виходів.

Profibus DP має режим синхронізації виведення. Для цього надсилається поширена керувальна команда синхронізації, при отриманні якої відбувається одночасне змінення стану виходів усіх пристроїв виведення. Є також команда «заморожування», під час роботи якої входи всіх пристроїв уведення зберігають свій поточний стан і перестають реагувати на зміни, що надходять на входи сигналів, поки не надійде команда «розморозування». Ці команди використовуються для синхронізації введення даних. Команди синхронізації можуть надсилатися всім пристроям мережі, групі або одному пристрою.

Для виявлення помилок у передавальних пристроях передбачено механізм тимчасового моніторингу (спостереження), який діє як у ведених, так і у ведучих пристроях. Проміжок моніторингу встановлюється при конфігуруванні системи. Ведучий пристрій (DPM1) контролює процес передавання даних веденим пристроєм

з допомогою таймера. Для кожного підпорядкованого пристрою використовується свій таймер. Якщо протягом інтервалу спостереження не надходять коректні дані, то видається діагностичне повідомлення для користувача. Якщо ввімкнено режим автоматичної реакції на помилки, то ведучий пристрій устанавлює всі виходи в безпечні стани, а сам переходить у стан «Clear».

Ведений пристрій також виконує контроль ведучого пристрою або лінії передання. Для цього використовується сторожовий таймер. Якщо від ведучого пристрою не надходять дані протягом періоду роботи сторожового таймера, то ведений пристрій автоматично переводить свої виходи в безпечний стан. Для більшого захисту виходів у багатомайстерній системі тільки один (уповноважений) майстер має прямий доступ до змінення станів виходів пристрою. Усі інші майстри можуть зчитувати тільки «зображення» сигналів на входах і виходах пристрою.

У Profibus є також розширені DP-функції, які дають змогу передавати аперіодичні функції читання й запису, а також сигнали тривоги паралельно й незалежно від періодичного пересилання даних, устанавленого значення користувачем при конфігуруванні системи. Це дає змогу, наприклад, з допомогою DPM2 змінювати параметри знову під'єднаних ведених пристроїв або зчитувати стан будь-яких пристроїв мережі без зупинення системи. Ці службові функції виконуються аперіодично з низьким пріоритетом паралельно з робочим процесом передання даних у системі. Для забезпечення такої можливості при параметризації в усій мережі встановлюють збільшений цикл звернення маркера, щоб шина була завантажена на 100 %.

10.11 Передання повідомлень Profibus

Profibus використовує два типи сервісів для передання повідомлень: **SRD** (Send and Receive Data with acknowledge – відправлення й приймання даних з повідомленням) і **SND** (Send Data with No acknowledge – передання даних без повідомлення).

Сервіс SRD дає змогу відправити й отримати дані в одному циклі обміну. Цей спосіб обміну є найбільш поширеним у Profibus і дуже зручним під час роботи з пристроями введення-виведення, оскільки в одному циклі можна і відправити, і отримати дані.

Сервіс SND використовується, коли треба відправити дані одночасно групі ведених пристроїв (багатоабонентський режим) або всім веденим пристроям (широкомовний режим). При цьому ведені пристрої не відправляють свої повідомлення майстру.

Повідомлення в Profibus називають телеграмою. Телеграма може містити до 256 байтів, з них 244 байти даних, плюс 11 службових байтів (заголовок телеграми). Усі телеграми мають заголовки однакової довжини, за винятком телеграми з назвою Data_Exchange. Зауважимо, що 11 байтів службової інформації роблять Profibus дуже неефективним під час передання коротких повідомлень. Однак при великих обсягах даних такий формат телеграми є досить ефективним.

Поля телеграми на рисунку 10.3 містять:

SD – стартовий роздільник, який використовується для зазначення початку телеграми та її формату. Є чотири типи роздільників для телеграм запиту й відповіді та один тип для короткого повідомлення. Коротке повідомлення має поле SD, але не на початку телеграми;

LE – довжина переданих даних (DA + SA + FC + DSAP + SSAP + DU);

LEr – повторення поля LE з метою його резервування;

DA – адреса пристрою – одержувача телеграми;

SA – адреса відправника;

FC – код типу телеграми (запит, повідомлення, відповідь, діагностичні дані, тип пристрою – майстер або ведений, пріоритет, повідомлення);

DSAP – пристрій-одержувач використовує це поле щоб визначити, який тип сервісу потрібно виконати;

SSAP – COM-порт відправника;

DU – дані довжиною від 1 до 244 байтів;

FCS – контрольна сума телеграми (сума значень полів DA + SA + FC + DU, за модулем 255);

ED – ознака кінця.

SD	LE	LEr	SD	DA	SA	FC	DSAP	SSAP	DU(1...244 байт)	FCS	ED
----	----	-----	----	----	----	----	------	------	------------------	-----	----

Рисунок 10.3 – Характеристика пакета даних

З метою підвищення надійності в Profibus передбачено резервування.

1. Ведені пристрої містять два різних Profibus-інтерфейси, основний й резервний. Вони можуть бути або у одному пристрої, або у двох однакових пристроях (основному й резервному).

2. Пристрої забезпечуються двома незалежними стеками протоколів зі спеціальним розширенням для резервування.

3. Процес резервування стеків протоколів здійснюється шляхом запуску спеціального програмного об'єкта резервування RedCom.

У нормальному режимі комунікація виконується тільки через основний пристрій, який посилає діагностичну інформацію резервному пристрою. У разі коли в основному пристрої відбувається збій, резервний пристрій бере на себе його функції. Крім того, майстер контролює всі ведені пристрої й видає діагностичне повідомлення на верхній рівень АСУ ТП, як тільки у системі вийшов з ладу основний пристрій і не залишилося резервного, або коли вийшло з ладу резервне. Резервний пристрій може працювати на основній Profibus-лінії або на двох, якщо є резервна.

Підхід до резервування у Profibus має такі властивості:

– одна й та сама модифікація пристроїв використовується для реалізації різних варіантів резервування;

- ведучий, ведений пристрої і шина можуть бути резервованими незалежно один від одного;
- не потрібно особливого додаткового конфігурування резервного пристрою;
- можливий повний моніторинг обох ведених пристроїв.

Резервування забезпечує високий коефіцієнт готовності, невеликий час відновлення, відсутність утрат даних і нечутливість системи до відмов.

Сучасні модулі введення-виведення є інтелектуальними пристроями й виконують багато функцій, які раніше виконували тільки контролери. Однак щоб виконати ці функції, пристрої потребують складного налаштування при інсталяції системи, обслуговуванні й параметризації. Тому необхідно мати точний і повний опис відомостей про пристрої, таких як тип виконуваних функцій, кількість входів/виходів, діапазон змінення змінних, одиниці виміру, значення за замовчуванням, що ідентифікують параметри пристрою й т. д.

Profibus пропонує кілька методів й засобів для опису пристроїв, які забезпечують уніфікацію опису. З історичних причин у промисловій автоматизації використовується в основному формат GSD (General Station Data – загальні дані про пристрій). Опис пристроїв у цьому форматі створюється їх виробником і поставляється разом з пристроєм.

Характеристики пристрою описуються з допомогою мови опису електронних пристроїв Electronic Device Description Language (EDDL) і поставляються у вигляді текстового файлу EDD (Electronic Device Description – опис електронного пристрою). Інтерпретатор цього опису дуже добре апробовано для додатків середньої складності. Для великих програм Profibus пропонує інший програмний компонент – Device Type Manager (DTM).

Текстовий файл GSD містить як загальну, так і специфічну для конкретного пристрою інформацію. З допомогою ключових слів засіб конфігурації може прочитати ідентифікаційні записи, налаштувати параметри, типи даних, допустимі значення параметрів. Деякі з ключових слів є обов'язковими (за стандартом), наприклад ім'я виробника, інші – опціональними. GSD-файл складається з трьох секцій:

- загальні параметри – ім'я постачальника та ім'я пристрою, версія апаратури й програмного забезпечення, ідентифікаційний номер, підтримувані швидкості передавання;
- специфікація ведучого пристрою – указуються допустима кількість під'єднаних ведених пристроїв, параметри передавання й приймання повідомлень;
- специфікація веденого пристрою – указуються кількість і тип каналів уведення-виведення, перелік діагностичних повідомлень і список модулів при модульній конструкції пристрою.

GSD-файл завантажується в засіб конфігурації системи «Profibus Configurator» і використовується при її інсталяції.

Більш потужний засіб опису пристроїв – мова EDD, яка є частиною міжнародного стандарту IEC 61804-2 і дає змогу описувати пристрої середньої складності.

Лекція 11 УНІВЕРСАЛЬНИЙ АСИНХРОННИЙ ІНТЕРФЕЙС UART

11.1 Загальні відомості

UART (Universal Asynchronous Receiver/Transmitter) – універсальний асинхронний приймач, інтерфейс для зв'язку цифрових пристроїв, призначений для передавання даних у послідовній формі. Цей пристрій є дуже поширеним і досить затребуваним, має апаратну реалізацію в багатьох мікроконтролерах. Наприклад, мікроконтролери STM32 із сім'ї STM32F100xx залежно від варіанта виконання містять два або три USART.

USART (Universal Synchronous-Asynchronous Receiver/Transmitter) – універсальний синхронно-асинхронний приймач – аналогічний до UART-інтерфейсу, але додатково до можливостей UART підтримує режим синхронного передавання даних – з використанням додаткової лінії тактового сигналу. Утім, синхронне передавання використовується набагато рідше за асинхронне.

UART може використовуватися як для взаємодії компонентів усередині одного пристрою, так і для з'єднання пристроїв між собою. Для зовнішніх під'єднань сигнали з рівнями логіки ТТЛ або КМОП є малоприсадними через низьку завадостійкість. Поширеним стандартом фізичного рівня для UART, придатним для під'єднання зовнішніх пристроїв, є RS-232. Цьому стандарту, зокрема, відповідає послідовний порт (COM-порт) комп'ютера. Мікроконтролер з допомогою схеми перетворення рівнів може обмінюватися інформацією з COM-портом комп'ютера.

В UART передавання даних відбувається у послідовній формі, тобто по одному біту. Тому для передавання в одному напрямку потребується один провідник, для повнодуплексного двобічного зв'язку – два.

Вихід позначають TD або TX (transmitted data), вхід – RD або RX (received data). Для з'єднання двох пристроїв вихід одного під'єднують до входу другого й вхід першого – до виходу другого (рисунок 11.1).

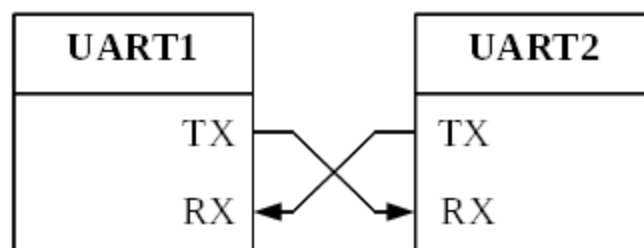


Рисунок 11.1 – Схема з'єднання двох пристроїв

Можливим є варіант використання UART для напівдуплексного двобічного зв'язку по одному дроту. У цьому випадку виводи TX і RX кожного пристрою з'єднують разом. Пристрій весь час, поки не передає даних, тримає вихід у відімкненому стані (переводить у Z-стан, або стан з високим опором; можна використовувати режим роботи з відкритим стоком – під час паузи в переданні на виході UART формується логічна одиниця, що рівносильно переходу в Z-стан). Пристрій може мати апаратну підтримку напівдуплексного обміну даними, тоді потрібно лише вибрати необхідний режим роботи. Якщо апаратної підтримки немає, то напівдуплексний режим легко реалізується програмно. Для цього потрібно від'єднати передавач, коли пристрій не передає даних, щоб звільнити лінію для здійснення передання іншими пристроями, і від'єднати приймач під час роботи свого передавача, щоб не приймати власне передання (або програмно відкидати дані, що передаються своїм передавачем).

До однопровідної лінії можна під'єднати кілька пристроїв, які будуть утворювати мережу для передання даних. Арбітраж у цій мережі має бути реалізовано програмно (рисунок 11.2).

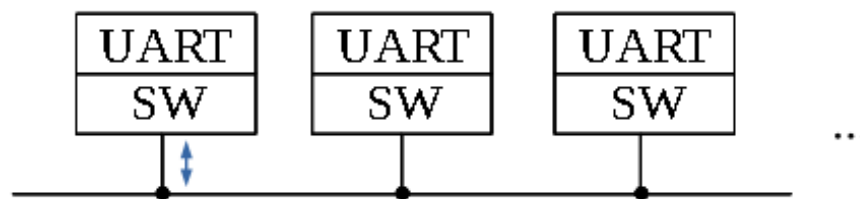


Рисунок 11.2 – Характеристика арбітражу

Як бачимо, з'єднувати пристрої з допомогою UART дуже просто. Для двонаправленого під'єднання потрібні тільки три провідники (з урахуванням загального проводу), а для одно- або двоспрямованого напівдуплексного – усього два.

11.2 Формат передання даних

Коли немає передання на виході UART, є рівень логічної одиниці. Дані передаються у вигляді послань (фреймів), кожне з яких складається зі стартового біта, бітів даних та одного або кількох стоп-бітів (рисунок 11.3). Тривалість усіх бітів є однаковою і пов'язана зі швидкістю передання співвідношення $T = 1/S$. Існує кілька стандартних швидкостей передання: 110, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600 бітів. Якщо всередині одного пристрою зв'язок можна здійснювати на довільній швидкості, то для зв'язку із зовнішніми пристроями слід дотримуватися стандартних величин.



Рисунок 11.3 – Характеристика фреймів

Посилання починається зі стартового біта, який завжди має значення логічного нуля. Після стартового біта передаються біти даних. Кількість бітів даних може становити від 5 до 9 залежно від налаштувань UART. Зазвичай передається вісім або дев'ять бітів даних (вісім бітів власне даних і один біт парності). Завершується посилання стоп-бітами, їх значення – завжди логічна одиниця, кількість зазвичай становить 1, 1.5 або 2. Під кількістю стоп-бітів розуміється тривалість відповідного ім'я одиничного імпульсу відносно тривалості бітів даних і старт-біта. Цим пояснюється можливість виражати кількість бітів дробовим числом. Відразу ж після стоп-бітів може починатися передання наступного посилання або може бути пауза довільної тривалості, під час якої на виході також формується рівень логічної одиниці.

Оскільки під час передання стоп-біта й поки лінія є вільною, на виході є середнє арифметичне значення, а старт-біт має значення логічного нуля, старт-біт дає змогу виявити момент початку передання даних, розділити два послідовні посилання й здійснити синхронізацію передавача й приймача.

Якщо передавач і приймач працюють на одній швидкості, їх налаштовано на роботу з однаковою кількістю бітів даних, стоп-бітів, однаково сконфігуровано відносно біта парності, то для обміну даними не потрібно передавати окремо тактовий сигнал – приймач його встановить самостійно.

Виявивши початок старт-біта, приймач очікує протягом половини тривалості передання біта, після чого починає зчитувати сигнал на вході з частотою, яка дорівнює швидкості передання даних. В ідеальному випадку момент кожного зчитування припадає на середину прийнятого біта. У реальності генератори тактових імпульсів передавача й приймача мають неузгодженість за частотою, унаслідок чого кожне нове зчитування все більше зміщується відносно середини чергового біта. Важливо, щоб за час передання одного посилання зсув не перевищував половини тривалості біта, а з урахуванням перехідних процесів зміщення не повинно перевищувати чверті тривалості біта, інакше замість зчитування одного біта буде зчитуватися сусідній (або вся лінія під час перехідного процесу) і посилання буде прийнято неправильно.

Для керування потоком даних UART використовується програмний або апаратний метод. Якщо застосовується програмний метод, то інформація про готовність пристрою приймати дані або необхідність зупинити передання передається по тих же каналах, що й дані. Приймальний пристрій програмно розділяє дані й керувальні сигнали відповідно до прийнятого протоколу.

В інтерфейсі UART передбачено можливість використання додаткових сигналів (CTS, RTS) (рисунок 11.4) для апаратного керування потоком даних. Апаратне керування може використовуватися деякими повільними пристроями або пристроями з простою схемною реалізацією, однак для нього необхідно дві додаткові лінії для під'єднання пристрою.

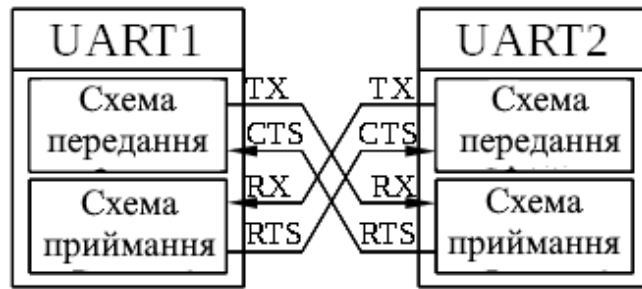


Рисунок 11.4 – Характеристика додаткових сигналів

Якщо в UART увімкнено контроль стану CTS, то передавач перед відправленням чергового фрейму перевіряє вхід CTS. Якщо на CTS рівень є низьким, то передавання відбувається, якщо високим, то не відбувається. Якщо сигнал CTS буде встановлено під час передавання посилення (фрейму), то поточне передавання все одно буде завершено перед припиненням роботи.

Приймач, зі свого боку, установлює на виході RTS значення логічного нуля, якщо він готовий приймати дані, і логічну одиницю, якщо ставить вимогу передавачу зупинити передавання.

11.3 USART у мікроконтролерах STM32

USART у мікроконтролерах STM32 надає гнучкі засоби для повнодуплексного обміну даними із зовнішніми пристроями в послідовному форматі з можливістю підтримки сигналів CTS/RTS, підтримує напівдуплексний обмін за однопровідною лінією, може працювати в широкому діапазоні швидкостей передавання. У мультибуферному режимі DMA набуває високої швидкості передавання даних, максимальне значення становить 3 Мбіт/с, також підтримується односпрямоване передавання в синхронному режимі, мультипроцесорний зв'язок. LIN (local interconnection network) – мережа для локального зв'язку, smartcard-протокол, інфрачервоний протокол відповідно до специфікації IrDA (infrared data association) SIR ENDEC.

Основні переваги USART:

- асинхронний повнодуплексний зв'язок;
- асинхронний однопровідний напівдуплексний зв'язок;
- налаштування методу оверсемплінгу (супердискретизації), який дає змогу вибирати між швидкістю передавання й допустимим відхиленням швидкості;
- використання передавачем і приймачем загальної програмованої швидкості передавання, яка може налаштовуватись у широких межах, максимальне значення становить 3 Мбіт/с при 8-кратному оверсемплінгу;
- програмована довжина слова – 8 або 9 бітів;
- кількість стоп-бітів – 1 або 2;

- у LIN-режимі підтримується відправлення й виявлення приймачем Break-посилання (генерується 13-бітова й детектується 10- або 11-бітова частота);
- є вихід тактового сигналу для синхронного передання;
- IrDA SIR-кодек для інфрачервоного зв'язку (підтримується тривалість біта 3/16 у нормальному режимі);
- підтримка інтерфейсом Smartcard асинхронного протоколу смарт-карт, як визначено в стандарті ISO 7816-3; використання 0,5 і 1,5 стоп-бітів в операціях зі смарт-картою;
- конфігурується мультибуферний зв'язок з використанням DMA (direct memory access);
- встановлення прапорців при виявленні подій під час обміну даними (приймальний буфер заповнено; буфер для передання є порожнім; передання завершено);
- контроль парності (можна налаштувати передавач на формування біта парності й приймач на контроль біта парності);
- встановлення чотирьох прапорців при виявленні помилок (помилка переповнення, виявлений шум у прийнятому сигналі, помилка фрейму, помилка парності);
- наявність 10 джерел переривання USART, пов'язаних з прапорцями регістру стану SR (змінення стану CTS; виявлення посилання LIN Break; регістр даних передавача є порожнім; передання завершено; регістр даних приймача заповнено; виявлення події «лінія є вільною» (Idle line); помилка переповнення; помилка фрейму; виявлення шуму; помилка парності);
- наявність мультипроцесорного зв'язку (перехід у тихий режим, якщо не відбулося зіставлення адреси);
- пробудження з тихого режиму при виявленні вільної лінії (Idle line) або адресної мітки;
- використання двох режимів пробудження приймача – за адресним бітом (9-й, старший біт) або при виявленні, що лінія є вільною.

Функціональну схему USART у мікроконтролерах STM32 зображено на рисунку 11.5, на якому наведено такі позначення:

SW: Single Wire – вивід для одноведучого під'єднання пристроїв. У звичайному двонаправленному повнодуплексному режимі потрібно щонайменше два виводи для під'єднання USART: RX (Receive Data In) й TX (Transmit Data Out).

RX: вхід для послідовних даних. Для вилучення даних використовується техніка оверсемплінгу (супердискретизації), коли зчитування входу здійснюється з частотою, яка в кілька разів перевищує швидкість передання даних (у цьому випадку – у 8 або 16 разів). Зіставлення зчитаної послідовності із зумовленими шаблонами дає змогу виокремити фронт сигналу, обчислити значення прийнятого біта й виявити шум при його наявності у прийнятому сигналі.

TX: вихід для передання даних у послідовній формі. Коли передавач вимкнено, вивід повертається у стан, заданий конфігурацією порту введення-виведення. Коли передавач увімкнено, але ніяких даних не передається, на виході

TX установлюється високий рівень (перерви у переданні даних або перерви у вільній лінії відповідає наявність логічної одиниці лінії). У однопровідному режимі або режимі смарт-карти цей вивід використовується як для передання, так і для приймання даних (тому на схемі його позначено як TX/SW).

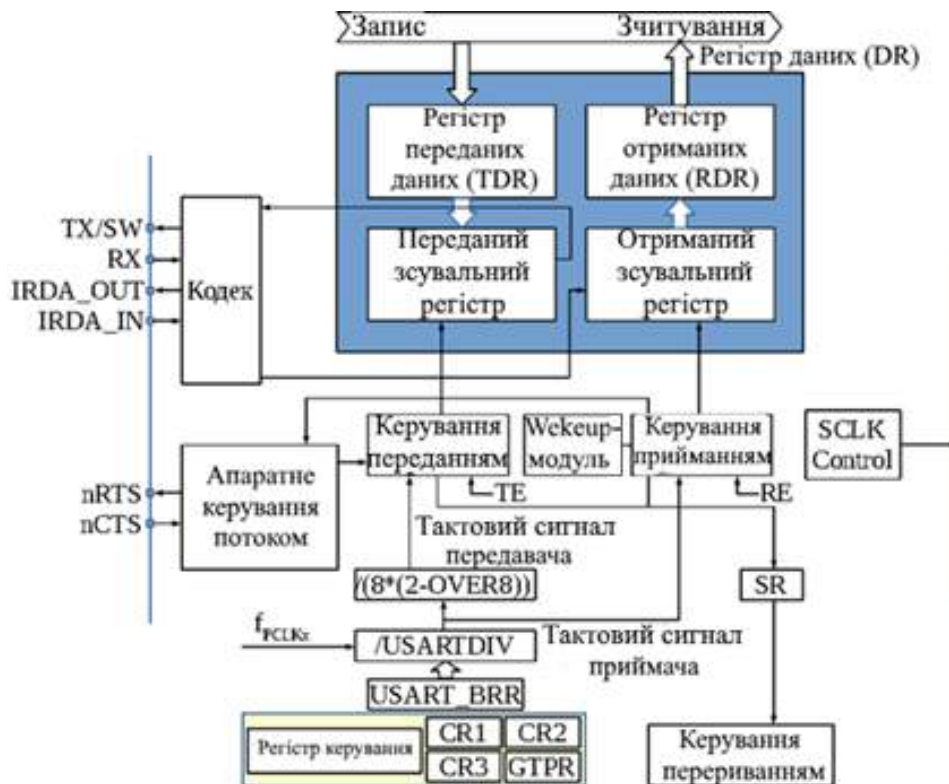
SCLK: вихід для тактового сигналу при синхронному переданні даних, відповідає майстер-режиму SPI. Тактові імпульси не формуються під час старт- і стоп-бітів. Наявність імпульсу під час передання останнього біта даних визначається програмно. Паралельно може відбуватися синхронне приймання даних на вході RX. Фаза й полярність імпульсів задаються програмно. У режимі смарт-карти вивід SCLK може формувати тактовий сигнал для неї.

IrDA_RDI: Receive Data Input – вхід для одержуваних даних в IrDA-режимі.

IrDA_TDO: Transmit Data Output – вихід для переданих даних в IrDA- режимі.

nCTS: Clear To Send – високий рівень на вході призупиняє передання даних після завершення поточного посилання.

nRTS: Request to send – низький рівень на цьому виході сигналізує про готовність USART приймати дані.



Регістр 11.5 – Функціональна схема USART

Програмне приймання й передання даних через інтерфейс USART здійснюється шляхом читання й запису регістра даних DR. Дані, що записуються в регістр (містяться в регістрі даних), передаються TDR, а після завершення поточного передання копіюються в передавальний зсувний регістр. Якщо в певний момент нічого не передається, то дані відразу розміщують у зсувний регістр. Зсувний регістр перетворює дані з паралельної форми подання на послідовну:

схема керування із заданою тактовою частотою зсуває вміст регістра на 1 біт управо. Зсунуті біти, починаючи з молодшого, подаються на вихід USART. Крім того, схема керування формує стартовий біт, біт парності (якщо використовується) і стоп-біти.

Після того як вміст TDR копіюється у зсувний регістр, у DR можна записувати черговий байт без ризику пошкодити попередній вміст.

Дані, що приймаються, потрапляють у приймальний зсувний регістр. Коли приймання фрейму завершується, дані з приймального зсувного регістра розміщуються у регістрі приймання даних (RDR), звідки їх може бути прочитано шляхом читання регістра даних DR. Такий пристрій регістра даних дає змогу одночасно передавати й отримувати дані.

11.4 Використання USART для обміну даними

USART дає змогу гнучкого налаштування конфігурації: можна в широкому діапазоні змінювати швидкість передання, кількість бітів у переданому одним фреймом слові, використовувати один з кількох можливих варіантів контролю парності або відмикати контроль, налаштовувати кількість стоп-бітів, можна використовувати або не використовувати лінії CT/RTS. Безумовно, безліч можливостей з налаштування надає USART велику гнучкість та універсальність, але, з іншого боку, створює певні труднощі під час обміну даними. Адже для того, щоб обмін даними був можливим, під'єднані пристрої мають мати однакові налаштування за всіма параметрами.

Після конфігурації USART можна розпочати обмін даними. Мікроконтролер має два варіанти роботи з USART: побайтове передання й приймання або з використанням DMA. У першому випадку для приймання й передання кожного байта процесор має виконати послідовність певних дій. Так, він може обробляти переривання, що виникають, коли звільняється регістр переданих даних або заповнюється регістр приймальних даних; установивши причину переривання, записати в регістр даних черговий байт для передання або зчитати з регістра прийнятий байт і обробити його. Якщо використовується режим роботи з DMA, то процесору необхідно задати розташування буферів для приймання й передання, їх розмір. USART і DMA далі будуть здійснювати передання й приймання самостійно, не витрачаючи обчислювальні ресурси процесора.

Використання DMA розвантажує процесор і дає змогу досягти дуже високої швидкості передання, але якщо не планується передавати великі обсяги даних на високій швидкості, то можна використовувати побайтове передання. Цей варіант є дещо простішим у програмуванні.

Лекція 12 МОДУЛІ АНАЛОГОВИХ ІНТЕРФЕЙСІВ

12.1 Модулі послідовного введення-виведення

Наявність у складі 8-розрядного мікроконтролера (МК) модуля контролера послідовного введення-виведення стала останнім часом звичайним явищем.

Завдання, які вирішуються засобами модуля контролера послідовного введення-виведення, можна поділити на три основні групи:

- зв'язок убудованої мікроконтролерної системи із системою керування верхнього рівня, наприклад з персональним комп'ютером; частіше для цієї мети використовуються інтерфейси RS-232C й RS-485;

- зв'язок із зовнішніми відносно МК периферійними інтерфейсами, а також з датчиками фізичних величин з послідовним виходом; для цього використовуються інтерфейси I2C, SPI, а також нестандартні протоколи обміну;

- інтерфейс зв'язку з локальною мережею у мультиконтролерних системах. У системах з не менш ніж п'ятьма МК звичайно використовуються мережі на основі інтерфейсів I2C, RS-232C і RS-485 з власними мережними протоколами високого рівня, у складніших системах найбільш популярним стає протокол CAN.

З огляду на організацію обміну інформацією наведені типи інтерфейсів послідовного зв'язку відрізняються режимом передання даних (синхронний або асинхронний), форматом кадру (кількістю бітів у посилянні під час передання байта корисної інформації) і часовими діаграмами сигналів на лініях (рівнями сигналів і положенням фронту під час перемикання).

Кількість ліній, по яких передаються дані в послідовному коді, зазвичай дорівнює двом (I2C, RS-232C, RS-485) або трьом (SPI, деякі нестандартні протоколи), що дає змогу спроектувати модулі контролерів послідовного обміну так, щоб з їх допомогою на апаратному рівні можна було реалізувати кілька типів послідовних інтерфейсів. При цьому режим передання (синхронний або асинхронний) і формат кадру підтримуються на рівні логічних сигналів, а реальні фізичні рівні сигналів для кожного інтерфейсу одержують з допомогою спеціальних ІС, які називають приймачами-передавачами, конверторами, трансиверами.

Серед різних типів убудованих контролерів послідовного обміну, які входять до складу тих або інших 8-розрядних МК, утворився стандарт «де-факто» – модуль UART (Universal Asynchronous Receiver and Transmitter) – універсальний асинхронний приймач-передавач. Проте більшість модулів UART окрім асинхронного режиму обміну можуть також реалізувати режим синхронного передання даних.

Не всі виробники МК використовують термін UART для позначення типу модуля контролера послідовного обміну. Так, у МК фірми Motorola модуль асинхронного приймання/передання, який підтримує ті ж режими асинхронного обміну, що й UART, прийнято називати SCI (Serial Communication Interface). Слід зазначити, що модуль типу SCI зазвичай реалізує тільки режим асинхронного обміну, тобто його функціональні можливості вже є порівнянними з модулями типу UART. Проте є й винятки: під тим же ім'ям SCI у МК MC68HC705B16 «ховається» модуль синхронно-асинхронного передання даних.

Модулі типу UART в асинхронному режимі роботи дають змогу реалізувати протокол обміну для інтерфейсів RS-232C, RS-422A, RS-485, у синхронному режимі – нестандартні синхронні протоколи обміну, у деяких моделях – SPI. У МК фірми Motorola традиційно передбачено два модулі послідовного обміну: модуль SCI з можливістю реалізації тільки протоколів асинхронного приймання/передання для інтерфейсів RS-232C, RS-422A, RS-485 і модуль контролера синхронного інтерфейсу і стандарті SPI.

Протоколи інтерфейсів локальних мереж на основі МК (I2C і CAN) відрізняє складніша логіка роботи. Тому контролери CAN інтерфейсу завжди виконуються у вигляді самостійного модуля. Інтерфейс I2C з можливістю роботи як у ведучому, так і веденому режимі також зазвичай підтримується спеціальним модулем (модуль послідовного порту в МК 89C52 фірми Philips). Однак якщо реалізується тільки ведений режим I2C, то в МК PIC16 фірми Microchip його успішно поєднано з SPI: налаштування одного й того ж модуля на один з протоколів здійснюється шляхом ініціалізації.

Останнім часом виникла велика кількість МК з убудованими модулями контролерів CAN і модулями універсального послідовного інтерфейсу периферійних пристроїв USB (Universal Serial Bus), кожний з яких має досить складні протоколи обміну.

12.2 Модулі аналогового введення-виведення

Необхідність приймання й формування аналогових сигналів потребує наявності в МК модулів аналогового введення-виведення.

Найпростішим пристроєм аналогового введення в МК є вбудований компаратор напруги. Компаратор порівнює вхідну аналогову напругу з опорним потенціалом VREF і встановлює на виході логічну одиницю, якщо вхідна напруга є більшою від опорної. Компаратори зручніше використовувати для контролю певного значення вхідної напруги, наприклад, у термостатах. У комбінації із зовнішнім генератором напруги, що лінійно змінюється, убудований компаратор дає змогу реалізувати на МК інтегровальний аналого-цифровий перетворювач (АЦП). Проте більших можливостей для роботи з аналоговими сигналами надає АЦП, убудований у МК. Частіше його реалізують у вигляді модуля багатоканального АЦП, призначеного для введення в МК аналогових сигналів з датчиків фізичних величин і перетворення цих сигналів на двійковий код. Структурну схему типового модуля АЦП зображено на рисунку 12.1.

Багатоканальний аналоговий комутатор призначено для під'єднання одного з джерел аналогових сигналів (PTx0...PTx7) до входу АЦП. Вибір джерела сигналу для перетворення здійснюється шляхом запису номера каналу комутатора у відповідні розряди регістра керування АЦП.

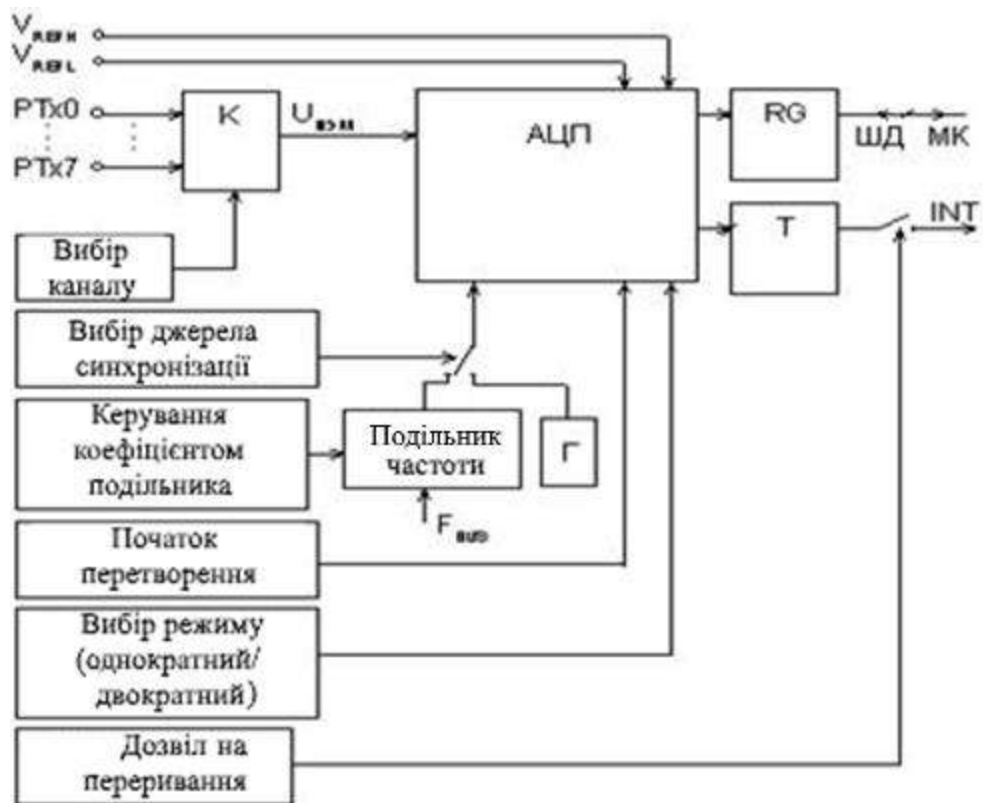


Рисунок 12.1 – Структурна схема типового модуля

Два виводи модуля АЦП використовуються для задання опорної напруги $U_{оп}$: V_{REFH} – верхня межа $U_{оп}$, V_{REFL} – нижня межа. Різниця потенціалу на входах V_{REFH} і V_{REFL} і становить $U_{оп}$. Роздільна здатність АЦП становить $U_{оп}/2^n$, де n – кількість двійкових розрядів у слові результату. Максимальне значення опорної напруги зазвичай дорівнює напрузі живлення МК. Якщо вимірювана напруга $U_{вим} > V_{REFH}$, то результат перетворення буде дорівнювати FF , код 00 відповідає напругам $U_{вим}$ і V_{REFL} . Для досягнення максимальної точності вимірювання слід вибрати максимально допустиме значення $U_{оп}$. У цьому випадку напруга зсуву нуля вхідного буфера й нелінійність передавальної характеристики АЦП привнесуть відносно малі погрішності.

Власне аналого-цифровий перетворювач виконано за методом послідовного наближення. Майже в усіх моделях 8-розрядних МК розрядність АЦП також становить 8 розрядів. Відповідно формат подання результатів вимірювання АЦП – однобайтовий. Виняток становлять лише модулі АЦП мікроконтролерів для керування перетворювачами частоти електроприводів, роздільна здатність яких дорівнює 10 розрядам. Два молодші розряди результату одержують з допомогою додаткової ємності подільника, не пов'язаної з регістром послідовного наближення.

Тривалість такту перетворення задає генератор синхронізації: один цикл дорівнює двом періодам частоти генератора t_{ADC} . Час перетворення для типових модулів АЦП мікроконтролерів становить від одиниць до десятків мікросекунд.

Джерелом синхронізації модуля АЦП може бути вбудований RC-генератор (Г) або імпульсна послідовність тактувальних міжмодульних магістралей МК. У першому випадку частота синхронізації АЦП обов'язково виявиться оптимальною, тобто такою, що рекомендується в технічному описі, у другому – її вибрано з інших міркувань: може виявитися, що fBUS не відповідає модулю АЦП. На цей випадок у складі деяких модулів передбачено програмований подільник частоти fBUS.

Момент завершення кожного циклу перетворення завершується установленням тригера готовності даних. Якщо переривання від модуля АЦП дозволено, то генерується запит на переривання. Зазвичай читання регістра результату скидає тригер готовності.

Більшість модулів АЦП має тільки режим програмного запуску: установлення одного з бітів регістра режиму запускає чергове вимірювання. Самі універсальні модулі АЦП мають також режим автоматичного запуску, при якому після завершення одного циклу перетворення негайно починається наступний. Проте дані вимірювання кожного циклу має бути зчитано програмним способом.

Цифро-аналогові перетворювачі в складі МК є великою рідкістю. Функція цифро-аналогового перетворювача реалізується засобами модуля програмованого таймера в режимі ШІМ. На одному з виводів МК формується високочастотна імпульсна послідовність з регульованою тривалістю імпульсу. Одержаний сигнал згладжується фільтром нижніх частот на операційному підсилювачі. Роздільна здатність такого ЦАП визначається дискретністю регулювання коефіцієнта заповнення в режимі ШІМ.

12.3 Інтерфейс «струмова петля»

Інтерфейс призначено для передання інформації між пристроями з радіально-послідовним зв'язком (ІРПС) (таблиця 12.1). Пристрої під'єднуються радіально з допомогою кабелю. Інтерфейс забезпечує асинхронне передання постійним струмом (струмова петля) по чотирьохпроводному двосторонньому колу. Формат переданої інформації (у бітах): старт-біт – 1, передані дані – 5, 7 або 8, парність – 1 або її немає, стоп – 1 або 2.

Таблиця 12.1 – Характеристика інтерфейсу «струмова петля»

Варіанти ІРПС	Стан	Струм, мА
40-міліамперна струмова петля	Логічна одиниця або логічний нуль	30...50/5...10
20-міліамперна струмова петля	Логічна одиниця або логічний нуль	15...25/0...3

Сполучені кінцеві пристрої мають гальванічний поділ, що здійснюється з боку ланцюга взаємозв'язку і не живиться струмом. Номінальне значення ізоляційної напруги гальванічного поділу – 500 В.

Струмова петля (англ. current loop) — послідовний асинхронний інтерфейс, який забезпечує передання інформації у старт-стоповому режимі (з буфером або без буфера) постійним струмом.

В електричній сигналізації аналогова струмова петля використовується, коли пристрій має контролюватися або керуватися дистанційно парою провідників. У будь-який момент у петлі є якийсь рівень струму, тому на неї не так впливають електромагнітні завади.

Аналогові струмові петлі легше зрозуміти й налагодити, ніж більш складні цифрові польові шини. У більшості ситуацій це потребує тільки кишенькового цифрового мультиметра. Використання польових шин і вирішення пов'язаних з цим проблем зазвичай потребує набагато більш високого рівня освіти й розуміння.

Над аналоговою струмовою петлею можна передавати цифрову інформацію. Такий спосіб передання даних описано в HART-протоколі. Конкурентними протоколами, що можуть у майбутньому витиснути HART, є різні цифрові польові шини, такі як Fieldbus Foundation або PROFIBUS.

12.4 HART-протокол

HART-протокол (Highway Addressable Remote Transducer – адресований дистанційний магістральний перетворювач) розроблено фірмою Fisher Rosemount Inc.

Швидкість передання даних за стандартом HART-протоколу становить до 1200 біт/с. Обмін реалізується за принципом Master/Slave. У мережі може бути до двох Master-вузлів, при цьому другий Master зазвичай призначено для зв'язку з будь-якою системою контролю або відображення даних.

Стандартну топологію організовано за принципом «точка-точка» або «зірка». Для передання даних по мережі використовуються два режими:

- за схемою «запит-відповідь» режим реалізує асинхронний обмін даними з циклом 500 мс;

- усі пасивні вузли безперервно передають свої дані на Master-вузол з поновленням даних у Master-вузлі за 250...300 мс.

Можна побудувати топологію типу «шина» (до 15 вузлів), коли кілька вузлів під'єднано до однієї пари проводів. Живлення здійснюється по шині.

Весь набір команд, реалізованих у HART-протоколі, умовно можна поділити на три групи:

- *універсальні команди* – команди загального призначення, які використовуються на рівні операторських станцій: код виробника пристрою в мережі, модель, серійний номер, стислий опис пристрою, діапазони обмежень, набір робочих змінних;

- *команди для груп пристроїв* фіксація значення струму на вихідному каналі, скидання і т. д;

- *команди, що залежать від пристрою* – старт/стоп, специфічні функції калібрування і т. д.

За одне посилання один вузол іншому може передати до чотирьох технологічних змінних, а кожен HART-пристрій може мати до 256 змінних, що описують його стан.

Структура інформаційного кадру має такий вигляд: один стартовий біт, вісім бітів даних, один біт контролю непарності, один стоповий біт. Метод контролю коректності даних, що передаються, базується на отриманні підтвердження.

У HART-протоколі використовується стандарт BELL202 кодування сигналу методом частотного зсуву (FSK) для посилання цифрової інформації через телефонні мережі, при цьому цифровий сигнал накладається на аналоговий вимірювальний сигнал 4...20 мА. Для подання двійкових «1» і «0» використовуються дві різні частоти: 1200 і 2200 Гц відповідно (рисунок 12.2).

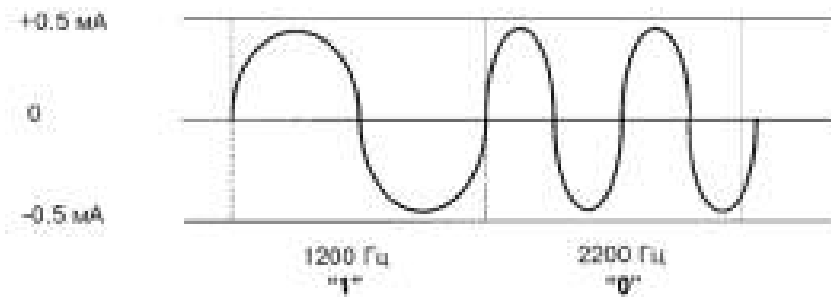


Рисунок 12.2 – Характеристика сигналу

Метод формування фізичних сигналів і середовища передання даних HART-протоколу відповідає фізичному рівню OSI-моделі протоколів.

Середнє значення синусоїдального сигналу за період дорівнює нулю. Тому, незважаючи на проходження цифрових даних, до сигналу 4...20 мА ніяка компонента постійного струму не додається. Отже, наявні аналогові прилади продовжують працювати у звичайному режимі, крім того, застосовується низькочастотна фільтрація, яка ефективно відкидає комунікаційний сигнал, наприклад, односмуговий низькочастотний фільтр 10 Гц зменшує комунікаційний сигнал до амплітуди коливань приблизно $\pm 0,01$ % від аналогового сигналу.

Оскільки виконавчі числа передаються на швидкості обміну даними 1200 біт, число «1» подано одним циклом 1200 Гц, а число «0» – приблизно двома циклами 2200 Гц.

У протоколі HART визначено, що головні пристрої (ведуча система керування або ручний комунікатор) передають сигнал у вигляді напруги, тоді як підпорядковані (первинні) пристрої передають струмовий сигнал. Токовий сигнал перетвориться на відповідну напругу з допомогою опору навантаження контуру, яке має бути в межах від 230 до 1100 Ом. Отже, усі пристрої мають використовувати такі приймачі, які можуть приймати напругу.

Рівні комунікаційного сигналу HART-протоколу: сигнал, переданий головним пристроєм, – min 400 мВ, max 600 мВ; сигнал, переданий підпорядкованим пристроєм – min 0,8 мА max 1,2 мА; чутливість приймача – від 120 мВ до 2,0 В; поріг приймача – від 0 до 80 мВ. Усі значення наведено між піками сигналу (подвійна амплітуда).

Чутливість приймача є такою, що відбувається деяке згасання сигналу через кабель або вплив інших складових. Характеристика межі приймача зменшує ймовірність завад зовнішніх сигналів і запобігає перетину з іншими HART-сигналами.

Під час роботи не в моноканалі (один підпорядкований пристрій) первинна змінна може бути зчитаною або як аналогова величина, або за цифровим зв'язком. У режимі моноканалу зчитувати первинну змінну можна тільки з допомогою цифрової комунікації, оскільки аналоговий сигнал більше не є доступним.

Для зв'язку з HART-пристроями призначено такі вироби:

1) HART-комунікатор – портативний пристрій, призначений для зчитування інформації, віддаленого налаштування та конфігурації інтелектуальних польових приладів;

2) HART-модем – призначено для зв'язку персонального комп'ютера або системних засобів АСУ ТП з інтелектуальними датчиками, він забезпечує високу надійність передання даних, використовується з програмним забезпеченням (AMS, VisualInstrument, H-Master) для налаштування інтелектуальних пристроїв;

3) HART-мультиплексор – забезпечує зв'язок персонального комп'ютера або засобів АСУ ТП з вісьма або шістнадцятьма інтелектуальними датчиками й будь-якими іншими пристроями, що підтримують HART-протокол. Мультиплексор забезпечує перетворення інформаційного сигналу HART на цифровий сигнал інтерфейсу RS-485 або RS-232, при цьому аналоговий сигнал 4...20 мА струмової петлі може використовуватися системою реєстрації й керування.

Лекція 13 **ІНТЕРФЕЙС МІКРОПРОЦЕСОРА**

13.1 Класифікація мікропроцесорних комплексів

Мікропроцесори (МП) і мікропроцесорні комплекти класифікують за такими ознаками: за призначенням; кількістю великих інтегральних схем (ВІС); способом керування; типом архітектури; типом системи команд. За призначенням мікропроцесори поділяють на універсальні й спеціалізовані.

Універсальні мікропроцесори є мікропроцесорами загального призначення, які вирішують широкий клас завдань обчислення, оброблення й керування.

Спеціалізовані мікропроцесори призначено для виконання завдань лише певного класу. Вирішення таких завдань є найбільш швидкодієним та ефективним. До спеціалізованих мікропроцесорів належать: сигнальні, медійні й мультимедійні та трансп'ютери.

Сигнальні процесори призначено для вирішення завдань цифрового оброблення сигналів у реальному масштабі часу, наприклад для фільтрації сигналів, обчислення згортки, обчислення кореляційної функції, підсилення, обмеження й трансформації сигналу, прямого й зворотного перетворення Фур'є. До сигнальних процесорів належать процесори TMS320C80 – (Texas Instruments), ADSP2106x – (Analog Devices), DSP560xx DSP9600x – (Motorola).

Медійні й мультимедійні процесори призначено для оброблення аудіосигналів, графічної інформації, відеозображень, а також для вирішення деяких завдань у мультимедіакомп'ютерах, іграшкових приставках, побутовій техніці. До медійних і мультимедійних процесорів належать процесори Mediaprocesor – (MicroUnity), Trimedia – (Philips), Mpract Media Engine – (Cromatic Reserch,) NV1 – (Nvidia), MediaGX – (Cyrix).

Трансп'ютери призначено для масово-паралельних обчислень і роботи в мультипроцесорних системах. Для них характерною є наявність внутрішньої пам'яті й убудованого міжпроцесорного інтерфейсу, тобто каналів зв'язку з іншими ВІС МП. До трансп'ютерів належать процесори фірми Inmos – T-2, T-4, T-8, T9000.

За кількістю ВІС у МПК розрізняють багатокристалні (МПК) та однокристалні (ОМК) мікроконтролери. До багатокристалних комплектів належать: МПК з однокристалними й секційними МП.

Однокристалний мікропроцесор є конструктивно завершеним виробом у вигляді однієї ВІС. Інша назва однокристалних мікропроцесорів – мікропроцесори з фіксованою розрядністю даних. До цього типу належать процесори Intel – (Pentium (P5, P6, P7)), AMD – (K5, K6), Cyrix – (6x86, M1, M2), Digital Equipment – (Alpha 21064, 21164A), Silicon Graphics – (MIPS R10000), Motorola – (Power PC 603, 604, 620), Hewlett Packard – (PA-8000, Sun Microsystems – Ultra SPARC II).

У **секційних** мікропроцесорах у одній ВІС реалізується лише деяка функціональна частина (секція) процесора. Інша назва секційних мікропроцесорів – розрядно-модульні, або мікропроцесори зі збільшенням розрядності. Секційність ВІС МП зумовлює значну гнучкість мікропроцесорних систем, можливість збільшення розрядності даних, створення специфічних технологічних команд із набору мікрокоманд. До секційних належать мікропроцесори серій K589, K1804.

Однокристалний мікроконтролер являє собою пристрій, який виконано конструктивно в одному корпусі ВІС і містить основні складові частини МПК. До однокристалних мікроконтролерів належать ОКМ фірм Intel – (MCS-196/296), MicroChip – (PIC17C4x PIC17C75x), Mitsubishi Electric – (M3820), Motorola – (MC33035, MC33039).

За способом керування розрізняють МП зі схемним і мікропрограмним керуванням. Мікропроцесори зі схемним керуванням мають фіксований набір

команд, розроблений фірмою-виробником, який споживач не може змінювати. У мікропроцесорах з мікропрограмним керуванням система команд розробляється при проектуванні конкретного МПК на базі набору найпростіших мікрокоманд з урахуванням класу завдань, для вирішення яких призначено МПК.

За типом архітектури, або принципом побудови, розрізняють МП з фоннейманівською і гарвардською архітектурою. За типом системи команд розрізняють CISC (Complete Instruction Set Command) – процесори з повним набором команд і RISC (Reduced Instruction Set Command) – процесори зі зменшеним набором команд. Слід зазначити, що багато мікропроцесорних комплектів підпадають під різні класифікаційні ознаки, оскільки можуть вирішувати завдання різних класів. Так, існують універсальні мікропроцесори з мультимедійним розширенням наборів команд, наприклад Pentium MMX, Pentium П, Cyrix 6x86MX, AMD K6, Ultra SPARC. У CISC-процесорах Pentium PRO реалізовано ядро з RISC-архітектурою.

13.2 Організація шин мікропроцесорних комплексів

Шина – це інформаційний канал, який об'єднує всі функціональні блоки мікропроцесорної системи й забезпечує обмін даними у вигляді двійкових чисел. Конструктивно шина складається з n провідників та одного спільного провідника (землі). Дані по шині передаються у вигляді слів, що є групою бітів. У паралельній шині n бітів передаються по окремих лініях одночасно, у послідовній шині – по єдиній лінії послідовно в часі. Паралельні шини виконуються у вигляді плоского кабелю, а послідовні – у вигляді коаксіального або волоконно-оптичного кабелю. Коаксіальний кабель використовується при передаванні даних на відстань до 100 метрів, причому треба узгоджувати передавальні й приймальні каскади з хвильовим опором лінії. Волоконно-оптичний кабель використовується для передавання даних на великі відстані.

Усі основні блоки мікропроцесорного комплексу під'єднуються до єдиної паралельної шини, яка має назву системної шини SB (System Bus). Системна шина містить три шини: адреси, даних і керування. Шина адреси AB (Address Bus) є односпрямованою, її призначено для передавання адреси комірки пам'яті або пристрою введення/виведення. Напрямок передавання по шині адреси – від мікропроцесора до зовнішніх пристроїв.

Шина даних DB (Data Bus) є двоспрямованою, її призначено для передавання даних між блоками мікропроцесорної системи. Інформація по одних і тих самих лініях DB може передаватися у двох напрямках – як до мікропроцесора, так й від нього.

Шину керування CB (Control Bus) призначено для передавання керувальних сигналів. Хоча напрямок керувальних сигналів може бути різним, шина керування не є двоспрямованою, адже для сигналів різного напрямку використовуються окремі лінії.

Узагальнену структурну схему мікропроцесорної системи зображено на рисунку 13.1.



Рисунок 13.1 – Структурна схема мікропроцесорної системи

До складу МПС входять: центральний процесор (ЦП), постійний запам'ятовувальний пристрій (ПЗП), оперативний запам'ятовувальний пристрій (ОЗП), система переривань, таймер, пристрої введення/виведення (ПВВ). Пристрої введення/виведення під'єднано до системної шини через інтерфейси введення/виведення. ПЗП та ОЗП складають систему пам'яті, яку призначено для збереження інформації у вигляді двійкових чисел. ПЗП призначено для збереження програм керування, таблиць, констант, ОЗП – для збереження проміжних результатів обчислень. Пам'ять організовано у вигляді масиву комірок, кожна з яких має свою адресу й містить байт або слово. Байтом називають групу з восьми бітів, а слово може мати будь-яку довжину в бітах. Найбільш часто під словом розуміють двійкове число довжиною два байти. Для звернення до комірки пам'яті необхідно видати її адресу на шину адреси.

Модуль центрального процесора (ЦП) обробляє дані й керує всіма іншими модулями системи. ЦП, крім ВІС мікропроцесора, містить схеми синхронізації та інтерфейсу із системною шиною. Він вибирає коди команд з пам'яті, дешифрує їх й виконує. Протягом часу виконання команди, який має назву командного циклу, ЦП виконує такі дії:

- 1) виставляє адресу команди на шину адреси АВ;
- 2) отримує код команди із пам'яті й дешифрує його;
- 3) обчислює адреси операнда й зчитує дані;
- 4) виконує операцію, визначену командою;
- 5) сприймає зовнішні керувальні сигнали, наприклад запити переривань;
- 6) генерує сигнали стану й керування, необхідні для роботи пам'яті й пристроїв введення/виведення.

Поняття архітектури мікропроцесора містить його складові частини, а також зв'язки та взаємодію між ними. Опис архітектури містить: опис структурної схеми самого мікропроцесора, програмну модель мікропроцесора (опис функцій регістрів), інформацію про організацію

пам'яті (місткість пам'яті та способи її адресації), опис організації процедур уведення/виведення. Існують два основних типи архітектури – фоннейманівська та гарвардська.

Фоннейманівську архітектуру (рисунок 13.2, *а*) запропонував 1945 р. американський математик Дж. фон Нейман. Особливістю цієї архітектури є те, що програма й дані знаходяться у спільній пам'яті, доступ до якої здійснюється по одній шині даних і команд.

Гарвардську архітектуру було вперше реалізовано 1944 р. в релейній обчислювальній машині Гарвардського університету (США). Особливістю цієї архітектури є те, що пам'ять даних і пам'ять програм розділено вони мають окремі шини даних і команд (рисунок 13.2, *б*), що дає змогу підвищити швидкодію мікропроцесорної системи.

Структурні схеми обох архітектур містять: процесор, пам'ять, інтерфейси введення/виведення (ІВВ) і пристрої введення/виведення (ПВВ). Пам'ять та ІВУ для різних типів МП можуть бути як внутрішніми, тобто розміщуватися на тому ж кристалі, що й процесорний елемент (ПЕ), так і зовнішніми. Процесорний елемент містить регістри, арифметико-логічний пристрій (АЛП) і пристрій керування та виконує функції оброблення даних і керування процесами обміну інформацією. Пам'ять забезпечує зберігання кодів команд програми й даних. Інтерфейси призначено для зв'язку з пристроями введення/виведення, наприклад з клавіатурою, дисплеєм, друкувальними пристроями, датчиками інформації. Усі елементи структурної схеми під'єднано з допомогою шин.

Більш докладну структурну схему з процесором фоннейманівської архітектури зображено на рисунку 13.3. Схема процесора містить пристрій керування, арифметико-логічний пристрій (АЛП) і такі регістри: акумулятор, лічильник команд, вказівник стеку, регістр адреси, регістри даних, регістр команд, регістр стану.

Пристрій керування відповідно до кодів команд і зовнішніх керувальних і синхросигналів виробляє керувальні сигнали для усіх блоків структурної схеми МП, а також керує обміном інформацією між МП, пам'яттю й пристроями введення/виведення. Пристрій керування реалізує такі функції: початкового встановлення МП, синхронізації, переривань, узгодження швидкодії модулів мікропроцесорної системи.

Функція початкового встановлення МП. Зовнішній сигнал початкового встановлення процесора RESET формується при ввімкненні джерела живлення МП або натисканні кнопки «RESET». При виникненні цього сигналу пристрій керування забезпечує завантаження нульового значення в програмний лічильник, що ініціює вибірку з пам'яті байта команди з нульовою адресою. Наприкінці вибірки вміст лічильника команд збільшується на одиницю і вибирається байт команд з наступною адресою.

Таким чином, виконується вся програма, записана в пам'яті програм.

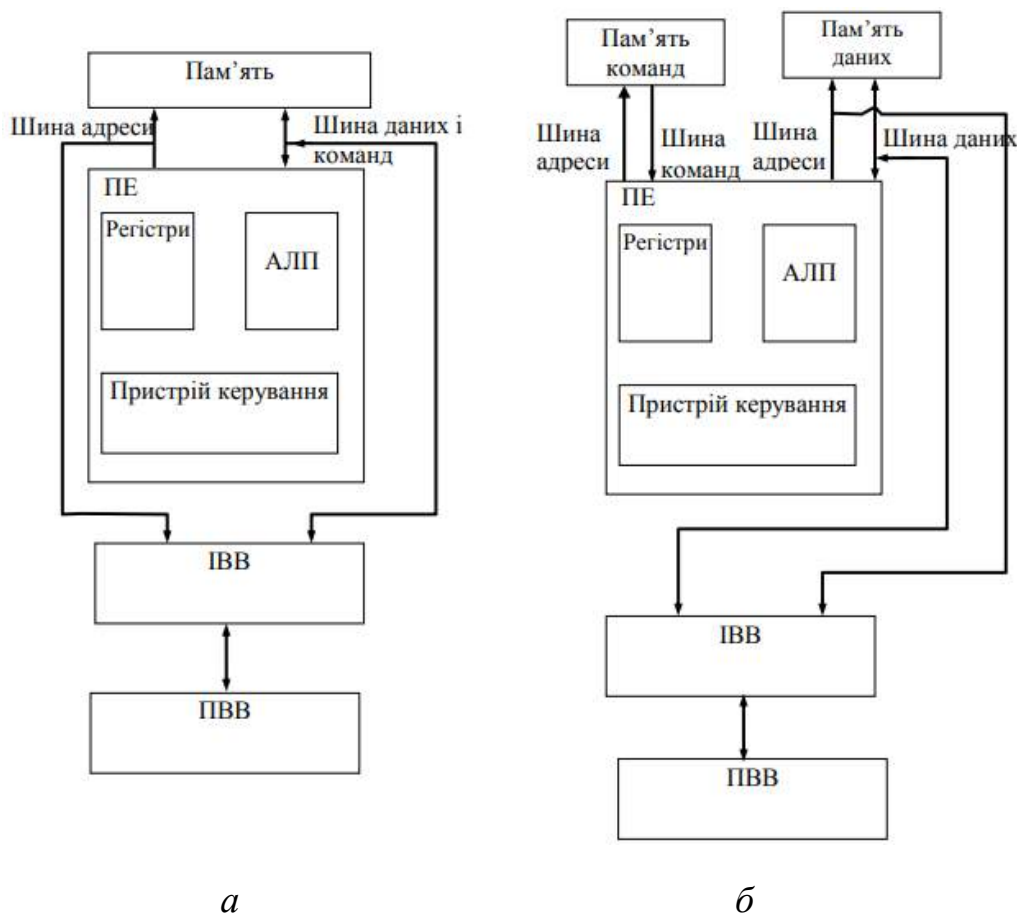


Рисунок 13.2 – Структура фоннейманівської (а) і гарвардської (б) архітектур

Функція синхронізації. Згідно із зовнішніми керувальними й синхросигналами пристрій керування синхронізує роботу всіх блоків МП.

Функція переривань. При надходженні сигналу переривання пристрій керування ініціює роботу підпрограми оброблення відповідного переривання. Необхідність реалізації функцій переривань виникає в тих випадках, коли при виконанні основної програми треба перевести МП на вирішення іншого завдання, наприклад оброблення аварійної ситуації або роботи з пристроями введення/виведення.

Функція узгодження швидкодії модулів мікропроцесорної системи. При обслуговуванні пам'яті й ПВВ, які мають значно меншу швидкодію, ніж МП, узгодження швидкодії вирішується шляхом генерації тактів очікування МП. При обслуговуванні пристроїв, які мають більшу швидкодію, ніж МП, використовується режим безпосереднього доступу до пам'яті.

Арифметико-логічний пристрій являє собою комбінаційну схему на основі суматора, який сигналами з виходів пристрою керування налагоджується на виконання певної арифметичної або логічної операції: додавання, віднімання, логічна одиниця, логічне АБО, логічне НІ, виключене АБО, зсуву, порівняння, десяткової корекції. Таким чином, АЛП виконує арифметичні або логічні операції над операндами, які пересилаються із пам'яті і/або регістру мікропроцесора. Операнд – це об'єкт у вигляді значення даних, вмісту регістрів або комірки пам'яті,

з яким оперує команда. Наприклад, у команді додавання операндами є доданки. Операнд може бути задано в команді у вигляді числа або знаходитися у регістрі або комірці пам'яті. Отриманий після виконання команди в АЛП результат пересилається в регістр або комірку пам'яті.

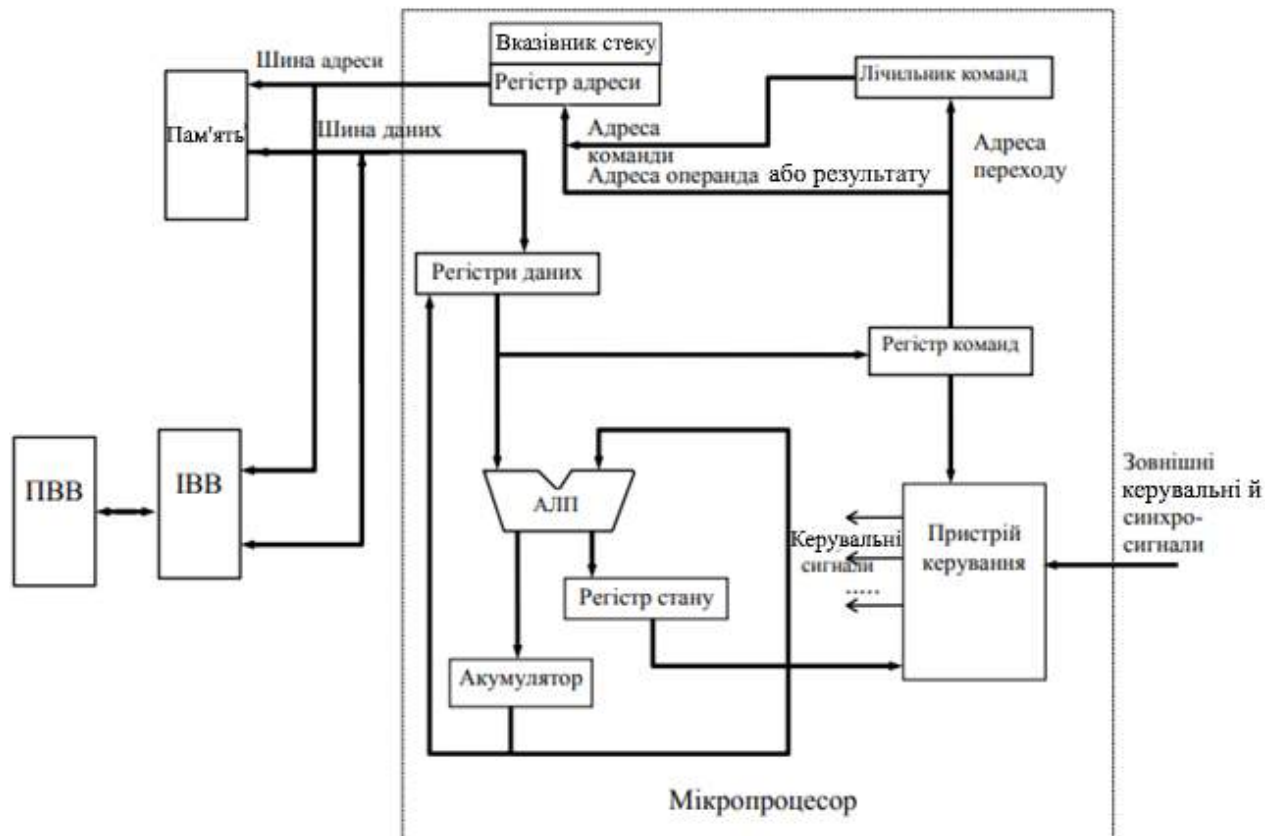


Рисунок 13.3 – Структурна схема з процесором фоннейманівської архітектури

Регістри призначено для зберігання n -розрядного двійкового числа і являють собою n тригерів зі схемами керування читанням/записом і вибірки. Регістри утворюють внутрішню пам'ять МП і використовуються для зберігання проміжних результатів обчислень. Акумулятор – це регістр, у якому зберігається один з операндів. Після виконання команди в акумуляторі замість операнда розміщується результат операції. У 8-розрядних процесорах акумулятор бере участь в усіх операціях арифметико-логічного пристрою. У 16-розрядних МП більшість команд виконуються без участі акумулятора, але в деяких командах (уведення, виведення, множення, ділення) акумулятор діє так само, як і у 8-розрядних МП, тобто зберігає один з операндів, а після виконання команди – результат операції. Лічильник команд, або програмний лічильник, призначено для зберігання адреси комірки пам'яті, яка містить код наступної команди. Програму дій мікропроцесора записано в пам'яті у вигляді послідовності кодів команд. Для переходу до наступної команди вміст лічильника збільшується на одиницю в момент вибірки команди з пам'яті. Таким чином, наприкінці виконання команди в лічильнику команд зберігається адреса команди, яка має виконуватися наступною.

Вказівник стеку – це регістр, який зберігає адресу останньої зайнятої комірки стеку. Стеком, або стековою пам'яттю, називають область пам'яті, організовану за принципом «останній прийшов – перший вийшов». Регістр команд зберігає код команди протягом усього часу виконання команди. Регістр адреси й регістри даних призначено для зберігання адрес і даних, які використовуються при виконанні поточної команди в МП. Регістр стану, або регістр прапорців (ознак) призначено для зберігання інформації про результат операції в АЛП і складається з кількох тригерів, що набувають одиничних або нульових значень.

13.3 Цикли шини процесора

Протягом циклу шини МП виставляє адресу комірки пам'яті або ПВУ на шину адреси, формує керувальні сигнали читання/запису, а після цього зчитує або записує дані. Окрім циклів читання й запису пам'яті або ПВВ, існують цикли підтвердження переривання й захоплення шин. Цикл шини може ініціювати не тільки незалежний процесор i8086, але й арифметичний співпроцесор або співпроцесор введення/виведення. Розрізняють цикли шини в мінімальному й максимальному режимах. Часові діаграми циклів читання й запису в мінімальному режимі зображено на рисунку 13.4.

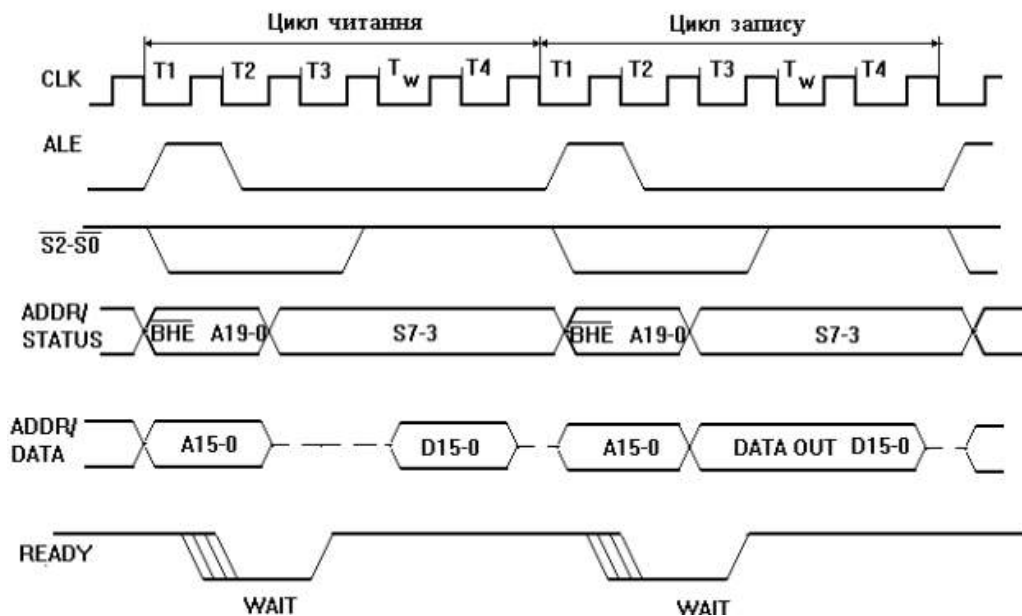


Рисунок 13.4 – Структура циклу шини процесора

Цикл шини складається щонайменше з чотирьох тактів. Такт визначається як проміжок часу між задніми фронтами двох сусідніх імпульсів CLK. Будь-який цикл шини може бути необмежено розтягнутим з допомогою сигналу готовності READY, при цьому процесор уводить необов'язкові такти очікування T_w . Цикли звернення до порту відрізняються від циклів пам'яті тим, що старші розряди шини адреси мають нульове значення (при непрямій адресації).

Процесор i8086 може обробляти до 256 типів переривань. Кожному перериванню відповідає свій вектор – подвійне слово, що містить адресу CS: IP-підпрограми, що викликається. Під вектори переривань у загальному просторі адрес пам'яті відводиться 1 Кбайт, починаючи з нульової адреси (рисунок 13.5).

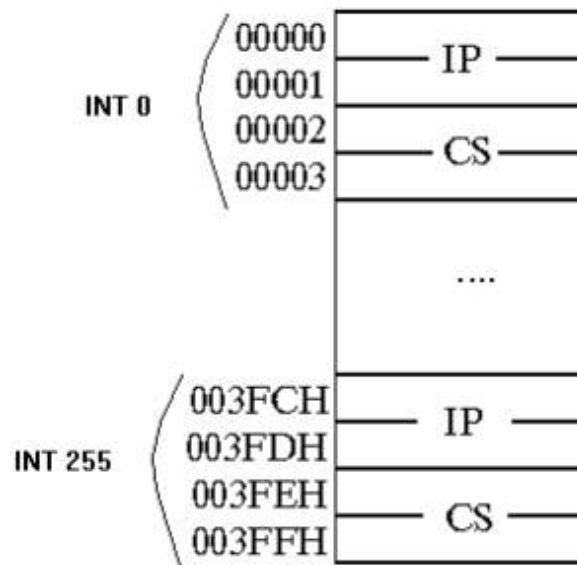


Рисунок 13.5 – Структура адреса пам'яті

Під час переходу на підпрограму оброблення переривань INT n (n – тип переривання) процесор переміщує в стек вміст регістрів IP, CS, регістру прапорців F і скидає прапорець дозволу переривання IF; обчислює адресу 4 x n і перше слово з цією адресою переміщує в IP, друге – у CS. Послідовність цих дій є еквівалентною командам: PUSHF, CALL FAR i_proc_4n, запам'ятовування в стеку прапорців, далекий виклик підпрограми оброблення, переривання.

Скидання прапорця переривання IF не дає змогу перервати виконання підпрограми оброблення переривання до її завершення або виконання команди дозволу STI. Останньою командою підпрограми оброблення переривання є команда IRET. За цією командою процесор витягає зі стеку адресу повернення (адресу команди, що є наступною за командою INT) і вміст регістру прапорців. Типи переривань показано на рисунку 13.6. Переривання поділяються на зовнішні апаратні й внутрішні. Запити IRQi зовнішніх апаратних переривань надходять на систему переривань або на вивід немаскованого переривання NMI мікропроцесора. Система переривання формує сигнал INTR маскованого переривання МП. Зазначимо, що масковане переривання відрізняється від немаскованого тим, що перше може бути заборонено програмним шляхом – командою скидання прапорця дозволу переривань IF. У цьому разі при надходженні запитів переривання вони будуть ігноруватися. Внутрішні переривання процесора поділяються на програмні й апаратні. Джерелами внутрішніх програмних переривань є помилка ділення (тип 0), покроковий режим (тип 1), команда INTO (тип 4).

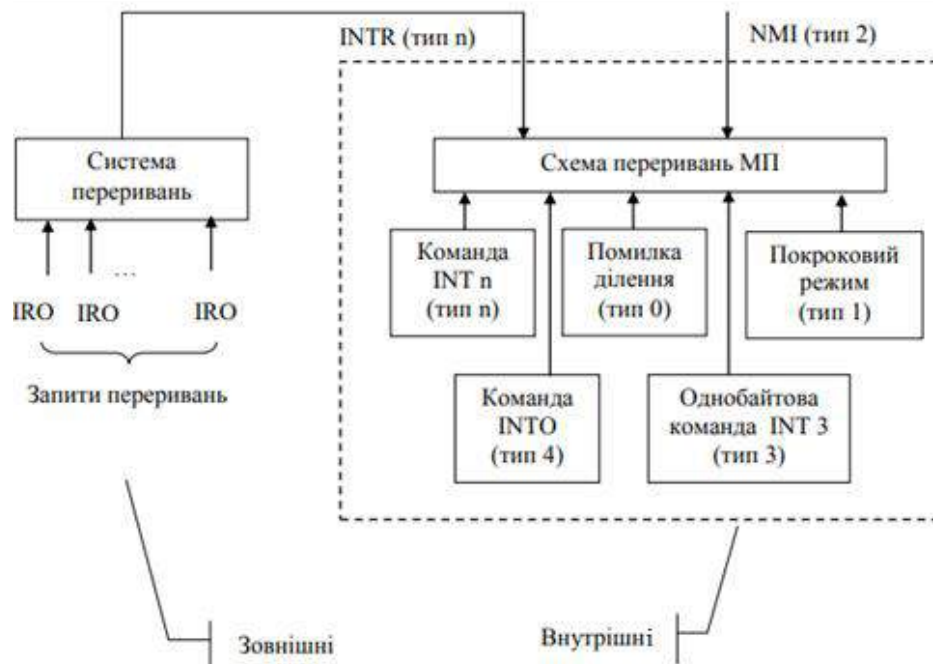


Рисунок 13.6 – Типи переривань

Внутрішні програмні переривання INT *n* і INT 3 виконуються з командами переривання й дають змогу викликати підпрограми оброблення переривань (наприклад, сервісні підпрограми BIOS і DOS) без застосування дальніх викликів. Переривання INT 3 є однобайтовою командою на відміну від INT *n* і зазвичай використовується для передання керування підпрограмою налагоджувачу. Виконання програмних переривань не залежить від прапорця дозволу переривань IF. Внутрішні апаратні переривання процесора виникають в таких особливих випадках: при діленні на 0 (тип 0), при встановленому прапорці трасування (тип 1). У цьому випадку переривання відбувається після виконання кожної команди, а після команди INTO (тип 4), якщо тільки встановлено прапорець переповнення OF.

Апаратні переривання виникають при активному рівні сигналів на контактах МП – NMI (немасковане переривання – тип 2) і INTR (масковані, типи 5...255). Масковані переривання виконуються при встановленому прапорці IF. При переході до підпрограми оброблення апаратного переривання процесор виробляє два цикли підтвердження переривання, що йдуть один з одним, у яких генерується сигнал INTA. За другим імпульсом INTA контролер переривань передає по шині даних номер вектора переривання *n*. Далі дії процесора є аналогічними до виконання програмного переривання. Оброблення поточного переривання може бути перервано немаскованим перериванням або іншим маскованим перериванням вищого пріоритету в тому разі, якщо підпрограма-обробник установить прапорець дозволу переривання IF. Немасковане переривання виконується незалежно від стану прапорця IF.

Лекція 14

ІНТЕРФЕЙСИ ПЕРИФЕРІЙНИХ ВУЗЛІВ КОМП'ЮТЕРНИХ СИСТЕМ

14.1 Концепція шини SCSI

Шина SCSI – це шина введення/виведення, а не системна шина й не інтерфейс приладового рівня. Інтерфейсні засоби типу шини SCSI є особливо ефективними для машин, які потребують під'єднання кількох дискових накопичувачів або інших периферійних пристроїв. Інтерфейс SCSI підвищує гнучкість та обчислювальну потужність системи, оскільки дає змогу під'єднати до однієї шини кілька різних ПП, які можуть безпосередньо взаємодіяти один з одним. Швидкість передання даних по шині безумовно не буде обмежувальним фактором, оскільки цей показник для шини SCSI сьогодні становить 40 Мбайт/с.

У шині SCSI передбачено можливість під'єднання до восьми пристроїв. На перший погляд це може здатися досить серйозним обмеженням, проте якщо врахувати, що кожен пристрій може складатися з восьми логічних блоків, а кожен логічний блок – з 256 логічних підблоків, то є очевидним, що можливостей розширення тут більш ніж достатньо.

Кожному з пристроїв шини SCSI має бути призначено індивідуальний ідентифікатор ID, значення якого зазвичай задається з допомогою комутаційних перемичок безпосередньо в пристрої. Ідентифікатор ID виконує дві функції: ідентифікує пристрій на шині й визначає його пріоритет в арбітражі за доступом до шини (чим більшим є номер пристрою, тим вищим буде його пріоритет).

Кожен з восьми комплексів пристроїв шини може виконувати функцію ініціатора (initiator), виконавця (target) або поєднувати їх. Ініціатор – це частина хосту (головного) адаптера SCSI, який призначено для під'єднання головного комп'ютера до шини SCSI. У типовій системі до одного ініціатора під'єднується один або кілька виконавців. Система підвищеної складності може містити більше від одного хост-адаптера SCSI (багато ініціаторів). У таких системах може установлюватися взаємодія не тільки будь-якого процесора з будь-яким периферійним пристроєм (ПП), але також хост-адаптерів один з одним, оскільки хост-адаптер сам є пристроєм шини SCSI і може бути як ініціатором, так і виконавцем. Однак два ПП (обидва виконавця) не можуть взаємодіяти один з одним, оскільки тільки пара ініціатор – виконавець може обмінюватися даними по шині у кожен конкретний момент часу.

Інтерфейс контролера SCSI і системної шини може бути як зовсім простим (будується з принципом програмного опитування каналу), так і більш складним (передбачається високошвидкісний обмін даними в режимі прямого доступу до пам'яті, ПДП). Такі контролери сприймають високорівневі команди й звільняють ЦП від необхідності оброблення й контролю сигналів шини SCSI.

Програмне забезпечення головного комп'ютера спрощується, оскільки йому не доводиться враховувати фізичні характеристики конкретного пристрою. В

інтерфейсі SCSI передбачається використання логічних, а не фізичних адрес для всіх блоків даних.

У протоколі шини SCSI передбачено вісім окремих фаз:

- Bus Free – вільна шина;
- Arbitration – арбітраж;
- Selection – вибірка;
- Reselection – зворотня вибірка;
- Command – команда;
- Data – дані;
- Status – стан;
- Message – повідомлення.

Останні чотири фази називають фазами передання інформації. Шина SCSI у кожен конкретний момент часу може знаходитися тільки в одній з цих восьми фаз.

Фаза «Вільна шина» означає, що не один пристрій у цей момент не працює з шиною SCSI в активному режимі й шина є вільною для звернення. Ця фаза зазвичай виникає після системного скидання або після скидання шини сигналом RST. Ознакою фази «Вільна шина» є відсутність сигналів зайнятості BSY та вибірки SEL.

Шина перемикається у фазу «Арбітраж», коли будь-який SCSI-пристрій хоче взяти на себе керування шиною, тобто стати ініціатором на шині. Це відбувається у випадках, коли ініціатор хоче вибрати виконавця або виконавець хоче зробити перевибірку й запитує його раніше від ініціатора. У фазу «Арбітраж» шина може перемикатися тільки з фази «Вільна шина». Після того як пристрій визначить, що шина є вільною, починається фаза «Арбітраж». Для цього формується сигнал BSY, на відповідну лінію даних видається ідентифікатор ID SCSI-пристрою (ID-біт). При цьому кожен з восьми можливих пристроїв шини SCSI може видавати свій ID-біт тільки на закріплену за ним лінію даних як ознаку своєї участі в арбітражі. Пристрій з максимальним значенням ідентифікатора ID виграє арбітраж і бере на себе керування шиною.

Фаза «Вибірка» дає змогу ініціатору вибрати виконавця, щоб ініціювати виконання ним відповідної функції, наприклад команди читання або запису. Згідно з протоколом специфікації SCSI-2 фаза «Вибірка» завжди настає після фази «Арбітраж». У специфікації SCSI-1 передбачається варіант системи з одним ініціатором, де необхідност арбітражу немає, і у фазу вибірки можна входити відразу ж після фази «Вільна шина». В обох випадках для вибору виконавця ініціатор видає його ID-біт на відповідну лінію даних шини SCSI і формує сигнал вибірки SEL.

Необов'язкова фаза перевибірки є можливою, коли виконавець хоче встановити зв'язок з тим ініціатором, який раніше послав йому команду. Ця фаза взагалі нагадує фазу «Вибірка», за винятком того, що разом із сигналом вибірки SEL переходить в активний стан лінія I/O, що дає змогу розрізняти ці дві фази.

Фази «Команда», «Дані», «Стан» і «Повідомлення» утворюють групу фаз передання інформації, оскільки всі вони використовуються для передання даних

або керувальної інформації по шині даних. Щоб їх розрізнити, використовуються сигнали C/D – керування, I/O – введення/виведення і MSG – повідомлення, що виробляються виконавцями, і тим самим керують усіма переходами з однієї фази в іншу. Для керування переданням даних між виконавцем та ініціатором у фазах передання інформації використовуються сигнали ліній REQ/ACK – запит/підтвердження (у версії SCSI-2 додатково застосовуються лінії REQV/ACKV).

Реальний обмін даними може здійснюватися синхронним та асинхронним способом. В обох випадках для виконання квитуювання використовуються сигнальні лінії ACK і REQ. Для виконавця режим синхронного передання є необов'язковим. Ініціатор може поставити вимогу, щоб виконавець здійснював синхронне передання, але якщо останній відкине цей запит, то буде використовуватися асинхронний режим.

14.2 Реалізація протоколу SCSI-шини

При використанні інтерфейсу SCSI на внутрішнє програмне забезпечення хост-адаптера або контролера ПП покладається також функція підтримки SCSI-операцій.

При цьому ступінь необхідної програмної підтримки залежить від рівня інтелектуальності використовуваної інтерфейсної системи контролера SCSI-шини. За деякими оцінками, при використанні SCSI-шини першого, другого й третього поколінь обсяг необхідного для реалізації SCSI-протоколу ВПО становить близько 400, 2500 і кілька сотень команд відповідно.

Контролери різних поколінь відрізняються кількістю й складністю покладених на них функцій прийняття рішення щодо ситуацій у SCSI-системі. Наприклад, контролер першого покоління NCR 5380 працює на рівні оброблення та формування логічних сигналів SCSI-шини. Функції аналізу ситуацій і прийняття рішень повністю покладаються на ВПО. Такий режим характеризується великим обсягом керувального коду, високою інтенсивністю переривань на рівні ВПО і, як наслідок, високими накладними витратами на час виконання операцій.

Подібна схема використовується в SCSI-підсистемах ПК Macintosh фірми Apple. Для програмування SCSI-підсистем фірмою розроблено спеціальне програмне забезпечення Mac's SCSI Manager, що містить 14 програм, які обслуговують різні фази протоколу SCSI-шини, серед них: читання/запис даних у різних режимах, участь в арбітражі, оброблення команд, статусу, повідомлень та ін.

Усі перелічені функції реалізуються шляхом безпосереднього програмування апаратних пристроїв SCSI-контролера. Оскільки в усіх комп'ютерах сім'ї Mac використовуються хост-адаптери SCSI-шини на базі контролера NCR 5380, ВПО для всіх моделей будується за одним принципом. Однак є нюанси, які є важливими для програмування на нижньому рівні. Наприклад, у різних моделях ПК контролер 5380 має різні адреси. Крім того, по-різному використовуються деякі режими роботи контролера 5380. Зокрема, є відмінність у реалізації операцій

читання/запису даних у синхронному режимі, що зумовлено різним ступенем апаратної підтримки процедури в різних моделях сім'ї Mac. Якщо в моделі Mac Plus синхронний режим потребує повного програмного контролю ліній читання/запису й даних, то в машинах Mac SE і Mac II реалізовано повну апаратну підтримку цього режиму. Особливістю моделі Mac IIx є використання спеціальної системи – SCSI-контролера, до складу якої входить контролер 5830.

Схожі функції виконує й програмний інтерфейс ASPI (Advanced SCSI programming interface), запропонований фірмою Adaptec для програмування SCSI-пристроїв у середовищі MS-DOS. Інтерфейс ASPI також забезпечує виконання нижнього рівня протоколу SCSI-шини й містить шість команд, що дають змогу визначити кількість хост-адаптерів у складі системи, тип периферійного SCSI-пристрою, операції введення/виведення на SCSI-шині, припинити виконання SCSI-операції, скинути пристрої на SCSI-операції, установити параметри хост-адаптера.

У контролерах другого покоління зазвичай в обов'язковому порядку апаратно підтримуються функції арбітражу, керування переданням даних по шині. У деяких моделях уводиться внутрішній набір команд, які, по-перше, спрощують процес програмування, а по-друге, маскують типові для контролерів першого покоління переривання при зміні фаз на SCSI-шині.

До третього покоління однокристальних SCSI-контролерів належать NCR 53C700 NCR53C710 фірми NCR, у яких упроваджено нову програмну концепцію фірми NCR – так званий Script-процесор. Обидва контролери містять потужний убудований процесор продуктивністю 2 млн операцій за секунду, що забезпечує автономне керування операціями на SCSI-шині.

Script-алгоритм містить програми керування такими операціями, як вибір/перевибір, від'єднання/повторне під'єднання, змінення фази SCSI-шини, передання інформації.

Реалізація розвинених керувальних алгоритмів потребує великої місткості пам'яті. Якщо цей алгоритм виконується засобами хост-системи (за типом BIOS), то він стає залежним від типу системної шини і, отже, утрачає свою інваріантність. Якщо ж він виконується самим контролером, то потрібна велика місткість внутрішньої пам'яті. Особливістю й незаперечною перевагою контролерів серії 53C7XX є їх здатність безпосередньо працювати з пам'яттю хост-системи, де й може бути розміщено Script-алгоритм.

Набір Script-команд містить три основних типи операцій.

1. Блокові пересилання, що виконують передання даних між SCSI-шиною й основною пам'яттю, при цьому забезпечується вільний доступ за адресами, а також рівнозначність керувальної інформації на SCSI-шині й користувачі інформації.

2. Команди введення/виведення виконують різні операції на SCSI-шині й безпосередньо адресують вузли SCSI-ядра контролера.

3. Команди керування дають змогу порівняти вміст внутрішніх регістрів із сигналами на SCSI-шині або з першим байтом переданої послідовності.

4. Керування може бути передано за іншою адресою. Таким чином, усі типи переходів (jump, call, return) у Script-програмі можуть базуватися на безпосередньому порівнянні фаз SCSI-шини, що зручно для прийняття рішень у реальному часі.

Потужні Script-команди істотно зменшують накладні витрати SCSI-шини. Наприклад, при використанні команд блокового пересилання під час роботи з розосередженими блоками даних за одну команду може бути передано сторінку даних для програми користувача.

14.3 Загальні відомості про шину I2C

Переваги шини I2C. Потрібно лише дві лінії – лінія даних (SDA) і лінія синхронізації (SCL). Кожен пристрій, під'єднаний до шини, може бути програмно адресованим з унікальною адресою. У кожен момент часу існує просте відношення ведучий/ведений: ведучі можуть працювати як ведучий-передавач і ведучий-приймач. Шина дає змогу мати кілька ведучих, надаючи засоби для визначення колізій та арбітраж для запобігання пошкодженню даних при ситуації, коли два або більше ведучих одночасно починають передання даних. У стандартному режимі забезпечується передання послідовних 8-бітових даних зі швидкістю до 100 кбіт/с і до 400 кбіт/с – у «швидкому» режимі. Убудований у мікросхеми фільтр заглушує сплески сигналів, забезпечуючи цілісність даних. Максимальна допустима кількість мікросхем, під'єднаних до однієї шини, обмежується максимальною місткістю шини 400 пФ.

До недоліків можна віднести недостатню швидкість передання даних. Дійсно, шина I2C є послідовною шиною і має застосовуватися тоді, коли система, що здійснює функції керування, не потребує високошвидкісного передання даних.

Хоча послідовні шини не мають пропускну здатності паралельних шин, вони потребують менше з'єднань і контактів мікросхем. Поняття шини складається не лише із призначення сполучних дротів, а також із усіх форматів і процедур для зв'язку всередині системи. Пристрої, що зв'язуються по шині, повинні мати деякий протокол, який попереджує усі можливості зіткнень, втрати даних і блокування інформації. Швидкі пристрої повинні мати змогу зв'язатися з повільними пристроями. Система не повинна залежати від пристроїв, під'єднаних до неї, інакше модифікації й поліпшення стануть неможливими. Має бути також розроблено процедуру, згідно з якою визначається, який пристрій керує шиною й коли. Крім того, якщо різні пристрої з різними тактовими частотами під'єднано до шини, то має бути визначено джерело синхронізації шини. Усі ці критерії задовольняє шина I2C.

Шина I2C підтримує будь-яку технологію виготовлення мікросхем (nМОП, КМОП, біполярну). Дві лінії даних (SDA) і синхронізації (SCL) призначено для перенесення інформації. Кожен пристрій розпізнається з унікальною адресою – будь то мікроконтролер, РКІ-буфер, пам'ять або інтерфейс клавіатури – і може працювати як передавач або приймач залежно від призначення пристрою. Зазвичай РКІ-буфер – лише приймач, а пам'ять може як приймати, так й передавати дані. Крім того, пристрої можна класифікувати як ведучі й ведені під час передання даних. Ведучий – це пристрій, який ініціює передання даних і виробляє сигнали синхронізації. При цьому будь-який пристрій, що адресується, вважається веденим відносно ведучого. Приклад конфігурації системи з двома мікроконтролерами по шині I2C показано на рисунку 14.1.

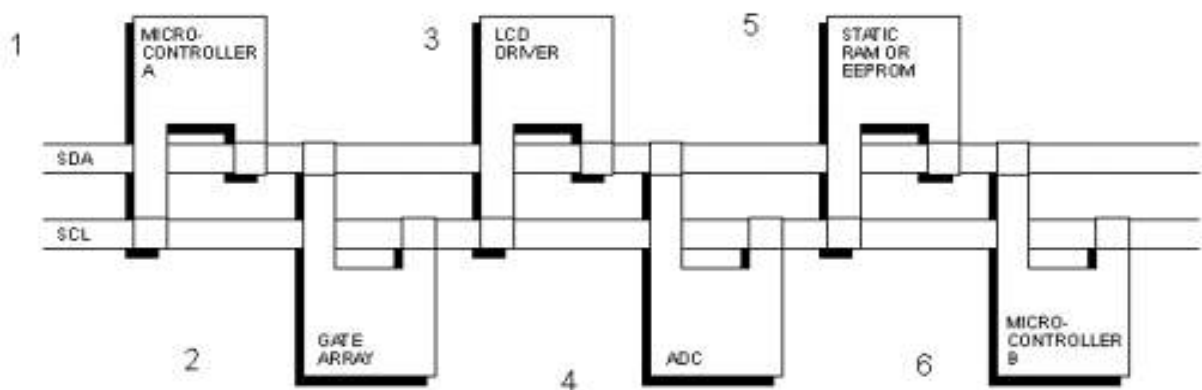


Рисунок 14.1 – Структура конфігурації системи з двома мікроконтролерами по шині I2C

У шині I2C допускається кілька ведучих. Це означає, що більш ніж один пристрій, який керує шиною, можна під'єднаним до неї. Розглянемо відношення передавач – приймач й ведучий -- ведений, що існують у шині I2C, на прикладі пересилання даних між двома мікроконтролерами А і В, під'єднаними до шини. Слід зазначити, що цей зв'язок не є постійним і залежить лише від напрямку пересилання даних у певний момент часу.

Кількість байтів, що передається за один сеанс зв'язку по лінії SDA, є необмеженою. Кожен байт має закінчуватися бітом підтвердження. Дані передаються, починаючи з найбільш значущого біта MSB (рисунок 14.2). Якщо приймач не може прийняти наступний байт, поки він не виконає яку-небудь іншу функцію (наприклад, обслужить внутрішнє переривання), то він може утримувати лінію SCL у низькому стані, переводячи передавач у стан очікування. Пересилання даних продовжиться, коли приймач буде готовим до наступного байта і звільнить лінію SCL. У деяких випадках необхідно використовувати інший формат даних (наприклад, CBUS). Посилання, яке передається з такою адресою, може бути закінчено виданням сигналу «стоп», навіть якщо це відбувається під час передання байта. У цьому випадку підтвердження не генерується.

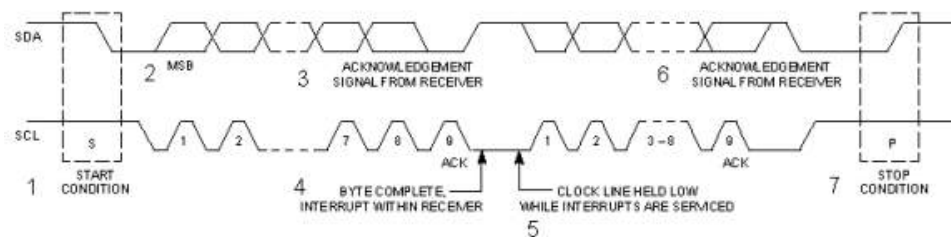


Рисунок 14.2 – Структура передання даних:

- 1 – сигнал СТАРТ (S);
- 2 – старший розряд байта (MSB – most significant bit);
- 3 – сигнал підтвердження від приймача (ACK);
- 4 – приймання байта завершено, переривання всередині приймача;
- 5 – синхролінія утримується в низькому стані, поки обслуговується переривання;
- 6 – сигнал підтвердження від приймача (ACK);
- 7 – сигнал СТОП (P)

Підтвердження під час передання даних є обов'язковим. Для цього ведучий пристрій генерує додатковий (9-й) імпульс синхронізації. Передавач відпускає (переводить у високий стан) лінію SDA протягом синхроімпульсу підтвердження. Приймач має утримувати лінію SDA протягом високого стану синхроімпульсу підтвердження в стабільно низькому стані (рисунок 14.3). Зазвичай приймач, який був адресованим, має генерувати підтвердження після кожного прийнятого байта, виключаючи ті випадки, коли посилання починається з адреси CBUS. У тому випадку, коли ведений пристрій не може підтвердити свою адресу (наприклад, коли він виконує у цей момент які-небудь функції реального часу), лінія даних має бути залишеною у високому стані. Після цього ведучий пристрій може видати сигнал «стоп» для переривання переслання даних. Якщо ведений пристрій підтвердив свою адресу, але через деякий час більше не може приймати дані, ведучий пристрій також має перервати переслання. Для цього ведений пристрій не підтверджує наступний байт, залишає лінію даних у високому стані й ведучий пристрій генерує сигнал «Стоп». Якщо у пересланні бере участь ведучий пристрій, то він має повідомити про закінчення передання веденому пристрою шляхом непідтвердження останнього байта. Ведений пристрій має звільнити лінію даних, щоб дати змогу ведучому пристрою видати сигнал «Стоп» або повторити сигнал «Старт».

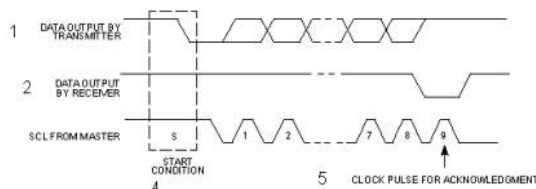


Рисунок 14.3 – Структура передання даних:

- 1 – дані, передані передавачем;
- 2 – дані, передані приймачем;
- 3 – синхроімпульси SCL від ведучого пристрою;
- 4 – сигнал «СТАРТ»;
- 5 – синхроімпульс підтвердження

При переданні послань по шині I2C кожен ведучий пристрій генерує свій синхросигнал на лінії SCL. Дані є дійсними лише під час високого стану синхроімпульсу. Синхронізація виконується з використанням під'єднання до лінії SCL. Це означає, що внаслідок переходу лінії SCL з високого стану в низький, спричиненого переходом синхросигналу одного з пристроїв у низький стан, відбудеться також перехід синхросигналу іншого пристрою в низький стан.

Цей стан лінії SCL утримується доти, доки не буде встановлено високий стан внутрішнього синхросигналу одного з пристроїв. Проте перехід синхросигналу з низького стану у високий може не викликати аналогічний перехід на лінії SCL, якщо синхросигнал іншого пристрою все ще перебуває в низькому стані. Таким чином, лінія SCL перебуватиме в низькому стані протягом найдовшого низького періоду з двох синхросигналів. Пристрої з коротшим низьким періодом входять у стан очікування на якийсь час, поки не закінчиться довгий період. Коли в усіх задіяних пристроїв закінчиться низький період синхросигналу, лінія SCL перейде у високий стан. Усі пристрої почнуть проходити високий період своїх синхросигналів. Перший пристрій, у якого закінчиться цей період, знову встановить лінію SCL у низький стан. Таким чином, низький період синхролінії SCL визначається найдовшим періодом синхронізації зі всіх задіяних пристроїв, а високий період – найкоротшим періодом синхронізації пристроїв.

Арбітраж відбувається на шині SDA у періоди, коли шина SCL перебуває у високому стані. Якщо один ведучий пристрій передає на лінію даних низький рівень, тоді як інший – високий, то останній від'єднується від лінії, оскільки стан SDL (низький) не відповідає високому стану його внутрішньої лінії даних. Арбітраж може продовжуватися протягом кількох бітів. Оскільки спочатку передається адреса, а потім дані, арбітраж може тривати до закінчення адреси, а якщо ведучі пристрої адресують один й той самий пристрій, то в арбітражі братимуть участь і дані.

БИБЛИОГРАФИЧНИЙ СПИСОК

Блек, Ю. Сети ЭВМ: протоколы, стандарты, интерфейсы : пер. с англ. / Ю. Блек. – М. : Мир, 1990. – 506 с.

Богуславский, Л. Б. Основы построения вычислительных сетей для автоматизированных систем / Л. Б. Богуславский, В. И. Дрожжинов. – М. : Энергоатомиздат, 1990. – 256 с.

Гук, М. Аппаратные интерфейсы ПК : энциклопедия / М. Гук. – СПб. : Питер, 2002. — 528 с.

Иди, Ф. Сетевой и межсетевой обмен данными с микроконтроллером : пер. с англ. / Ф. Иди. – М. : ДОДЕКА, 2007. – 370 с.

Новиков, Ю. В. Разработка устройств сопряжения для персонального компьютера типа IBM PC / Ю. В. Новиков, О. А. Калашников, С. Е. Гуляев. – М. : ЭКОМ, 2000. – 221 с.

Овчинников, В. В. Техническая база интерфейсов локальных вычислительных сетей / В. В. Овчинников, И. И. Рыбкин. – М. : Радио и связь, 1989. – 271 с.

Огик, П. Использование LPT-порта ПК для ввода/вывода информации / П. Огик. – М. : НТ Пресс, 2006. – 106 с.

Фурман, І. О. Організація й програмування мікроконтролерів : підруч. для студентів вузів / І. О. Фурман та ін. – Харків : Еспада, 2005. – 248 с.

ЗМІСТ

Лекція 1 Вступ. Класифікація інтерфейсів	3
1.1 Загальні відомості про інтерфейси	3
1.2 Класифікація інтерфейсів	5
1.3 Гальванічна розв'язка.....	9
1.4 Інтерфейси блоку живлення портативного комп'ютера.....	11
1.5 Аналогова та цифрова земля.....	12
1.6 Джерела шумів і завад та методи боротьби з ними	13
Лекція 2 Паралельний інтерфейс LPT-порт.....	16
2.1 Загальні відомості про LPT-порт.....	16
2.2 Інтерфейс Centronics.....	16
2.3 Стандартний паралельний порт ПК.....	18
2.4 Стандарт IEEE 1284	19
2.5 Функції BIOS для LPT-порту.....	21
2.6 Режими передавання даних.....	22
Лекція 3 Послідовний інтерфейс COM-порт	28
3.1 Загальні відомості про COM-порт.....	28
3.2 Фізичні реалізації послідовного інтерфейсу	30
3.3 Інтерфейс RS-232C	31
3.4 Керування потоком даних	35
3.5 Робота з COM-портом на низькому рівні	37
Лекція 4 Універсальна послідовна шина.....	38
4.1 Загальні відомості про USB.....	38
4.2 Структура USB	38
4.3 Фізичний інтерфейс USB.....	40
4.4 Протокол.....	43
4.5 Функції та хаби	43
4.6 USB-пакети	45
4.7 Пріоритети передавання даних по USB-шині.....	47
Лекція 5 Послідовний інтерфейс – «шина FireWire».....	48
5.1 Загальні відомості про стандарт IEEE 1394.....	48
5.2 Компоненти FireWire	50
5.3 Протокол IEEE1394	53
5.4 Специфікація FireWire	56
Лекція 6 Шина PCI. Порт – AGP	57

6.1	Основні відомості про шину PCI.....	57
6.2	Сигнали шини PCI.....	58
6.3	Порт AGP.....	60
Лекція 7	Шина розширення ISA, EISA та PCI/104	64
7.1	Основні властивості шини ISA.....	64
7.2	Основні властивості шини EISA.....	66
7.3	Сумісність шини EISA та PCI.....	67
7.4	Характеристики шини PC-104	68
7.5	Стандарт на промислові PC-комп'ютера для вбудованих додатків (розширення стандарту IEEE-P996 (ISA)).....	70
7.6	Види використання модулів PC-104.....	72
Лекція 8	Інтерфейси 1-Wire, SPI, JTAG.....	72
8.1	Однопровідний інтерфейс 1-Wire.....	72
8.2	Лінії зв'язку й топологія інтерфейсу 1-Wire.....	75
8.3	Послідовний периферійний інтерфейс SPI.....	77
8.4	Послідовний периферійний інтерфейс JTAG	80
Лекція 9	Бездротові інтерфейси периферійних пристроїв	85
9.1	Інфрачервоний інтерфейс IrDA.....	85
9.2	Радіоінтерфейс Bluetooth	89
Лекція 10	Промислові мережі.....	92
10.1	Загальні характеристики шини CAN.....	92
10.2	Топологія мережі CAN	93
10.3	Фізичний рівень каналу CAN	94
10.4	Арбітраж шини CAN	95
10.5	Структура формату передання даних	95
10.6	Механізм оброблення помилок.....	97
10.7	Адресація і протоколи високого рівня.....	98
10.8	Інтерфейс PROFIBUS	102
10.9	Фізичний рівень інтерфейсу PROFIBUS	103
10.10	Канальний рівень Profibus DP	105
10.11	Передання повідомлень Profibus	108
Лекція 11	Універсальний асинхронний інтерфейс UART	111
11.1	Загальні відомості.....	111
11.2	Формат передання даних	112
11.3	USART у мікроконтролерах STM32	114

11.4 Використання USART для обміну даними	117
Лекція 12 Модулі аналогових інтерфейсів	117
12.1 Модулі послідовного введення-виведення.....	117
12.2 Модулі аналогового введення-виведення.....	119
12.3 Інтерфейс «струмова петля»	121
12.4 HART-протокол.....	122
Лекція 13 Інтерфейс мікропроцесора	124
13.1 Класифікація мікропроцесорних комплексів.....	124
13.2 Організація шин мікропроцесорних комплексів	126
13.3 Цикли шини процесора.....	131
Лекція 14 Інтерфейси периферійних вузлів комп'ютерних систем	134
14.1 Концепція шини SCSI.....	134
14.2 Реалізація протоколу SCSI-шини	136
14.3 Загальні відомості шини I2C	138
Бібліографічний список.....	142

Навчальне видання

**Бреславець Марія Володимирівна
Білоконська Юлія Валентинівна**

ІНТЕРФЕЙСИ ТА ЗАСОБИ ВЗАЄМОДІЇ ПРИСТРОЇВ АВТОМАТИЗАЦІЇ

Редактор О. Ф. Серьожкіна

Зв. план, 2020

Підписано до видання 28.12.2020

Ум. друк. арк. 8,1. Обл.-вид. арк. 9,13. Електронний ресурс

Видавець і виготовлювач
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»
61070, м. Харків, вул. Чкалова, 17
<http://www.khai.edu>
Видавничий центр «ХАІ»
61070, м. Харків, вул. Чкалова, 17
izdat@khai.edu

Свідоцтво про внесення суб'єкта видавничої справи
до Державного реєстру видавців, виготовлювачів і розповсюджувачів
видавничої продукції сер. ДК № 391 від 30.03.2001