

Технології передавання даних

Лекція 1

Дисципліна «Мікроконтролери та технології передавання даних в енергетиці»

ЗМ 3. Семестр 4

Проф. Райтер П.М.

Зміст

- Поняття обміну даними
- Стандарти
- Модель взаємодії відкритих систем (OSI)
- Протоколи
- Фізичні стандарти. Стандарт інтерфейсу RS-232
- Інтелектуальні вимірювальні системи
- Принципи обміну даними
- Режими передачі сигналів
- Асинхронні системи
- Синхронні системи
- Виявлення помилок
- Характеристики процесу передачі

Поняття обміну даними

- Обмін даними є передачею інформації від однієї точки системи контролю до іншої. Далі розглядається власне цифровий обмін даними.
- У цьому контексті поняття «дані» відноситься до інформації, яка представлена послідовністю нулів та одиниць; з аналогічним типом даних має справу також комп'ютер.
- Багато систем обміну інформацією працюють з аналоговими даними – прикладом є телефонна мережа, радіо та телебачення.
- Але все більше сучасних вимірювальних приладів пов'язані з передачею цифрових даних.
- Для будь-якої системи обміну даними необхідні передавач, щоб відправляти дані, приймач, щоб приймати їх, і канал зв'язку, що знаходиться між ними.
- Каналом зв'язку може бути мідний провід, оптоволокно, радіоканал чи мікрохвильове випромінювання.
- Іноді на коротких відстанях використовується паралельна передача даних; під цим мається на увазі, що для передачі сигналу потрібно кілька дротів. Таке підключення обмежується такими пристроями, як наприклад локальний принтер.

Поняття обміну даними

- Практично всі сучасні системи обміну даними використовують канали зв'язку, у яких дані передаються послідовно з одного ланцюга.
- Іноді цифрові дані передаються за допомогою системи, яка переважно початково була пристосована для забезпечення аналогових комунікацій.
- Наприклад, так працює модем, використовуючи потік цифрових даних для модуляції аналогового сигналу, що передається телефонною лінією.
- На приймальному кінці інший модем демодулює цей сигнал та отримує вихідні цифрові дані. Слово «модем» виходить із двох слів — модулятор та демодулятор.
- Між сторонами, що передає та приймає, має бути встановлена взаємна домовленість про те, як треба кодувати\декодувати дані.
- Приймач повинен вміти розуміти те, що надсилає передавач.
- Правило, за яким пристрої спілкуються один з одним, називається протоколом.
- Ще десять років тому було визначено багато стандартів і протоколів, які дозволяють технології обміну даними ефективно застосовувати в промисловості, але зараз кількість найбільш вживаних стандартів обмежена.

Стандарти

- Протоколи є структурами, що використовуються в системах обміну даними для того, щоб комп'ютер міг, наприклад, спілкуватися з принтером.
- Зазвичай розробники програмного та апаратного забезпечення використовують протоколи, які можуть використовувати лише їх пристрої.
- Щоб розробляти більш універсальні системи контролю та управління, необхідна стандартизація протоколів зв'язку.
- Стандарти можуть розвиватися від протоколу, що широко використовується, одного виробника (що є стандартом de facto) або можуть бути спеціально розроблені організацією, що представляє будь-яку галузь промисловості.
- Стандарти дозволяють виробникам розробляти продукцію, яка може використовуватися разом із вже існуючим обладнанням, що спрощує користувача інтеграцію пристроїв різних виробників.

Модель взаємодії відкритих систем (OSI)

- Модель OSI, розроблена Міжнародною організацією зі стандартизації (ISO), швидко набуває підтримки у промисловості.
- Модель OSI зводить усі проблеми розробки та обміну даними до кількох рівнів, показаних на рис. 1.1.
- Стандарт на фізичний інтерфейс, такий як, наприклад, RS-232, добре вписується у «фізичний рівень», інші рівні відносяться до інших протоколів.
- Повідомлення або дані зазвичай надсилаються у вигляді пакетів, які є простою послідовністю байтів.
- Довжину пакета, зазвичай фіксована, визначає протокол.
- Кожен пакет вимагає адреси джерела та адреси призначення, щоб система знала, куди його передавати, і щоб приймач знав, від кого прийшов пакет.

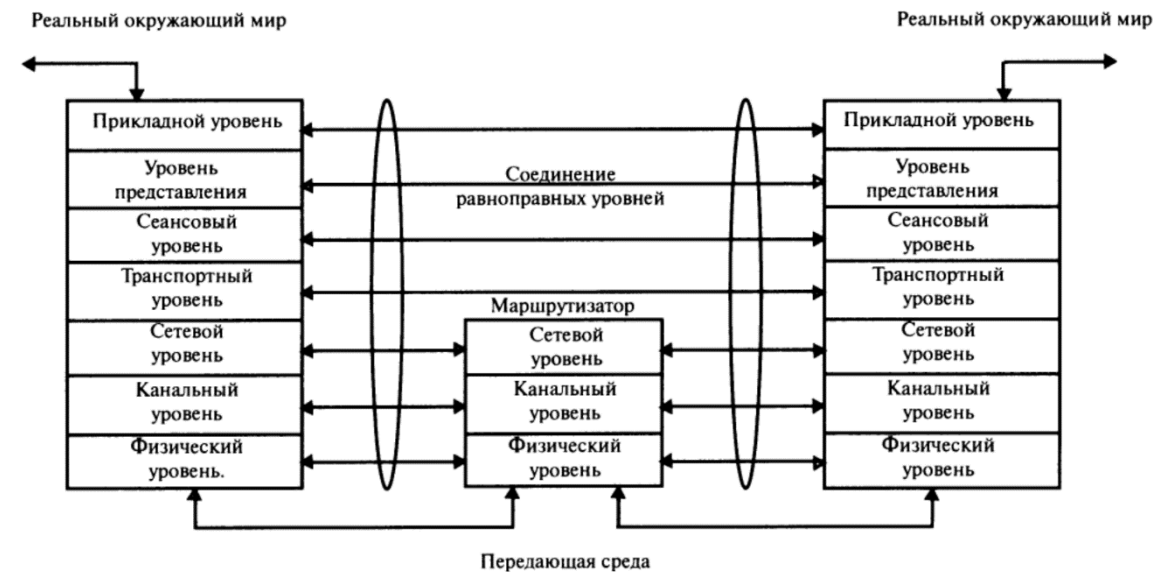


Рис. 1.1 Подання моделі OSI

Модель взаємодії відкритих систем (OSI)

- Передача пакета починається з верху стека протоколу, з рівня програми і потім проходить вниз через решту рівня, доки не досягне фізичного рівня.
- Саме тоді пакет пересилається каналом зв'язку.
- Зміщуючись вниз по стеку, на кожному рівні пакет отримує додатковий заголовок, який повідомляє нижньому рівню, що робити з пакетом.
- На приймаючому кінці пакет переміщується вгору стеком, і тут, навпаки, при переміщенні він відкидає інформацію заголовків.
- Прикладний рівень приймає лише дані, передані прикладним рівнем сторони, що передає.
- Стрілки між рівнями на мал. 1.1 показують, що кожен рівень зчитує пакет як він приходить від або йде до відповідного рівня на протилежному кінці каналу зв'язку.
- Це називається з'єднанням рівноправних рівнів каналу комунікації, хоча насправді пакет передається за допомогою фізичної ланки.

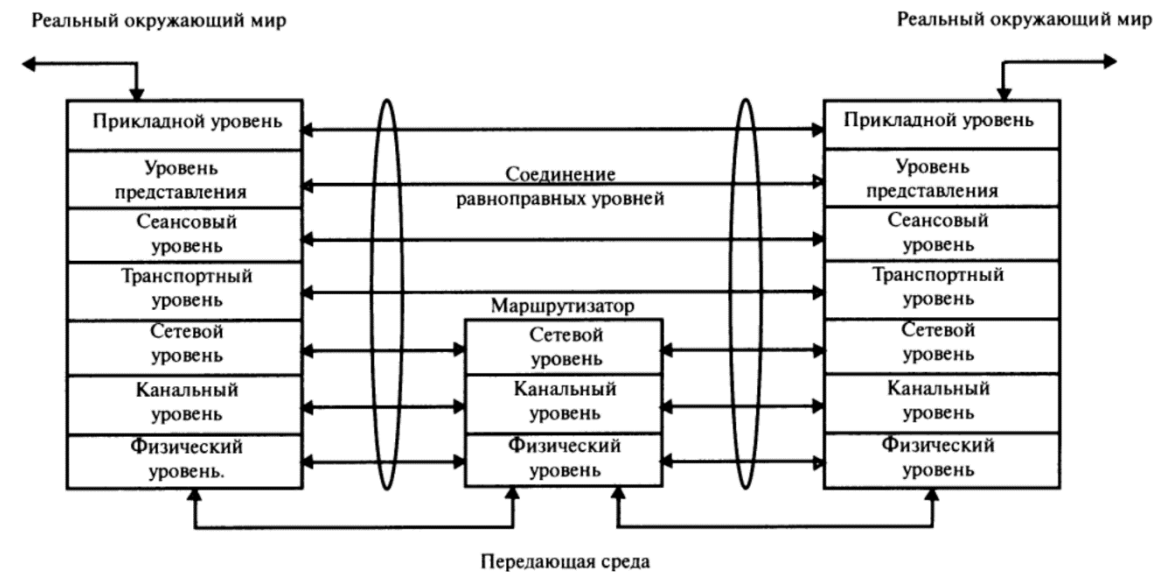


Рис. 1.1 Подання моделі OSI

Модель взаємодії відкритих систем (OSI)

- Стек, зображений посередині, у цьому конкретному випадку є маршрутизатором, має лише три нижніх рівня, які реалізують усе, що потрібно для коректної передачі пакета між пристроями.
- Модель OSI корисна у забезпеченні універсальної основи для всіх комунікаційних систем.
- Проте вона не визначає реальний протокол, який має використовуватися кожному рівні.
- Передбачається, що групи виробників у різних галузях промисловості, співпрацюючи, вироблять стандарти програмного та апаратного забезпечення, що найбільш підходять саме для їхньої галузі.
- Ті ж, хто шукає єдиний базис для своїх специфічних вимог до обміну даними, з ентузіазмом скористалися моделлю OSI і використовують її як основу вироблення своїх промислових стандартів, таких як Fieldbus та HART.
- Прийняття цих стандартів усім ринком було уповільнено невпевненістю у широкому схваленні будь-якого конкретного стандарту, додатковою вартістю запровадження стандарту та занепокоєнням про адекватну підтримку та підготовку фахівців з експлуатації систем.

Протоколи

- Модель OSI забезпечує базис, для якого може бути визначений будь-який специфічний протокол.
- Кадр (пакет) може складатися з наступних елементів:
- Перший байт може бути рядком одиниць та нулів, необхідних для синхронізації приймача або для забезпечення прапорів, що є індикатором початку пакета (для використання приймачем).
- Другий байт може містити адресу призначення, яка вказує місце, куди має піти повідомлення.
- Третій байт може містити адресу джерела, що вказує, звідки виходить повідомлення.
- Байти посеред повідомлення можуть бути реальними даними, які необхідно переслати від передавача до приймача.
- Кінцевий байт є індикаторами кінця пакета, який може містити коди виявлення помилки та/або кінцеві прапори.
- Протоколи можуть бути дуже простими (наприклад, протокол на основі коду ASCII), так і дуже складними, які працюють з дуже високими швидкостями, передаючи мегабіти даних в секунду.
- Не може бути хорошого чи поганого протоколу – вибір залежить від конкретного застосування.

перше

останнє

Байт синхронізації	Адреса призначення	Адреса джерела	Власне дані що передаються	Байт виявлення помилок
--------------------	--------------------	----------------	----------------------------	------------------------

Фізичні стандарти

Стандарт інтерфейсу RS-232

- Стандарт інтерфейсу RS-232C був випущений у США в 1969 р., щоб визначити електричні та механічні параметри інтерфейсу між термінальним обладнанням (DTE) та апаратурою передачі даних (DCE), які використовували послідовний двійковий обмін даними.
- При послідовному обміні даними комунікаційна система може містити такі компоненти:
 - ✓ DTE — термінал, що надсилає дані, наприклад, комп'ютер, який є джерелом даних (зазвичай передаються послідовності символів, закодовані з використанням відповідного цифрового формату);
 - ✓ DCE - пристрій, що діє як перетворювач даних (наприклад, модем), який перетворює сигнал у форму, що підходить для каналу зв'язку, наприклад аналоговий сигнал для телефонної лінії;
 - ✓ сам канал зв'язку, наприклад, телефонна лінія;
 - ✓ відповідний приймач, такий як модем, також є DCE пристроєм, який перетворює аналоговий сигнал знову у форму, що підходить для приймаючого терміналу;
 - ✓ термінал прийому даних, наприклад принтер, також DTE пристрій, який приймає цифрові імпульси та декодує їх знову у послідовності символів.



Стандарт інтерфейсу RS-232

- Стандарт на інтерфейс RS-232C описує взаємодію терміналу (DTE) та модему (DCE) саме при передачі послідовних двійкових чисел.
- Він надає широку свободу розробникам протоколів апаратного та програмного забезпечення.
- Згодом цей стандарт на інтерфейс був пристосований для роботи з безліччю іншого обладнання, наприклад, комп'ютери (ПК), принтери, програмовані контролери, програмовані логічні контролери (PLC), вимірювальні прилади тощо.
- Щоб розрізнити ці додаткові застосування, остання версія стандарту RS-232E розширила значення акроніму DCE від простого "data communication equipment (апаратура передачі даних)" до узагальненого "data circuit-terminating equipment (кінцеве обладнання передачі даних)".
- Інтерфейс RS-232 має ряд властивих йому слабких місць, які не дозволяють використовувати його для обміну даними із системами контролю та управління у промислових умовах, тому для подолання цих обмежень було розроблено інші інтерфейсні стандарти RS.
- Найбільш часто використовуваними з них для систем вимірювання та контролю є RS-423, RS-422 та RS-485.

Інтелектуальні вимірювальні системи

- У 1960-х роках як стандарт для вимірювальних приладів de facto був прийнятий аналоговий інтерфейс “струмова петля” 4-20 мА.
- В результаті цього виробники вимірювального обладнання отримали стандартний комунікаційний інтерфейс, який використовували при виробництві своїх приладів. Користувачі мали великий вибір приладів та датчиків від великої кількості виробників, які могли бути використані в їх системах керування.
- З приходом мікропроцесорів та розвитком цифрових технологій ситуація змінилася.
- Більшість користувачів оцінили переваги цифрових вимірювальних приладів. До них відноситься великий обсяг інформації, що відображається на одному приладі, місцева та віддалена індикація, надійність, економічність, автоматичне налаштування та можливість діагностування. В даний час спостерігається поступовий перехід від аналогових до цифрових технологій.
- Є ряд інтелектуальних цифрових датчиків, що підтримують обмін цифрових даних і підходять для традиційних застосувань. До них відносяться датчики для вимірювання температури, тиску, рівнів, потоку, маси (ваги), щільності та параметрів систем живлення. Ці нові цифрові датчики стали відомі як інтелектуальні прилади.
- До головних особливостей, що характеризують інтелектуальний вимірювальний прилад, належать:
 - ✓ інтелектуальні цифрові датчики;
 - ✓ можливість передачі цифрової інформації;
 - ✓ здатність поєднання з іншими приладами.
- З'явився ряд інтелектуальних цифрових пристроїв, що підтримують передачу даних, які можна назвати інтелектуальними виконавчими пристроями. Прикладами таких пристроїв є різні приводи з регульованою швидкістю, плавні пускачі, захисні реле та комутаційна апаратура, які забезпечують обмін даними.

Принципи обміну даними

- Кожна система обміну даними вимагає:
 - ✓ джерела даних (передавач або лінійний формувач), який перетворює інформацію на форму, зручну передачі по каналу зв'язку;
 - ✓ приймача, який приймає сигнали та перетворює їх назад у вихідні дані;
 - ✓ каналу зв'язку, яким передаються сигнали. Це може бути мідний дріт, оптоволокно, радіо або супутниковий зв'язок.
- Крім того, передавач та приймач повинні розуміти один одного. Для цього має бути вироблена угода щодо ряду факторів:
 - ✓ тип використовуваних сигналів;
 - ✓ визначення логічної «одиниці» та логічного «нуля»;
 - ✓ коди, що становлять ці символи;
 - ✓ забезпечення синхронізації між передавачем та приймачем;
 - ✓ керування потоком даних, щоб не переповнити приймач;
 - ✓ спосіб виявлення та виправлення помилок, що виникають під час передачі.
- Фізичні фактори зазвичай називають стандартом інтерфейсу, всі інші фактори відносяться до протоколу.
- Фізичний спосіб передачі даних каналом зв'язку залежить від використовуваного середовища.
- Наприклад, двійкові значення 0 і 1 можуть виражатися присутністю або відсутністю напруги на мідному дроті, двома звуковими частотами, що генеруються та декодуються модемом, як це відбувається в телефонній системі, або шляхом модуляції світла, як відбувається при передачі сигналу по оптоволокну.

Режими передачі сигналів

- Через будь-який канал зв'язку, що з'єднує два пристрої, можна передавати дані з використанням одного з трьох режимів:
 - ✓ симплексний;
 - ✓ напівдуплексний;
 - ✓ повнодуплексного
- Симплексна система - це така система, яка пристосована для передачі повідомлень лише в одному напрямку (рис.).



Режими передачі сигналів

- Дуплексна система призначена для передачі повідомлень в обох напрямках.
- Напівдуплексною вважається передача, при якій дані можуть передаватися в обох напрямках, але лише у різний час (рис).



- У повнодуплексній системі дані можуть передаватися в обох напрямках одночасно (рис.).



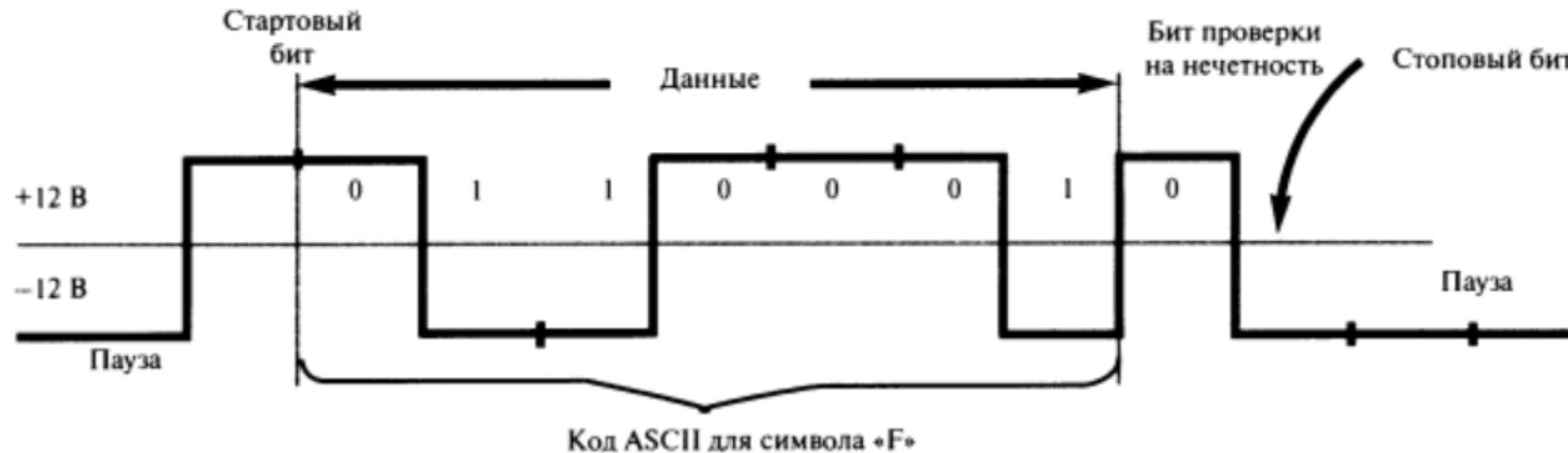
Асинхронні системи

- Асинхронна система - це система, в якій кожен символ або байт надсилається у вигляді кадру.
- Приймач не починає виявлення даних доти, доки він не прийме перший біт, званий стартовим бітом.
- Стартовий біт представляється напругою, протилежною до напруги паузи, і дозволяє приймачеві синхронізуватися з передавачем для отримання даних, що містяться в кадрі.
- Приймач зчитує окремі біти кадру в міру їх надходження, контролюючи у часі напругу логічного 0 або логічного 1.
- Тактова частота повинна бути однаковою на обох кінцях лінії, щоб приймач прочитував кожен біт у той момент, коли передавач посилає його.
- Оскільки тактові частоти синхронізуються на початку кожного кадру, то за низьких швидкостей передачі допускаються деякі відхилення.
- У міру збільшення швидкості передачі відхилення повинні зменшуватися, а при високих швидкостях (понад 100 кбіт/с) асинхронна передача даних може стикатися з проблемами синхронізації.

Формат повідомлення

- Асинхронний кадр може мати такий формат:
- Стартовий біт: сигнали на початку кадру
- Дані: зазвичай 7 або 8 бітів, але може бути 5 або 6 бітів
- Біт парності: біт виявлення операційної помилки
- Стоповий біт (біти): зазвичай 1, 1,5 або 2 біти. Величина 1,5 означає, що рівень утримується протягом часу, півтора рази більшого часу передачі одного біта

Формат асинхронного кадру



Формат повідомлення

- Передавач та приймач повинні бути налаштовані точно на одну конфігурацію, щоб з кадру можна було коректно вийняти дані.
- Кожен символ має власний кадр, причому реальна швидкість передачі виходить менше швидкості передачі бітів за секунду.
- Наприклад, з урахуванням стартового біта, семи бітів даних, одного біта парності та одного стопового біта для передачі семи бітів даних всього необхідно передавати десять бітів.
- Таким чином, передача корисних даних становить 70% від усієї швидкості передачі бітів.

Синхронні системи

- У синхронних системах приймач спочатку синхронізується за тактовими імпульсами передавача, які містяться в потоці даних, що передаються.
- Це дозволяє приймачеві підтримувати синхронізацію протягом довгих повідомлень, які зазвичай можуть бути до 4500 байтів (36000 бітів).
- Це забезпечує ефективну передачу великих кадрів із високою швидкістю.
- Синхронні системи пакують безліч символів разом і посилають їх у вигляді безперервного потоку, званого пакетом (або кадром).

Формат повідомлення

Типовий кадр синхронної системи

Преамбула	Начало разграничителя данных (SFD)	Преамбула	Адрес назначения	Источник сообщения	Длина данных	Данные	Контрольная последовательность кадра (FCS)
-----------	------------------------------------	-----------	------------------	--------------------	--------------	--------	--

Типовий кадр синхронної системи

- Преамбула: сюди входить один або кілька байтів, які дозволяють пристрої, що приймає, синхронізуватися з кадром.
- SFD: початок розмежувача даних сигналізує про початок кадру.
- Адреса призначення: адресу, за якою відправляється кадр,
- Джерело: адресу, з якої виходить кадр,
- Довжина даних: кількість байтів у полі даних,
- FCS: контрольна послідовність кадру, призначена для виявлення помилок.

Кожен із цих елементів називається полем.

Виявлення помилок

- Будь-яка практична передача даних піддається впливу перешкод, особливо мідні дроти, прокладені в промислових умовах, де можуть бути великі електричні перешкоди.
- Наявність перешкод може призводити до спотворення отриманих даних.
- Основним принципом виявлення помилок є обчислення передавачем контрольного символу, що базується на вмісті оригінального повідомлення.
- Цей символ посилається приймачеві в кінці повідомлення, і приймач повторює ті ж обчислення з бітами, які він отримує.
- Якщо обчислений контрольний символ відповідає переданому символу, можна припустити, що з передачі відбулася помилка.
- Найпростішою формою контролю над помилками в асинхронних системах є додавання біта парності, який може бути парним чи непарним.

Виявлення помилок

- Перевірка на парність вимагає, щоб сумарна кількість бітів даних, що є логічною 1, плюс біт парності дорівнювало парному числу.
- Комунікаційне обладнання на передавальній стороні обчислює необхідну парність та встановлює біт парності, щоб забезпечити парне число логічних одиниць.
- Перевірка на непарність працює аналогічно до перевірки на парність, за винятком того, що біт парності встановлюється таким чином, щоб повна кількість логічних одиниць, включаючи біт парності, дорівнювала непарному числу.
- Апаратне забезпечення на стороні, що приймає, визначає сумарну кількість логічних одиниць і повідомляє про помилку, якщо воно не відповідає парному або непарному числу.
- Апаратура приймача також виявляє переповнення та помилки кадру.
- Статистично використання біта парності при високій швидкості передачі має можливість виявлення помилки лише 50%.
- Цей спосіб дозволяє виявити парне число помилкових бітів і дозволяє виявляти непарне число помилкових бітів.
- Біт парності зазвичай опускається, якщо замість цього способу використовуються складніші схеми контролю над помилками.

Характеристики процесу передачі

- Швидкість передачі сигналів (у бодах)
- Швидкість передачі сигналів через канал зв'язку визначається тим, скільки разів фізичний сигнал зміниться на секунду, і вимірюється в бодах.
- Якщо передачу даних проконтролювати з допомогою осцилографа, то екрані будуть імпульси, частота яких відповідає швидкості в бодах.
- Наприклад, швидкість 1000 бод відповідає імпульсам, що впливають через 1 мс.
- Для асинхронних систем швидкість у бодах на обох кінцях каналу зв'язку встановлюється так, щоб тривалість імпульсів була однаковою.

Швидкість передачі даних

- Швидкість передачі даних або швидкість передачі бітів виражається в бітах в секунду (біт/с) або в кратних одиницях: кбіт/с, Мбіт/с та Гбіт/с (кіло, мега та гігабіт за секунду).
- Ця швидкість представляє фактичне число біт даних, що передаються в секунду.
- Для прикладу розглянемо 1000 бод, що передаються каналом RS-232 вигляді кадру з 10 біт, який складається з 7 біт даних, стартового, стопового біта та біта парності.
- У цьому прикладі швидкість передачі сигналу становить 1000 бод, швидкість передачі даних - 700 біт/с.
- Хоча є тенденція плутати швидкості передачі сигналу і передачі бітів, це не те саме.
- У той час як швидкість передачі сигналу вказує на кількість змін сигналу в секунду, швидкість передачі бітів вказує на кількість бітів, представлених кожною зміною сигналу.
- У простих системах передачі, таких як RS-232, швидкість передачі сигналу дорівнює швидкості передачі бітів.
- Для синхронних систем швидкість передачі бітів майже перевищує швидкість передачі сигналу.
- Але для ВСІХ систем швидкість передачі даних менша, ніж швидкість передачі бітів, що обумовлено передачею додаткових бітів - стартового та стопового, а також бітів парності (в асинхронних системах) або додаткових полів - адресного поля та поля виявлення помилки у кадрах синхронної системи.

Ширина смуги

- Єдиним найважливішим фактором, який обмежує швидкість передачі даних, є ширина смуги каналу зв'язку.
- Ширина смуги зазвичай виявляється у герцах (Гц) і означає кількість коливань на секунду.
- Ширина смуги представляє максимальну частоту, з якою може змінюватися сигнал, коли ослаблення погіршує сигнал.
- Ширина смуги близько пов'язана з середовищем і може бути від 5000 Гц для телефонної мережі до декількох гігагерц для оптоволоконного кабелю.
- Оскільки сигнал має тенденцію послаблюватися з відстанню, то канали зв'язку можуть вимагати установки через деякий інтервал повторювачів (репітерів), які необхідні для посилення сигналу, що загасає.
- Обчислення теоретичної максимальної швидкості передачі даних використовує формулу Найквіста та включає ширину смуги та кількість рівнів, що використовуються для кодування кожного елемента сигналу.

Відношення сигнал/шум

- Відношення сигнал/шум (S/N) каналу зв'язку є ще одним важливим фактором, що обмежує швидкість передачі.
- Джерела перешкод можуть бути як зовнішніми, так і внутрішніми.
- Максимальна реальна швидкість передачі даних для комунікаційного каналу математично пов'язана із шириною смуги, відношенням сигнал/шум та кількістю рівнів, за допомогою яких кодується кожний елемент сигналу.
- При зменшенні відношення сигнал/шум зменшується швидкість передачі бітів.
- Взаємозв'язок цих властивостей описується законом Шеннона-Хартлі (Shannon-Hartley).

Пропускна спроможність

- Оскільки дані завжди передаються в протокольній оболонці, яка може бути кадром символу або представляти набагато складніші схеми передачі повідомлення, швидкість передачі даних завжди буде меншою за швидкість передачі бітів.
- Кількість надлишкових даних, що входять до пакету повідомлення, збільшується в міру його проходження вниз стеком протоколів мережі.
- Це означає, що ставлення неінформативних даних до «реальної» інформації може бути важливим фактором у визначенні ефективної швидкості передачі, яка іноді називається пропускнуою спроможністю.

Частота появи помилок

- Частота появи помилок пов'язана з такими факторами, як відношення сигнал/шум, так і наявністю шумів і перешкод.
- Зазвичай існує компроміс між швидкістю передачі та допустимою частотою помилок, яка залежить від типу програми.
- Зазвичай, в промислових системах управління не дозволяється наявність помилок і тому розробляються з урахуванням максимальної надійності передачі.
- Це означає, що промислова система буде порівняно повільно діючою у сенсі передачі.
- У міру збільшення швидкості передачі є точка, в якій кількість помилок стає занадто великою.
- Протоколи опрацьовують помилки шляхом запиту повторної передачі пакетів.
- Очевидно, що кількість повторних передач врешті-решт досягне того, що фактично висока швидкість передачі даних знижує реальну швидкість передачі повідомлень, оскільки потрібно багато часу на повторні передачі збійних повідомлень.

Кодування даних

- Узгоджений стандартний код дозволяє приймачеві розуміти повідомлення, надіслані передавачем.
- Кількість бітів у коді визначає максимальну кількість унікальних символів або знаків, які можуть бути представлені кодом.
- Найпоширеніші коди описуються далі.

Код Бодо

- Хоча код Бодо використовується нечасто, він має історичне значення. Цей код був винайдений в 1874 Бодо (Maurice Emile Baudot) і вважається першим кодом, всі символи якого мали однакову довжину.
- Використовуючи п'ять бітів, він може представляти 32 (25) символи та підходить для систем, що передають лише літери, кілька знаків пунктуації та керуючі коди.
- Код Бодо використовувався головним чином перших телетайпах.
- Сучасна версія коду Бодо була адаптована ІТУ (Міжнародний союз телекомунікацій) як стандарт для телеграфних повідомлень і використовує два символи "shift" (перемикання регістру) для букв і цифр.
- Цей код став попередником сучасних кодів ASCII та EBCDIC.

Код ASCII

- Найбільш поширеним набором символів у країнах є Американський стандартний код обміну інформацією, або ASCII (табл. Наст. слайд).
- Цей код використовує рядок з 7 бітів, що дає можливість подати у закодованому вигляді 128 (2⁷) символів, до яких входять:
 - верхній та нижній регістри літер,
 - цифри від 0 до 9,
 - пунктуаційні знаки та символи,
 - набір керуючих кодів, що становлять перші 32 символи, які використовуються самим каналом зв'язку і під час друку не виводяться.
- Приклад: символ D відповідає двійковому коду ASCII 1000100.

Код ASCII

- Канал зв'язку, налаштований на рядки даних, що складаються з 7 бітів, може працювати лише з шістнадцятковими значеннями від 00 до 7F.
- Для повної шістнадцяткової передачі даних необхідний 8-розрядний канал зв'язку, в якому кожен пакет складається з байта (два шістнадцяткових знака), який може бути від 00 до FE.
- Восьмирозрядний канал зв'язку часто називають прозорим, оскільки він може передати будь-яке значення. У такому каналі зв'язку символ можна інтерпретувати як значення ASCII, якщо це необхідно, але в цьому випадку восьмий біт ігнорується.
- Повний шістнадцятковий набір можна передати через 7-розрядний канал зв'язку шляхом подання кожного шістнадцятирозрядного знака у вигляді його ASCII еквіваленту.
- Таким чином, шістнадцяткове число 8E буде представлено у вигляді двох ASCII значень 3845 (шістнадцяткових) («8» «E»).
- Недоліком такого способу є те, що кількість даних, що передаються, майже подвоюється, при цьому на кожній стороні потрібна додаткова обробка даних.

Код ASCII

- Доступ до керуючого коду ASCII можна отримати безпосередньо з клавіатури ПК, натиснувши [Ctrl] разом з іншою клавішею.
- Наприклад, поєднання Control-A(^A) генерує ASCII код початку заголовка (SOH).
- ASCII є найпоширенішим кодом, який використовується для кодування символів під час передачі даних.
- Він є 7-розрядним кодом і, отже, допускає лише $2^7 = 128$ можливих комбінацій із семи двійкових знаків (бітів) у діапазоні двійкових величин від 0000000 до 1111111 або в діапазоні шістнадцяткових величин від 00 до 7F.
- Кожному з цих 128 кодів надано специфічні керуючі коди або символи, що визначаються такими стандартами:
 - ANSI-X3.4.
 - ISO-646.
 - ITU алфавіту № 5.
- Таблиця ASCII є довідковою таблицею, яка використовується для запису бітового значення кожного символу, визначеного кодом.
- Існує безліч різних форм цієї таблиці, але всі вони містять одну й ту саму базову інформацію, що відповідає стандартам.

Приклад частини таблиці кодів ASCII

		MSB							
		0	1	2	3	4	5	6	7
HEX	HEX	0	1	2	3	4	5	6	7
	BIN	000	001	010	011	100	101	110	111
0	0000	(NUL) Пустой символ	(DEL) Авторегистр	(Space) Пробел	0	@	P	`	p
1	0001	(SOH) Начало заголовка	(DC1) Управление устройством	!	1	A	Q	a	q
2	0010	(STX) Начало текста	(DC2) Управление устройством	“	2	B	R	b	r
3	0011	(ETX) Конец текста	(DC3) Управление устройством	#	3	C	S	c	s
4	0100	(EOT) Конец передачи	(DC4) Управление устройством	\$	4	D	T	d	t
5	0101	(ENQ) Символ запроса	(NAK) Отсутствие подтверждения приема	%	5	E	U	e	u

Приклад кодування слова ASCII кодом

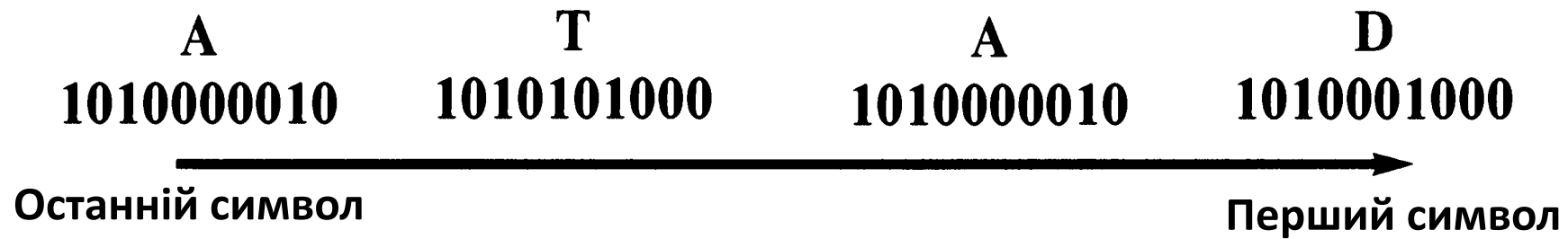
- Для подання слова «DATA» у двійковому вигляді за допомогою 7-розрядного коду ASCII (Код ASCII) кожна буква повинна кодуватись наступним чином:

	Двійковий код	Шіснадцятковий код
D	100 0100	44
A	100 0001	41
T	101 0100	54
A	100 0001	41

Приклад кодування слова ASCII кодом

- Повертаючись до таблиці ASCII, можна побачити, що двійкові числа колонці з правого боку змінюються на одну одиницю при кожному зміщенні вниз по таблиці.
- Тобто біт з правої сторони числа є **молодшим бітом (LSB)**, оскільки його зміна не набагато змінює загальне значення. Біт з лівого боку називається **старшим бітом (MSB)**, оскільки його зміна суттєво змінює загальне значення.
- Відповідно до традицій західного світу слова та речення читаються зліва направо.
- Дивлячись на ASCII код символу, спочатку необхідно читати старший біт, який знаходиться з лівого боку, **але при передачі даних прийнята інша угода - ПЕРШИМ передається молодший біт, який знаходиться з правого боку, а останнім передається старший біт.**
- Проте символи зазвичай посилаються у звичайній послідовності — у вигляді, як вони генеруються.
- Наприклад, якщо слово D-A-T-A передається, то символи передаються в цій же послідовності, але 7-розрядний ASCII код для кожного символу має «зворотний» порядок.

- Отже, послідовність бітів, що спостерігаються у каналі зв'язку, буде наступною (зчитування кожного біта в порядку справа наліво):



	Старший біт	Молодший біт
Код ASCII для «А»	1 0 0 0 0 0 1	

Додаючи до ASCII символу стоповий біт (1), біт парності (1 або 0) та стартовий біт (0), можна отримати показану знизу послідовність імпульсів (з перевіркою на парність).

Наприклад, ASCII код символу "A" буде передаватися у вигляді:

STOP	P	MSB	LSB	START
Стоповий	біт	старший	молодший	стартовий
Біт	парності	біт	біт	біт
1	0	1 0 0 0 0 0 0	1	0

Код Грея

- Двійковий код неідеальний для деяких типів пристроїв, оскільки при послідовному збільшенні рахунку має змінюватися кілька цифр.
- Код Грея можна використовувати для інкрементних пристроїв, таких як перетворювачі кутового положення валу.
- Перевагою цього коду в порівнянні з двійковим кодом є те, що зі зростанням значення (інкрементне прирощення) за один раз змінюється лише один біт.
- Код Грея показаний у табл

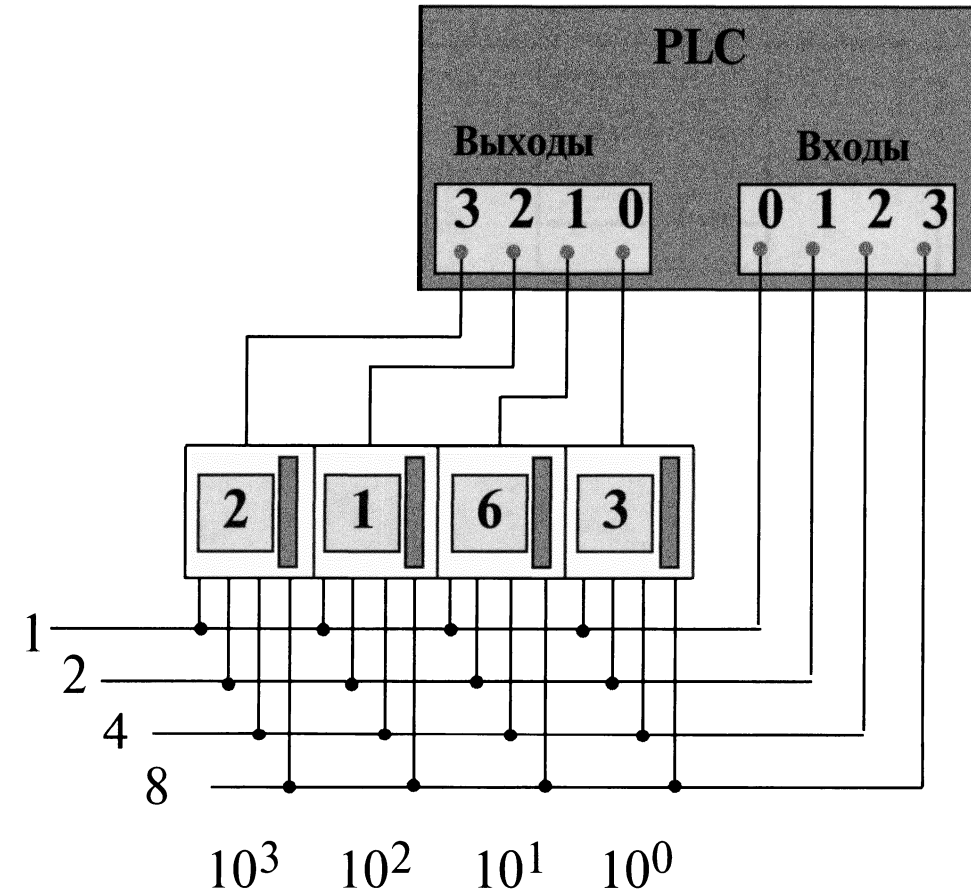
Десятиричне число	Код Грея
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1100
9	1101
10	1111
11	1110
12	1010
13	1011
14	1001
15	1000

Двійково-десятковий код

- Двійково-десятковий код (BCD) є розширенням 4-розрядного двійкового коду.
- При двійково-десятковому кодуванні кожна окрема цифра десяткового числа перетворюється на 4-розрядний двійковий код.
- Тобто для представлення однієї десятичної цифри двійково-десятковий код використовує чотири розряди.
- Хоча чотири розряди у двійковому коді можуть представляти 16 чисел (від 0 до 15), для двійково-десятькового кодування дійсні лише перші 10, від 0 до 9 (табл).

Десятиричне число	Двоичный код	Код Грея	Двоично-десятичный код
0	0000	0000	0000
1	0001	0001	0001
2	0010	0011	0010
3	0011	0010	0011
4	0100	0110	0100
5	0101	0111	0101
6	0110	0101	0110
7	0111	0100	0111
8	1000	1100	1000
9	1001	1101	1001
10	1010	1111	0001 0000
11	1011	1110	0001 0001
12	1100	1010	0001 0010
13	1101	1011	0001 0011
14	1110	1001	0001 0100
15	1111	1000	0001 0101

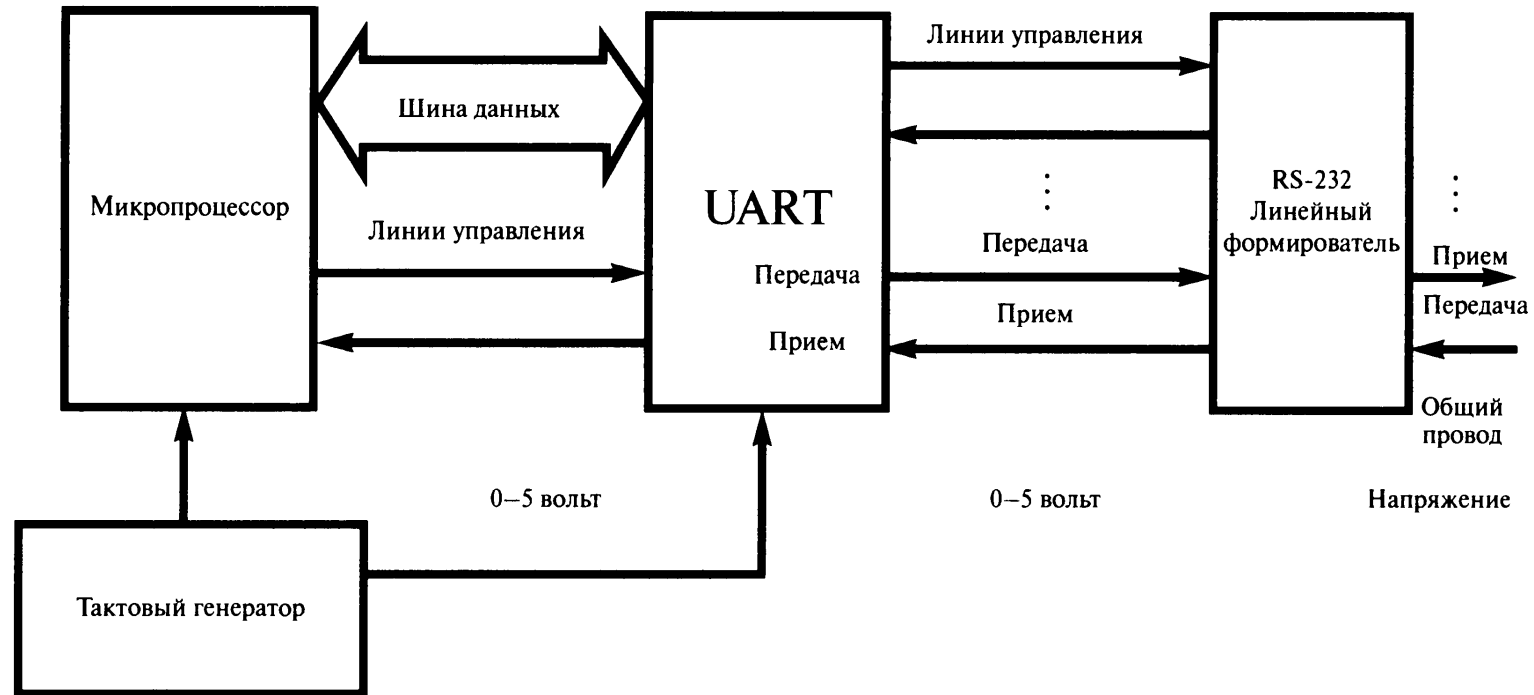
- Двійково-десятковий код зазвичай використовується у відносно простих системах, таких як невеликі вимірювальні прилади, координатні маніпулятори та цифрових панельних вимірювальних приладів.
- Для підключення двійково-десяткових компонентів до інших інтелектуальних пристроїв є спеціальні інтерфейсні плати та інтегральні мікросхеми (IC). Вони можуть підключатися безпосередньо до входів або виходів PLC.
- Типовим застосуванням двійково-десяткового коду є налаштування параметрів на панелі, що управляє, за допомогою групи перемикачів у вигляді коліщаток.
- Кожен перемикач представляє одну десяткову цифру (зліва направо: тисячі, сотні, десятки та одиниці). Інтерфейсне підключення кожної цифри до PLC вимагає чотирьох проводів плюс ще один провід для десяткової коми, що в цілому для чотирирозрядного набору перемикачів дає 20 проводів.
- Кількість проводів та їх підключень до PLC може бути зменшено до восьми, використовуючи розділену в часі мультиплексуючу систему, показану на рис
- Кожний вихід PLC підключається по черзі, і двійковий код зчитується PLC по чотирьох входах.
- Аналогічна система використовується (назад) для цифрового дисплея панельного вимірювального приладу, що використовує групу з чотирьох 7-сегментних рідкокристалічних або світлодіодних дисплеїв.



Універсальний асинхронний приймач-передавач (UART)

- Старт/стопові біти та біти парності, що використовуються в асинхронній системі передачі, зазвичай реально генеруються стандартною інтегральною мікросхемою, яка є частиною інтерфейсу між шиною мікропроцесора та лінійним формувачем (або приймачем) каналу зв'язку.
- Такий тип ІС називається UART (універсальний асинхронний приймач) або іноді ACE (адаптер асинхронного зв'язку).
- Різні форми UART використовуються також у синхронній передачі даних і називаються USRT (універсальний синхронний приймач). Разом всі ці ІС називаються USART (універсальний синхронно-асинхронний приймач).
- Вихід ІС UART не призначений безпосереднього підключення до каналу зв'язку.
- Щоб видавати і приймати рівні напруг, відповідні даному каналу зв'язку, використовуються додаткові пристрої, які називаються лінійними формувачами та лінійними приймачами.
- Прикладами UART є ІС 8250, 16450, 16550, а прикладом USART є ІС 8251.

Типове включення універсального асинхронного приймача-передавача (UART)



Головним завданням UART є догляд за всіма місцевими «рутинними» процедурами, пов'язаними з підготовкою 8-розрядного паралельного вихідного сигналу мікропроцесора до асинхронної передачі послідовної даних. Як імпульси синхронізації використовуються тактові імпульси мікропроцесора, що підключаються через зовнішній роз'єм.

При передачі UART виконує наступне:

- встановлює швидкість двійкової передачі;
- приймає біти символу від мікропроцесора у вигляді паралельної групи;
- генерує стартовий біт;
- додає біти даних у послідовну групу;
- визначає парність та додає біт парності (якщо необхідно);
- закінчує передачу стоповим бітом (іноді двома стоповими бітами);
- сигналізує процесору, що він готовий до наступного символу;
- у разі потреби координує квітування.

- IC UART має окрему сигнальну лінію для передачі (TX) та одну для прийому (RX), тому вона може працювати у повнодуплексному або напівдуплексному режимі.
- Інші лінії UART забезпечують апаратні сигнали квитування - методу забезпечення деякої форми взаємодії між пристроями, розташованими на двох кінцях каналу зв'язку.

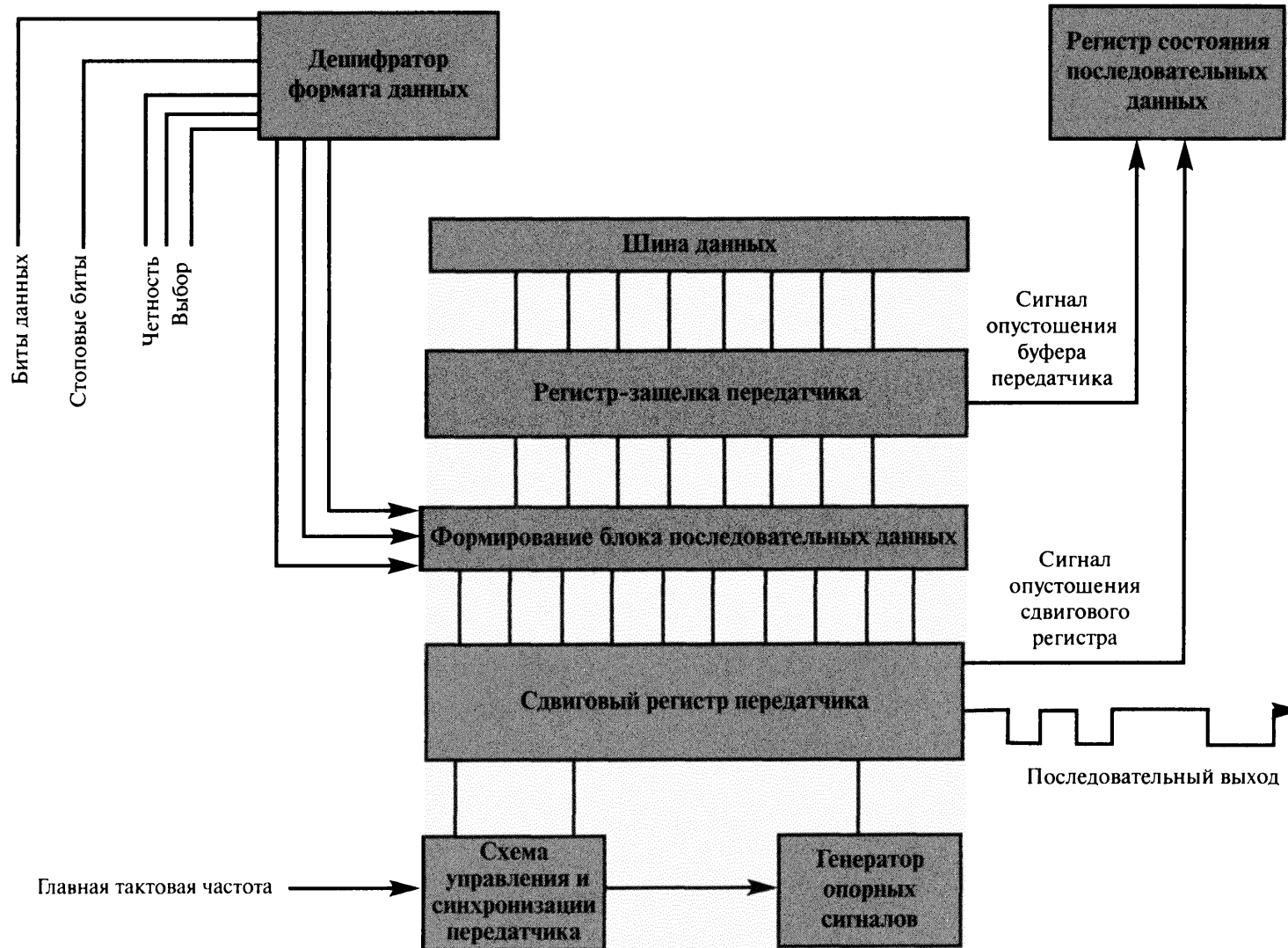
В процесі приймання UART виконує наступне:

- встановлює швидкість двійкового прийому;
 - розпізнає стартовий біт;
 - зчитує біти даних, що у вигляді послідовної групи бітів;
 - зчитує біт парності та проводить перевірку на парність;
 - розпізнає стоповий біт (біти);
 - передає символ у паралельному вигляді до мікропроцесора для подальших перетворень;
 - за необхідності координує квітування;
 - перевіряє дані на наявність помилки та при її виявленні помічає в регістрі стану біт помилки.
- ✓ Застосування UART усуває необхідність програмування описаних вище процедур у мікропроцесорі, натомість вони прозоро реалізуються UART.
- ✓ Програма з послідовними даними робить лише одне: вона просто записує/читає байти в/з UART.

Передавач UART

- Байт, прийнятий від мікропроцесора на передачу, записується за адресою вводу/виводу передавальної схеми UART.
- Біти, які необхідно передати, завантажуються в регістр зсуву, звідки вони потім виштовхуються негативними перепадами тактових імпульсів.
- Частота тактових імпульсів визначає швидкість передачі бітів.
- Коли всі біти будуть виштовхнуті зі зсувного регістру передавача, у нього завантажуються наступний пакет, і процес повторюється спочатку.
- Слово пакет в даному випадку використовується для позначення сукупності стартового біта, бітів інформації, що передається, біта парності і стопового біта. Деякі автори називають пакетом блок послідовних даних (SDU).
- Між регістром-засувкою передавача та зсувним регістром знаходиться схема, звана блоком формування послідовних даних (SDU).
- Ця схема збирає реальний пакет, який завантажуються у зсувний регістр.

Передавач UART

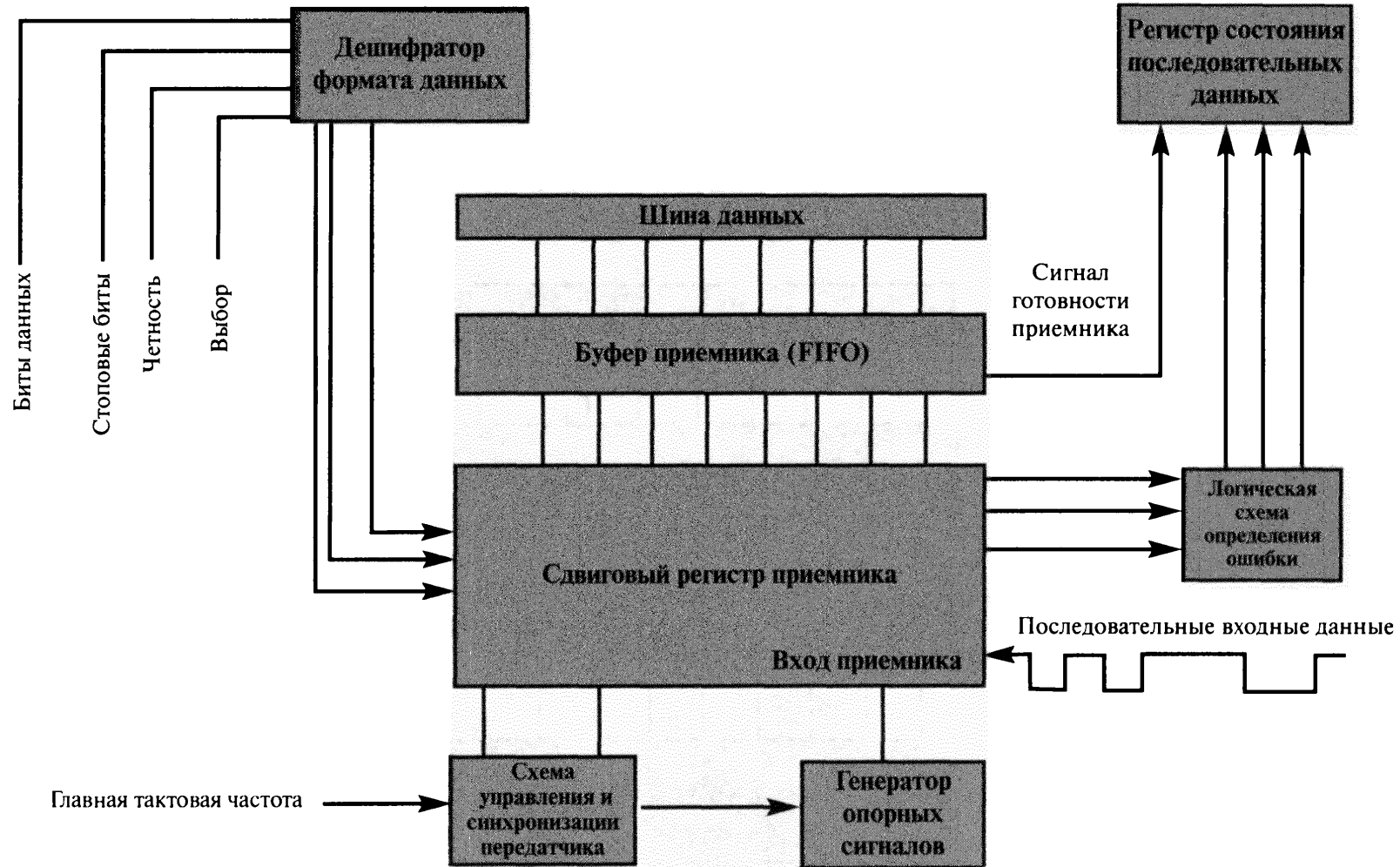


- У повнодуплексному режимі програмному забезпеченню необхідно лише перевіряти значення прапорця, який сигналізує про спустошення буфера передавача (TBE), щоб визначити необхідність запису наступного байта в UART.
- У напівдуплексному режимі модем повинен перемикатися між станами передавача та приймача, тому програма повинна перевіряти як буфер передавача, так і буфер приймача щодо наявності в них даних.

Приймач UART

- Приймальна схема UART безперервно відстежує вхідну послідовну лінію, очікуючи на прихід стартового біта.
- При появі стартового біта лінія приймача контролюється з обраною швидкістю передачі бітів, і послідовні біти поміщаються в регістр зсуву приймача.
- Це відбувається відповідно до формату, описаного за допомогою регістру формату даних, програмованого користувачем.
- Після того, як весь байт буде отримано, він переміщається в буфер типу FIFO («першим прийшов, першим пішов»).
- На цьому етапі прапорець RxRDY (готовність приймача) переводиться в стан істини і залишається таким, доки буфер FIFO не спорожніє.

Приймач UART



Типи помилок

- Інший головною функцією UART є виявлення помилок у даних, що приймаються.
- Більшість помилок є помилками прийому.
- Зустрічаються такі типові помилки:
 - Переповнення приймача: байти приймаються швидше, ніж можуть бути прочитані.
 - Помилка парності: невідповідність біта парності.
 - Помилка кадру: це відбувається, якщо лічені біти не відповідають вибраному формату.
 - Помилка розриву: це відбувається в тому випадку, якщо стартовий біт діє протягом більшого часу кадру.

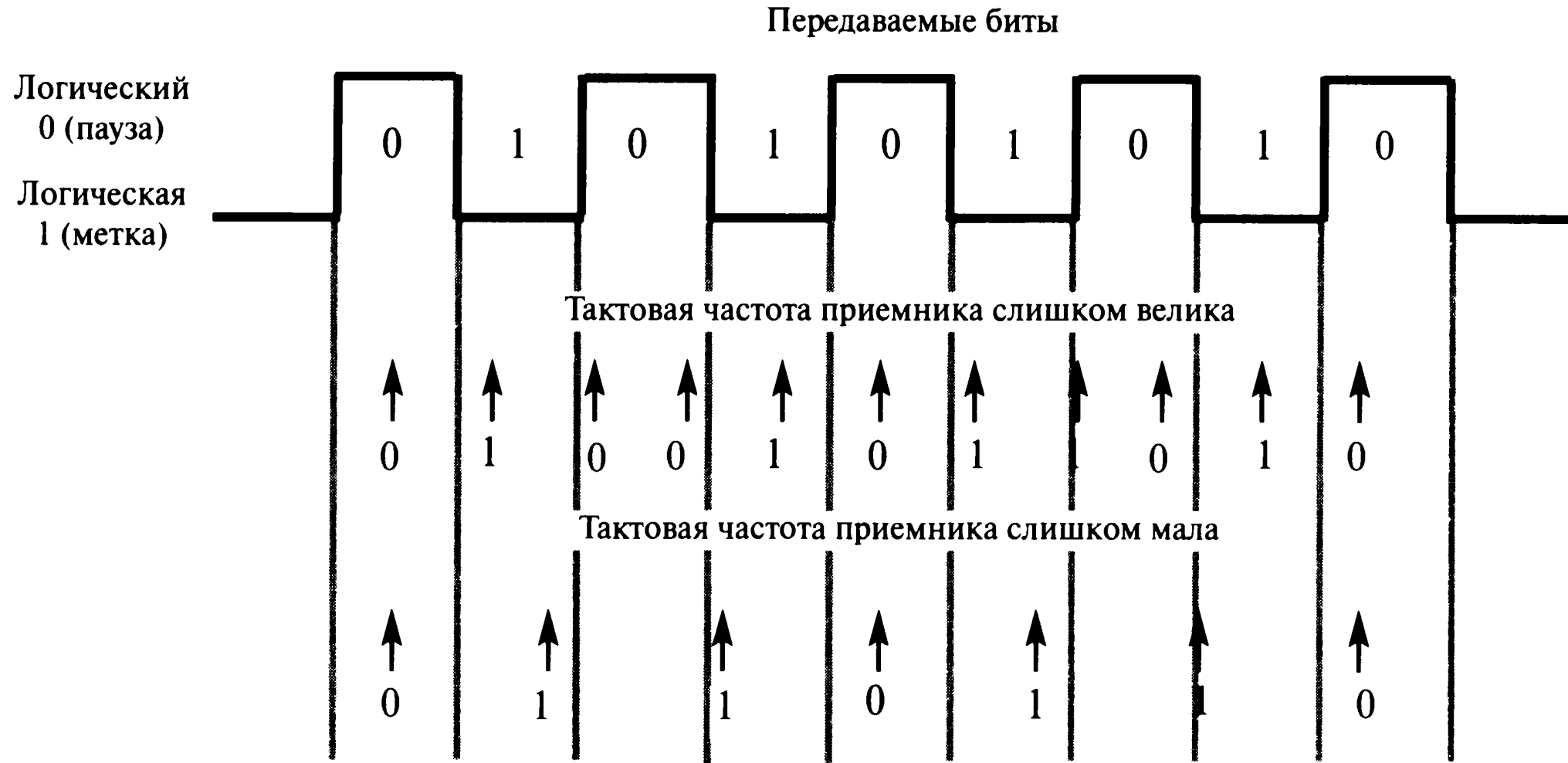
Виявлення розриву

- Щоб заволодіти увагою приймача, передавач може утримувати лінію даних у стані паузи (позитивна напруга) протягом більш тривалого часу, ніж необхідно для передачі цілого символу.
- Цей стан називається розривом і деякі приймачі можуть бути забезпечені схемою виявлення розриву.
- Ця умова корисна для переривання роботи приймача, що можна робити навіть серед переданого потоку знаків.
- Час виявлення розриву залежить від швидкості передачі.
- Наявність помилки створення послідовності даних передається в регістр стану даних, показаний на рисунках попередніх слайдів .

Синхронізація роботи приймача

- Для внутрішніх операцій UART, а також управління операціями зсуву в схемах передавача і приймача необхідно мати окремий тактовий сигнал.
- Частота головного тактового сигналу зазвичай набагато більша, ніж швидкість послідовної передачі.
- Відношення частоти головного тактового сигналу до частоти передачі бітів називається коефіцієнтом тактування (зазвичай він дорівнює 16).
- Замість вибірки вхідної лінії з частотою послідовної передачі покращений детектор стартового біта здійснює вибірку приймальної лінії з частотою головного тактового сигналу. Це мінімізує ймовірність появи помилки, викликані розбіжністю вибірки з бітами, що передаються, що призводить до вибірки неправильних бітів.
- Перші послідовні порти використовували IC 8250 або 8251, які переривали роботу головного процесора для прийому або передачі кожного символу.
- Такий спосіб добре працював для швидкостей на той час.
- Потім ці IC були замінені на IC 16450, які працюють аналогічно, але підтримують швидші швидкості шини ПК, а ще пізніше IC 16550, яка має 16-байтовий буфер і, отже, зменшує кількість переривань ЦП на коефіцієнт рівний 16.
- Пізніші розробки використовують покращений послідовний порт, який має буфер ємністю близько 1000 байт і свій власний процесор, що зменшує кількість переривань основного ЦП у 1000 разів.

Приклад неточної синхронізації джерела і приймача



Мінімізація похибок з використанням коефіцієнта тактування 16



Швидкодіючий UART (16550)

- IC 16550 є швидкодіючим послідовним асинхронним приймачем (UART).
- Ця IC використовується практично на всіх IBM-сумісних комп'ютерах і портах COM, що продаються в даний час.
- Вона дуже відрізняється від старого UART 8259 двома характеристиками – швидкістю та розміром FIFO буфера.
- Перевагою IC 16550 над старішими UART 16450 та 8250 є його 16-байтний буфер.

Швидкість

- IC 16550 може працювати із швидкостями від 1 до 115 кбод. Вона зазвичай використовується для інтерфейсу RS-232, хоча стандарт RS-232 допускає швидкість до 19,2 кбод.
- Через доступність і низьку вартість IC 16550 виробники комп'ютерів та додаткових портів COM використовують IC 16550 як стандартне обладнання.

Буфер FIFO

- Старий UART типу 8250 (швидкість 19,2 кбод) мав лише одnobайтний буфер FIFO.

Шістнадцятибайтний буфер IC 16550 має

дві переваги:

- IC 16550 робить швидкодіючі комунікації надійнішими. На старих IC, що мають однобайтний буфер, UART міг втратити дані, якщо другий байт приходив на UART до того, як ЦП витягував перший байт. IC 16550, маючи 16-байтний буфер, надає ЦП 16 шансів витягти дані, перш ніж символ буде втрачено. Наступне допоможе зрозуміти це значення.
- Коли UART працює зі швидкістю 19200 біт/с (з кадром, що містить 10-бітовий символ), то ЦП повинен обслуговувати COM-порт 1920 разів на кожну секунду, або кожні 0,0005 секунди. Якщо трапиться так, що ЦП знадобиться 0,006 секунди, щоб обслужити COM-порт, то в однобайтному буфері UART перший байт буде втрачено. При використанні IC 16550, яка має 16-байтний буфер, обслуговування COM-порту тепер можна проводити через 0,008 секунд.
- Нова IC допомагає реалізувати багатозадачну систему ефективніше.
- Коли COM-порт передає дані, він повинен переривати ЦП і заповнювати буфер передавача UART.
- Це означає, що якщо ЦП виробляє фонове сканування каталогу, то сканування піде більше часу, поки COM-порт намагається надіслати дані зовнішньої системі.
- У UART з однобайтним буфером при швидкості 19200 біт/с COM-порт повинен переривати ЦП 1920 разів кожну секунду, щоб передати дані із COM-порту.
- Однак при використанні IC 16550 він може помістити до 16 байтів даних у буфер за один раз і, отже, переривати ЦП лише 120 разів на секунду.
- Це збільшує продуктивність ЦП під час роботи з COM-портом.