МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ПОЛІТЕХНІКА»

кафедра Комп'ютерних Інтелектуальних Систем та Мереж (КІСМ)

Р. Шапорін, І. Мілейко, Є.Шендрик, І. Копитчук, В. Шапорін, О. Шапоріна

ПРОМИСЛОВІ МЕРЕЖІ

Навчальний посібник

Одеса - 2023

Шапорін Р.О. та ін.
Промислові мережі: навч. посібник / уклад.: Р.О. Шапорін, І.Г. Мілейко, Є.В. Шендрик, І.М. Копитчук, В.О. Шапорін, О.Л. Шапоріна – Одеса: Нац. ун-т "Одес. політехніка, 2023. – 307 с

Рекомендовано Вченою Радою
Інституту комп'ютерних систем
Національного університету «Одеська політехніка»
протокол № 6 від 17 травня 2023 р.

Навчальний посібник містить інформацію щодо роботи з сучасним обладнанням промислових мереж, системами і комплексами, які використовуються у промисловості, системах зв'язку, телекомунікаційних системах та формує у здобувачів вищої освіти інтелектуальні вміння, які дозволяють вирішувати різноманітні задачі з дослідження та розробки інформаційно-вимірювальних та інфокомунікаційних систем зі застосуванням сучасного устаткування у галузі промислових мереж, методик та принципів структурно-алгоритмічного синтезу систем промислової автоматики.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

PNO (Profibus Network Organisation) - Організація мережіProfibus

DDLM (Direct Data Link Mapper) - Прямий перетворювач дляканального рівня

DP (Decentralized Peripherals) — Децентралізована периферіяPA (Process Automation) — Автоматизація процесів

LAS (List of Active Station) - Список активних станційTTR (Time Target Rotation) - Тимчасове обертання

HSA (Highest Station Address) - Адреса найвищої станції

HMI (Human Machine Interface) - Дюдино-машинний інтерфейс

SSAP (Source Service Access Point) - Точка доступу до початкової службі

DSAP (Destination Service Access Point) - точка доступу приймачаNRZ-коде (Non Return to Zero) - не повернення до нуля

OLM (Optical Link Module) - Модуль оптичного зв'язку

OLP (Optical Link Plug) - Оптичний з'єднувач

FDL (Fieldbus Data Link) – посилання на польову шину

OPC – OLE for Process Control

OPC DA –OLE for Process Control Data Access

OPC HDA – OLE for Process Control Historical Data Access

OPC UA – OLE for Process Control Unified Architecture

РОЗДІЛ 1
БАГАТОРІВНЕВИЙ ПІДХІД. МОДЕЛЬ OSI

Теза щодо користі стандартизації, що справедлива для всіх галузей, в комп'ютерних та промислових мережах набуває особливого значення. Суть мережі – це з'єднання різного обладнання, тобто проблема сумісності є однією з найбільш гострих. Без дотримання всіма виробниками загальноприйнятих правил розробки обладнання прогрес у «будівництві» мереж був би неможливий. Тому розвиток комп'ютерної галузі, що відображений в стандартах, - будь-яка нова технологія тільки тоді набуває «законного» статусу, коли її зміст закріплюється у відповідному стандарті.

У комп'ютерних та промислових мережах ідеологічною основою стандартизації є багаторівневий підхід до розробки засобів мережевої взаємодії. Саме на основі цього підходу була створена стандартна модель взаємодії відкритих систем (сім рівнів), що стала свого роду універсальною мовою мережевих фахівців.

1.1. Багаторівневий підхід

1.1.1. Декомпозиція задачі мережевої взаємодії

Організація взаємодії між пристроями мережі є складним завданням. Для вирішення складних завдань використовується універсальний прийом – декомпозиція, тобто розбиття однієї задачі на кілька задач-модулів (див. рис. 1.1).

Декомпозиція полягає в чіткому визначенні функцій кожного модуля, а також порядку їх взаємодії (інтерфейсів). Отже досягається логічне спрощення задачі, а, крім того, з'являється можливість модифікації окремих модулів без зміни іншої частини системи.

Під час декомпозиції зазвичай використовують багаторівневий підхід:

– всю безліч модулів, що вирішують часткові завдання, розбивають на групи і впорядковують за рівнями, що створює ієрархію;

– відповідно до принципу ієрархії для кожного проміжного рівня можна вказати безпосередньо сусідні рівні, що розміщено вище, та ніжні рівні, які примикають до нього (див. рис. 1.2);

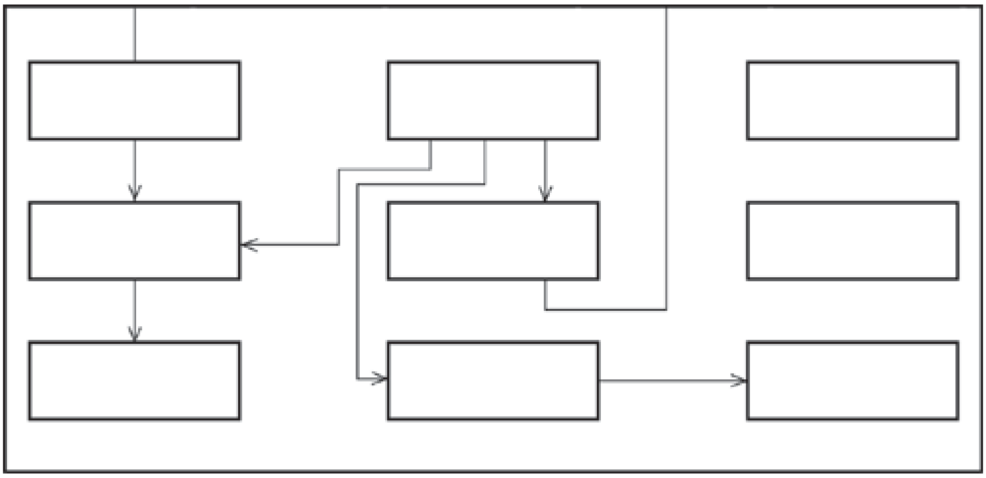


Рисунок 1.1 – Приклад декомпозиції задачі

– група модулів, що складають кожний рівень, повинна бути сформована таким чином, щоб усі модулі цієї групи для виконання своїх завдань зверталися із запитами тільки до модулів сусіднього нижчого рівня;

– з іншого боку, результати роботи всіх модулів, віднесених до деякого рівня, можуть бути передані тільки модулям сусіднього вищого рівня.

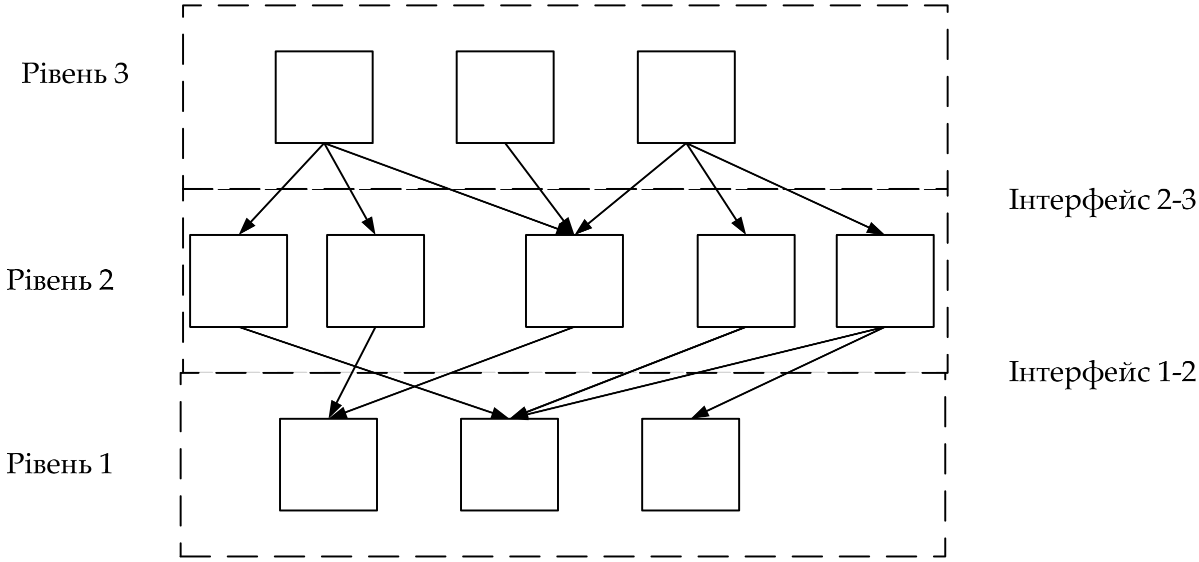


Рисунок 1.2 – Багаторівневий підхід – створення ієрархії задач

Така ієрархічна декомпозиція задачі передбачає чітке визначення функції кожного рівня та інтерфейсів між рівнями. Інтерфейс визначає набір функцій, які рівень, що розміщений нижче, надає розміщеному вище. Під час ієрархічної декомпозиції досягається відносна незалежність рівнів, а значить, можливість їх автономної розробки та модифікації.

Засоби вирішення задачі організації мережевої взаємодії зазвичай теж можуть бути представлені у вигляді ієрархічно організованої безлічі модулів. Наприклад, модулю нижнього рівня можна доручити питання, що пов'язані з надійною передачею інформації між двома сусідніми вузлами, а наступному модулю, більш високого рівня – транспортування повідомлень у межах усієї мережі. Останнє завдання – організація зв'язку двох будь-яких, не обов'язково сусідніх, вузлів – є більш загальною і тому її можна вирішити за допомогою багаторазових звернень до нижчого рівня.

Зв'язок вузлів A і B (див. рис. 1.3) можна звести до послідовного зв'язку пар проміжних суміжних вузлів. Таким чином, модулі вищого рівня під час вирішення своїх завдань розглядають засоби нижчого рівня, як інструмент.

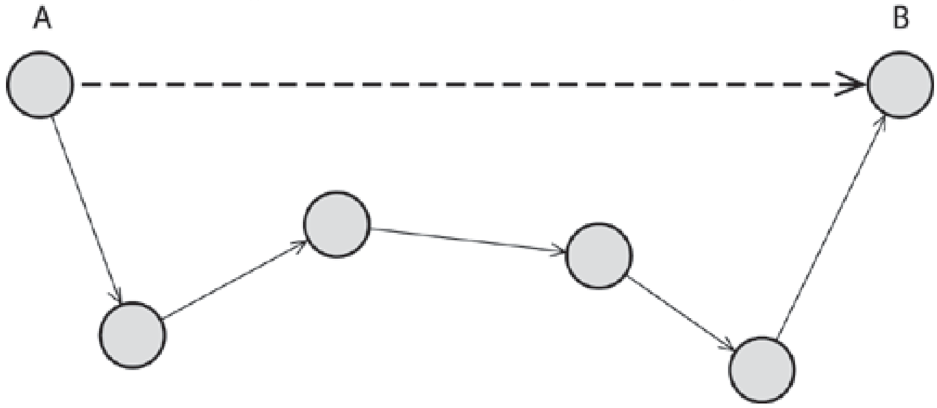


Рисунок 1.3 – Декомпозиція задачі зв'язку довільної пари вузлів на більш часткові задачі зв'язування пар сусідніх вузлів

1.1.2. Протокол. Інтерфейс. Стек протоколів

Багаторівневі представлення засобів мережевої взаємодії мають свою специфіку, яка пов'язана з тим, що під час обміну повідомленнями беруть участь дві сторони. Тобто в даному випадку необхідно організувати узгоджену роботу двох «ієрархій», що працюють на різних комп'ютерах. Обидва учасники мережевого обміну повинні прийняти безліч угод. Наприклад, вони повинні узгодити рівні та форму електричних сигналів, спосіб визначення довжини повідомлень, домовитися щодо методів контролю достовірності та т.п. Іншими словами, угоди повинні бути прийняті для всіх рівнів, починаючи від найнижчого рівня передачі бітів – до найвищого, що реалізовує сервіс для користувачів мережі.

На рис. 1.4 показана модель взаємодії двох вузлів. З кожної сторони засобу взаємодії представлені чотирма рівнями. Процедура взаємодії цих двох вузлів може бути описана у вигляді набору правил взаємодії кожної пари відповідних рівнів обох сторін-учасниць.

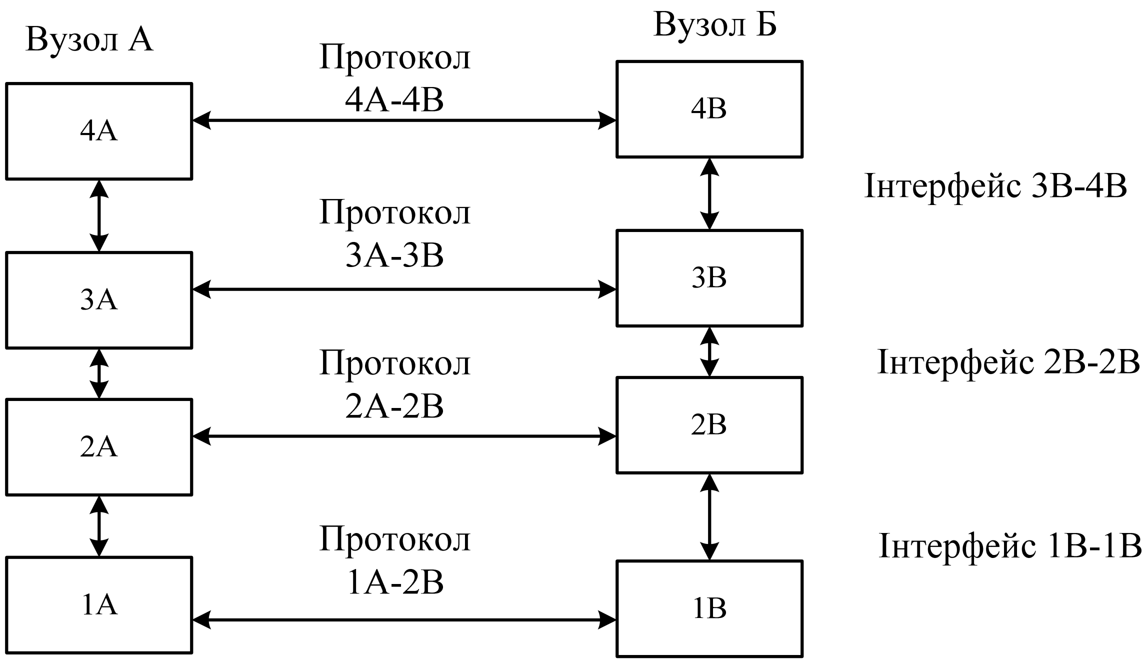


Рисунок 1.4 – Взаємодія двох вузлів

Формалізовані правила, що визначають послідовність і формат повідомлень, якими обмінюються мережеві компоненти, лежать на одному рівні, але в різних вузлах, називаються ***протоколом***.

Модулі, що реалізують протоколи сусідніх рівнів і, що знаходяться в одному вузлі, також взаємодіють один з іншим відповідно за чітко встановленими правилами за допомогою стандартизованих форматів повідомлень. Ці правила прийнято називати ***інтерфейсом***.

Інтерфейс визначає послідовність і формат повідомлень, якими обмінюються мережеві компоненти, що знаходяться на сусідніх рівнях у одному вузлі. Інтерфейс визначає набір послуг, що надається цим рівнем сусідньому рівню.

Протокол та інтерфейс виражають одне й те саме поняття, але традиційно в мережах за ними закріплені різні області дії: протоколи визначають правила взаємодії модулів одного рівня в різних вузлах, а інтерфейси – модулів сусідніх рівнів в одному вузлі.

Засоби кожного рівня повинні відпрацьовувати, по-перше, власний протокол, а по-друге, інтерфейси з сусідніми рівнями.

Ієрархічно організований набір протоколів, що достатній для організації взаємодії вузлів в мережі, називається ***стеком комунікаційних протоколів***.

Комунікаційні протоколи можуть бути реалізовані як програмно, так і апаратно. Протоколи нижніх рівнів зазвичай реалізуються комбінацією програмних і апаратних засобів, а протоколи верхніх рівнів – програмними засобами.

Програмний модуль, що реалізує деякий протокол, зазвичай скорочено також називають протоколом. При цьому, співвідношення між протоколом як формально певною процедурою і протоколом – програмним модулем, що реалізує цю процедуру, – аналогічно співвідношенню між алгоритмом розв'язання деякої задачі та програмою, що вирішує цю задачу.

Один і той самий алгоритм може бути запрограмований з різним ступенем ефективності. Так само і протокол може мати декілька програмних реалізацій. Саме тому під час порівняння протоколів потрібно враховувати не тільки логіку їх роботи, але і якість програмних рішень. Більш того, на ефективність взаємодії пристроїв у мережі впливає якість усієї сукупності протоколів, що складають стек. Зокрема, те, наскільки раціонально розподілені функції між протоколами різних рівнів і наскільки добре визначені інтерфейси між ними.

Протоколи реалізуються не тільки комп'ютерами, але й іншими мережевими пристроями – ***концентраторами, мостами, комутаторами, маршрутизаторами*** і т.д. У загальному випадку зв'язок комп'ютерів у мережі здійснюється не безпосередньо, а через різні комунікаційні пристрої. Залежно від типу пристрою в ньому повинні бути вбудовані засоби, що реалізують той чи інший набір протоколів.

Щоб ще раз пояснити поняття «протокол» та «інтерфейс», розглянемо приклад, який не має відношення до обчислювальних мереж, а саме, обговоримо взаємодію двох підприємств, **А** і **В**. Між цими підприємствами існують численні домовленості та угоди, наприклад щодо регулярних поставок продукції. Відповідно до домовленості, начальник відділу продажів підприємства **А** регулярно на початку кожного місяця посилає офіційне повідомлення начальнику відділу закупівель підприємства **В** про те, скільки якого товару може бути поставлено в цьому місяці. У відповідь на це повідомлення начальник відділу закупівлі підприємства **В** посилає заявку встановленого зразка на потрібну кількість продукції. Можливо, подібна процедура включає додаткові узгодження; в будь-якому випадку, існує встановлений порядок взаємодії, який можна вважати «протоколом рівня начальників». Начальники посилають свої повідомлення і заявки через секретарів. Порядок взаємодії начальника і секретаря відповідає поняттю міжрівневого інтерфейсу «начальник - секретар». На підприємстві **А** обмін документами між начальником і секретарем проходить крізь спеціальну папку, а на підприємстві **В** начальник спілкується з секретарем по факсу. Таким чином, інтерфейси «начальник - секретар» на цих двох підприємствах відрізняються.

Після того як повідомлення передані секретарям, начальників не хвилює, яким чином ці повідомлення будуть переміщатися далі – звичайною поштою або електронною, факсом або іншим способом. Вибір способу передачі – це рівень компетенції секретарів, вони можуть вирішувати це питання, не повідомляючи про те своїх начальників, так як їх протокол взаємодії пов'язаний тільки з передачею повідомлень, що надходять зверху, і не стосується змісту цих повідомлень. На рис. 1.5 показано, що як протоколом взаємодії «секретар – секретар» використовується обмін листами.

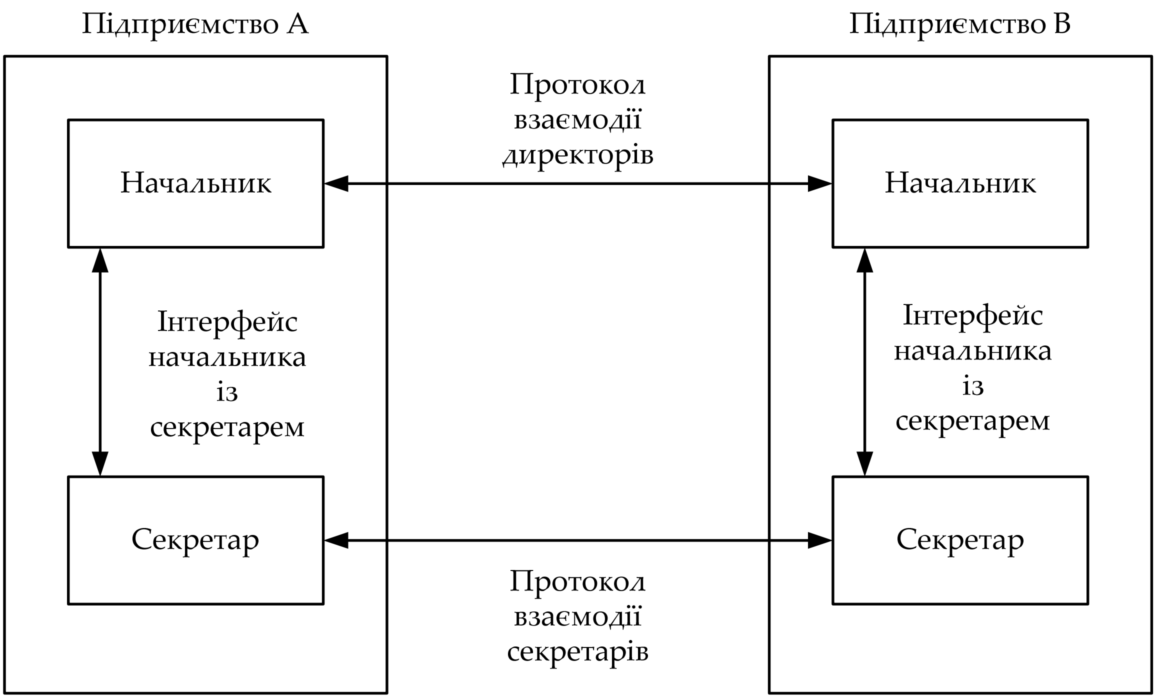


Рисунок 1.5 – Приклад багаторівневої взаємодії підприємств

При вирішенні інших питань начальники можуть взаємодіяти з іншими правилами-протоколам, але це не вплине на роботу секретарів, для яких не важливо, які повідомлення відправляти, а важливо, щоб вони дійшли до адресата. Отже, в цьому випадку маємо справу з двома рівнями – рівнем начальників і рівнем секретарів, і кожен з них має власний протокол, який може бути змінений незалежно від протоколу іншого рівня. У цій незалежності протоколів один від одного і полягає перевага багаторівневого підходу.

1.2. Модель OSI

1.2.1. Загальна характеристика моделі OSI

З того, що протокол є угодою, яка прийнята двома взаємодіючими об'єктами, у цьому випадку двома працюючими в мережі комп'ютерами, зовсім не випливає, що він обов'язково був стандартним. Але на практиці під час реалізації мереж зазвичай використовуються стандартні протоколи. Вони можуть бути фірмові, національні або міжнародні стандарти.

На початку 80-х років ряд міжнародних організацій по стандартизації – ISO, ITU-T і деякі інші – розробили модель, яка зіграла значну роль у розвитку мереж. Ця модель називається моделлю ISO/OSI.

Модель взаємодії відкритих систем (Open System Interconnection, OSI) визначає різні рівні взаємодії систем у мережах з комутацією пакетів, дає їм стандартні імена та вказує, які функції повинен виконувати кожен рівень. Модель OSI була розроблена на підставі великого досвіду, який отримано під час створення глобальних комп'ютерних мереж у 70-і роки. Повний опис цієї моделі займає більше 1000 сторінок тексту.

У моделі OSI (див. рис. 1.6) засоби взаємодії поділяють на сім рівнів:

– прикладний,
– представницький,
– сеансовий,
– транспортний,
– мережевий,
– канальний
– фізичний.

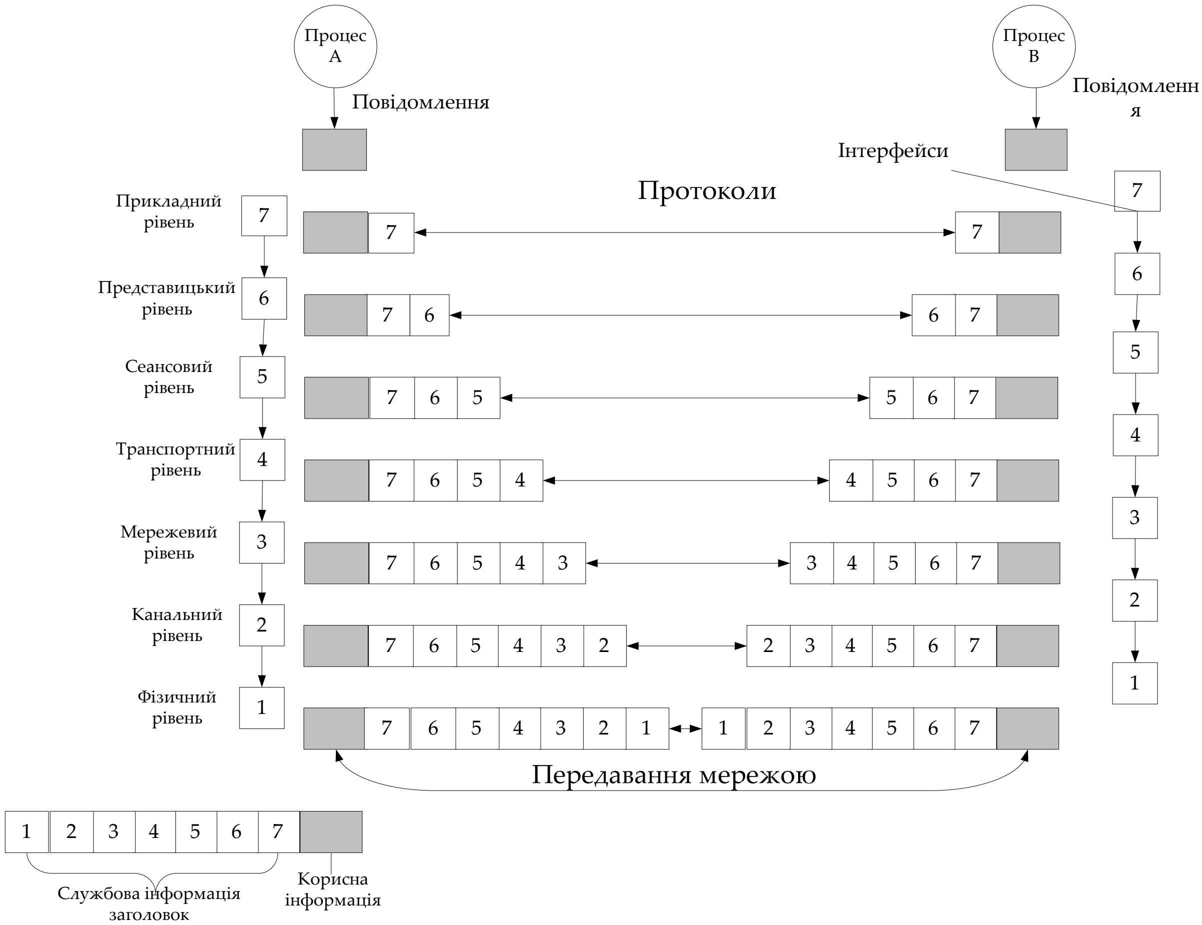


Рисунок 1.6 – Модель взаємодії відкритих систем ISO/OSI

Кожен рівень має справу з певним аспектом взаємодії мережевих пристроїв. Модель OSI описує тільки системні засоби взаємодії, що реалізуються операційною системою, системними утилітами та апаратними засобами. Модель не включає засоби взаємодії додатків кінцевих користувачів. Власні протоколи взаємодії додатку реалізують під час звернень до системних засобів. Тому необхідно розрізняти рівень взаємодії додатків і прикладний рівень.

Додаток може взяти на себе функції деяких верхніх рівнів моделі OSI. Наприклад, деякі СУБД мають вбудовані засоби віддаленого доступу до файлів. У цьому випадку додаток, який виконує доступ до віддалених ресурсів, не використовує системну файлову службу; він обходить верхні рівні моделі OSI і звертається безпосередньо до системних засобів, які відповідають за транспортування повідомлень мережею, що розташовуються на нижніх рівнях моделі OSI. Нехай додаток звертається із запитом до прикладного рівня, наприклад до файлової служби. На підставі цього запиту програмне забезпечення прикладного рівня формує повідомлення стандартного формату. Звичайне повідомлення складається зі заголовка і поля даних. Повідомлення містить службову інформацію, яку необхідно передати через мережу прикладному рівню машини-адресата, щоб повідомити йому, яку роботу треба виконати.

Заголовок повинен містити інформацію щодо місцезнаходження файлу і тип операції, яку необхідно виконати. Поле даних повідомлення може бути порожнім або містити будь-які дані, наприклад ті, що необхідно записати до віддаленого файлу. Але для того щоб доставити цю інформацію за призначенням, необхідно вирішити ще багато завдань, відповідальність за які несуть рівні, що розміщено нижче.

Після формування повідомлення прикладний рівень направляє його вниз за стеком до представницького рівня. Протокол представницького рівня на підставі інформації, яку отримано із заголовка прикладного рівня, виконує необхідні дії й додає до повідомлення власну службову інформацію – заголовок представницького рівня, у якому містяться вказівки для протоколу представницького рівня машини-адресата. Отримане повідомлення передається вниз сеансовому рівню, який у свою чергу додає свій заголовок, і т.д. Деякі протоколи розміщують службову інформацію не тільки на початку повідомлення у вигляді заголовка, але і наприкінці, у вигляді так званого «кінцевіка». Нарешті, повідомлення досягає нижнього фізичного рівня, який і передає його за лініями зв'язку машині-адресату. До цього моменту повідомлення «обростає» заголовками всіх рівнів (див. рис. 1.7).

Коли повідомлення за мережею надходить до машини-адресата, воно приймається її фізичним рівнем і послідовно переміщується вгору з рівня на рівень. Кожен рівень аналізує та обробляє заголовок свого рівня, виконуючи відповідні цьому рівню функції, а потім видаляє цей заголовок і передає повідомлення рівню.

Поряд з терміном повідомлення (***message***) існують й інші терміни, що застосовуються мережевими фахівцями для позначення одиниць даних у процедурах обміну. У стандартах ISO для позначення одиниць даних, з якими мають справу протоколи різних рівнів, використовується загальна назва протокольний блок даних (Protocol Data Unit, PDU). Для позначення блоків даних певних рівнів зазвичай використовуються спеціальні назви: кадр (***frame***), пакет (***packet***) , дейтаграма (***datagram***), сегмент (***segment***).

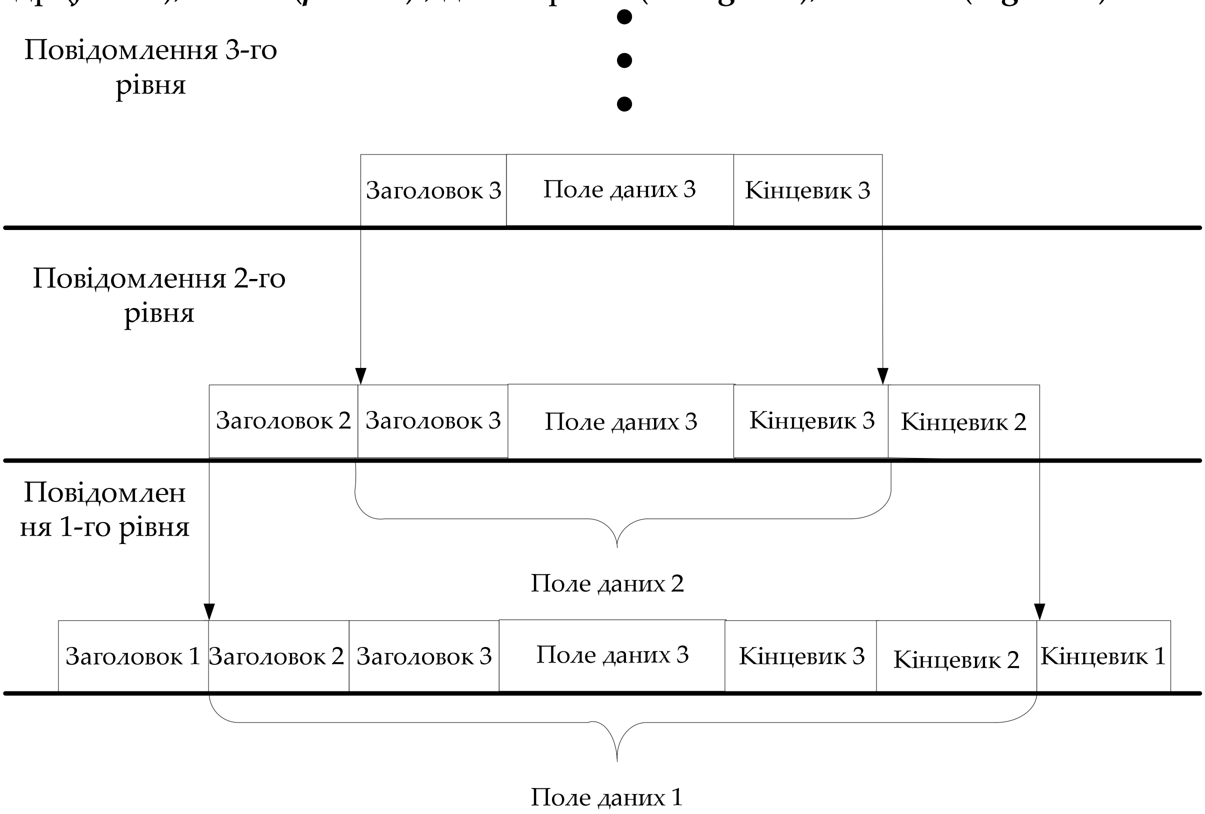


Рисунок 1.7 – Вкладеність повідомлень різних рівнів

1.2.2. Фізичний рівень

Фізичний рівень (Physical layer) має справу з передачею бітів по фізичним каналам зв'язку, таким, як коаксіальний кабель, кручена пара, оптоволоконний кабель або цифровий територіальний канал. До цього рівня мають відношення характеристики фізичних середовищ передачі даних, такі як смуга пропускання, перешкодозахищеність, хвильовий опір та інші. На цьому ж рівні визначаються характеристики електричних сигналів, що передають дискретну інформацію, таку як крутизна фронтів імпульсів, рівні напруги або струму сигналу, що передається, тип кодування, швидкість передачі сигналів. Крім того, тут стандартизуються типи з'єднувачів і призначення кожного контакту.

Фізичний рівень реалізується апаратно та включає:

– передачу бітів по фізичним каналам;

– формування електричних сигналів;

– кодування інформації;
– синхронізацію;
– модуляцію.

Функції фізичного рівня реалізуються у всіх пристроях, які підключені до мережі. З боку комп'ютера функції фізичного рівня виконуються мережевим адаптером або послідовним портом.

Прикладом протоколу фізичного рівня може служити специфікація 10Base-T технології Ethernet, яка використовує як кабель неекрановану кручену пару категорії 3 з хвилевим опором 100 Ом, з'єднувач RJ-45, максимальну довжину фізичного сегмента 100 метрів, манчестерський код для представлення даних у кабелі, а також деякі інші характеристики середовища та електричних сигналів.

1.2.3. Канальний рівень

На фізичному рівні пересилаються біти. При цьому, не враховується, що в тих мережах, в яких лінії зв'язку використовуються (розділяються) пари взаємодіючих комп'ютерів, фізичне середовище передачі може бути зайняте. Тому одним із завдань канального рівня (Data Link layer) є перевірка доступності середовища передачі. Інше завдання канального рівня – реалізація механізмів виявлення та корекції помилок. Для цього на канальному рівні біти групуються у набори, які називаються кадрами (frames). Канальний рівень забезпечує коректність передачі кожного кадру. Рівень поміщає спеціальну послідовність біт у початок і кінець кожного кадру, для його виділення, а також розраховує контрольну суму, обробляючи всі байти кадру певним способом, і додає контрольну суму до кадру. Коли кадр приходить мережею, одержувач знову розраховує контрольну суму отриманих даних і порівнює результат з контрольною сумою з кадру. Якщо вони збігаються, кадр вважається правильним і приймається. Якщо ж контрольні суми не збігаються, то фіксується помилка. Канальний рівень може не тільки виявляти помилки, але і виправляти їх завдяки повторної передачі пошкоджених кадрів. Необхідно відзначити, що функція виправлення помилок для канального рівня не є обов'язковою, тому в деяких протоколах цього рівня вона відсутня, наприклад у Ethernet та Frame Relay.

Функції канального рівня реалізують програмно-апаратно:

1. Надійну доставку пакета:
   – між двома сусідніми станціями в мережі з довільною топологією;

– між будь-якими станціями в мережі з типовою топологією.

2. Перевірку доступності середовища, що розділяється.
3. Виділення кадрів з потоку даних, що надходять мережею.
4. Формування кадрів при відправці даних.
5. Підрахунок і перевірку контрольної суми.

У протоколах канального рівня, що використовуються в локальних мережах, закладена певна структура зв'язків між комп'ютерами і способи їх адресації. Хоча канальний рівень і забезпечує доставку кадру між будь-якими двома вузлами локальної мережі, він це робить тільки у мережі з певною топологією зв'язків, саме тією топологією, для якої він був розроблений. До таких типових топологій, що підтримуються протоколами канального рівня локальних мереж, відносяться «загальна шина», «кільце» і «зірка», а також структури, отримані з них за допомогою мостів і комутаторів. Прикладами протоколів канального рівня є протоколи Ethernet, Token Ring, FDDI, 100VG-AnyLAN.

У локальних мережах протоколи канального рівня використовуються комп'ютерами, мостами, комутаторами і маршрутизаторами. У комп'ютерах функції канального рівня реалізуються спільними зусиллями мережевих адаптерів та їх драйверів.

У глобальних мережах, які рідко мають регулярну топологію, канальний рівень зазвичай забезпечує обмін повідомленнями тільки між двома сусідніми комп'ютерами, які з’єднано індивідуальною лінією зв'язку. Прикладами протоколів «точка-точка» (як зазвичай називають такі протоколи) можуть служити широко поширені протоколи PPP і LAP-B. У таких випадках для доставки повідомлень між кінцевими вузлами через усю мережу використовуються засоби мережевого рівня. Саме так організовані мережі X.25. Іноді в глобальних мережах функції канального рівня в явному вигляді виділити важко, так як в одному і тому ж протоколі вони об'єднуються з функціями мережевого рівня. Прикладами такого підходу можуть служити протоколи технологій ATM і Frame Relay.

Канальний рівень є доволі потужний набір функцій щодо пересилки повідомлень між вузлами мережі. У деяких випадках протоколи канального рівня виявляються самодостатніми транспортними засобами, і тоді поверх них можуть працювати безпосередньо прикладний рівень або додатки, без залучення засобів мережевого і транспортного рівнів. Наприклад, існує реалізація протоколу управління мережею SNMP, безпосередньо без

Ethernet, хоч стандартно цей протокол працює на основі мережевого протоколу IP і транспортного протоколу UDP. Природно, що застосування такої реалізації буде обмеженим – вона не підходить для складних мереж різних технологій, наприклад Ethernet і X.25, і навіть для такої мережі, в якій у всіх сегментах застосовується Ethernet, але між сегментами існують зв'язки у вигляді петлі. У двохсегментній мережі Ethernet, що об'єднана мостом, реалізація SNMP над канальним рівнем буде цілком працездатна.

Проте, для забезпечення якісного транспортування повідомлень в мережах будь-яких топології та технології функцій канального рівня виявляється недостатньо, тому в моделі OSI розв'язання цієї задачі покладається на два наступних рівні – мережевий і транспортний.

Канальний рівень забезпечує передачу пакетів даних, що надходять від протоколів верхніх рівнів, вузлу призначення, адреса якого також вказує протокол верхнього рівня. Протоколи канального рівня оформлюють передані їм пакети в кадри власного формату. Вони розміщують вказану адресу призначення в одне з полів такого кадру, а також супроводжують кадр контрольної сумою. Протокол канального рівня має локальний сенс, він призначений для доставки кадрів даних у межах мереж з простою топологією зв'язків і однотипною або близькою технологією, наприклад в односегментній мережі Ethernet або в багатосегментних мережах Ethernet і Token Ring ієрархічних топології, розділених тільки мостами і комутаторами. У всіх цих змінах адреса призначення має локальний сенс для даної мережі й не змінюється під час проходження кадру від вузла-джерела до вузла призначення. Можливість передавати дані між локальними мережами різних технологій пов'язана з тим, що в цих технологіях використовуються адреси однакового формату, до того ж виробники мережевих адаптерів забезпечують унікальність адрес незалежно від технології.

Іншою областю дії протоколів канального рівня є зв'язок типу «точка-точка» глобальних мереж, коли протокол канального рівня відповідальний за доставку кадру безпосереднього сусіда. Адреса в цьому випадку не має принципового значення, а на перший план виходить здатність протоколу відновлювати перекручені й загублені кадри, так як погана якість територіальних каналів, особливо комутованих телефонних, зазвичай вимагає виконання таких дій. Якщо ж перераховані вище умови не дотримуються, наприклад, зв'язок між сегментами Ethernet має структуру петлі, або об'єднані мережі використовують різні способи адресації, як у мережах Ethernet і X.25, то протокол канального рівня не може самотужки впоратися зі завданням передачі кадру між вузлами і вимагає допомоги протоколу мережевого рівня.

1.2.4. Мережевий рівень

Мережевий рівень (Network layer) необхідний для утворення єдиної транспортної системи, що об'єднує декілька мереж, причому ці мережі можуть використовувати різні принципи передачі повідомлень між кінцевими вузлами і мають довільну структуру зв'язків. Функції мережевого рівня досить різноманітні. Розглянемо їх на прикладі об'єднання локальних мереж.

Протоколи канального рівня локальних мереж забезпечують доставку даних між будь-якими вузлами тільки в мережі з відповідною типовою топологією, наприклад топологією ієрархічної зірки. Це жорстке обмеження, яке не дозволяє будувати мережі з розвиненою структурою, наприклад, мережі, що об'єднують декілька мереж підприємства до єдиної мережі, або високонадійні мережі, в яких існують надлишкові зв'язки між вузлами. Можна було б ускладнювати протоколи канального рівня для підтримки надлишкових зв'язків у вигляді петлі, але принцип розподілу обов'язків між рівнями призводить до іншого рішення. Щоб, з одного боку, зберегти простоту процедур передачі даних для типових топологій, а з іншого – допустити використання довільних топологій, вводиться додатковий мережевий рівень.

На мережевому рівні сам термін «мережа» наділяють специфічним значенням. У цьому випадку під мережею розуміється сукупність комп'ютерів, які з'єднано між собою відповідно до однієї зі стандартних типових топологій і використовується для передачі даних один з протоколів канального рівня, що визначений для цієї топології.

Усередині мережі доставка даних забезпечується відповідним канальним рівнем, а ось доставкою даних між мережами займається мережевий рівень, який і підтримує можливість правильного вибору маршруту передачі повідомлення навіть у тому випадку, коли структура зв'язків між складовими мережами має характер, який відмінний від прийнятого в протоколах канального рівня.

Мережі з'єднуються між собою спеціальними пристроями, так званими маршрутизаторами. ***Маршрутизатор*** – це пристрій, який збирає інформацію щодо топології міжмережевих з'єднань і пересилає пакети мережевого рівня до мережі призначення. Щоб передати повідомлення від відправника, що знаходиться в одній мережі, одержувачу, що знаходиться в іншій мережі, потрібно здійснити деяку кількість транзитних передач між мережами, чи хопів (від слова hop – стрибок), кожен раз вибираючи відповідний маршрут. Таким чином, ***маршрут*** є послідовність маршрутизаторів, через які проходить пакет.

Мережевий рівень – доставка пакета:

– між будь-якими двома вузлами мережі з довільною топологією;

– між будь-якими двома мережами у складовій мережі;

– мережа – сукупність комп'ютерів, що використовують для обміну даними єдину мережеву технологію.

На рис. 1.8 показані чотири мережі, які пов'язані трьома маршрутизаторами. Між вузлами А і В цієї мережі пролягає два маршрути: перший – через маршрутизатори 1 і 3, а другий – через маршрутизатори 1, 2 і 3.

Проблема вибору найкращого шляху називається ***маршрутизацією***, та її рішення є однією з головних задач мережевого рівня. Ця проблема ускладнюється тим, що найкоротший шлях – не завжди найкращий. Зазвичай критерієм під час вибору маршруту є час передачі даних; він залежить від пропускної здатності каналів зв'язку та інтенсивності трафіку, яка може змінюватися у часі. Деякі алгоритми маршрутизації намагаються пристосуватися до зміни навантаження, в той час як інші приймають рішення на основі середніх показників за тривалий час. Вибір маршруту може здійснюватися і за іншими критеріями, таким як надійність передачі.

У загальному випадку функції мережевого рівня ширше, ніж функції передачі повідомлень по зв'язках зі нестандартною структурою, які були розглянуті на прикладі об'єднання декількох локальних мереж. Мережевий рівень також вирішує завдання узгодження різних технологій, спрощення адресації у великих мережах і створення надійних і гнучких бар'єрів на шляху небажаного трафіку між мережами.

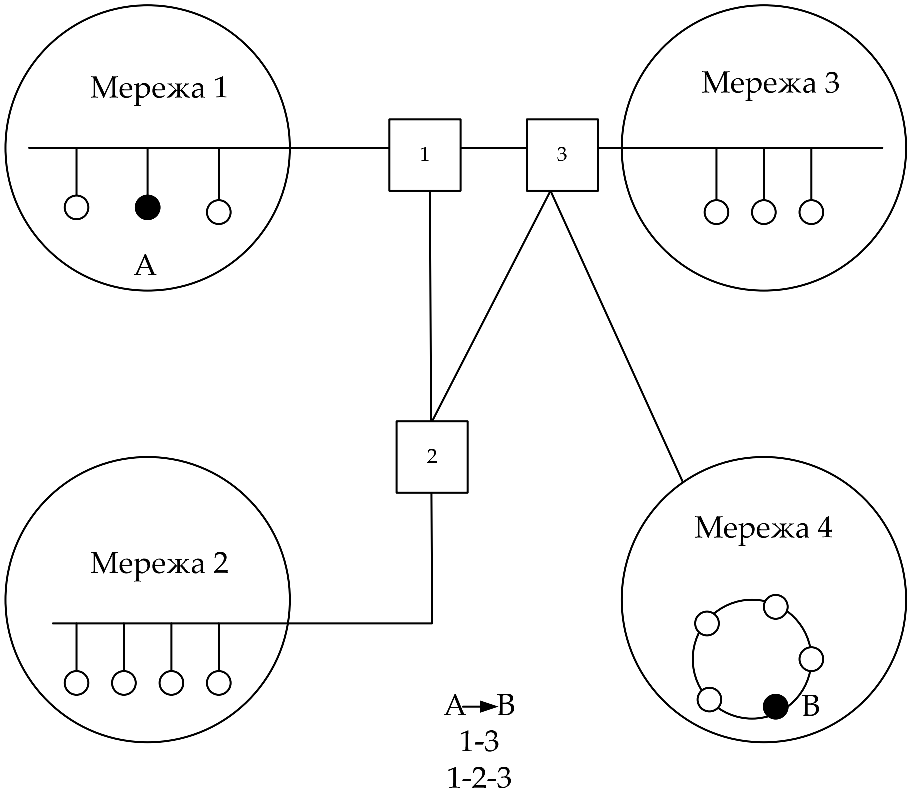


Рисунок 1.8 – Приклад складної мережі

Повідомлення мережевого рівня прийнято називати ***пакетами*** (***packet***). Під час організації доставки пакетів на мережевому рівні використовується поняття «номер мережі». У цьому випадку адреса одержувача складається з старшої частини – номера мережі і молодшої – номера вузла в цій мережі. Усі вузли однієї мережі повинні мати одну і ту ж старшу частину адреси, тому терміну «мережа» на мережевому рівні можна дати й інше, більш формальне, визначення: мережа – це сукупність вузлів, мережева адреса яких містить один і той же номер мережі.

На мережевому рівні визначається два види протоколів. Перший вид – мережеві протоколи (***routed protocols***) реалізують передачу пакетів крізь мережу. Саме ці протоколи зазвичай мають на увазі, коли говорять щодо протоколів мережевого рівня. Однак зазвичай до мережевого рівня відносять і інший вид протоколів, так звані протоколи обміну маршрутною інформацією або просто протоколами маршрутизації (***routing protocols***). За допомогою цих протоколів маршрутизатори збирають інформацію щодо топології міжмережевих з'єднань. Протоколи мережевого рівня реалізуються програмними модулями операційної системи, а також програмними і апаратними засобами маршрутизаторів.

На мережевому рівні працюють протоколи ще одного типу, які відповідають за відображення адреси вузла, що використовується на мережевому рівні, в локальну адресу мережі. Такі протоколи зазвичай називають протоколами дозволу – Address Resolution Protocol, ARP. Іноді їх відносять не до мережевого рівня, а до канального, хоча тонкощі класифікації не змінюють суті.

Прикладами протоколів мережевого рівня є протокол міжмережевої взаємодії IP стека TCP / IP і протокол міжмережевого обміну пакетами IPX стека Novell.

1.2.5. Транспортний рівень

На шляху від відправника до одержувача пакети можуть бути ушкоджені або загублені. Хоча деякі додатки мають власні засоби обробки помилок, існують і такі, які вважають за краще відразу мати справу з надійним з'єднанням. Транспортний рівень (Transport layer) забезпечує додаткам або верхнім рівням стека – прикладному і сеансовому – передачу даних з тим ступенем надійності, яка їм потрібна. Модель OSI визначає п'ять класів сервісу, наданих транспортним рівнем. Ці види сервісу відрізняються якістю наданих послуг: терміновістю, можливістю відновлення перерваного зв'язку, наявністю засобів мультиплексування декількох з'єднань між різними прикладними протоколами через загальний транспортний протокол, а головне – здатністю до виявлення і виправлення помилок передачі, таких як ушкодження, втрата і дублювання пакетів.

Вибір класу сервісу транспортного рівня визначається, з одного боку, тим, якою мірою завдання забезпечення надійності вирішується самими додатками і протоколами більш високих, ніж транспортний, рівнів, а з іншого боку, залежить від того, наскільки надійною є система транспортування даних у мережі, забезпечується рівнями, розташованими нижче транспортного – мережевим, канальним і фізичним. Так, наприклад, якщо якість каналів передачі зв'язку є дуже високою, і ймовірність наявності помилок, не виявлених протоколами нижчих рівнів, невелика, варто скористатися одним з полегшених сервісів транспортного рівня, не обтяжених численними перевірками, квотуванням та іншими прийомами підвищення надійності. Якщо ж транспортні засоби нижніх рівнів спочатку дуже ненадійні, то доцільно звернутися до найбільш розвиненого сервісу транспортного рівня, який працює, під час використання максимум засобів для виявлення та усунення помилок, – за допомогою попереднього встановлення логічного з'єднання, відстеження доставки повідомлень щодо контрольних сум і циклічній нумерації пакетів, встановлення тайм-аутів доставки і т.п.

Транспортний рівень – забезпечення доставки інформації з необхідною якістю між будь-якими вузлами мережі:

– розбивка повідомлення сеансового рівня на пакети, їх нумерація;
– буферизація прийнятих пакетів;
– упорядкування пакетів, що прибувають;
– адресація прикладних процесів;
– управління потоком.

Зазвичай усі протоколи, починаючи з транспортного рівня і вище, реалізуються програмними засобами кінцевих вузлів мережі – компонентами їх мережевих операційних систем. Як приклад транспортних протоколів можна привести протоколи TCP і UDP стека TCP / IP і протокол SPX стека Novell.

Протоколи чотирьох нижніх рівнів узагальнено називають мережевим транспортом або транспортною підсистемою, так як вони повністю вирішують задачу транспортування повідомлень із заданим рівнем якості в складних мережах з довільною топологією і різними технологіями. Решта три верхніх рівні вирішують задачі надання прикладних сервісів на основі транспортної підсистеми.

1.2.6. Сеансовий рівень

Сеансовий рівень (Session layer) забезпечує управління діалогом: фіксує, яка зі сторін є активною у цей момент, надає засоби синхронізації. Останні дозволяють вставляти контрольні точки під час довгих передач, щоб у разі відмови можна було повернутися назад до останньої контрольної точки, а не починати все спочатку.

Сеансовий рівень – управління діалогом об'єктів прикладного рівня:

– встановлення способу обміну повідомленнями (двобічний або напівдуплексний);
– синхронізація обміну повідомленнями;
– організація «контрольних точок» діалогу.

1.2.7. Представницький рівень

Представницький рівень (Presentation layer) має справу з формою подання інформації, що передається за мережею інформації, не змінюючи при цьому її змісту. Представлення інформації прикладним рівнем однієї системи завжди зрозуміла прикладному рівню іншої системи. За допомогою засобів цього рівня протоколи прикладних рівнів можуть подолати синтаксичні відмінності в представлених даних або ж відмінності в кодах символів, наприклад в кодах ASCII і EBCDIC. На цьому рівні може виконуватися шифрування і дешифрування даних, завдяки якому секретність обміну даними забезпечується відразу для всіх прикладних служб. Прикладом такого протоколу є протокол Secure Socket Layer (SSL), який забезпечує секретний обмін повідомленнями для протоколів прикладного рівня стека TCP/IP.

Рівень представлення – погоджує подання (синтаксис) даних під час взаємодії двох прикладних процесів:

– перетворення даних зі зовнішнього формату у внутрішній;

– шифрування і розшифрування даних.

1.2.8. Прикладний рівень

Прикладний рівень (Application layer) – це насправді просто набір різноманітних протоколів, за допомогою яких користувачі мережі отримують доступ до таких ресурсів як файли, принтери або гіпертекстові Web-сторінки, а також організують спільну роботу, наприклад за допомогою протоколу електронної пошти. Одиниця даних, якою оперує прикладний рівень зазвичай називається повідомленням (message).

Прикладний рівень – набір усіх мережевих сервісів, які надає система кінцевому користувачеві:

– ідентифікація, перевірка прав доступу;

– принт і файл-сервіс, пошта, віддалений доступ.

Існує дуже багато різних служб прикладного рівня. Наведемо як приклад хоч би декілька найбільш поширених реалізацій файлових служб: NCP в операційній системі Novell NetWare, SMB в Microsoft Windows NT, NFS, FTP і TFTP, що входять в стек TCP/IP.

1.2.9. Мережезалежні та мереженезалежні рівні

Функції всіх рівнів моделі OSI можуть бути віднесені до однієї з двох груп: або до функцій, котрі залежать від конкретної технічної реалізації мережі, або до функцій, що орієнтовані на роботу з додатками.

Три нижніх рівні – фізичний, канальний і мережевий – є мережезалежними, тобто протоколи цих рівнів тісно пов'язані з технічною реалізацією мережі та комунікаційним обладнанням. Наприклад, перехід на обладнання FDDI означає повну зміну протоколів фізичного і канального рівнів у всіх вузлах мережі.

Три верхніх рівня – прикладний, представницький і сеансовий – орієнтовані на додатки і мало залежать від технічних особливостей побудови мережі. На протоколи цих рівнів не впливають які б то не було зміни в топології мережі, заміна обладнання або перехід на іншу мережеву технологію. Так, перехід від Ethernet до високошвидкісної технології 100VG-AnyLAN не зажадає ніяких змін у програмних засобах, що реалізовують функції прикладного, представницького і сеансового рівнів.

Транспортний рівень є проміжним, він приховує всі деталі функціонування нижніх рівнів від верхніх. Це дозволяє розробляти додатки, які не залежать від технічних засобів безпосереднього транспортування повідомлень.

На рисунку 1.9 показані рівні моделі OSI, на яких працюють різні елементи мережі. Комп'ютер з встановленою на ньому мережевою ОС взаємодіє з іншим комп'ютером за допомогою протоколів усіх семи рівнів. Цю взаємодію комп'ютери здійснюють опосередковано, через різні комунікаційні пристрої: концентратори, модеми, мости, комутатори, маршрутизатори, мультиплексори.

Залежно від типу комунікаційний пристрій може працювати або тільки на фізичному рівні (повторювач), або на фізичному і канальному (міст), або на фізичному, канальному і мережевому, іноді захоплюючи і транспортний рівень (маршрутизатор). На рис. 1.10 показано відповідність функцій різних комунікаційних пристроїв рівням моделі OSI.

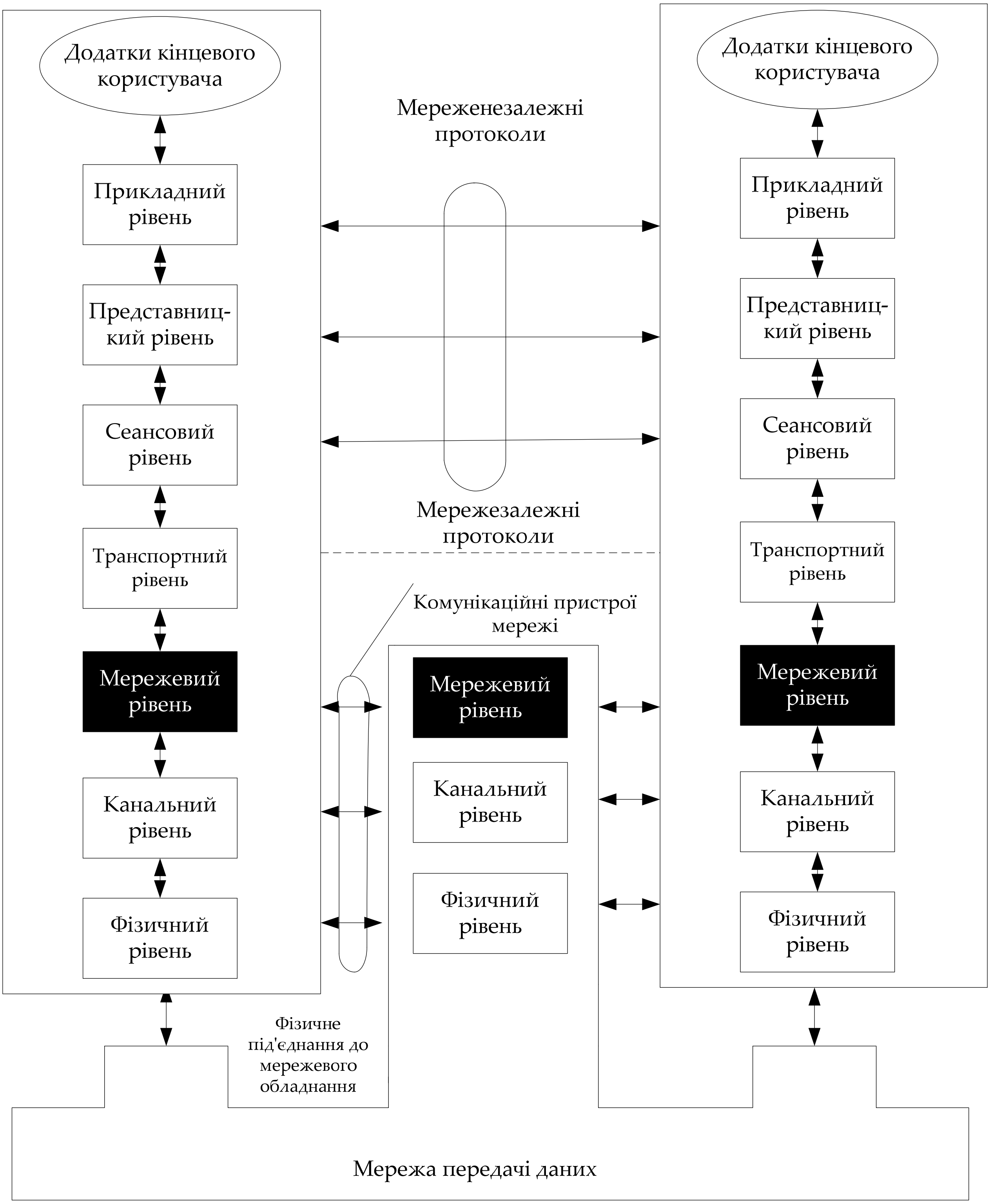


Рисунок 1.9 – Мережезалежні та мереженезалежні рівні моделі OSI

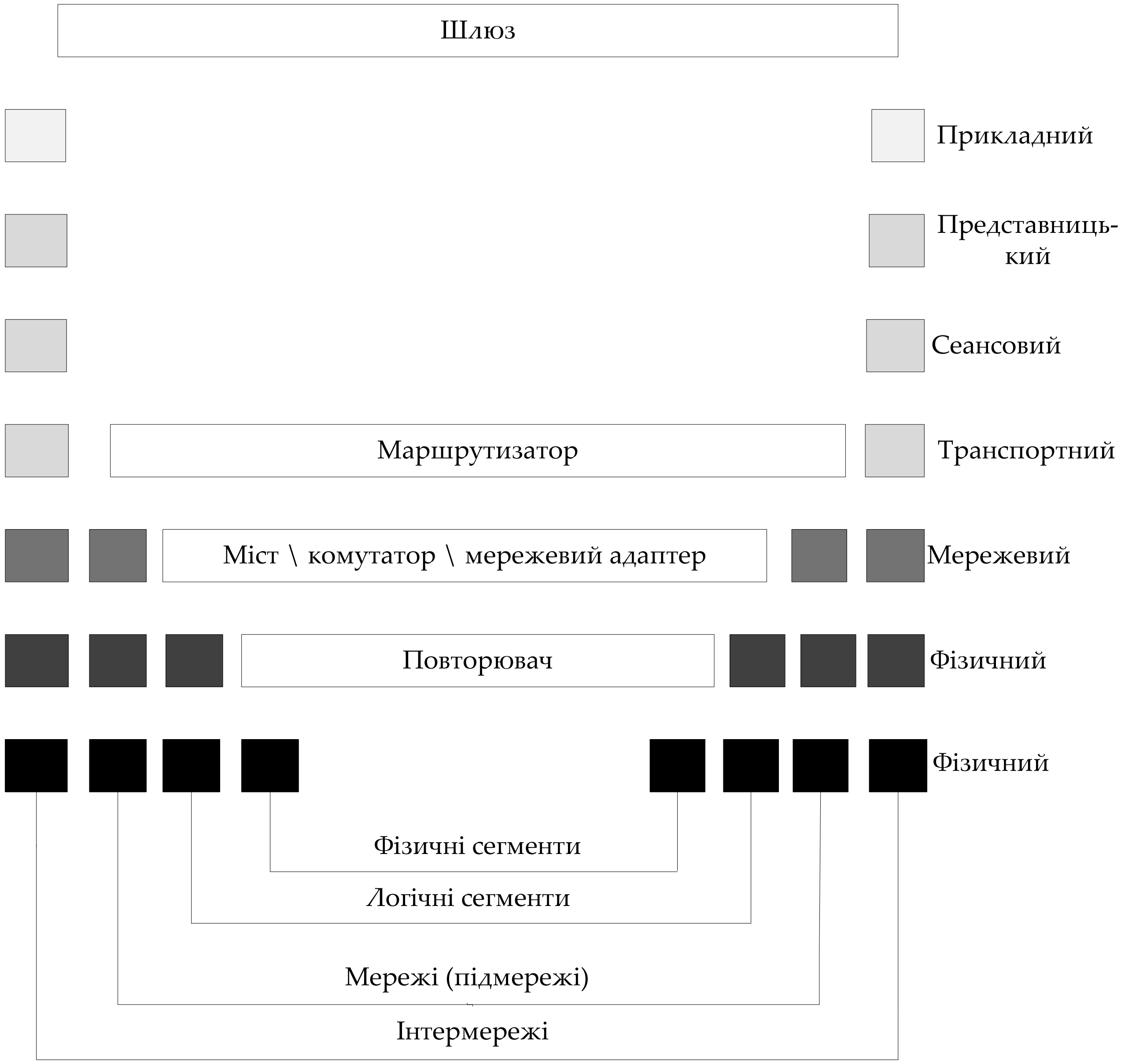


Рисунок 1.10 – Відповідність функцій різних пристроїв мережі рівням моделі OSI

Модель OSI представляє хоча і дуже важливу, але тільки одну з багатьох моделей комунікацій. Ці моделі і пов'язані з ними стеки протоколів можуть відрізнятися кількістю рівнів, їх функціями, форматами повідомлень, службами, що підтримується на верхніх рівнях та ін.

Контрольні запитання та завдання

1. Надайте визначення поняттям: рівень OSI, протокол рівня, сутність рівня, сервіс рівня, інтерфейс рівня, архітектура мережі, стек протоколів.
2. Поясніть призначення фізичного, канального, мережевого, транспортного та прикладного рівнів у контексті моделі OSI.
3. Наведіть перелік робочих характеристик, за допомогою якого можна оцінити промислову мережу.
4. Які сервіси може підтримувати промислова мережа на прикладному рівні? Як може бути реалізований інтерфейс прикладного рівня?
5. Які моделі обміну між прикладними процесами зазвичай використовуються в промислових мережах? Як вони функціонують.
6. Яким чином визначається місцезнаходження необхідних даних, які приймають участь в обміні? Що таке прикладний профіль пристрою?
7. Які моделі функціонування сервісів прикладного рівня можна виділити в промислових мережах? Як вони функціонують?

РОЗДІЛ 2
СФЕРА ПРИЗНАЧЕННЯ ПРОТОКОЛІВ ПРОМИСЛОВИХ МЕРЕЖ, ОСНОВНА ВІДМІННІСТЬ ВІД ПУБЛІЧНИХ МЕРЕЖ

Найбільшого поширення в автоматизованих системах управління технологічних процесів (АСУ ТП) отримали розподілені системи, елементи яких (контролери, модулі вводу-виводу) рознесено у просторі, незалежні один від одного, але взаємодіють між собою для виконання спільного завдання. Основою для побудови розподілених систем збору даних та управління є промислова мережа.

Промислові мережі – мережі, що забезпечують комунікацію між пристроями, що входять до складу автоматизованої системи.

2.1. Загальні відомості про промислові мережі

Обмін інформацією між пристроями, що входять до складу автоматизованої системи (комп'ютерами, контролерами, датчиками, виконавчими пристроями), відбувається зазвичай через ***промислову мережу*** (***fieldbus***, «польову шину»). Промислові мережі відрізняються від офісних такими властивостями:

– спеціальним конструктивним виконанням, що забезпечує захист від пилу, вологи, вібрації, ударів;

– широким температурним діапазоном (зазвичай від – 40 до +70 ºС);

– підвищену міцність кабелю, ізоляції, контактів, елементів кріплення; підвищеною стійкістю до дії електромагнітних перешкод;

– можливістю резервування для підвищення надійності;

– підвищеною надійністю передачі даних;

– можливістю самовідновлення після збою;

– детермінованістю (визначеністю) часу доставки повідомлень;

– можливістю роботи у реальному часі (з малою, постійною і відомою величиною затримки);

– роботою з довгими лініями зв'язку (від сотень метрів до кількох кілометрів).

Промислові мережі зазвичай не виходять за межі одного підприємства. Проте з появою Ethernet і Internet для промислових мереж стали застосовувати ту ж класифікацію, що і для офісних:

LAN (Local Area Network) – мережі, що розташовано на обмеженій території (у цеху, офісі, у межах заводу);

MAN (Metropolitan Area Networks) – мережі міст;

WAN (Wide Area Network) – глобальна мережа, що охоплює кілька міст або континентів. Зазвичай для цього використовують Internet-технологію.

У теперішній час налічується більше 50 типів промислових мереж (ModBus, ProfiBus, DeviceNet, CANopen, LonWorks, ControlNet, SDS, Seriplex, ArcNet, BACnet, FDDI, FIP, FF, ASI, Ethernet, WorldFIP, Foundation Fieldbus, Interbus, BitBus та ін.). Однак широко поширеними є тільки частина з них. Переважна більшість АСУ ТП використовують мережі ModBus і ProfiBus. У останні роки зріс інтерес до мереж на основі CANopen і DeviceNet.

Залежно від області застосування весь спектр промислових мереж можна розділити на два рівні (див. рис. 2.1):

– ***контролерні мережі***, які вирішують завдання з управління процесом виробництва, збору й обробки даних на рівні промислових контролерів;

– ***польові*** або ***сенсорні мережі***, які вирішують завдання опитування датчиків і управління роботою різноманітних виконавчих механізмів).

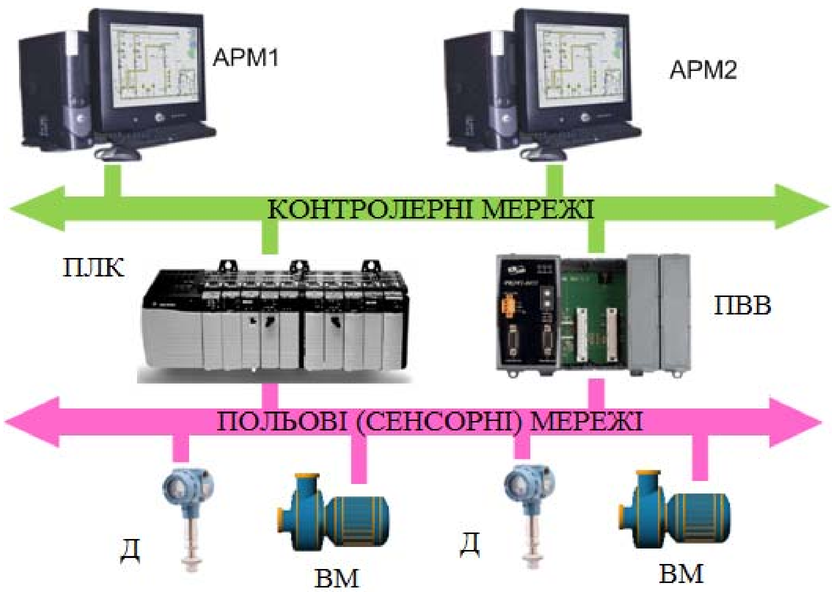


Рисунок 2.1 – Рівні промислових мереж

Іноді польові мережі називають мережами низової автоматики. Для польових мереж характерна передача невеликого обсягу даних на невеликі відстані за короткий час. Для контролерних мереж навпаки характерна передача великих обсягів даних на великі відстані.

Оскільки у промисловій автоматизації мережеві інтерфейси можуть бути невід'ємною частиною пристроїв з'єднання, а мережеве програмне забезпечення рівня додатків моделі OSI виконується на основному процесорі промислового контролера, то відокремити мережеву частину від пристроїв, що об'єднуються до мережі, іноді фізично неможливо. З іншого боку, зміну однієї мережі на іншу зазвичай можна виконати за допомогою заміни електричного кабелю і мережевого адаптера або введенням перетворювача інтерфейсу, тому зазвичай один і той самий тип ПЛК може використовуватися в мережах різних типів.

З'єднання промислової мережі з її компонентами (пристроями, вузлами мережі) виконується за допомогою інтерфейсів. ***Мережевим інтерфейсом*** називають логічну і (або) фізичну межу між пристроєм і середовищем передачі інформації. Зазвичай цією межею є набір електронних компонентів і пов'язаного з ними програмного забезпечення. Під час істотної модифікації внутрішньої структури пристрою або програмного забезпечення інтерфейс залишається без змін, що є однією з ознак, які дозволяють виділити інтерфейс у складі обладнання.

Найбільш важливими параметрами інтерфейсу є пропускна здатність і максимальна довжина кабелю, що підключається. Промислові інтерфейси зазвичай забезпечують гальванічну розв'язку між пристроями, що з'єднуються. Найбільш поширені у промислової автоматизації послідовні інтерфейси RS-485, RS-232, RS-422, Ethernet, CAN, HART, AS-інтерфейс.

Для обміну інформацією пристрої, що взаємодіють, повинні мати однаковий протокол обміну. ***Протокол*** – це набір правил, які керують обміном інформацією. Він визначає синтаксис і семантику повідомлень, операції управління, синхронізації й стану під час комунікації. Протокол може бути реалізований апаратно, програмно або комбіновано. Назва мережі зазвичай збігається з назвою протоколу. Порівняння рівнів промислових мереж наведено в таблиці 2.1.

Таблиця 2.1 – Порівняння рівнів промислових мереж

Характеристика	Польовий	Контролерний
Максимальна довжина мережі, м	100	1000
Час циклу передачі, с	0,001 – 1	0,01 – 10
Об'єм даних, що передаються, байт	1 – 8	8 – 1000
Доступ до шини	вільний	Фіксований/вільний
Ціна середовища передачі	низька	низька

2.2. HART-протокол

HART-протокол (Highway Addressable Remote Transducer – «віддалений адресний перетворювач з високою пропускною спроможністю») є відкритим стандартом на метод мережевого обміну, який включає в себе не тільки протокол взаємодії пристроїв, але і вимоги до апаратури каналу зв'язку. Стандарт HART розроблений у 1980 році фірмою Rosemount Inc., яка пізніше зробила його відкритим. У теперішній час стандарт підтримується міжнародною організацією HART Communication Foundation (HCF). HART застосовується для зв'язку контролера з датчиками і вимірювальними перетворювачами, електромагнітними клапанами, локальними контролерами, а також з іскробезпечним обладнанням.

Незважаючи на свою низьку швидкодію (1200 біт/с) і ненадійний аналоговий спосіб передачі даних, а також появу більш досконалих мережевих технологій, пристрої з HART-протоколом розробляються досі й обсяг цього сегменту ринку продовжує зростати. Однак застосування HART досить обмежено, оскільки впровадження датчиків з HART-протоколом вимагає одночасного застосування HART-сумісних контролерів і спеціалізованого програмного забезпечення. Стандарт HART включає в себе 1-й, 2-й і 7-й рівні моделі OSI (див. табл. 2.2).

При створенні HART-протоколу в 1980 році переслідувалася мета зробити його сумісним з широко поширеним у той час стандартом «струмова петля», але додати можливості, необхідні для управління інтелектуальними пристроями. Тому аналогова «струмова петля» (4 – 20) мА була модернізована таким чином, що отримала можливість полудуплексного цифрового обміну даними.

Таблиця 2.2 – Модель OSI HART-протоколу

Номер рівня	Назва рівня	HART
7	прикладний	HART-команди, відповіді, типи даних
6	представницький	Ні
5	сеансовий	Ні
4	транспортний	Ні
3	мережевий	Ні
2	канальний (передачі даних)	Ведучий/ведений, контрольна сума, контроль парності, організація потоку бітів у повідомленні, контроль прийому повідомлень.
1	фізичний	Накладання цифрового ЧМ сигналу на аналоговий (4 – 20) мА; мідна кручена пара

Для цього аналоговий сигнал *A(t)* підсумовується з цифровим сигналом *D(t)* та передаються за допомогою джерела струму (4 – 20) мА в лінію зв'язку. Завдяки великій відмінності діапазонів частот аналогового (0 – 10 Гц) і цифрового (1200 Гц і 2200 Гц) сигналів вони легко можуть бути розділені фільтрами низьких і високих частот у пристроях прийому. Під час передачі цифрового двійкового сигналу логічна одиниця кодується синусоїдальним сигналом з частотою 1200 Гц, нуль – 2200 Гц (див. рис. 2.2).

Під час зміни частоти фаза коливань залишається безперервною. Такий спосіб формування сигналу називається частотною маніпуляцією з безперервною фазою.

Переваги HART-протоколу:

– сумісність з «струмовою петлею» (4 – 20) мА;

– використання у вибухонебезпечному обладнанні.

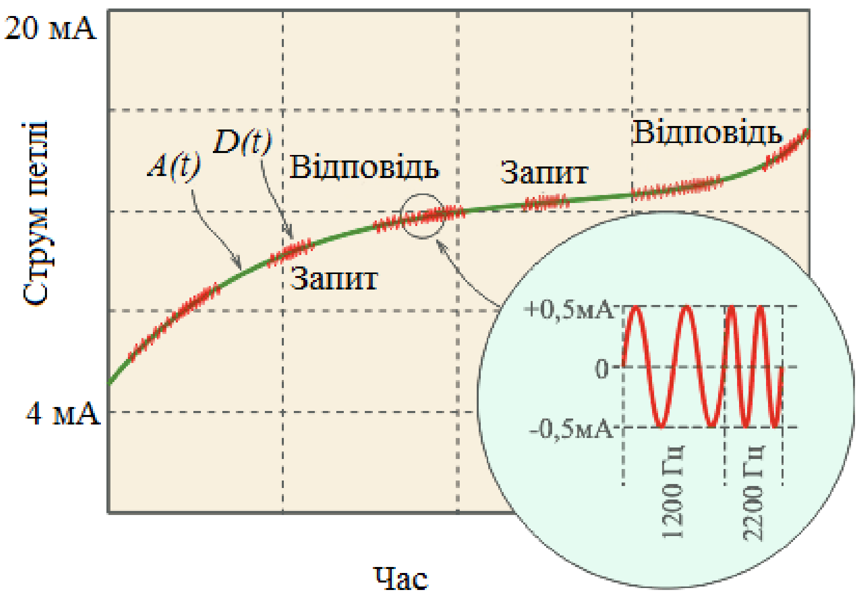


Рисунок 2.2 – Сума аналогового та цифрового сигналу в HART-протоколі

Недоліки HART-протоколу:

– низька швидкість передачі;

– ненадійний аналоговий спосіб передачі даних;

– обмін за типом «ведучий/ведений»;

– необхідність застосування HART-сумісних контролерів і спеціалізованого програмного забезпечення.

2.3. Modbus

Протокол ModBus і мережа ModBus є найпоширенішими у світі. Незважаючи на свій вік (стандартом де-факто ModBus став ще у 1979 році), ModBus не тільки не застарів, але, навпаки, істотно розширив кількість нових розробок та обсяг організаційної підтримки цього протоколу. Одним з переваг ModBus є відсутність необхідності у спеціальних інтерфейсних контролерах (ProfiBus і CAN вимагають для своєї реалізації мікросхеми), простота програмної реалізації та елегантність принципів функціонування. Все це знижує витрати на освоєння стандарту як системними інтеграторами, так і розробниками контролерного обладнання. Високий ступінь відкритості протоколу забезпечується також повністю безкоштовними текстами стандартів, які можна скачати з сайту www.modbus.org.

ModBus за поширеністю конкурує тільки з ProfiBus. Популярність протоколу у теперішній час пояснюється, перш за все, сумісністю з великою кількістю обладнання, яке має протокол ModBus. Крім того, ModBus має високу достовірність передачі даних, що пов'язано зі застосуванням надійного методу контролю помилок. ModBus дозволяє уніфікувати команди обміну завдяки стандартизації номерів (адрес) регістрів і функцій їх читання-запису. Основним недоліком ModBus є мережевий обмін за типом «ведучий/ведений», що не дозволяє веденим пристроям передавати дані при їх появі й тому вимагає інтенсивного опитування ведених пристроїв ведучими (див. рис. 2.3).

Різновидами ModBus є протоколи ModBus Plus – многомастерний протокол з кільцевої передачею маркера і ModBus TCP (див. рис. 2.4), який розрахований на використання в мережах Ethernet та Internet.

Стандарт ModBus передбачає застосування фізичного інтерфейсу RS-485, RS-422 або RS-232. Найбільш поширеним для організації промислової мережі є 2-провідний інтерфейс RS-485. Для з'єднань точка – точка може бути використаний інтерфейс RS-232 або RS-422.

Модель OSI протоколу ModBus (див. табл. 2.3) містить три рівні: фізичний, канальний і прикладний.

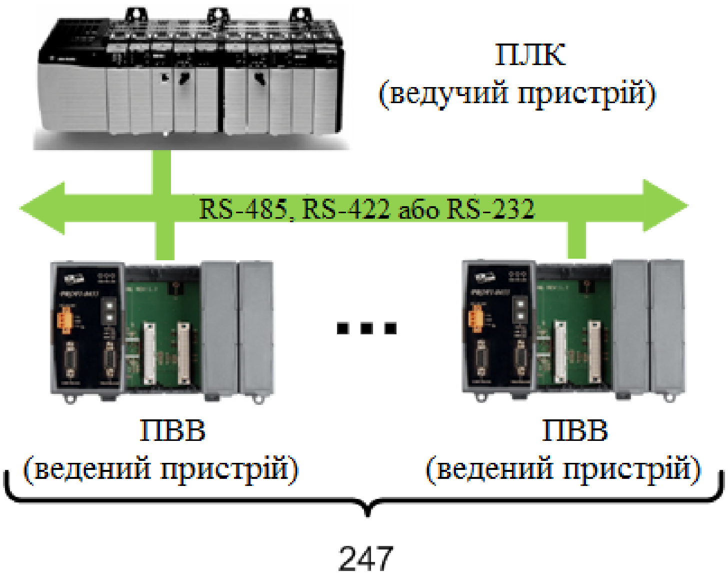


Рисунок 2.3 – Організація протоколу ModBus

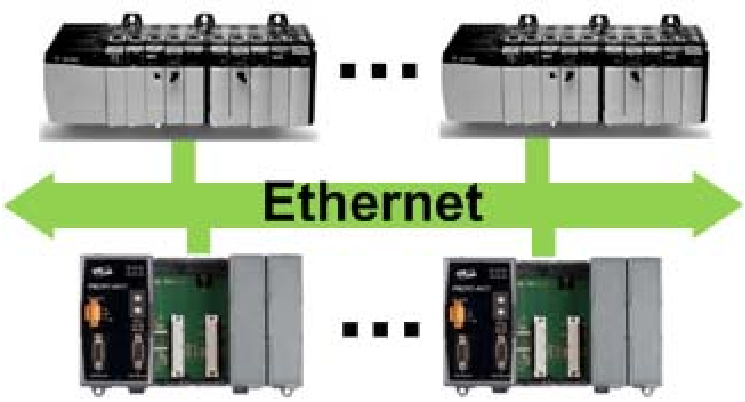


Рисунок 2.4 – ModBus TCP

Таблиця 2.3 – Модель OSI для ModBus

Номер рівня	Назва рівня	Реалізація
7	прикладний	MODBUS Application Protocol
6	представницький	Ні
5	сеансовий	Ні
4	транспортний	Ні
3	мережевий	Ні
2	канальний (передачі даних)	Протокол «ведучий / ведений» Режими RTU и ASCII
1	Фізичний	RS-485 або RS-232

У нових розробках на основі ModBus стандарт рекомендує використовувати інтерфейс RS-485 з двохпровідною лінією передачі, але допускається застосування чотирьох лінії та інтерфейсу RS-232.

ModBus-шина повинна складатися з одного магістрального кабелю, від якого можуть бути зроблені відводи. Магістральний кабель ModBus повинен містити 3 провідника в загальному екрані, два з яких представляють собою кручену пару, а третій загальний («земляний») вивід всіх інтерфейсів RS-485 у мережі. Загальний провідник та екран повинні бути заземлені в одній точці, бажано близько ведучого пристрою.

Пристрої можуть підключатися до кабелю трьома способами:

– безпосередньо до магістрального кабелю;

– через пасивний перехідник (трійник);

– через активний перехідник (повторювач інтерфейсу).

ModBus-пристрій обов'язково має підтримувати швидкості обміну 9600 біт/с та 19200 біт/с, з них 19200 біт/с встановлюється «за замовчанням». Допускаються також швидкості 1200, 2400, 4800, ..., 38400 біт/с, 65 кбіт/с, 115 кбіт/с. Швидкість передачі має бути витримана в передавачі з похибкою не більш 1 %, а приймач повинен приймати дані щодо відхилення швидкості передачі до 2 %.

Сегмент мережі, що не містить повторювачів інтерфейсу, повинен допускати підключення до 32 пристроїв, проте їх кількість може бути збільшена, якщо це допустимо виходячи з навантажувальної здатності передавачів і вхідного опору приймачів, які повинні бути приведені у документації на інтерфейси.

Протокол ModBus передбачає, що тільки один ведучий пристрій (контролер) і до 247 ведених (модулів введення-виведення) можуть бути об'єднані до промислової мережі. Обмін даними завжди ініціюється ведучим. Ведені пристрої не можуть обмінюватися даними один з іншим. Тому в будь-який момент часу в мережі ModBus може відбуватися тільки один акт обміну.

Переваги ModBus:

– відсутність необхідності у спеціальних інтерфейсних контролерах;
– простота програмної реалізації;
– елегантність принципів функціонування;
– високий ступінь відкритості протоколу.

Недолік ModBus: мережевий обмін по типу «ведучий / ведений».

2.4. Profibus

Слово PROFIBUS отримано від зобов'язань щодо зменшення PROcess FIeld BUS, що приблизно перекладається як «промислова шина для технологічних процесів». Стандарт ProfiBus був спочатку прийнятий у Німеччині в 1987 році, потім, у 1996 році, він став міжнародним (EN 50170 і EN 50254). Мережа ProfiBus (як і інші промислові мережі, крім Industrial Ethernet) використовує тільки перший і другий рівні моделі OSI. ProfiBus має три модифікації: ProfiBus DP, ProfiBus FMS та ProfiBus PA (див. рис. 2.5).

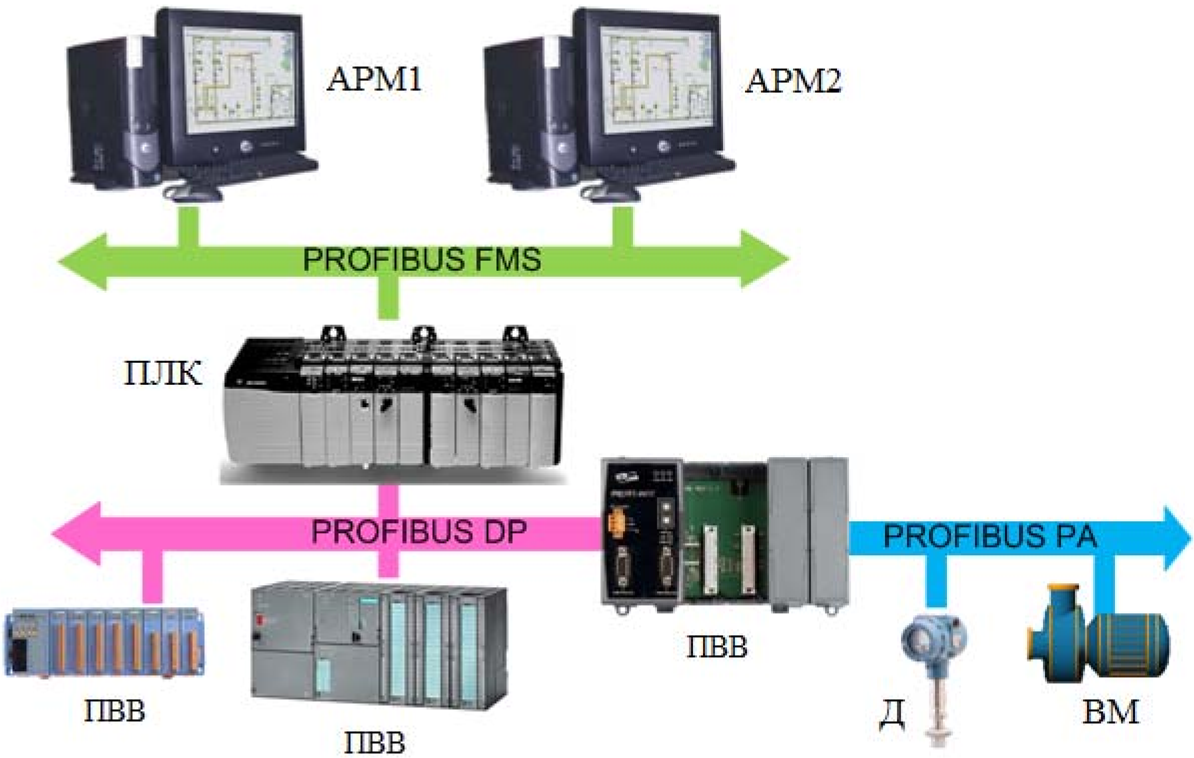


Рисунок 2.5 – Модифікацій мережі ProfiBus

ProfiBus DP (ProfiBus for Decentralized Peripherals – «ProfiBus для децентралізованої периферії») використовує рівні 1 і 2 моделі OSI, а також призначений для користувача інтерфейсу, який у модель OSI не входить. Безпосередній доступ з призначеного для користувача додатка до канального рівня здійснюється за допомогою DDLM (Direct Data Link Mapper – «прямий перетворювач для канального рівня»). Інтерфейс забезпечує функції, які необхідні для зв'язку з пристроями вводу-виводу і контролерами. ProfiBus DP на відміну від FMS і PA побудований так, щоб забезпечити найбільш швидкий обмін даними з пристроями, підключеними до мережі.

ProfiBus FMS (ProfiBus з FMS протоколом) використовує рівень 7 моделі OSI і застосовується для обміну даними з контролерами і комп'ютерами на регістровому рівні. ProfiBus FMS надає велику гнучкість при передачі великих обсягів даних, але програє протоколу DP в популярності внаслідок своєї складності. ProfiBus FMS і DP використовують один і той фізичний рівень, який заснований на інтерфейсі RS-485, і можуть працювати у загальній мережі.

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

У останні роки з'явився стандарт PROFInet, який заснований на Industrial Ethernet і технологіях COM, DCOM. Він легко забезпечує зв'язок промислової мережі ProfiBus з офісною мережею Ethernet.

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

Для збільшення дальності передачі в ProfiBus передбачена можливість роботи з оптоволоконним кабелем. При використанні скляного оптоволокна дальність зв'язку може бути збільшена до 15 км. Оптоволоконні інтерфейси виконуються у вигляді змінних модулів для контролерів.

Для ProfiBus розроблена модифікація 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, які встановлені стандартом на іскробезпечні електричні кола.

Канальний рівень моделі OSI в ProfiBus називається FDL- рівнем (Fieldbus Data Link – «промисловий канал зв'язку»). Об'єкт MAC (Medium Access Control – «управління доступом до каналу») на канальному рівні визначає процедуру передачі даних пристроями, які включають управління правами на передачу даних через мережу. Протокол канального рівня забезпечує виконання таких важливих вимог:

– під час комунікації між провідними пристроями необхідно забезпечити виконання кожним з них свого завдання протягом заздалегідь визначеного інтервалу часу;

– взаємодія провідних пристроїв (контролерів) з відомими має відбуватися максимально швидко.

Переваги PROFIBUS:

– однокабельне з'єднання пристроїв і невеликих підмереж;
– широкі можливості при обробці даних;
– забезпечення іскробезпеки;

– швидкісний обмін на рівні контролерів;
– велика довжина з'єднань.
Недоліки PROFIBUS: відносно висока ціна.

2.5. CAN

CAN (Controller Area Network – «мережа контролерів») є комплекс стандартів для побудови розподілених промислових мереж, який використовує послідовну передачу даних у реальному часі з дуже високим ступенем надійності й захищеності. Центральне місце в CAN займає протокол канального рівня моделі OSI. Спочатку CAN був розроблений для автомобільної промисловості, але у теперішній час швидко впроваджується в область промислової автоматизації. Початок розвитку CAN було покладено компанією Bosch в 1983 р, перші мікросхеми CAN-контролерів були випущені фірмами Intel і Philips у 1987 році, у теперішній час контролери та трансивери CAN випускаються багатьма фірмами, а саме Analog Devices, Inc., Atmel Corp. Cast, Dallas Semiconductor та ін.

CAN охоплює два рівні моделі OSI: фізичний і канальний. Стандарт не передбачає ніякого протоколу прикладного (7-го) рівня моделі OSI. Тому для його втілення у життя різні фірми розробили кілька таких протоколів: CANopen (організації CiA), SDS (фірми Honeywell Micro Switch Division), CAN Kingdom (фірми Kvaser), DeviceNet (фірми Allen-Bradley) і ряд інших.

CAN характеризується такими основними властивостями:
– кожному повідомленню (а не пристрою) встановлюється свій пріоритет;
– гарантована величина паузи між двома актами обміну;
– гнучкість конфігурації й можливість модернізації системи;
– широкомовний прийом повідомлень з синхронізацією часу;
– несуперечливість даних на рівні всієї системи;
– допустимість кількох провідних пристроїв у мережі;
– здатність до виявлення помилок і сигналізації щодо їх наявність;
– автоматичний повтор передачі повідомлень, доставлених з помилкою, відразу, як тільки мережа стане вільною;
– автоматичне розпізнавання збоїв і відмов з можливістю автоматичного відключення модулів, що відмовили.

До недоліків можна віднести порівняно високу вартість CAN-пристроїв, відсутність єдиного протоколу прикладного рівня, а також надмірну складність і заплутаність протоколів канального і прикладного рівня, що викладені у стандартах організації CAN in Automation (CiA).

Фізичний рівень моделі OSI забезпечує надійну передачу бітів, ігноруючи зміст інформації, що передається. Основними поняттями фізичного рівня є лінії передачі (у більшості випадків це кручена пара, хоча допускається використання плоского кабелю або один провід і «корпусну землю, оптоволокно, радіоканал), часові діаграми, система синхронізації, формат даних, забезпечення достовірності передачі (контрольна сума, методи кодування, виявлення та відновлення помилок). Характеристики передавача і приймача стандартом не встановлюються, оскільки вони можуть бути обрані для кожного конкретного випадку виходячи з вимог застосування.

Кабель крученої пари у мережі CAN повинен мати загальний (третій) провідник; на обох кінцях крученої пари повинні бути резистори погодження, опір яких дорівнює однохвильовому опору кабелю. Максимальна довжина кабелю становить 1 км. Для збільшення довжини, кількості вузлів або гальванічної розв'язки можуть бути використані повторювачі інтерфейсу, мережеві мости і шлюзи.

Стандарт встановлює такі швидкості обміну: 1 Мбіт/с, 800 кбіт/с, 500 кбіт/с, 250 кбіт/с, 125 кбіт/с, 50 кбіт/с, 20 кбіт/с. CAN-модулі можуть підтримувати не всі швидкості, але бажано, щоб їх кількість була найбільшим.

Недоліки CAN:

– порівняно висока вартість CAN-пристроїв;

– відсутність єдиного протоколу прикладного рівня;

– надмірна складність і заплутаність протоколів канального і прикладного рівня.

2.6. Промисловий Ethernet

Ethernet з'явився більше 30 років тому. У теперішній час під Ethernet розуміють сімейство продуктів для локальних мереж, які відповідають стандарту IEEE 802.3. Промисловому застосуванню стандарту довгий час заважав метод випадкового доступу до мережі, що не гарантував доставку повідомлення у короткий і заздалегідь відомий час. Однак це проблема була вирішена застосуванням комутаторів. Недоліком промислового Ethernet є відносно висока ціна: Ethernet модулі вводу-виводу у середньому у 2 рази дорожче аналогічних ModBus-пристроїв.

Впровадженню Ethernet у промисловість сприяли такі його якості:

– висока швидкість передачі (до 10 Гбіт/с) і відповідність вимогам жорсткого реального часу при високій швидкодії;

– простота інтегрування з Internet і Intranet, у тому числі за протоколами прикладного рівня SNMP (Simple Network Management Protocol), FTP, MIME (Multipurpose Internet Mail Extensions), HTTP;

– простота інтегрування з офісними мережами;

– наявність великої кількості фахівців з обслуговування Ethernet;

– по-справжньому відкриті рішення;

– можливість організації багатомайстерних мереж;

– необмежені можливості щодо організації мереж найрізноманітніших топологій;

– широке застосування в офісних мережах, що забезпечило економічну ефективність технічної підтримки стандарту з боку міжнародних організацій по стандартизації;

– поява недорогих комутаторів, які вирішили проблему недетермінованості Ethernet.

Існують 4 стандартні швидкості передачі даних у мережах Ethernet за оптоволоконним кабелем і крученою парою дротів: 10 Мбіт/с, 100 Мбіт/с, 1 Гбіт/с і 10 Гбіт/с. Ethernet має кілька модифікацій, структура найменувань яких має такий вигляд: <швидкість передачі> Base <додаткові позначення>. Швидкість вказується цифрою в Мбіт/с або в Гбіт/с, в останньому випадку до цифри додається буква G. Букви після «Base» позначають тип кабелю (T – «Twisted pair» – «кручена пара», F – «Fiber optic» – «оптоволоконний», S – «Short wavelength optic» – «оптичний короткохвильовий», L – «Long wavelength» – «довгохвильовий», C – «short Copper cable»). Символ «X» означає наявність блоку кодування на фізичному рівні. У тексті стандарту IEEE 802.3-2005 [IEEE] наводиться 44 варіанта таких позначень. Наприклад, 10BaseT означає специфікацію фізичного рівня для швидкості 10 Мбіт/с з методом доступу CSMA/CD і з використанням двох кручених пар провідників; 100BaseFX – для швидкості 100 Мбіт/с, CSMA/CD з застосуванням двох багатомодових оптичних кабелів.

У таблиці 2.4 наведена типова модель OSI для локальної мережі LAN, в якій Ethernet, як стандарт IEEE 802.3, охоплює тільки фізичний і канальний рівень. На інших рівнях представлено протоколи Internet. LLC – «Logical Link Control» («управління логічними зв'язками») – клієнт підрівня MAC; MAC («Medium Access Control») – підрівень управління доступом до лінії передачі; PHY (від «PHYsical») – фізичний рівень (лінія передачі). Рівні 1 і 2 Ethernet-протоколу зазвичай реалізуються апаратно, інші – програмно.

При використанні стандартного Ethernet час реакції мережі зазвичай перевищує 100 мс. У локальних сегментах мережі з малою кількістю пристроїв він знижується до 20 мс при використанні протоколу TCP до 10 мс – для UDP. Використання прямої MAC-адресації у локальних сегментах мережі може зменшити час реакції до 1 мс. Цей час у Ethernet є випадковою величиною. Тому основною відмінністю промислового Ethernet від офісної реалізації є наявність комутаторів, які перетворюють Ethernet у мережу з детермінованою поведінкою. Відмінними ознаками промислового Ethernet є:

– відсутність колізій і детермінованість поведінки завдяки застосуванню комутаторів;
– індустріальні кліматичні умови;
– стійкість до вібрацій;
– відсутність вентиляторів в обладнанні;
– підвищені вимоги до електромагнітної сумісності;
– компактність, кріплення на ДІН-рейку;
– зручне підключення кабелів;
– діагностична індикація на панелі приладу;
– електроживлення від джерела напруги у діапазоні від 10 до 30 В;
– можливість резервування;
– контакти та обладнання зі ступенем захисту до IP67;
– захист від електростатичних зарядів, електромагнітних імпульсів, від перевищення напруги живлення;
– повнодуплексна передача.

Таблиця 2.4 – Рівні моделі OSI для мережі Ethernet

<table>
<tr><th>№ рівня</th><th>OSI модель</th><th>Мережа Ethernet</th><th colspan="2">Варіанти реалізації</th></tr>
<tr><td>7</td><td>прикладний</td><td rowspan="3">прикладний</td><td rowspan="3">HTTP, FTP, SMTP, DNS</td><td rowspan="3">NFS, XDR, RPC</td></tr>
<tr><td>6</td><td>представницький</td></tr>
<tr><td>5</td><td>сеансовий</td></tr>
<tr><td>4</td><td>транспортний</td><td>транспортний</td><td>TCP</td><td>UDP</td></tr>
<tr><td>3</td><td>мережевий</td><td>Internet-протокол</td><td colspan="2">IP</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">канальний
(передачі даних)</td><td>LLC або інші клієнти MAC</td><td rowspan="2" colspan="2">Ethernet IEEE 802.3</td></tr>
<tr><td>MAC</td></tr>
<tr><td>1</td><td>Фізичний</td><td>PHY</td><td colspan="2">Кручена пара, оптичний кабель, коаксіальний кабель</td></tr>
</table>

Недоліком Ethernet є значний рівень «накладних витрат» у протоколах TCP/IP, які розраховані на пересилку даних великих обсягів. Якщо ж пристрій пересилає всього 8 байт, що типово для АСУ ТП, то корисна інформація в протоколі становить всього 11 %, оскільки кожен пакет TCP/IP міститься 68 байт службової інформації. Однак це зазвичай можна компенсувати великою пропускною здатністю мережі.

2.7. FOUNDATION Fieldbus

Польова шина FOUNDATION fieldbus є повністю цифровою послідовною двобічною комунікаційною системою зі швидкістю обміну 31,25 кбіт/с, за допомогою якої обмінюються даними такі польові пристрої, як датчики, приводи і контролери. Вона є локальною обчислювальною мережею для приладів, які використовуються в автоматичному управлінні технологічним процесом, з вбудованими можливостями розподілу керуючих додатків по всій мережі. Foundation Fieldbus зберігає всі переваги «струмового петлі», такі як:

– використання стандартного фізичного інтерфейсу з ведучою мережею зв'язку;
– подача живлення за шиною на пристрої, що знаходяться на одній провідній парі;
– забезпечення іскробезпеки.

Технологія Fieldbus дозволяє передавати безліч змінних з кожного приладу до керуючої системи в обох напрямках. Висока роздільна здатність і відсутність спотворень досягається використанням повністю цифрової системи зв'язку. Під час виявленні позаштатних ситуацій або при необхідності проведення профілактичного технічного обслуговування пристрої Fieldbus можуть передати відповідні повідомлення оперативно-технологічного та обслуговуючому персоналу.

Польова шина Fieldbus дозволяє підключати кілька пристроїв до однієї провідної пари. Це призводить до зменшення протяжності кабелів, зниження числа бар'єрів іскрозахисту і кросових шаф. Основною перевагою Foundation Fieldbus є можливість розподілу функцій управління процесом серед пристроїв нижнього рівня (рівня польових пристроїв) АСУ ТП.

У FOUNDATION fieldbus використовуються уніфіковані «функціональні блоки», що представляють собою стандартні функції задач автоматизації. З використанням функціональних блоків у польових пристроях можуть виконуватися багато функцій системи управління, наприклад аналогове введення (AI), аналоговий вивід (AO) і ПІД-регулювання.

Основними недоліками протоколу Foundation Fieldbus є:

– висока вартість обладнання;
– обмежений набір сумісних пристроїв;
– розтягнутість у часі процесів стандартизації та прийняття в промисловості.

2.8. Порівняльний аналіз протоколів

Порівняльний аналіз протоколів наведено в таблиці 2.5.

Найбільш простим і найбільш поширеним у світі є мережевий протокол ModBus, популярність якого пояснюється простотою як для розробників, так і споживачів, а також низькою вартістю реалізації. Проте відсутня можливість використання у вибухонебезпечних зонах.

Таблиця 2.5 – Порівняльний аналіз протоколів

Протокол	Переваги	Недоліки
HART	– повна сумісність з (4 – 20) мА; – забезпечення іскробезпеки.	– низька швидкодія; – ненадійний аналоговий спосіб передачі даних.
ModBus	– дуже низька вартість; – простота реалізації.	– відсутнє забезпечення іскробезпеки.
PROFIBUS	– забезпечення іскробезпеки; – швидкісний обмін на рівні контролерів.	– відносно висока ціна.
CAN	– надійний спосіб передачі даних.	– дуже висока вартість; – висока складність.
Foundation Filedbus	– розподіл функцій управління; – забезпечення іскробезпеки.	– висока вартість
Industrial Ethernet	– висока швидкість передачі; – широкі можливості по організації мереж.	– відносно низька ціна висока чутливість до електромагнітних завад.

Дуже ефективним і широко застосовуваним протоколом є ProfiBus, проте його поширеність пояснюється потужнім брендом «Siemens». Найбільш продуманим, універсальним і багатообіцяючим протоколом є Foundation Fieldbus, однак висока вартість заважає його швидкому поширенню. Для передачі великих обсягів даних найбільш підходящими є протоколи на основі промислового Ethernet.

Контрольні запитання та завдання

1. Чи відрізняються промислові від офісних мереж?
2. На які рівні розділяються промислові мережі?
3. Які найбільш важливі параметрами інтерфейсу?
4. Проведіть порівняльний аналіз рівнів промислових мереж.
5. Як модернізовано аналогову «токову петлю» у HART-протоколі?
6. Наведіть переваги та недоліки HART-протоколу.
7. Чому у теперішній час найбільш поширеним є протокол ModBus?
8. Наведіть переваги та недоліки ModBus.
9. Охарактеризуйте модифікації ProfiBus-протоколу.
10. Які переваги та недоліки має ProfiBus?
11. Які основні властивості має CAN-протокол?
12. Наведіть переваги та недоліки CAN-протоколу.
13. Які якості Ethernet сприяли впровадженню його у промисловість?
14. Назвіть відмінні ознаки промислового Ethernet-протоколу.
15. Що дозволяє технологія Fieldbus?
16. Проведіть порівняльний аналіз протоколів промислових мереж.

РОЗДІЛ 3
СТРУКТУРА ПРОТОКОЛІВ ТА МЕРЕЖ MODBUS І MODBUS TCP

3.1. ModBus: загальні положення

Протокол ModBus і однойменна мережа є найпоширенішими у світі серед протоколів і мереж. Незважаючи на свій вік (ModBus став стандартом де-факто ще в 1979 році) він не тільки не застарів, але, навпаки, істотно зросла кількість орієнтованих на нього нових розробок і збільшується обсяг організаційної підтримки протоколу. Мільйони ModBus-пристроїв по всьому світу продовжують успішно працювати, оновлюються версії опису протоколу.

Одним з головних переваг ModBus є відсутність необхідності в спеціальних інтерфейсних контролерів (PROFIBUS і CAN вимагають для своєї реалізації замовних мікросхем), також до переваг необхідно віднести простоту програмної реалізації та елегантність принципів функціонування. Все це знижує витрати на освоєння стандарту як системними інтеграторами, так і розробниками контролерного обладнання. Високий ступінь відкритості протоколу забезпечується повністю безкоштовними текстами стандартів.

ModBus за поширеністю конкурує тільки з PROFIBUS. Популярність протоколу в даний час пояснюється, перш за все, сумісністю з великою кількістю обладнання, яке підтримує протокол ModBus. Крім того, ModBus має високу достовірність передачі даних, пов'язану із застосуванням надійного методу контролю помилок. ModBus дозволяє уніфікувати команди обміну завдяки стандартизації номерів (адрес) регістрів і функцій їх читання-запису.

Основним недоліком ModBus є мережевий обмін за типом «ведучий/ведений», що не дозволяє веденим пристроям передавати дані з їх появою і тому вимагає інтенсивного опитування ведених пристроїв ведучими. Різновидами ModBus виступають ModBus Plus, що представляє собою багатомайстерний протокол з кільцевою передачею маркера, і протокол ModBus TCP, який розраховано на використання в мережах

Ethernet і Інтернет.

Протокол ModBus має два режими передачі: RTU (Remote Terminal Unit – віддалений термінальний пристрій) і ASCII. Стандарт передбачає, що режим RTU у протоколі ModBus повинен бути присутнім обов'язково, а режим ASCII є опціонним. Користувач може вибирати будь-який з них, але всі модулі, включені в мережу ModBus, повинні мати один і той же режим передачі.

На сьогоднішній день MODBUS підтримує та розвиває організація MODBUS – IDA, вона забезпечує відкритість цього протоколу та розробляє готові компоненти для спрощення реалізації. Згідно стандартів MODBUS – IDA – MODBUS є протоколом прикладного рівня для зв'язку типу Клієнт-Сервер між прикладними процесами пристроїв, що під'єднанні до різноманітних типів шин або мереж. У контексті OSI – моделі, ці мережі мають архітектуру, що наведена на рис. 3.1.

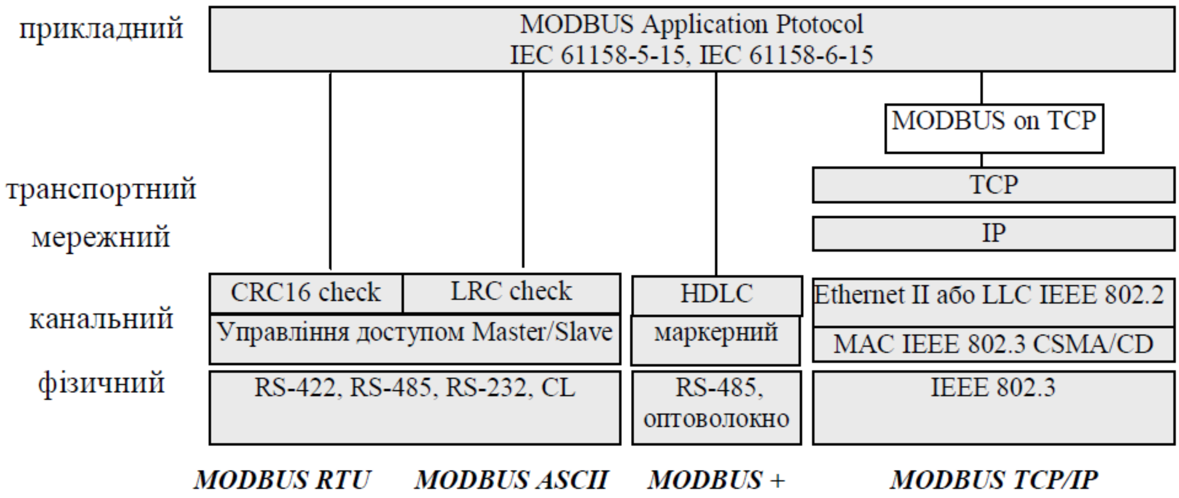


Рисунок 3.1 – MODBUS в контексті моделі OSI

Стандарт ModBus передбачає застосування фізичного інтерфейсу RS-485, RS-422 або RS-232. Зазвичай використовується для організації промислової мережі 2-провідний інтерфейс RS-485. Для з'єднань точка-точка може бути використаний інтерфейс RS-232 або RS-422.

У стандарті ModBus є вимоги обов'язкові, рекомендовані та опціонні (необов'язкові). Існує три ступеня відповідності стандарту:

– повністю відповідає (коли протокол відповідає всім обов'язковим і всім рекомендованим вимогам);

– умовно відповідає (коли протокол відповідає тільки обов'язковим вимогам і не відповідає рекомендованим);

– не відповідає.

Модель OSI протоколу ModBus містить три рівні: фізичний, канальний і прикладний.

3.2. Фізичний рівень

У нових розробках на основі ModBus стандарт рекомендує використовувати інтерфейс PS-485 з двохдротяною лінією передачі, але допускається застосування чотирьох ліній та інтерфейсу PS-232.

У документації на пристрій і на розгалужувач повинні бути вказані найменування підключених кіл.

На кожному кінці магістрального кабелю повинні бути встановлені резистори для узгодження лінії передачі, як це потрібно для інтерфейсу RS-485. На відміну від RS-485 наявність термінальних резисторів відповідно до стандарту ModBus є обов'язковим та не залежить від швидкості обміну. Їх номінал може дорівнювати 150 Ом при потужності 0,5 Вт. Термінальні резистори, а також резистори, що усувають невизначеність стану лінії при високоомному стані передавачів, встановлюються так само, як і в інших мережах на основі фізичного інтерфейсу RS-485. Стандарт вимагає, щоб у документації з експлуатації пристроїв ModBus було сказано, чи є зазначені резистори всередині пристрою або їх необхідно встановлювати під час монтажу мережі. Якщо потрібні зовнішні резистори, то вони повинні мати номінал в інтервалі від 450 до 650 Ом і бути встановлені тільки в одному місці в межах кожного сегмента мережі (сегментами вважаються частини мережі між повторювачами інтерфейсу).

ModBus-пристрій обов'язково має підтримувати швидкості обміну 9600 і 19 200 біт/с. Допускаються також швидкості 1200, 2400, 4800,, ... 38 400 біт/с, 65 і 115 кбіт/с,...

Швидкість передачі має бути витримана в передавачі з похибкою не більше 1 %, а приймач повинен приймати дані під час відхилення швидкості передачі до 2 %.

Сегмент мережі, що не містить повторювачів інтерфейсу, повинен допускати підключення до 32 пристроїв, проте їх кількість може бути збільшена, якщо це допустимо виходячи з навантажувальної здатності передавачів і вхідного опору приймачів.

Максимальна довжина магістрального кабелю при швидкості передачі 9600 біт/с і перетині жил більш 0,13 мм2 (AWG 26) складає 1 км. Відводи від магістрального кабелю не повинні бути довше 20 м. Під час використання багатопортового пасивного розгалуджувача з N відводами довжина кожного відведення не повинна перевищувати значення 40/N м.

ModBus не встановлює конкретних типів з'єднувачів, але якщо використовуються з'єднувачі RJ-45, mini-DIN або D-shell, вони повинні бути екранованими, а їх цокольовка повинна відповідати стандарту.

Для мінімізації помилок під час монтажу рекомендується використовувати дроти таких кольорів: жовтий – для позитивної клеми RS-485 (на якому встановлюється логічна 1, коли через інтерфейс виводиться логічна 1), коричневий – для другої клеми інтерфейсу RS-485, сірий – для загального провідника.

Типовим перетином кабелю є AWG 24 (0,2 мм2, діаметр дроту 0,51 мм). Під час використання кабелю категорії 5 його довжина не повинна перевищувати 600 м. Хвильовий опір кабелю бажано вибирати більш 100 Ом, особливо для швидкості обміну більш 19 200 біт/с.

3.3. Канальний рівень

Протокол ModBus передбачає, що тільки один ведучий пристрій (контролер) і до 247 ведених (модулів вводу-виводу) можуть бути об'єднані до промислової мережі. Обмін даними завжди ініціюється ведучим. Ведені пристрої ніколи не починають передачу даних, поки не отримають запит від ведучого. Також ведені пристрої не можуть обмінюватися даними один з іншим. Тому в будь-який момент часу в мережі ModBus може відбуватися тільки один акт обміну.

Адреси з 1 по 247 є адресами ModBus-пристроїв у мережі, а з 248 по 255 зарезервовані. Ведучий пристрій не повинен мати адреси, і в мережі не повинно бути двох пристроїв з однаковими адресами.

Ведучий пристрій може надсилати запити всіх пристроїв одночасно (широкомовний режим) або тільки одному. Для широкомовного режиму зарезервовано адресу 0 (під час використання у команді цієї адреси вона приймається всіма пристроями мережі).

3.3.1. Опис кадру (фрейму) протоколу ModBus

У протоколі ModBus RTU повідомлення починає сприйматися як нове після паузи (тиші) на шині тривалістю не менше 3,5 шістнадцятирічних символів (14 біт), тобто величина паузи в секундах залежить від швидкості передачі.

Формат кадру показаний на рис. 3.2. Поле адреси завжди (навіть у відповідях на команду, яку надіслану ведучим) містить тільки адресу відомого пристрою. Завдяки цьому ведучий пристрій знає, від якого модуля прийшла відповідь.

ADU			
Адреса 1 байт	Код функції 1 байт	Дані 0…255 байт	Контрольна сума 2 байт
	PDU		

Рисунок 3.2 – Формат кадру протоколу ModBus RTU: PDU (protocol data unit) – елемент даних протоколу; ADU (application data unit) – елемент даних додатку

Поле «Код функції» показує модулю те, яку дію потрібно виконати. Поле «Дані» може мати довільну кількість байтів у діапазоні від 0 до 255. У ньому може міститися інформація щодо параметрів, які використовуються в запитах контролера або відповідях модуля. Поле «Контрольна сума» містить контрольну суму CRC довжиною 2 байта.

3.3.2. Структура даних у режимі RTU

У режимі RTU дані передаються молодшими розрядами (див. рис. 3.3). Під час замовчування в режимі RTU біт паритету встановлюють 1, якщо кількість двійкових одиниць в байті непарна, та 0, якщо вона парна. Такий паритет називають парним (even parity), а метод контролю називають контролем парності. Якщо парна кількість двійкових одиниць у байті – біт паритету може дорівнювати 1. У цьому випадку говорять, що паритет є непарним (odd parity).

Стартовий біт	1 МЗР	2	3	4	5	6	7	8	Біт паритету	Стоп-біт

Рисунок 3.3 – Послідовність бітів у режимі RTU
(МЗР – молодший значущий розряд; під час відсутності біта паритету на його місце записується другий стоп-біт)

Контроль парності може бути відсутнім взагалі. У цьому випадку замість біта паритету повинен використовуватися другий біт. Для забезпечення максимальної сумісності з іншими пристроями рекомендується використовувати можливість заміни біта паритету на другий біт. Ведені пристрої можуть сприймати будь-який з варіантів: парний, непарний паритет або його відсутність.

3.3.3. Структура повідомлення ModBus RTU

Повідомлення ModBus RTU передаються у вигляді кадрів, для кожного з яких відомі початок і кінець. Ознакою початку кадру є пауза (тиша) тривалістю не менше 3,5 шістнадцятирічних символів (14 біт). Кадр повинен передаватися безперервно. Якщо під час передачі кадру виявляється пауза тривалістю більше 1,5 шістнадцятирічних символів (6 біт), то вважається, що кадр містить помилку і повинен бути відхилений модулем приймача. Ці величини пауз повинні строго дотримуватися при швидкостях нижче 19 200 біт/с, однак при більш високих швидкостях рекомендується використовувати фіксовані значення паузи – 1,75 мс і 750 мкс відповідно.

3.3.4. Контроль помилок

У режимі RTU є два рівня контролю помилок у повідомленні:
– контроль паритету для кожного байта (опційно);
– контроль кадру в цілому за допомогою CRC-методу.

CRC-метод використовується незалежно від перевірки паритету. Значення CRC встановлюється у ведучому пристрої перед передачею. Під час прийому повідомлення розраховується код CRC для всього повідомлення і порівнюється з його значенням, зазначеним у полі CRC кадру. Якщо обидва значення збігаються, то вважається, що повідомлення не містить помилки. Стартові, стопові біти і біт паритету під час розрахунку CRC не беруть участі.

3.4. Прикладний рівень

Прикладний рівень ModBus RTU забезпечує комунікацію між пристроями типу «ведучий/ведений». Прикладний рівень є незалежним від фізичного і канального, зокрема, він може використовувати протоколи Ethernet TCP/IP (ModBus TCP/IP), ModBus Plus (багатомастерна мережа з передачею маркера), інтерфейси RS-232, RS-422. RS-485, оптоволоконні лінії, радіоканали та інші фізичні середовища для передачі сигналів.

Прикладний рівень ModBus заснований на запитах за допомогою кодів функцій. Код функції вказує веденому пристрою, яку операцію він повинен виконати.

Під час використання протоколу прикладного рівня з різними протоколами транспортного і канального рівня зберігається незмінним основний блок ModBus – повідомлення, разом із кодом функції та дані (цей блок називається PDU – Protocol Data Unit – елемент даних протоколу). До блоку PDU можуть додаватися додаткові поля під час використання його в різних промислових мережах, і тоді він називається ADU – Application Data Unit - елемент даних програми.

3.4.1. Коди функцій

Стандартом ModBus передбачені три категорії кодів функцій: встановлені стандартом, що задаються користувачем і зарезервовані.

Коди функцій є числами у діапазоні від 1 до 127, причому коди у діапазоні від 65 до 72 та від 100 до 110 відносяться до заданих користувачем функцій. Коди у діапазоні від 128 до 255 зарезервовані для пересилання кодів помилок у відповідному повідомленні. Код 0 не використовується. Коди помилок використовуються веденим пристроєм, щоб визначити, яку дію зробити для їх обробки. Значення кодів і їх зміст описані в стандарті на ModBus RTU.

Поле даних (див. рис. 3.2) у повідомленні, яке надіслано від ведучого пристрою веденому, містить додаткову інформацію, яку ведений пристрій використовує, щоб виконати функцію, зазначену в полі «Код функції». Поле даних може містити значення станів дискретних входів/виходів, адреси регістрів, з яких треба зчитувати (записувати) дані, кількість байтів даних, посилання на змінні, кількість змінних, код підфункцій і т.п. Якщо ведений пристрій нормально виконав прийняту від ведучого функцію, то у відповіді поле «Код функції» містить ту ж інформацію, що і у запиті. У іншому випадку ведений видає код помилки. У разі помилки код функції у відповіді дорівнює коду функції у запиті, збільшеному на 128.

3.4.2. Зміст поля даних

У повідомленні ведучого пристрою веденому поле даних містить додаткову інформацію, яка необхідна для виконання зазначеної функції. Наприклад, якщо код функції вказує, що необхідно вважати дані з групи регістрів пристрою введення (код функції 03 hex), то поле даних містить адресу початкового регістра і кількість регістрів. Якщо ведучий пристрій посилає команду запису даних до групи регістрів (код функції 10 hex), то поле даних повинне містити адресу початкового регістра, кількість регістрів, кількість байтів даних і дані для запису до регістру. Конкретний зміст поля даних встановлюється стандартом для кожної функції окремо. У деяких повідомленнях поле даних може мати нульову довжину.

3.5. Список кодів ModBus

У табл. 3.1 наведено приклад кодів ModBus RTU для модуля дискретного введення і виведення типу RealLab типу NL-1 601 (фірми НДЛ АП). Для читання логічних станів входів модуля через інтерфейс 115-485 необхідно надіслати команду в форматі, який наведено на рис. 3.2, де в полях «Звідки» і «Код функції» вказуються значення з відповідних граф табл. 3.1.

Таблиця 3.1 – Приклад кодів ModBus RTU для модуля RealLab типу NL-1601

Позначка регістра	Hex-адреса регістра	Що читається або записується	Код функції читання регістра	Код функції запису до регістру	Примітка
00001	00h 00h	Дискретний вихід 0	01	05	1 або 0
00002	00h 01h	Дискретний вихід 1	01	05	1 або 0
10001	00h 00h	Дискретний вхід 0	02	–	1 або 0

10002	00h 01h	Дискретний вхід 1	02	–	1 або 0
10003	00h 02h	Дискретний вхід 2	02	–	1 або 0
10004	00h 03h	Дискретний вхід 3	02	–	1 або 0
10005	00h 04h	Дискретний вхід 4	02	–	1 або 0
10006	00h 05h	Дискретний вхід 5	02	–	1 або 0
10007	00h 06h	Дискретний вхід 6	02	–	1 або 0
10008	00h 07h	Дискретний вхід 7	02	–	1 або 0
10009	00h 08h	Дискретний вхід 8	02	–	1 або 0
10010	00h 09h	Дискретний вхід 9	02	–	1 або 0
10011	00h OAh	Дискретний вхід 10	02	–	1 або 0
10012	00h OBh	Дискретний вхід 11	02	–	1 або 0
10013	00h OCh	Дискретний вхід 12	02	–	1 або 0
10014	00h ODh	Дискретний вхід 13	02	–	1 або 0
10015	00h OEh	Дискретний вхід 14	02	–	1 або 0
10016	00h OFh	Дискретний вхід 15	02	–	1 або 0
40201	00h C8h	Ім'я модуля	03	10	–
40213	00h D4h	Версія програми	03	–	–
40513	02h 00h	Адреса модуля	03	06	000111-00 P711 (допустимий діапазон значень)
40513	02h 00h	Адреса модуля	03	06	000111-00P711 (дозволений діапазон значень)
40514	02h 01h	Швидкість UART	03	06	000311-000A11 (дозволений діапазон значень)
40518	02h 05h	Протокол	03	06	000011 – АБСИ, 000111 – ти
40769	03h 00h	Значення на виході після включення живлення модуля Power On Value	03	06	000011-000311 (дозволений діапазон значень)

3.6. ModBus TCP

Протокол ModBus TCP 15 (або ModBus TCP/IP) використовується для того, щоб підключити пристрої з протоколом ModBus до Ethernet або мережі Internet. Він використовує кадри ModBus RTU на 7-му (прикладному) рівні моделі OSI, протоколи Ethernet на 1-му і 2-му рівнях моделі OSI і TCP/IP на 3-му і 4-му рівнях, тобто Ethernet TCP/IP використовується для транспортування модифікованого кадру ModBus RTU. Кадр ModBus RTU (див. рис. 3.2) у цьому випадку не має поля контрольної суми, оскільки використовується стандартна контрольна сума Ethernet TCP/IP; немає також поля адреси, оскільки в Ethernet використовується інша систем адресації. Таким чином, тільки два поля – «Код функції» і «Дані» (блок PDU) вбудовуються в протокол Ethernet TCP/IP. Перед ними вставляється нове поле (див. рис. 3.4) – MBAP (ModBus Application Protocol – прикладний протокол ModBus).

<table>
<tr><td colspan="6">ADU для ModBus TCP</td></tr>
<tr><td>Ідентифікатор обміну 2 байта</td><td>Ідентифікатор протоколу 2 байта</td><td>Довжина 2 байта</td><td>Ідентифікатор пристрою 1 байт</td><td>Код функції 1 байт</td><td>Дані (змінної довжини)</td></tr>
<tr><td colspan="4"></td><td colspan="2"></td></tr>
<tr><td colspan="4">MBAP</td><td colspan="2">DPU</td></tr>
</table>

Рисунок 3.4 – Частина фрейма ModBus TCP, вбудована в поле «Дані» фрейму Ethernet

Поле «Ідентифікатор обміну» використовується для ідентифікації повідомлення, коли в межах одного TCP-з'єднання клієнт посилає серверу кілька повідомлень без очікування відповіді після кожного повідомлення. Поле «Ідентифікатор протоколу» містить нулі та зарезервовано для майбутніх застосувань. Поле «Довжина» вказує кількість наступних за ним байтів. Поле «Ідентифікатор пристрою» ідентифікує віддалений сервер, розташований поза мережею Ethernet (наприклад, у мережі ModBus RTU, яка з'єднана з Ethernet за допомогою мережевого моста). Зазвичай це поле містить нулі або одиниці, ігнорується сервером і відправляється назад у тому ж вигляді (як відлуння).

Зображений на рис. 3.4 фрейм називається фреймом ADU, вбудовується в поле «Дані» фрейму Ethernet і надсилається через TCP-порт 502, спеціально зарезервований для ModBus TCP (порти призначаються і контролюються організацією IANA - Internet Assigned Numbers Authority, www.iana.org). Клієнти і сервери ModBus посилають, отримують і прослуховують повідомлення через TCP – порт 502.

Таким чином, структура кадру і сенс його полів «Код функції» і «Дані» для ModBus і ModBus TCP абсолютно ідентичні, тому для роботи з ModBus TCP не потрібно додаткового навчання при знанні ModBus RTU. Ті ж самі коди функцій і дані, що і в ModBus RTU, передаються по черзі з прикладного (7-го) рівня моделі OSI (див. рис. 3.5) на транспортний рівень, який додає до блоку PDU кадру ModBus RTU (див. рис. 3.2) заголовок з протоколом TCP.

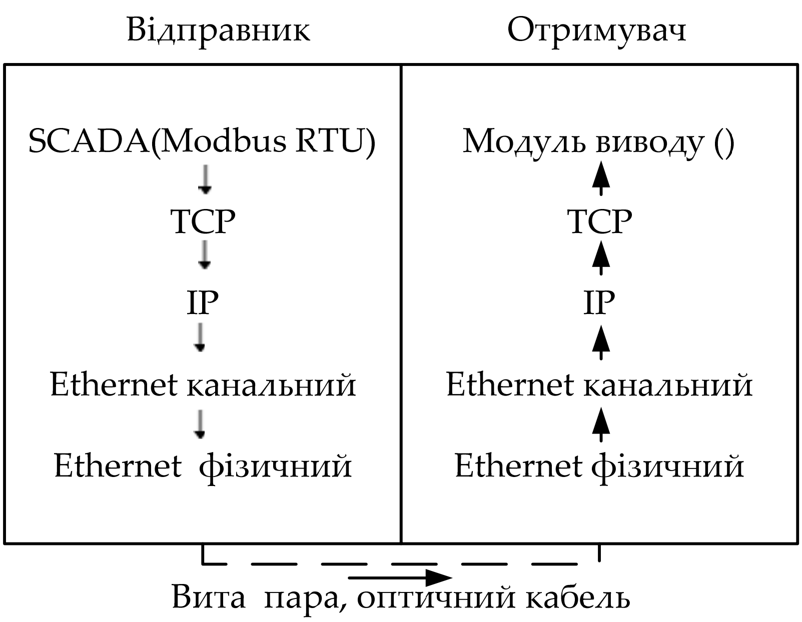


Рисунок 3.5 – Процес передачі кадру ModBus RTU за рівнями моделі OSI через стек протоколів Ethernet TCP / IP у мережах з протоколом ModBus TCP

Далі новий отриманий кадр передається на мережевий рівень, де до нього додається заголовок IP, потім він передається на канальний рівень Ethernet і на фізичний. Дійшовши до фізичного рівня, блок PDU виявляється отримав заголовки протоколів всіх рівнів, через які він пройшов. Пройшовши по лінії зв'язку, повідомлення просувається від низу до верху за стеком протоколів (рівням моделі OSI) у пристрої одержувача, де на кожному рівні з нього видаляється відповідний заголовок, а на прикладному рівні виділяється блок PDU (код функції і дані) кадру протоколу ModBus RTU.

У мережі з протоколом ModBus TCP пристрої взаємодіють за типом «клієнт-сервер», де як клієнт виступає ведучий пристрій, як сервер – ведений. Сервер не може ініціювати зв'язок у мережі, але деякі пристрої в мережі можуть виконувати роль як клієнта, так і сервера.

ModBus TCP не має багатоабонентського режиму, він здійснює з'єднання тільки між двома пристроями.

3.7. Формат основних функцій

Повний список кодів, а також специфікацію протоколу можна знайти на офіційному Web сайті MODBUS-IDA – www.MODBUS.org. Номер функції представлений у шістнадцятирічному форматі. Скорочення в дужках *Hi* та *Lo*, що вказують відповідно на старший та молодший байти. Тобто, якщо для вказівки адреси початкової змінної необхідно двохбайтне слово, то значення старшого байта буде передаватись у полі з позначенням *Hi*, а молодшого – відповідно *Lo*.

Код функції 01_{16} – читання статусу Coils (дискретних вихідних бітів). Повідомлення-запит вміщує адресу початкового біту і кількість бітів для читання. Біти нумеруються починаючи з 0. У повідомленні-відповіді кожне значення змінної передається одним бітом, тобто в одному байті пакується статус 8 бітових змінних. Якщо кількість їх не кратна восьми, інші біти у байті заповнюються нулями. Лічильник вміщує кількість байт у полі даних.

Запит:

Код функції	01
Адреса початкового біту (Hi)	0 до $FFFF_{16}$
Адреса початкового біту (Lo)	
Кількість біт (Hi)	1 до $7D0_{16}$ (2000)
Кількість біт (Lo)	

Відповідь:

Код функції	01
Лічильник байт	N
Значення бітів (перші 8 біт)	0 до FF_{16}
Значення бітів (наступні 8 біт)	0 до FF_{16}
...	
Значення бітів (N-ні 8 біт)	0 до FF_{16}

Код функції 0216 – читання статусу дискретних входів. Формат цього запиту такий же як попередній, за винятком поля функції.

Код функції 0316 – читання значення вихідних/внутрішніх регістрів. Повідомлення-запит вміщує адресу початкового вихідного/внутрішнього регістру (двохбайтове слово), і кількість регістрів для читання. Регістри нумеруються починаючи з 0.

У відповідному повідомленні в полі даних кожний регістр передається двома байтами.

Запит:

Код функції	03
Адреса початкового регістру (Hi)	від 0 до FFFF^{16}
Адреса початкового регістру (Lo)	
Кількість регістрів (Hi)	від 1 до 7D^{16} (125)

Відповідь:

Код функції	01
Лічильник байт	N*2
Значення 1-го регістру (Hi)	0 до FFFF^{16}
Значення 1-го регістру (Lo)	
…	
Значення N-го регістру	0 до FFFF^{16}
Значення N-го регістру	

Код функції 0416 – читання значення вхідних регістрів. Формат цього запиту такий же як попереднього, за винятком поля функції.

Код функції 0516 – запис вихідного/внутрішнього біту. У запиті вказується номер бітової змінної та значення: 0 – 0000, а 1 – FF00, всі інші значення не змінюють стан змінних. У широкомовній передачі клієнтський запит виставляє значення цієї змінної для всіх серверів.

Функція	05
Адреса біту (Hi)	від 0 до FFFF16
Адреса біту (Lo)	
Значення біту (Hi)	0000 або FF0016
Значення біту (Lo)	

Нормальна відповідь серверу являється повторенням запиту до клієнта.

Код функції 0616 – запис вихідного/внутрішнього регістру. Функція аналогічна попередній, але оперує з регістрами (словами). У запиті вказується номер вихідного/внутрішнього регістру та його значення. У широкомовній передачі запит виставляє значення цієї змінної для всіх серверів.

Функція	06
Адреса регістру (Hi)	від 0 до FFFF16
Адреса регістру (Lo)	
Значення регістру (Hi)	від 0 до FFFF16
Значення регістру (Lo)	

Нормальна відповідь сервера є повторенням запиту клієнту.

Код функції 0F16 – запис декількох вихідних/внутрішніх бітів. У запиті вказується початкова адреса біту, кількість бітів для запису, лічильник байтів і безпосередньо значення. У широкомовній передачі біти записуються всім серверам. Розглянемо приклад для встановлення таких бітових вихідних/внутрішніх змінних:

Байт 1								Байт 2							
6	5	4	3	2	1	0	9							8	7

У таблиці показана відповідність адреси змінної, починаючи з 19-ї, і значення біту. Для зручності біти розміщені у тому порядку, що і передаються. В другому байті корисні тільки 2 перші біти, значення інших не буде прийнято до уваги, оскільки кількість бітів вказані у кадрі. Запит та відповідь будуть мати такий вигляд:

Запит:

Функція	0F
Адреса початкового біту (Hi)	00
Адреса початкового біту (Lo)	13
Кількість бітів (Hi)	00
Кількість бітів (Lo)	0A
Лічильник байтів	02
Дані(змінні 19-26)	CD
Дані(змінні 27-28)	01

Відповідь:

Функція	0F
Адреса початкового біту (Hi)	00
Адреса початкового біту (Lo)	13
Кількість бітів (Hi)	00
Кількість бітів (Lo)	0A

Код функції 10_{16} – запис декількох вихідних/внутрішніх регістрів.

Запит:

Функція	10_{16}
Адреса початкового регістру (Hi)	0 до $FFFF_{16}$
Адреса початкового регістру (Lo)	
Кількість регістрів (Hi)	1 до $007B_{16}$
Кількість регістрів (Lo)	
Лічильник байтів	2*N
Дані (1-й регістр Hi)	0 до $FFFF_{16}$
Дані (1-й регістр Lo)	
…	
Дані (N-й регістр Hi)	0 до $FFFF_{16}$
Дані (N-й регістр Lo)	

Відповідь:

Функція	10
Адреса початкового регістру (Hi)	0 до $FFFF_{16}$
Адреса початкового регістру Lo	
Кількість регістрів (Hi)	1 до $007B_{16}$ (123)
Кількість регістрів (Lo)	

3.8. MODBUS TCP/IP

3.8.1. Комунікаційна архітектура MODBUS TCP/IP

Мережі MODBUS TCP/IP базуються на стику протоколів TCP/IP і перш за все призначені для роботи на базі Ethernet. MODBUS TCP/IP описаний в специфікаціях MODBUS-IDA, в яких комунікаційна система MODBUS TCP/IP може включати різні типи пристроїв (див. рис. 3.6): MODBUS TCP/IP Клієнти і Сервери підключені до TCP/IP мережі; міжмережні пристрої типу мостів, маршрутизаторів або шлюзів для з'єднання TCP/IP мережі з послідовними лініями підмереж, що дозволяє обмінюватися даними MODBUS Serial Клієнтськими і Серверними пристроями.

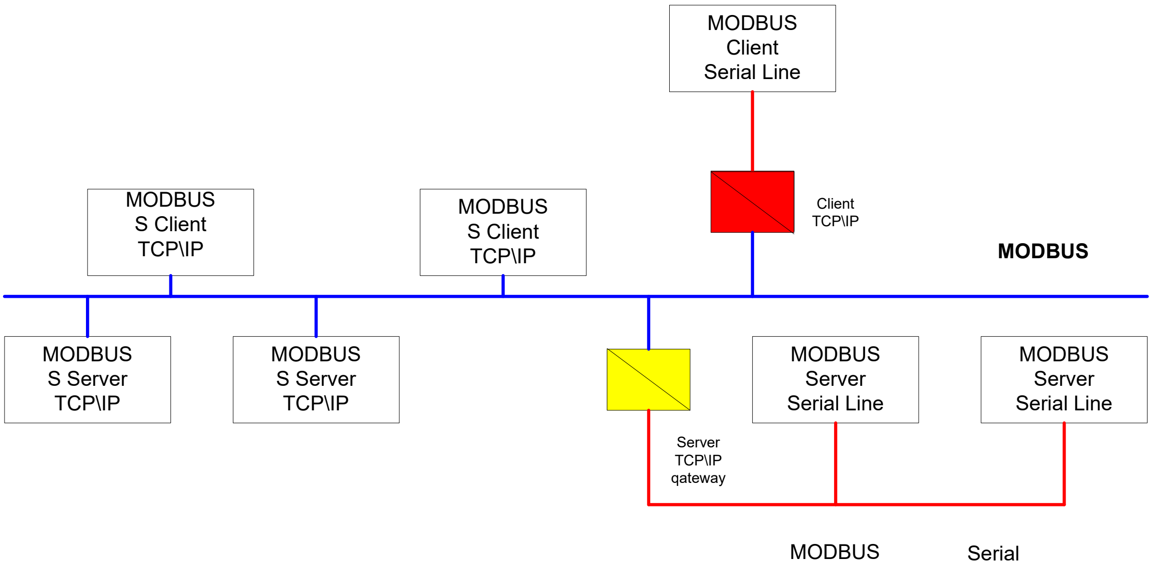


Рисунок 3.6 – Комунікаційна архітектура MODBUS TCP/IP

Таким чином комунікаційна система MODBUS TCP/IP дозволяє обмінюватися пристроям не тільки на мережах зі стеком TCP/IP, а і з пристроями на послідовних лініях зв'язку (MODBUS RTU/ASCII або MODBUS+). Як приклад можна привести рисунок 3.7, який взятий зі специфікації протоколу прикладного рівня.

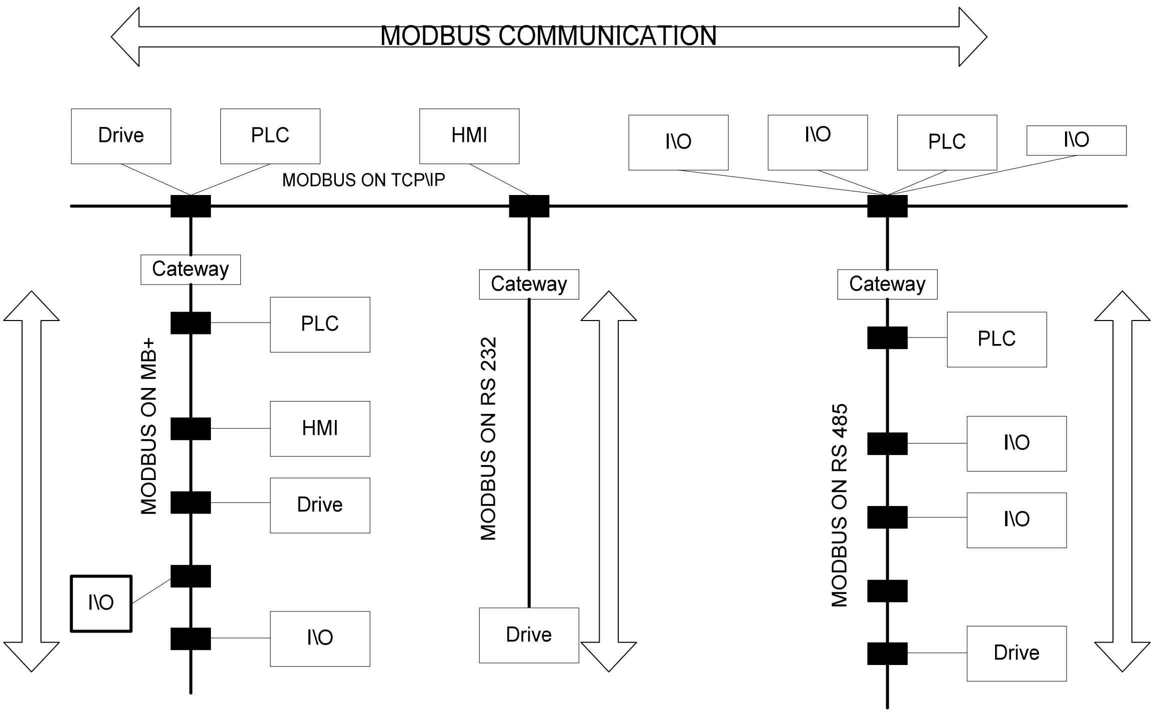


Рисунок 3.7 – Приклад архітектури мережі MODBUS

3.8.2. Особливості реалізації протоколу

Аналогічно всім мережам MODBUS, дані мережі використовують MODBUS Application Protocol. У MODBUS Serial на канальному рівні до PDU додається адреса веденого і контрольна сума, сам PDU не модифікується. Однак у MODBUS TCP/IP перед попаданням на транспортний рівень, до PDU (код функції та дані) додається додатковий MBAP-заголовок (див. рис. 3.8). Заголовок складається з полів, які описані в табл. 3.2. Отриманий модуль передається рівню TCP.

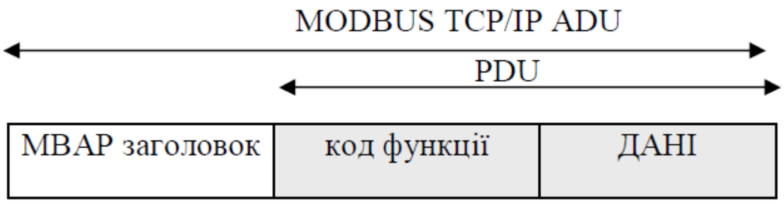


Рисунок 3.8 – Формат модуля даних прикладного рівня (APDU) MODBUS TCP/IP

Таблиця 3.2 – Поля MBAP заголовка

Поле	Довжина (байт)	Пояснення	Клієнт	Сервер
TransactionID	2	ідентифікація транзакцій запитів/ відповідей	ініціалізує Клієнт в запиті	копіює з запиту у повідомлення - відповідь
ProtocolID	2	тип протоколу, 0=MODBUS протокол	ініціалізує Клієнт в запиті	копіює з запиту у повідомлення - відповідь
Length	2	кількість наступних байтів	ініціалізує Клієнт в запиті	ініціалізує Сервер у відповіді
UnitID	1	адреса Веденого, який підключений до вузла	ініціалізує Клієнт в запиті	копіює з запиту у повідомлення - відповідь

Зверніть увагу, що за допомогою поля UnitID можна вказати адресу вузла в MODBUS Serial, наприклад. адресу веденого в MODBUS RTU. Якщо потрібно адресувати вузол, безпосередньо підключений по TCP/IP то UnitID=0.

Поле ProtocolID використовується для міжсистемного мультиплексування. Так TCP порт для MODBUS Серверу має номер 502, однак цей же порт, наприклад, використовує Schneider Electric для UNITE-Серверу. Таким чином замінивши поле ProtocolID, Ethernet модулі ПЛК Schneider Electric одночасно підтримують два протоколи: MODBUS та UNI-TE.

Для ідентифікації пристрою, якому передається запит, вказується його IP-адреса. Для ідентифікації TPDU, які направляються MODBUS-Серверу, як прикладному об'єкту, виділений 502-гий TCP-порт.

Контрольні запитання та завдання

1. Якими мережами на сьогоднішній день представлений MODBUS? Охарактеризуйте їх у контексті моделі OSI.
2. Розкажіть про основи функціонування MODBUS Application Protocol. Який формат повідомлення MODBUS PDU?
3. До яких змінних (даних процесу) можна доступитися через MODBUS функції?
4. Як MODBUS-серверний процес дізнається про помилковий результат обробки повідомлення-запиту?
5. Які функції використовуються для доступу до даних процесу?
6. Як формуються повідомлення-запити та повідомлення відповіді для читання та запису діапазону вхідних та вихідних регістрів?
7. Які ситуації можливі під час обробки запиту MODBUS Клієнта? Наведіть приклади відповідей про помилку.
8. Прокоментуйте основи функціонування MODBUS RTU/ASCII в контексті моделі OSI. Як пов'язана модель функціонування обміну на прикладному рівні з функціонуванням на канальному?
9. Яка модель адресації та який метод доступу до шини використовується в MODBUS RTU/ASCII на канальному рівні? Як на цьому рівні проводиться контроль за правильністю доставки бітової послідовності?

10. Розкажіть про принципи побудови кадрів для MODBUS RTU.

11. Прокоментуйте основи функціонування MODBUS TCP/IP в контексті моделі OSI. Які типи пристроїв передбачає комунікаційна архітектура MODBUS TCP/IP?

12. Які поля включає формат модуля даних прикладного рівня MODBUS TCP/IP? Поясніть призначення полів заголовка MBAP?

13. Яка адреса порту TCP використовується для MODBUS Сервера?

14. Яким чином забезпечується доступ до необхідного вузла MODBUS RTU/ASCII, підключених через шлюз до MODBUS TCP/IP в заголовку MBAP?

РОЗДІЛ 4
СТРУКТУРА ПРОТОКОЛІВ ТА МЕРЕЖі PROFIBUS

4.1. Загальна характеристика Profibus, відповідність моделі OSI

PROFIBUS (англ. Process Field Bus) — незалежний від виробника відкритий стандарт польової шини для широкого спектру застосувань у виробничій автоматизації. Назву мережі можна приблизно перекласти, як "промислова шина для технологічних процесів". Стандарт Profibus було спочатку прийнято в Німеччині в 1987 році, потім, в 1996 році, він став міжнародним (EN 50170 и EN 50254).

Прототип мережі було розроблено компанією Siemens AG для своїх промислових контролерів SIMATIC. PROFIBUS поширений в Європі, особливо в машинобудуванні та управлінні промисловим обладнанням. Підтримкою, стандартизацією та розвитком мереж цього стандарту займається Profibus Network Organisation (PNO).

PROFIBUS об'єднує технологічні та функціональні особливості послідовного зв'язку рівня польової шини. Вона дозволяє об'єднувати розрізнені пристрої автоматизації в єдину систему на рівні давачів і приводів. Мережа PROFIBUS — це комплексне поняття, вона базується на декількох стандартах і протоколах.

Мережа побудована згідно з багаторівневою моделлю ISO 7498-OSI та складається з трьох рівнів моделі:

1 — фізичного рівня;

2 — канального рівня;

7 — прикладного рівня.

Profibus існує в трьох основних різновидах, які називаються PNO-профілями — DP, FMS та PA. Рівень 7 використовує лише один один з профілей— Profibus FMS, 1 і 2 рівні використовують всі профілі. Відповідність моделі OSI показано у таблиці 1.

Фізично Profibus — це електрична мережа, в основі якої є екранована кручена пара або оптична мережа на основі оптоволоконного кабелю. Швидкість передачі по ній може змінюватися від 9.6 кбіт/с до 12 Мбіт/с. Найчастіше використовується кручена пара з інтерфейсом RS-485.

Приклад структури мережі Profibus-DP показано на рисунку 4.1.

Таблиця 4.1 Відповідність Profibus моделі OSI

<table>
<tr><td rowspan="3"></td><td>PNO-профіль для DP-пристроїв</td><td rowspan="2">PNO-профіль для FMS-пристроїв</td><td>PNO-профіль для PA-пристроїв</td></tr>
<tr><td>Основні функції
Розширені функції</td><td>Основні функції
Розширені функції</td></tr>
<tr><td>DP User Interface
Direct Data ink Mapper (DDLM)</td><td>Application-Layer Interface (ALI)</td><td>DP User Interface
Direct Data ink Mapper (DDLM)</td></tr>
<tr><td>Layer 7 (Прикладний)</td><td rowspan="2">Не використовується</td><td>Fieldbus Message Specification (FMS)</td><td rowspan="2">Не використовується</td></tr>
<tr><td>Layer 3-6</td><td>Не використовується</td></tr>
<tr><td>Layer 2 (Канальний)</td><td>Data Link Layer
Fielbus Data Link</td><td>Data Link Layer
Fielbus Data Link</td><td>IEC-interface</td></tr>
<tr><td>Layer 1 (Фізичний)</td><td>Physical Layer (RS485/LWL)</td><td>Physical Layer (RS485/LWL)</td><td>IEC 1185-2</td></tr>
</table>

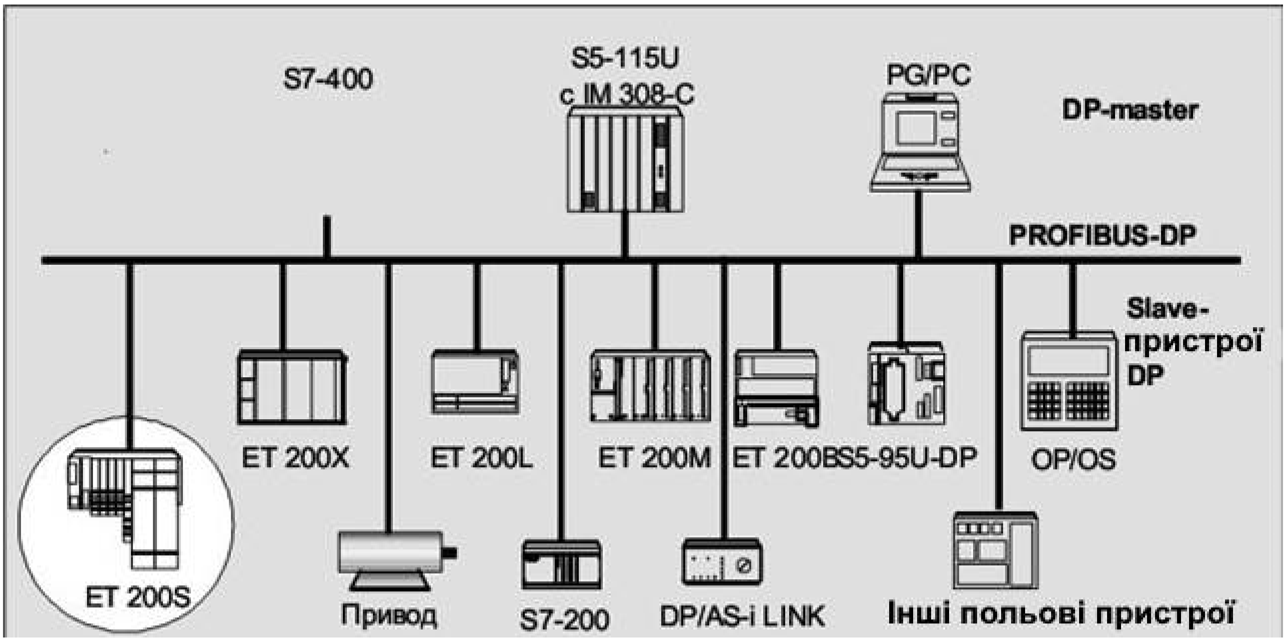


Рисунок 4.1 - Структура мережі Profibus-DP

4.2. Загальна характеристика профілей мережі Profibus

Профілі мережі Profibus сумісні між собою та знайшили широке застосування в сучасних виробництвах:

- PROFIBUS DP (Decentralized Peripherals — Децентралізована периферія) використовується для керування давачами та приводами керування через централізований прилад керування на виробництві. В цьому стандарті приділяється велика увага стандартним функціям діагностики обладнання. Безпосередній доступ з програми користувача до канального рівня здійснюється за допомогою елемента DDLM (Direct Data Link Mapper - "прямий перетворювач для канального рівня")
- PROFIBUS PA (Process Automation — Автоматизація процесів) використовується для спостереження за датчиками в системах автоматизації керування виробничими процесами. Цей варіант розроблений для використання у небезпечних/вибухонебезпечних зонах. Фізичний рівень (кабель передачі даних) відповідає стандарту IEC 61158-2, що дозволяє передачу даних по ньому до виробничого обладнання, та обмежити силу струму для запобігання вибухонебезпечних ситуацій навіть у випадку збоїв. Це накладає обмеження на кількість підключених

приладів автоматизації процесу виробництва. Прилади автоматизації процесу виробництва передають дані зі швидкістю 31.25 кбіт/с. Однак, прилади варіанту PA використовують протокол ідентичний варіанту DP, що дозволяє підключати прилади до мережі DP через перехідник. Набагато швидший варіант DP може служити центральною шиною для передачі сигналів на прилади керування. Завдяки цьому, обладнання DP та PA можна поєднувати в єдину мережу, особливо в гібридних системах, в яких обладнання автоматизації процесів виробництва та керування тісно пов'язані між собою.

– PROFIBUS FMS (Field bus Message Specification — Специфікація повідомлень польової шини) - протокол призначений в основному для зв'язку програмованих контролерів один з одним і станціями оператора на верхньому рівні. Він використовується в тих областях, де високий ступінь функціональності важливіший ніж ніж швидкий час реакції системи. При зв'язку через FMS використовуються відносини типу клієнт-сервер. Як приклад пристроїв, сполучених по FMS протоколу можна узяти з устаткування фірми Siemens, - SIMATIC S7 з FMS-CP або, наприклад, SIMATIC S5 з CP 5431FMS.

Дуже часто використовується комбінований режим роботи пристроїв PROFIBUS FMS і PROFIBUS DP, в цьому випадку між ведучими і веденими пристроями використовується протокол DP, а між самими ведучими - протокол FMS. В наш час застосування протоколу Profibus FMS скорочується, у зв'язку з переходом до промислового Ethernet і Profinet.

4.3. Вимоги та реалізація фізичного рівня

На цьому рівні можуть використовуватись три різні методи передачі:

– Передача електричних сигналів відповідно до EIA-485 витою парою з імпедансом 150 Ом у топології шини. Швидкості передачі становлять від 9,6 Кбіт/с до 12 Мбіт/с. Довжина кабелю між двома ретрансляторами обмежена від 100 до 1200 м, у залежності від потрібної швидкості передачі даних. Цей метод передачі використовується головним чином з PROFIBUS DP.
– Передача через оптичне волокно у топології зірки, шини чи кільця. Відстань між ретрансляторами може становити до 15 км. Кільцева топологія також може бути виконана надлишково.

– При використанні MBP (Manchester Bus Powered) передача даних і живлення польової шини здійснюються одним і тим же кабелем. Потужність може бути зменшена таким чином, щоб була можливість використання у вибухонебезпечних середовищах. Топологія шини передбачає до 1900 м довжини кабелю і дозволяє розгалуження для периферійних пристроїв (максимально до 60 м на відгалуження). Швидкість передачі даних тут є фіксованою 31,25 кбіт/с. Ця технологія була спеціально створена для використання PROFIBUS PA.

4.3.1. Реалізація для DP/FMS на крученій парі — RS485

Найбільш поширено використання на рівні 1 симетричної передачі даних за стандартом EIA RS485 (також позначається H2). Дроти шинних сегментів замкнені з обох кінців, скручені та екрановані — рисунок 4.2.

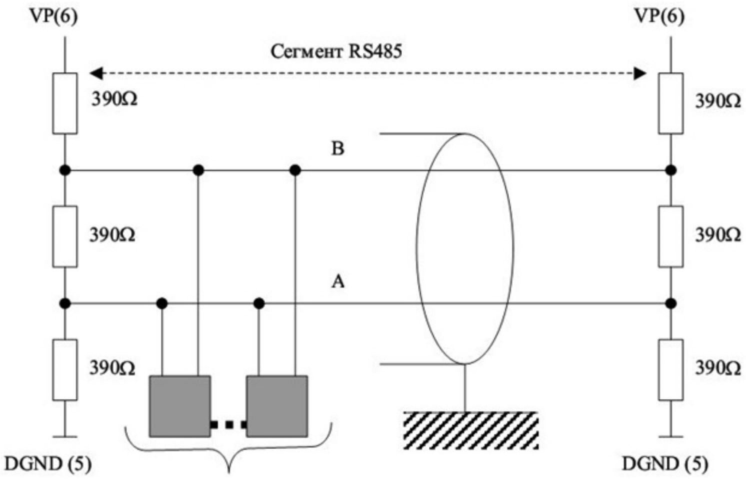


Рисунок 4.2 - Структура шинного сегмента RS485

Визначено полудуплексний, асинхронний спосіб передачі RS485. Дані передаються внутрі 11-розрядного кадру (рис. 4.3) в NRZ-коде (Non Return to Zero) , осцилограму якого показано на рисунку 4.4. Значення сигналу (біти) не міняються під час передачі сигналу. Кадри згруповано в логічні блоки, що називаються телеграмами.

В той час, як передача двійкового значення "1" відповідає позитивному потенціалу на дроті RxD/TxD-P (Receive/Transmit-Data-P), навпаки, на дроті RxD/TxD-N (Receive/Transmit-Data-N) в наявності "0".

Стану спокою між окремими телеграмами відповідає двійковий сигнал "1".

Рисунок 4.3 Структура кадру

В літературі часто також обидва дроти позначають як А-дріт і В-дріт. При цьому А-дріт відповідає RxD/TxD-N, а В-дріт — RxD/TxD-P.

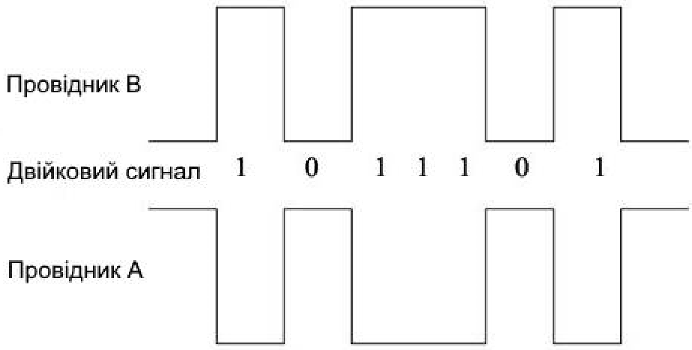


Рисунок 4.4 Осцилограма коду Non Return to Zero (NRZ)

Технічні вимоги для побудування фізичного рівня наведено у таблиці 4.2.

Таблиця 4.2 - Вимоги до фізичного рівня

Швидкість передачі даних	9,6 ... 12000 кБіт/сек
Дистанція зв'язку	100 ... 1200 м
Кількість пристроїв у сегменті	до 32
Волновий опір кабелю	135 ... 165 Ом
Погонна ємність кабелю	до 35пФ/м
Погонний опір кабелю	До 110 Ом/м
Кількість дротів	2 або 4
Наявність екрану	фольга або обплітка
Рекомендоване рознімання	DE-9 або M-12

Додаткові вимоги до фізичної організації мережі на крученій парі такі:

- на стороні пристрою має бути встановлене рознімання "гніздо", на стороні кабелю - "штекер"
- в самому рознімання встановлюється термінальний резистор з вимикачем
- для роботи з швидкістю більше 1,5 Мбіт/сек в рознімані встановлюється друкована індуктивність

4.3.2. Реалізація для DP/FMS на оптоволокні

Завдяки оптоволокну дистанцію між абонентами може бути збільшено до 15 км. Оптоволоконна мережа стійка до електромагнітних перешкод та до різниці потенціалів між пристроями. На малих дистанціях (до 80 м) можуть використовуватися дешеві пластикові оптичні волокна, на великих — скляні. Для підключення пристроїв можуть використовуватися різні оптичні модулі:

- OLM (Optical Link Module) має два електричні канали та один або два оптичні канали, фактично є оптичним повторювачем для електричного сегмету RS-485; схему організації мережі з OLM показано на рисунку 4.5
- OLP (Optical Link Plug) має тільки 2 оптичні канали, а електрично підключається прямо на рознімання DE-9 і отримує живлення від нього; модуль OLP використовується тільки для підключення ведених пристроїв (slave), а провідний пристрій (master) завжди підключається до оптичної мережі через OLM; схему організації мережі з OLP показано на рисунку 4.6
- LWL (Licht-Wellen Leiter) – безпосереднє підключення пристроїв Profibus до оптоволокна; використовується для пристроїв, обладнаних вбудованим оптичним модулем.

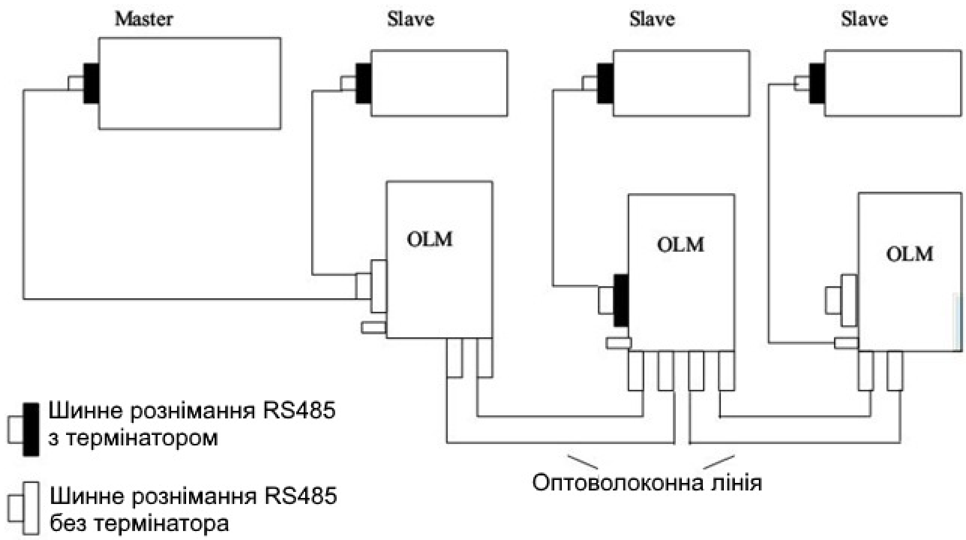


Рисунок 4.5 - Організація мережі з використанням OLM

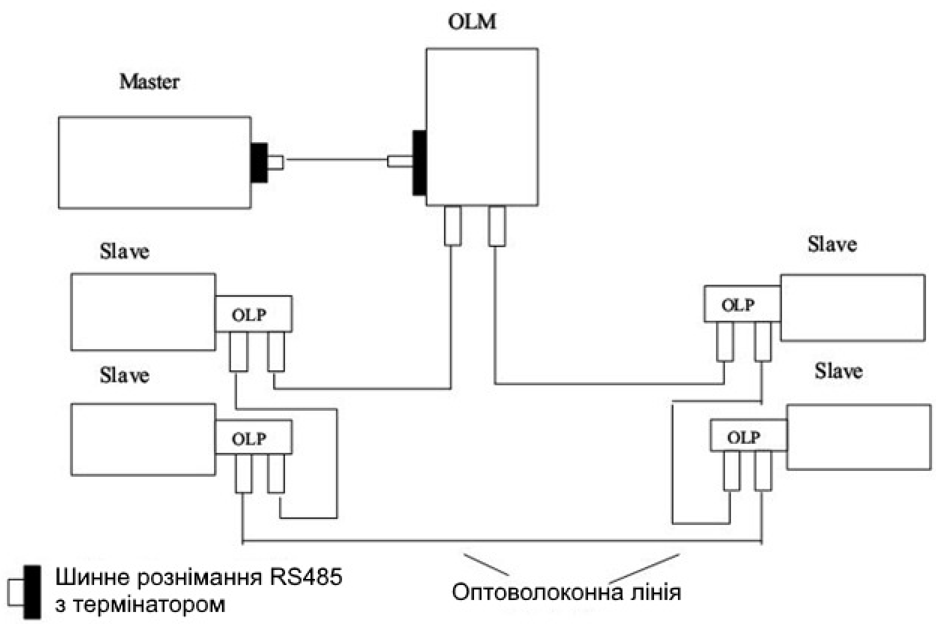


Рисунок 4.6 - Організація мережі з використанням OLP

4.3.3.Реалізація у вибухобезпечному виконанні (профіль PA)

В профілі Profibus-PA використовується передавальна техніка за стандартом IEC 1158-2. Вона забезпечує живлення польових приладів електробезпеку прямо через шину, а також задовольняє вимогам електробезпеки. Для передачі даних використовується біт-синхронізований, з манчестерським кодом протокол передачі без постійної складової (позначається також як H1). При передачі даних за допомогою манчестерського коду двійковий "0" передається, як зміна потенціалу з 0 на 1, а двійкова "1" – як зміна потенціалу з 1 на 0. Дані передаються за допомогою модуляції +/-9mA основного струму шинної системи I_b (рис. 4.7).

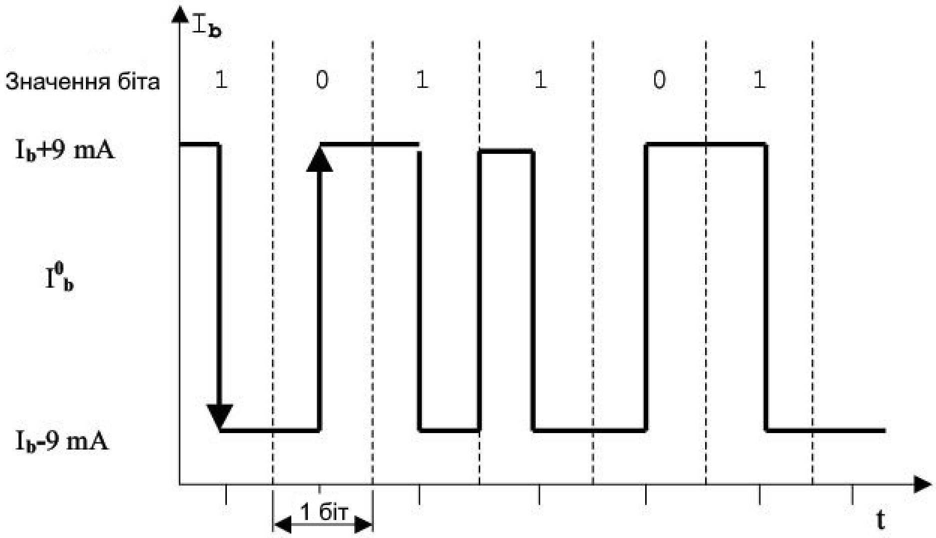


Рисунок 4.7 - Передача даних за допомогою модуляції струму

Швидкість передачі складає 31,25 кбіт/с. Середовищем передачі може бути екранована або неекранована кручена дротова пара, її характеристики не нормуються. Шина складається з сегментів, до яких підключені пристрої, сегменти замкнені на RC-ланцюжки. До одного сегменту шини PA може бути підключено до 32 приладів. Найбільша довжина сегменту сильно залежить від джерела живлення, типа дроту та споживання струму підключеними приладами. Організацію мережі для профілю Profibus-PA показано на рисунку 4.8.

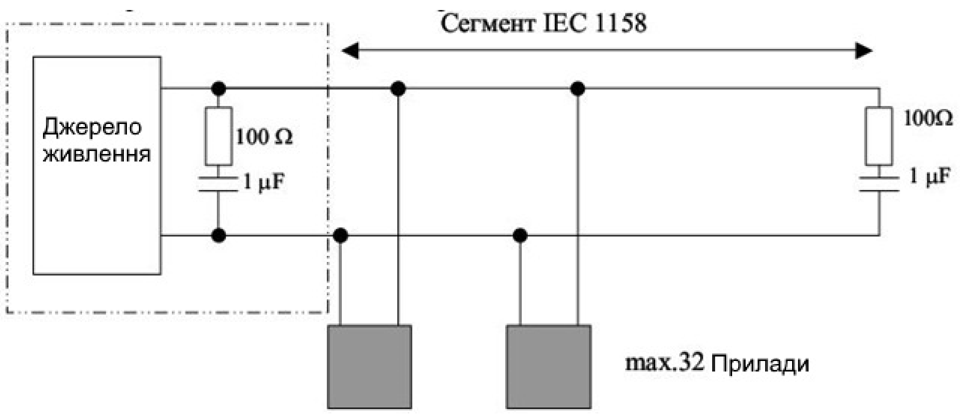


Рисунок 4.8 Організація мережі для профілю Profibus-PA

4.4. Логіка взаємодії на канальному рівні

Для версій Profibus існує єдиний протокол доступу до шини. Цей протокол реалізується на другому рівні моделі ISO (який називається FDL - Fieldbus Data Link). Він реалізує процедуру доступу за допомогою маркера (token). Мережа Profibus складається з провідних (master) і ведених (slave) станцій. Провідна станція може контролювати шину, тобто, може передавати повідомлення (без віддалених запитів), коли вона має право на це (тобто, коли вона утримує маркер). Ведена станція може лише розпізнавати отримані повідомлення чи передавати дані після відповідного запиту. Маркер обертається в логічному кільці, що складається лише з провідних пристроїв. Якщо система складається лише з одного провідного пристрою, то маркер не передається (буде у чистому вигляді система "master-slave"). Мережа з мінімальною конфігурацією може складатися або з двох провідних пристроїв, або з одного провідного та одного веденого пристрою.

Типову DP-конфігурацію має одномайстерна структура (Mono-Master-Struktur). Обмін між DP-Master'ом і DP-Slave'ом здійснюється за таким принципом. DP-Master зберігає в своїй пам'яті список викликів (Polling-Liste), з якого він вибирає адреси DP-Slave'ів; вони розпочинають передачу тільки на вимогу DPMaster'a. Обмін даними між DP-Master'ом і DP-Slave'ом відбувається циклічно, без урахування змісту даних користувача. Логіку цього обміну показано на рисунку 4.9.

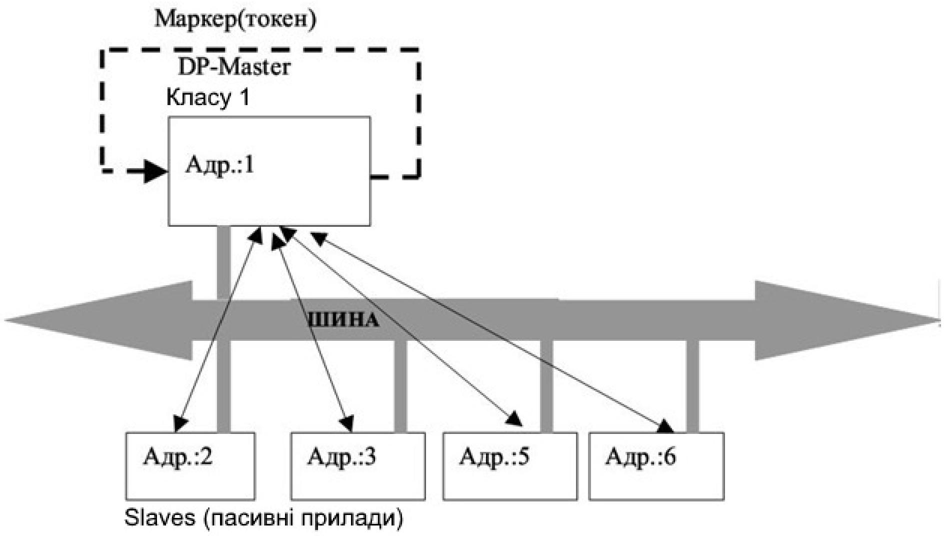


Рисунок 4.9 - Логіка обміну в системі з одним провідним пристроєм

Цикл обміна між DP-Master'ом і DP-Slave'ом складається з кадру запита (Request Frame) DP-Master'а та підтвердження (Response Frame), яке відправляє DP-Slave. Обробку списка викликів показано на рисунку 4.10.

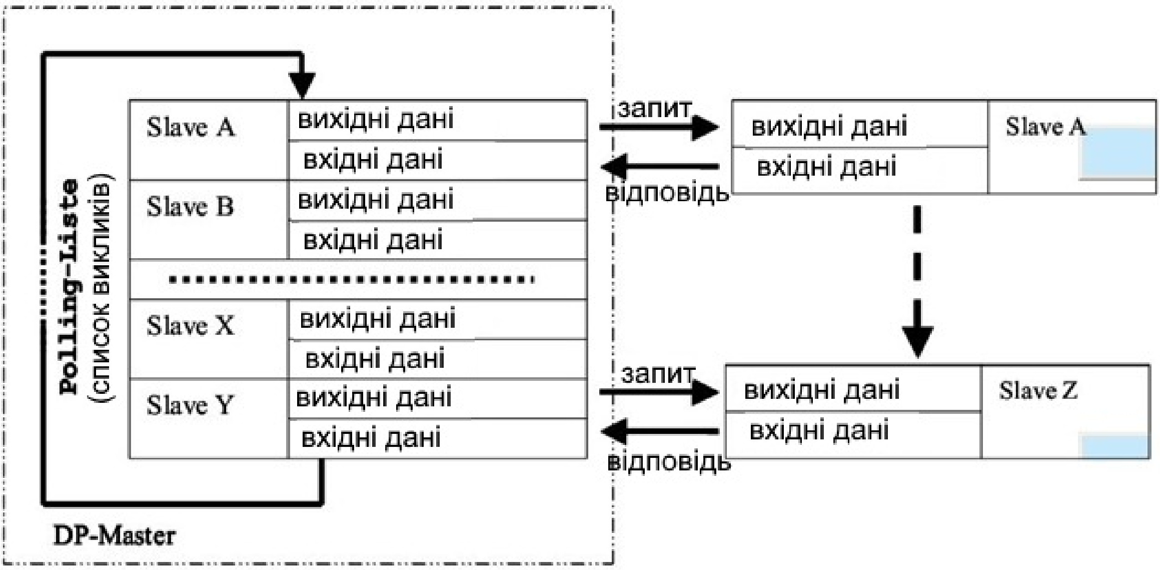


Рисунок 4.10 – Обробка списку викликів

В системі може бути кілька провідних пристроїв; в такому випадку вони передають один одному маркер (право на зайняття шини). Кожен провідний пристрій має право на володіння маркером; час володіння маркером встановлюється, як параметр конфігурації мережі. Ведений пристрій може включатися в цикл опитування кількох провідних, таким чином, наприклад, кілька систем можуть зчитувати значення параметрів з одних і тих самих датчиків. Але записувати дані в ведений пристрій має право тільки один провідний — той, якому його призначено. Логіку обміну в мережі з кількома провідними пристроями показано на рисунку 4.11.

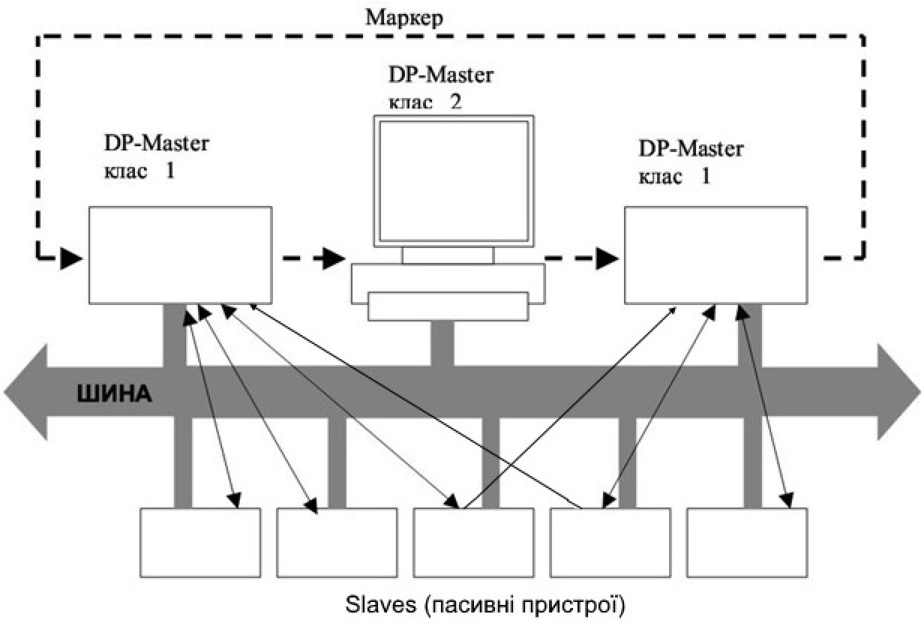


Рисунок 4.11– Обмін в мережі з кількома провідними пристроями

4.5. Базові та розширені функції обміну даними в Profibus

Провідні пристрої можуть бути 1 класу і 2. Різниця між ними полягає в тому, що пристрої 1 класу ведуть циклічний обмін даними, а пристрої 2 класу призначено для програмування, діагностики та управління шиною. В зв'язку з цим визначено кілька стандартних функцій, які можуть виконувати ці пристрої. Пристрої 1 класу виконують такі функції:

1. Set_Prm: встановлення параметрів — це, головним чином, визначення кількості вхідних та вихідних байтів для інформаційного обміну
2. Chk_Cfg: перевірка встановлених параметрів конфігурації
3. Data_Exchange: циклічний обмін вхідними та вихідними даними між DP-Slave'ом та DP-Master'ом, якому він призначений
4. Slave_Diag: зчитування діагностичної інформації під час запуску або циклічного обміну

5. Global_Control: встановлення робочого стану водночас всім пристроям мережі.

Пристрої 2 класу, крім тих, що зазначені вище, можуть виконувати ще такі додаткові функції:

i RD_Inp: зчитування вхідних даних з DP-Slave

i RD_Outp: зчитування вихідних даних з DP-Slave

i Get_Cfg: зчитування актуальних конфігураційних даних з DP-Slave

i Set_Slave_Add: встановлення нової шинної адреси DP-Slave'у та DP-Master'у, якщо вони підтримують таку функцію

Також можуть існувати комбіновані пристрої, які виконують функції DP-Master класу 1 і DP-Slave, або DP-Master класу 1 і класу 2.

Всі функції Profibus реалізуються за допомогою службових повідомлень, які передаються через мережу і називаються телеграмами. Процес відправлення та опрацювання прийнятих телеграм підрозділяється на 4 види служб, які відрізняються логікою залежно від того, яка є потреба в підтвердженні прийнятих даних і повторній відправці; вони перераховані в таблиці 3. Служби профілів DP та PA є підмножиною служб профілю FMS.

Таблиця 4.3 - Види сервісів у протоколі Profibus

Служба	Функції, які виконує служба	DP	DA	FMS
SDA	Посилання даних з квітуванням			+
SRD	Посилання та приймання даних з квітуванням	+	+	+
SND	Посилання даних без квітування	+	+	+
CSRD	Циклічне посилання і приймання з квітуванням			+

Телеграми передаються з використанням коду Хеммінга з відстанню HD=4; завдяки цьому може бути розпізнано до 3 помилок в тексті. В якості контрольних елементів використовуються початковий та кінцевий знаки телеграми, біт парності та контрольний байт, а також ковзна синхронізація. При цьому можуть бути розпізнані такі помилки:

\endash Помилка символьного формату (парність, переповнення, помилка кадру)

\endash Помилка протоколу

\endash Помилка роздільника початку та кінця кадру
\endash Помилка байта перевірки кадру
\endash Помилка довжини телеграми

Телеграма, у якої розпізнано помилку, повторюється принаймні, один раз. Це задається шинним параметром "Retry" - він може задавати від 2 до 8 повторних передач. Рівень 2 може виконувати не тільки передачу даних "від точки до точки", але й в багато точок: Broadcast и Multicast. При передачі Broadcast активний пристрій посилає повідомлення всім іншим (Master'ам и Slave'ам). Приймання даних не квитується. При передачі Multicast активний пристрій посилає повідомлення групі пристроїв (Master'ам і Slave'ам), приймання даних так само не квитується.

4.6. Формати телеграм в Profibus

Для обміну використовуються телеграми одного з стандартних форматів, які показано на рисунку 4.12. Позначення інформаційних полів у форматах телеграм наведені у таблиці 4.

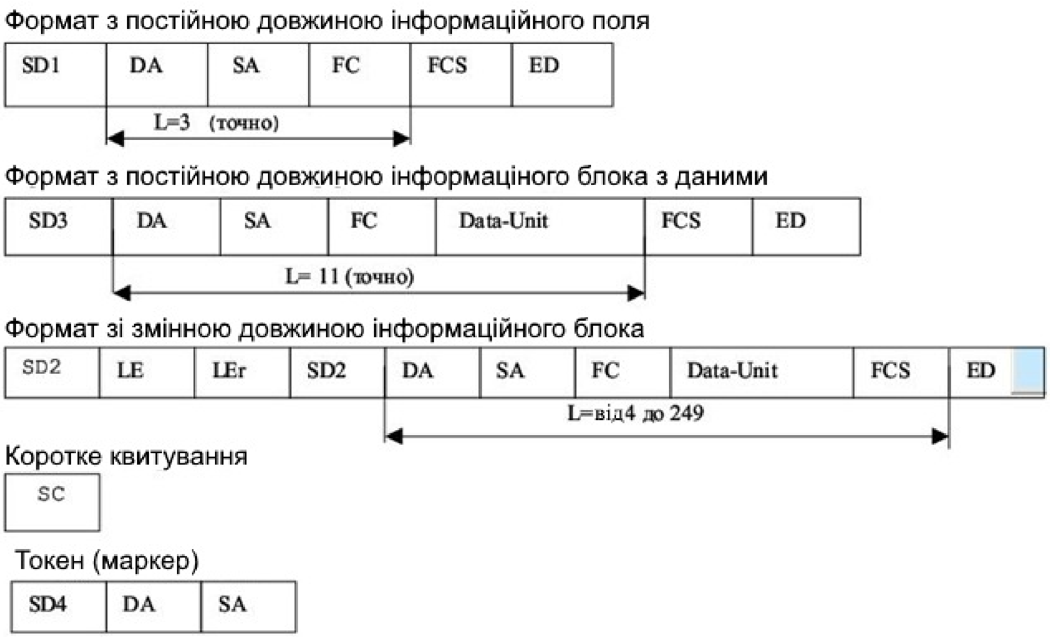


Рисунок 4.12 - Стандартні формати телеграм

Таблиця 4.4 - Позначення інформаційних полів у форматах телеграм

L		Довжина інформаційного поля
SC	Single Character	Окремий символ, використовується тільки для квитування SC=E5h
SD1-SD4	Start Delimiter	Стартовий байт для відміни різних форматів телеграм SD1=10h, SD2=68h, SD3=A2h, SD4=DCh
LE/LEr	LEngth	Байт, який показує довжину інформаційного поля телеграм зі змінною довжиною
DA	Destination Adress	Байт — адреса приймача телеграми
SA	Source Adress	Байт — адреса відправника телеграми
FC	Frame Control	Контрольний байт, містить інформацію про службу та пріоритет повідомлення
Data Unit		Поле даних, також може містити можливі розширення адреси даних користувача
FCS	Frame Check Sequence	Контрольна сума телеграми, яка розраховується операцією "AND" без переповнення
ED	End Delimiter	Кінцевий байт, позначає кінець телеграми (ED=16h)

4.7. Служби передачі PROFIBUS

Таблица 4.5

Служба	Функції	DP	PA	FMS
SDA	Посилка даних з квитуванням			x
SRD	Дані надсилаються и приймаются з квитуванням	x	x	x
SDN	Дані надсилаються без квитування	x	x	x
CSRD	Цикліче здійснення та прийом з квитуванням			x

Служби викликаються через точки доступу до служби (SAP, Service Access Point) канального рівня вищого рівня. У PROFIBUS-FMS ці точки доступу використовуються для адресації логічних комунікаційних зв'язків. У протоколах DP і PA застосовуються точки доступу строго впорядковані. У всіх активних і пасивних учасників можна використовувати паралельно кілька точок доступу. Розрізняються точки доступу джерела (SSAP, Source Service Access Point) і точки доступу приймача (DSAP, Destination Service Access Point).

Прикладний рівень FMS

Прикладний рівень визначено тільки для PROFIBUS-FMS. З допомогою введення спеціальних профілів для FMS-пристроїв, в якості надбудови до прикладного рівня, забезпечується можливість узгодженого взаємодії з пристроями різних виробників. В профілі пристроїв, що підключаються до PROFIBUS-FMS, визначений обов'язковий набір функцій, необхідний для конкретного типу пристроїв, завдяки чому пристрої різних виробників мають одні і ті ж комунікаційні функції. Для FMS визначені наступні профілі:

- комунікації між контролерами (3.002). Цей комунікаційний профіль встановлює, які FMS-служби застосовуються для комунікацій між PLC. В профілі визначені служби, параметри і типи даних, які повинен підтримувати кожен ПЛК;
- профіль для автоматизації будівель (3.011). Це галузевий профіль і основа для багатьох відкритих стандартів в автоматизації будівель. Він описує, як здійснюється обмін, управління, регулювання, обслуговування, обробка та архівування сигналів (Alarm) в системах автоматизації будівель через FMS;
- комутаційні низьковольтні прилади (3.032). Це галузевої користувальницький FMS-профіль. Він визначає правила обміну даними з низьковольтними комутаційними приладами через FMS.

Користувальницький інтерфейс і DP DP-профілі

Для PROFIBUS-DP визначено користувальницький інтерфейс, який позиціонується як надбудова над моделлю ВОС. Протокол PROFIBUS-DP визначає, як дані передаються по шині між учасниками. При цьому сенс передаються по протоколу користувальницьких даних протоколом не визначається (оскільки немає прикладного рівня!). У PROFIBUS-DP також як і в PROFIBUS - FMS визначено ряд профілів, завдяки чому прилади різних виробників можуть обмінюватися інформацією. В даний час встановлено наступні профілі PROFIBUS-DP:

профіль для керування роботами через PROFIBUS-DP (3.052). Профіль дозволяє описати рух і програмне керування роботом;

– профіль для підключення енкодера (перетворювача кут-код) (3.062).

Для енкодерів визначені два апаратних класу основних і додаткових функцій, як, наприклад, масштабування сигналів і розширена діагностика;

– профіль для приводів з регульованою швидкістю – PROFIDRIVE-профіль (3.072). Профіль встановлює, як приводи параметрируються і передають задані і дійсні значення, завдяки цьому стає можливим обмін даними з приводами різних виробників;

– профіль для управління і спостереження, HMI (Human Machine Interface) (3.082). Цей профіль використовується при підключенні до мережі пристроїв для реалізації людино-машинного інтерфейсу (операторські панелі, персональні комп'ютери). Профіль встановлює правила підключення цих пристроїв через PROFIBUS-DP до компонентів автоматизації .

Управління доступом до шини в PROFIBUS

До управління доступом до шини PROFIBUS пред'являються два істотних вимоги. З одного боку для надійних комунікацій між рівноправними ПЛК та/або ПК необхідно, щоб кожен учасник протягом певного часового вікна отримував доступ до шини для вирішення своїх комунікаційних завдань.

З іншого боку для обміну даними між ПЛК або ПК і простий децентралізованої периферією потрібно швидкий обмін даними з можливо малими витратами протоколу. Це досягається завдяки використанню гібридного управління доступом до шини, що складається з децентралізованого обміну маркером між активними учасниками (Master-ами) та централізованого обміну Master-Slave для обміну даними між активними і пасивними учасниками шини PROFIBUS.

Активний учасник, який володіє маркером, бере на себе функції майстра на шині, щоб проводити комунікації з пасивними і активними учасниками.

Обмін повідомленнями по шині відбувається при цьому через адресацію учасників. Кожному PROFIBUS-учаснику призначається однозначний адресу. Адреса призначається з області від 0 до 126. При цьому максимальне число учасників, що знаходяться на шині, не перевищує 127.

Використання гібридного управління доступом до середовища передачі дозволяє реалізувати наступні конфігурації системи:
– «чиста» система Multimaster (обмін маркером);
– «чиста» система Master-Slave;
– комбінація обох методів.

Метод доступу до PROFIBUS не залежить від середовища передачі.

Активні учасники, підключені до PROFIBUS, утворюють впорядкований за зростанням їх адрес логічне маркерне кільце (Token Ring) (рис. 13) по якому циркулює маркер (token). Маркер, а з ним і право на доступ до середовища передачі, передається в спеціальному маркер-сполучення між активним, тема учасниками. Активний учасник з найвищим на шині адресою HSA (Highest Station Address) передає маркер активному учаснику з найменшим шинним адресою, щоб замкнути маркерне кільце. Час проходження маркера через всіх активних учасників називається часом звернення маркера і обмежується за допомогою параметра TTR (Time Target Rotation) , який задається при конфігурації мережі.

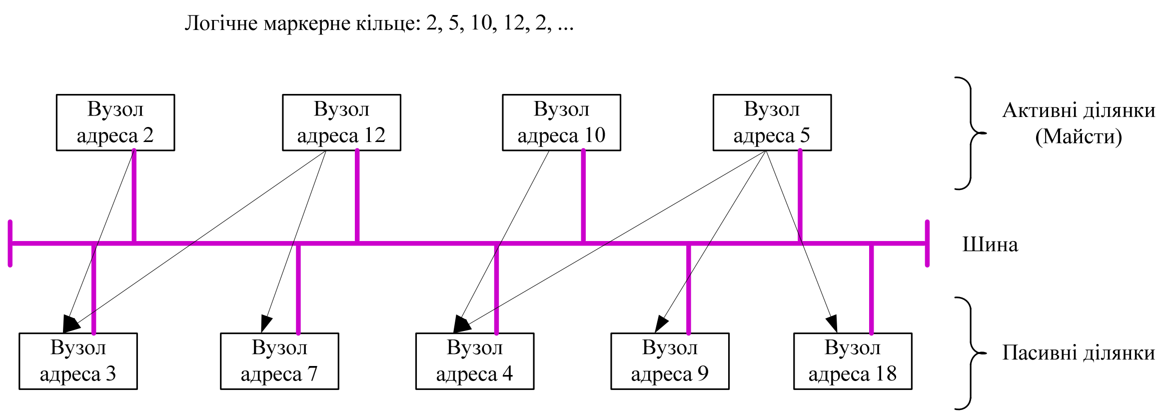


Рисунок 4.13 – Метод обміну маркером

Управління доступом до шини активних учасників здійснюється як на етапі ініціалізації під час функціонування маркерного кільця. При цьому встановлюються адреси всіх наявних на шині активних учасників і заносяться в список активних станцій (LAS, List of Active Station). Для управління маркером необхідно знати адреси попередньої станції, від якої маркер надсилається, і наступної станції, якій маркер призначається. Крім того, список LAS потрібен для того, щоб при поточній роботі виключати з кільця вийшли з ладу або дефектних активних учасників і, відповідно, приймати знову з'явилися учасників без перешкод поточного обміну даними по шині..

Якщо логічне маркерне кільце складається лише з одного активного і декількох пасивних учасників, то це відповідає «чистої» системи Master - Slave. Метод Master-Slave дозволяє майстру (активному учаснику), опитувати призначених йому Slave (пасивних учасників).

Контрольні запитання та завдання

1. Які вимоги пред'являють до промислової мережі PROFIBUS?
2. На яких рівнях типової структури АСУ ТП застосовуються промислові мережі PROFIBUS?
3. Які характеристики промислової мережі PROFIBUS дозволяють судити про відповідність мережі вимогам задач реального часу?
4. До якого рівня слід віднести протоколи PROFIBUS?
5. Скільки провідних і ведених пристроїв може бути в мережі на базі PROFIBUS?
6. Який максимальний об'єм даних, що передаються в повідомленні протоколу PROFIBUS?
7. Яку структуру пам'яті має будь-який пристрій, підключений до мережі з точки зору PROFIBUS протоколу ?
8. Як реалізується маршрутизація повідомлень в протоколі PROFIBUS?
9. В якій послідовності здійснюється обхід логічного кільця маркером в мережі PROFIBUS?
10. Що може використовуватися в якості середовища передачі для протоколу PROFIBUS-DP/FMS?

РОЗДІЛ 5
Основи PROFINET

Якщо говорити про промислову автоматизацію, перш за все можна подумати про мережеві системаи, такі як PROFIBUS, INTERBUS, DeviceNet or CANopen і т.д. Широке використання технології польових шин вже вкоренилося в області промислової автоматизації в останні кілька років. Однією з важливих переваг систем польових шин є те, що є можливість переміститися з централізованих систем автоматизації до децентралізованих.

Більшість Інтернет комунікацій засновані на стандарті Ethernet, який було надано IEEE 802.3 (Інститутом інженерів з електротехніки та електроніки). Однією з істотних властивостей Ethernet є проста структура. Вона може бути легко реалізована, і тому дуже популярна.

PROFINET є відкритим і інноваційним стандартом для промислової автоматизації на базі промислового Ethernet, тобто існує можливість обмінюватися даними процесу з машинами, як і раніше, але замість того, щоб використовувати мережеві системи, можна використовувати Ethernet як засіб комунікації

Концепція PROFINET задовольняє всім вимогам, що пред'являються до технології промислової автоматизації. Можна використовувати PROFINET для заводської і технологічної автоматики, які вимагають часу відгуку менше 100 мілісекунд. Інші програми, такі як технологія приводу додатків або годинник для синхронізації управління рухом, може бути поданий з часом відгуку до менш 1 мілісекунди. Для запуску додатків безпеки, можна використовувати PROFIsafe, який є частиною PROFINET.

PROFINET технологія розроблена і опублікована PROFIBUS / PROFINET International. Вона доповнює затверджену технологію PROFIBUS, особливо з точки зору швидкості передачі даних і використання інформаційних технологій (IT). PROFINET використовує IT-стандарти, такі як TCP / IP і XML для зв'язку, конфігурації і діагностики машин або польових пристроїв.

5.1 Функціональні класи PROFINET

У PROFINET виділяють два класи функцій, які не залежать одне від одного: PROFINET IO і PROFINET CBA (Component Based Automation).

PROFINET IO - розподілене введення / виведення (Input / Output).

PROFINET CBA - розподілена автоматизація. На рисунку 5.1 показані функціональні класи PROFINET.

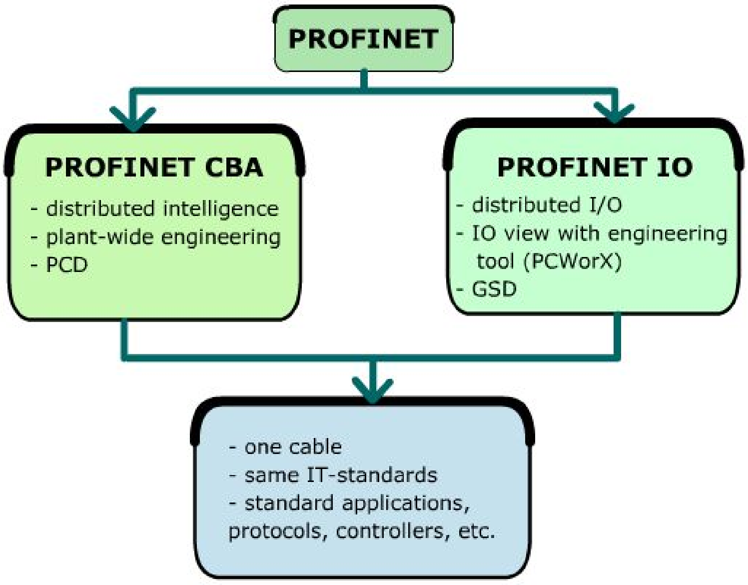


Рисунок 5.1 – Функціональні класи PROFINET

PROFINET IO: Розподілені польові пристрої підключаються через PROFINET IO. З PROFIBUS є значна схожість із PROFINET IO системами; наприклад, оброблені дані з польових пристроїв передаються в систему управління періодично.

1. PROFINET IO використовує три різних канали зв'язку для обміну даними з системами управління та іншими пристроями. Стандартний канал TCP/IP використовується для параметризації, конфігурації і ациклічних операцій читання/запису. NRT (Non-Real Time) використовується для не критичних до часу процесів. Канал RT (Real Time) використовується для стандартної циклічної передачі даних і сигналів тривоги. Третій канал, IPT (Isochronous in Real Time) є високошвидкісним каналом, який використовується для додатків

управління рухом. Технічні характеристики польових пристроїв описуються так званим GSD-файлом (General Station Description), який заснований на XML (eXtensible Markup Language);

2. PROFINET CBA. Ця концепція призначена для розподілених додатків промислової автоматизації. PROFINET CBA побудований на стандартній моделі DCOM (Distributed Component Object Model) і RPC (Remote Procedure Call) технологій. DCOM є об'єктно-орієнтованим механізмом, де структури, такі як клієнт, можуть знайти, запитувати і отримувати дані з сервера. DCOM спочатку була розроблена корпорацією Майкрософт. Об'єкти DCOM, які відключені від PROFINET CBA, називаються технологічні компоненти. Ці компоненти діють незалежно один від одного і координують свою діяльність між собою автономно. Інкапсульовані технологічні компоненти називаються компоненти PROFINET, які описуються за допомогою PCD (PROFINET Component Description). Можна уявити їх як чорний ящик з інтерфейсами на зовнішній стороні. Вони спілкуються один з одним через ці інтерфейси. Це також дозволяє з'єднати їх так як потрібно.

Пристрої PROFINET IO можна комбінувати і розглядати як компонент PROFINET, використовуючи відповідний інженерний інструмент. З концепцією PROFINET CBA можна поєднувати компоненти PROFINET в режимі обміну даними між машинами. Таким чином, ми будемо мати розподілене автоматичне з'єднання через PROFINET (рис. 5.2)

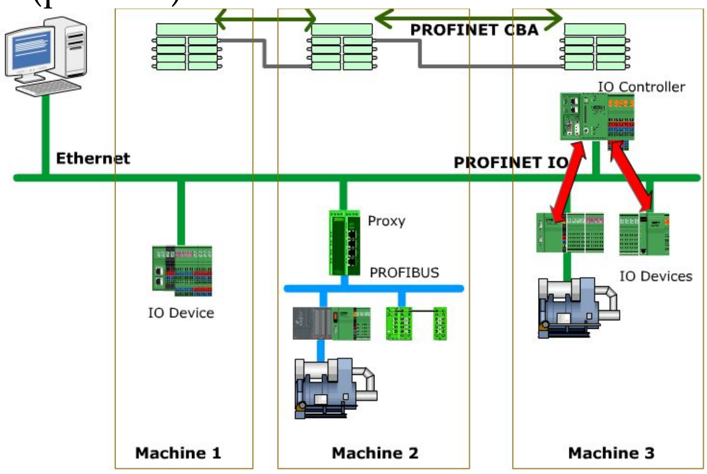


Рисунок 5.2 – Розподілені системи автоматизації PROFINET IO и CBA

5.2 Комунікації та безпека

Всі види транзакцій даних через Інтернет можуть бути перехоплені і записані. PROFINET використовує Ethernet в якості комунікаційної основи. Щоб можна було використовувати Ethernet, слід реалізувати купу протоколів, визначених у стандарті IEEE 803.2. Для транспортування даних використовуються протоколи TCP / IP або UDP / IP. Однак їх недостатньо для обробки даних. Таким чином, слід реалізувати більше протоколів, так званих прикладних протоколів, таких як HTTP, SNMP, SMTP, FTP і т. п. Таким чином PROFINET використовує три канали для зв'язку з різними класами продуктивності у відповідності до припустимого використання (рис. 5.3):

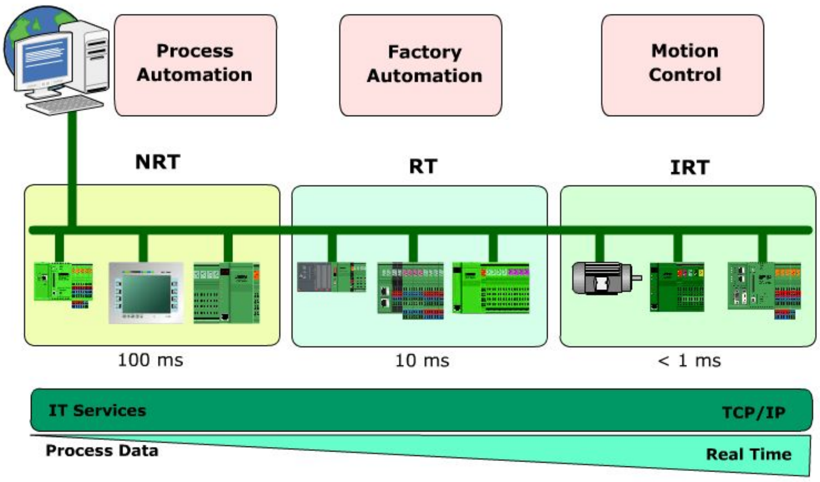


Рисунок 5.3 – Приклад комунікації PROFINET

– NRT (Non Real Time) для некритичних за часом процесів: PROFINET використовує стандартні TCP / IP і UDP / IP для передачі пакетів даних;

– RT (Real Time) для оптимізації обміну даними: операція читання-запису для промислової автоматизації вимагає, в основному, високошвидкісної передачі даних, в той час як стандартні

TCP / IP або UDP / IP не можуть задовольнити цю вимогу

– IRT (Isochronous Real Time) для синхронізованого по годинах зв'язку: запити додатків-приводів, таких як Motion Control, повинні бути негайно задоволені. IRT може зробити це з часом відгуку менше 1 мілісекунди.

Ядро концепції безпеки PROFINET зосереджено на захисті мереж в розділених зонах безпеки, так званих осередках (рис.4). Різні осередки можуть бути пов'язані між собою магістральною мережею. Кожна клітинка може бути підключена до магістралі через точку входу і захищена за допомогою спеціальних компонентів мережі безпеки, таких як комутатори або інші пристрої безпеки. Ці компоненти перевіряють дані, що передаються, для їх авторизації та контролю цілісності. Якщо ви хочете отримати доступ до захищених пристроїв автоматизації, ви можете використовувати спеціальне клієнтське програмне забезпечення безпеки

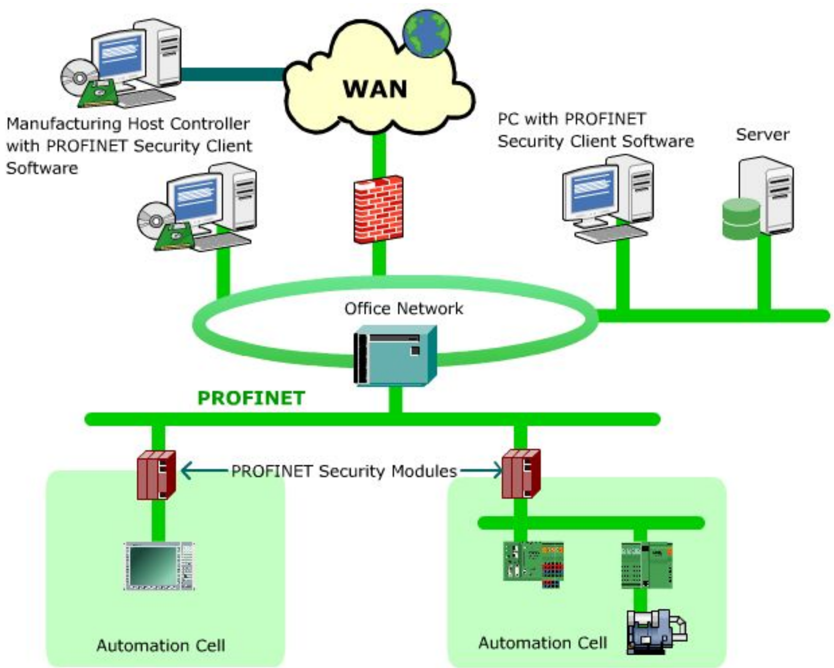


Рисунок 5.4 – Приклад безпеки PROFINET

Трафік даних між безпечними комірками або між програмним забезпеченням клієнта і сотою також може бути зашифрований з використанням звичайних алгоритмів шифрування. Це з'єднання може захистити дані від маніпуляцій або стеження. Це особливо необхідно, якщо ви хочете отримати доступ до захищених пристроїв автоматизації через небезпечну мережу, таку як Інтернет.

5.3 Порівняння PROFINET та PROFIBUS

Для тих, хто вже знайомий з PROFIBUS, може побачити PROFINET як подальший розвиток PROFIBUS DP. Однак, у порівнянні з PROFIBUS, PROFINET має ще багато переваг (Таблиця 1). У цьому розділі розглянуті деякі переваги PROFINET.

Переваги PROFINET

PROFINET пропонує кілька переваг з різних точок зору. Як користувач, ви можете володіти більшою гнучкістю, наприклад, управляти своїми пристроями автоматизації. Будучи виробником пристроїв PROFIBUS, все одно можна виготовляти старі добрі польові пристрої PROFIBUS при плануванні виробництва пристроїв PROFINET:

– висока швидкість: канал зв'язку реального часу забезпечує високошвидкісний обмін даними процесу, минаючи час, необхідний для обробки стека TCP / IP. Після цього дані про критичні за часом процеси можна обмінювати з використанням протоколів RT або IRT. Критичні по часу сигнали тривоги або некритичні діагностичні дані можуть бути встановлені однією установкою на одному носії;

– проста структура мережі. Конструкція стандарту Ethernet дуже проста в реалізації. Крім того, цей стандарт широко використовується в корпоративному секторі. Таким чином, можна забезпечити сумісність з'єднань з адміністративними відділами та офісами;

– економічні технології. Якщо порівняти вартість установки і продуктивність Ethernet з іншими мережевими системами, такими як ATM (асинхронний режим передачі) або Frame Relay, побачимо, що Ethernet набагато економічний, ніж інші. Більш того, Ethernet дасть кожному користувачеві однакову швидкість незалежно від того, де додаток реально розміщено;

– мультиплексування: використання Ethernet означає, що ви можете використовувати тільки один кабель для передачі всіх видів даних. Це, звичайно ж, призводить до зниження вартості кабелю. За допомогою перемикачів ви можете отримати доступ до потрібних пристроїв;

– програмне забезпечення IT. Для доступу до даних процесу або з діагностичними цілями можна використовувати звичайне програмне забезпечення IT-сфери, таке як веб-браузери, такі як Internet Explorer, Firefox, Opera і т. п., або офісні додатки, такі як Microsoft Office.

– просте масштабування мережі. Наприклад, в PROFINET IO ви можете додати нові польові пристрої, використовуючи бібліотеку пристроїв, яка дозволяє легко розширювати мережу. Такий підхід дозволяє вам розширити свою мережу, як ви побажаєте;

– індивідуальні налаштування: польові пристрої можуть бути встановлені індивідуально, наприклад, ви можете налаштувати різні польові пристрої з різним часом циклу в тій же мережі. Це дозволяє вам мати циклічні дані в різний час в залежності від ваших потреб;

– Повна інтеграція з системами польової шини. Для тих, у кого вже є система польової шини, наприклад, PROFIBUS, DeviceNet, INTERBUS і т. п., встановлена на заводі, PROFINET пропонує так званий проксі-сервер, який був розроблений для з'єднання PROFINET і PROFIBUS. Таким чином, інвестиції в уже встановлені польові шини захищені.

Порівняння PROFIBUS з PROFINET

Таблиця 5.1 - Порівняння PROFIBUS з PROFINET

Особливості	**PROFIBUS DP**	**PROFINET IO**
	Master class 1	IO controller
	Master class 2	IO supervisor
	Slave	IO device

Технологія дротової передачі	PROFIBUS по мідному або оптоволоконному кабелю	Промисловий Ethernet по мідному або оптоволоконному кабелю
Технологія бездротової передачі	Можливе інфрачервоне випромінювання	Можливих промисловий WLAN (бездротова локальна мережа)
Обмін даними	По запиту	Після параметризації пристрою вводу-виводу він може працювати сам по себе. Дані можуть обмінюватись циклічно або ациклічно (по запиту)
Кількість пристроїв	Максимум 126 пристроїв	Довільне, залежить від мережі
Канали даних	Один точно визначений канал даних між ведучим и підлеглим	Кілька каналів даних між контролером/супервізором і пристроєм
IT-услуги	Неможливо	Можуть бути інтегровані
Опис пристрою	Ключове слово	XML на основі визначення схеми
Пріоритет даних	Той же пріоритет	Можна встановлювати різні пріоритети
Доступ до польового пристрою	Читання тільки від кількох користувачів	Можливе читання и запис від кількох користувачів
Призначення адрес	Кодування адреси PROFIBUS через DIP-перемикач або інструмент проектування	• Призначення IP-адрес контролеру вводу-виводу за допомогою інструменту «Інжиніринг» • Призначення IP-адрес пристроям вводу-виводу за допомогою контролеру вводу-виводу • Призначення імені пристрою вводу-виводу за допомогою інструменту «Інжиніринг» • Призначення IP-адрес за допомогою веб-сторінки, яка інтегрована в деякі комутатори
Швидкість передачі даних	Максимум 12 Мбіт/с	100 Мбіт /с з повним дуплексом

Топологія	Стандарт: зірка і дерево Можливе: шини і кільце	Стандарт: лінія Можливе: дерево и кільце

Інтеграція з PROFIBUS

Концепція PROFINET дозволяє використовувати так званий проксі-сервер в разі, якщо ви хочете інтегрувати існуючу систему fieldbus, таку як PROFIBUS, INTERBUS, ASI і т. п., в систему PROFINET. З цією архітектурою у вас буде гібридна система, що складається з підсистем fieldbus і Ethernet. Таким чином, можливий плавний безперервний технологічний перехід від системи fieldbus до PROFINET.

На рисунку 5 показана інтеграція PROFINET і системи польової шини PROFIBUS. Як ви можете бачити, проксі знаходиться між цими двома системами і з'єднує їх разом. Це дозволяє інтегрувати існуючі PROFIBUS-конфігурації в PROFINET

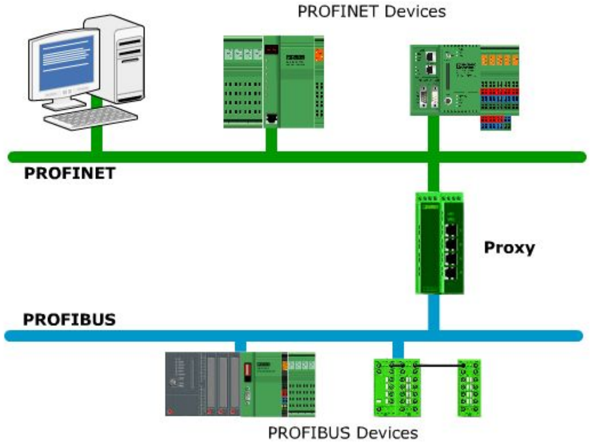


Рисунок 5.5 - Интеграція PROFINET и PROFIBUS систем

Основними причинами розробки нової технології є зниження вартості і поліпшення існуючих технологій і/або співвідношення «витрати/продуктивність». PROFINET виконує цей попит в усіх відношеннях. Він використовує стандартну технологію, яка проста в реалізації і економічно ефективна, а обробка промислових рішень автоматизації простіше, ніж раніше.

5.4. Основи комп'ютерних мереж

5.4.1 Еталонна модель ISO/OSI

Еталонна модель ISO/OSI поділяє мережні протоколи на рівні, та можна використовувати для розмежування функцій передачі даних. Ці два акроніма означають Міжнародну організацію стандартизації та взаємодія відкритих систем. Відповідно до еталонної моделі ISO/OSI, також званої моделлю рівня, ви можете розділити комунікаційну операцію на сім рівнів. Кожен рівень має певне завдання, яке може бути передано на рівень вище або нижче, відповідно до завдання, в якості служби. Зв'язок між двома рівнями може бути виконаний з використанням спеціального інтерфейсу, званого SAP (Service Access Point).

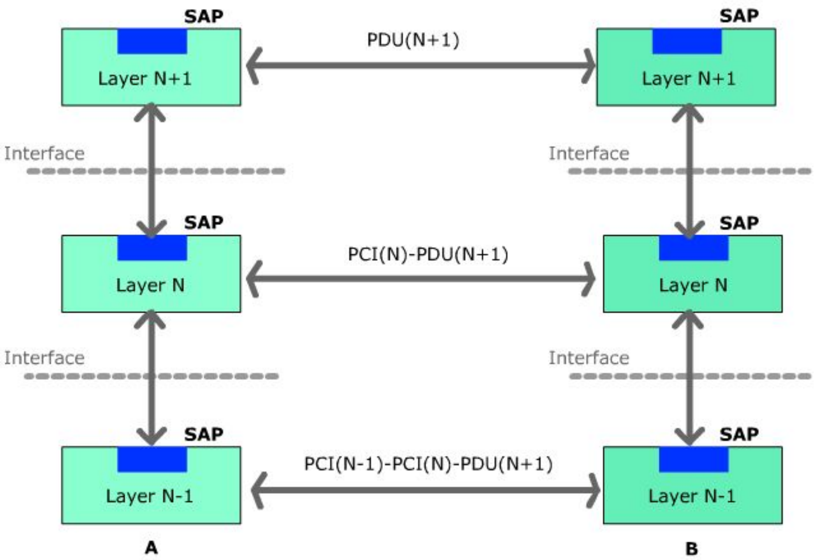


Рисунок 5.6 - Передача даних згідно моделі ISO / OSI

На рис. 5.6 показане фізичне середовище передачі даних усередині одного користувача, яке виконується у вертикальному напрямку. З точки зору шару N ви можете побачити потік даних на цьому рівні. Потік даних містить служби, які можна обмінювати в інтерфейсі SAP.

Логічна передача даних між двома користувачами проводиться в горизонтальному напрямку. Інформація буде обмінюватися з використанням стандартизованих протоколів. Цими стандартизованими протоколами є PDU (Protocol Data Unit) і PCI (інформація управління протоколом)

Таблиця 5.2 – рівні еталонної моделі ISO/OSI

Рівень моделі ISO/OSI	**Інтеграція з PROFINET**
Рівень 1: фізичний рівень На цьому рівні біти передаються по кабелю. Електричні і механічні властивості середовища передачі контролюються, щоб переконатися, що всі біти повністю надходять на приймач. Електричні властивості - це, наприклад, відповідність логічної «1». Механічними властивостями є, наприклад, типи кабелів, призначення роз'ємів, напрямок передачі даних	При передачі даних PROFINET використовує Fast Ethernet зі швидкістю 100 Мбіт/с відповідно до стандарту Ethernet від IEEE 802.3. Працює в повнодуплексному режимі, тобто користувач може одночасно відправляти і отримувати дані (двонаправлений). Типи кабелів: – 10BaseTX: 10 Мбіт/с в мідний кабель (вита пара). – 10BaseFX: 10 Мбіт/с у волоконно-оптичному кабелі. – 100BaseTX: 100 Мбіт/с в мідний кабель (вита пара). – 100BaseFX: 100 Мбіт/с в волоконно-оптичному кабелі
Рівень 2: Рівень каналу передачі даних Забезпечує безпомилкову передачу даних між двома користувачами на рівень 3. Вхідні дані розбиваються на фрейми. Ресивер може перевірити, чи була передача правильної чи ні. Якщо ні, відповідний кадр буде запитано знову. Крім того, цей рівень керує доступом до середовища. MAC (управління доступом до середовища) визначає унікальну ідентифікацію пристрою. MAC-адреса	Організація PROFIBUS Nutzer e. v. (PI) пропонує виробникам пристроїв частину ідентифікатора виробника MAC-адреси. Ця частина також називається OUI (організаційно унікальний ідентифікатор). Виробники пристроїв можуть придбати цю частину у PI, а потім додати свою власну послідовну

вказана виробником пристрою. MAC-адреса складається з двох частин: ідентифікатора виробника і послідовної нумерації.	нумерацію
Рівень 3: мережевий рівень Цей рівень забезпечує зв'язок між користувачами на великих відстанях і різних типах мереж. Мережевий рівень передає пакети даних з використанням їх IP-адреси (інтернет-протоколу). Підбирається підходяща маршрутизація. Пакети тимчасово зберігаються на Підвузли. Субноди будуть дивитися на поточну таблицю маршрутизації, для пошуку найкращого маршруту для пакетів	У разі розширення установки необхідно використовувати інженерний інструмент, щоб дати мережевих адрес пристрою PROFINET. У межах однієї мережі IP-адреса має бути унікальним. Ми можемо отримати доступ до відповідного пристрою PROFINET, використовуючи його IP-адресу. І пакети можуть бути передані на цей пристрій правильно
Рівень 4: транспортний рівень Завдання цього рівня полягає в забезпеченні з'єднання. Пакети даних можуть бути розбиті на дрібні шматочки (фрагментація) і знову зібрані з боку одержувача (дефрагментація). Якщо є помилки, вони будуть виправлені Можливі помилки: Втрата даних, помилкові дані, помилковий порядок фрагментації і т. п. Ці помилки будуть виправлені за допомогою, наприклад, квитанцій або повторних запитів. Найбільш відомим протоколом для управління потоком даних є TCP (Transmission Control Protocol). Іншим протоколом є UDP (User Datagram Protocol)	В PROFINET IO UDP реалізований для обміну даними процесів. Застосування PROFINET використовують порт 34962 - 34964 для зв'язку.
Рівень 5: сеансовий рівень На цьому рівні актуалізуються служби, такі як діалогове управління для спостереження за напрямком потоку даних або точками відновлення. Якщо	Не використовується

з'єднання перервано, його можна відновити в останній точці без повторення з самого початку	
Рівень 6: рівень представлення Дані перетворюються в загальний формат. На стороні відправника дані кодуються і стискаються. На стороні приймача дані декодуються і розпаковуються. Потім вони передаються відповідними програмами	Не використовується
Рівень 7: прикладний рівень На цьому рівні визначені певні інтерфейси. Ці інтерфейси забезпечують зв'язок між додатками. Зв'язок між різними протоколами актуалізується за допомогою так званих шлюзів або проксі	Можливий наступний сценарій: Вже встановлена система шин, наприклад, PROFIBUS. Ви хочете мати PROFINET, а потім інтегрувати пристрої PROFIBUS в PROFINET. Завдяки певному інтерфейсу на рівні 7 і використання проксі-сервера ми можемо легко реалізувати це. Тому обмін даними між пристроями PROFIBUS і PROFINET можливий без проблем

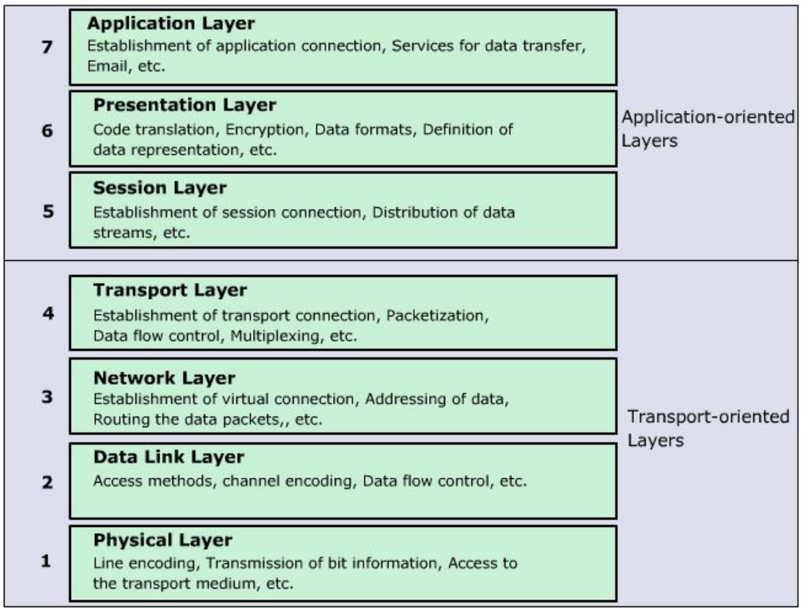


Рисунок 5.7 – Багаторівнева схема еталонної моделі ISO/OSI

Процес обміну даними між двома користувачами здійснюється в горизонтальному напрямку. Поодинокі біти будуть передаватися від користувача A користувачеві B через фізичне середовище в рівні 1 еталонної моделі ISO / OSI. Однак зв'язок в межах користувача A, в разі відправки, відбувається у вертикальному напрямку. На шарі 7 шар 7-заголовка додається до пакетів даних до того, як вони будуть передані на рівень 6. На рівні 6 додається заголовок рівня 6. Потім вони передаються на рівень 5 і так далі. На рівні 1 передаються пакети даних. На стороні приймача попередня процедура працює в зворотному порядку. В кінці шляху ви отримуєте передані пакети даних.

Зверніть увагу, що в процесі спілкування беруть участь тільки основні шари. Якщо для певного шару не використовується, заголовок цього шару не додається.

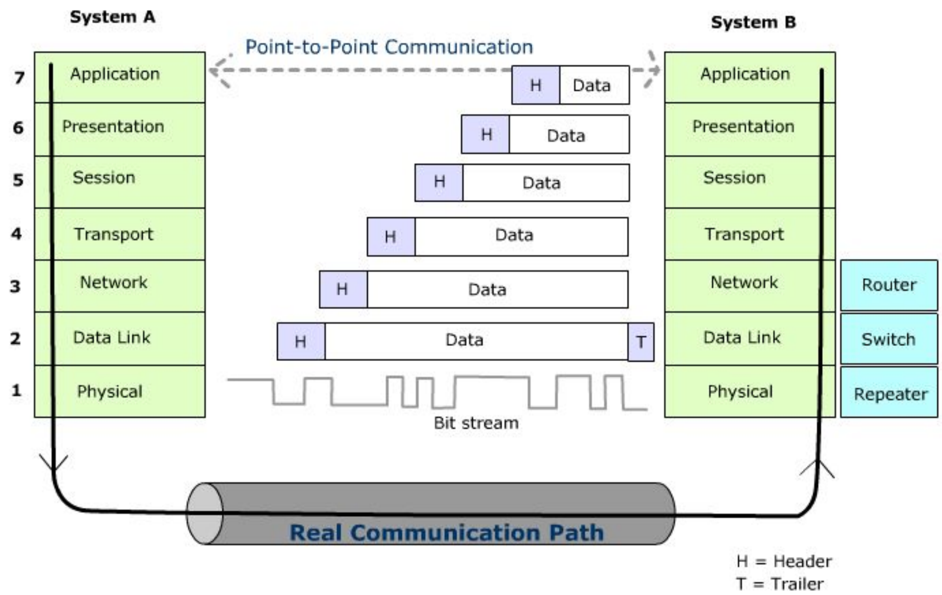


Рисунок 5.8 – Канал связи в эталонной модели ISO/OSI

5.4.2 Основи Ethernet

Ethernet - це різновид комп'ютерної мережі, яка є частиною найбільш часто використовуваних локальних мереж в світі. Найбільш відомими прикладами є LAN (локальна мережа) і WLAN (бездротова локальна мережа).

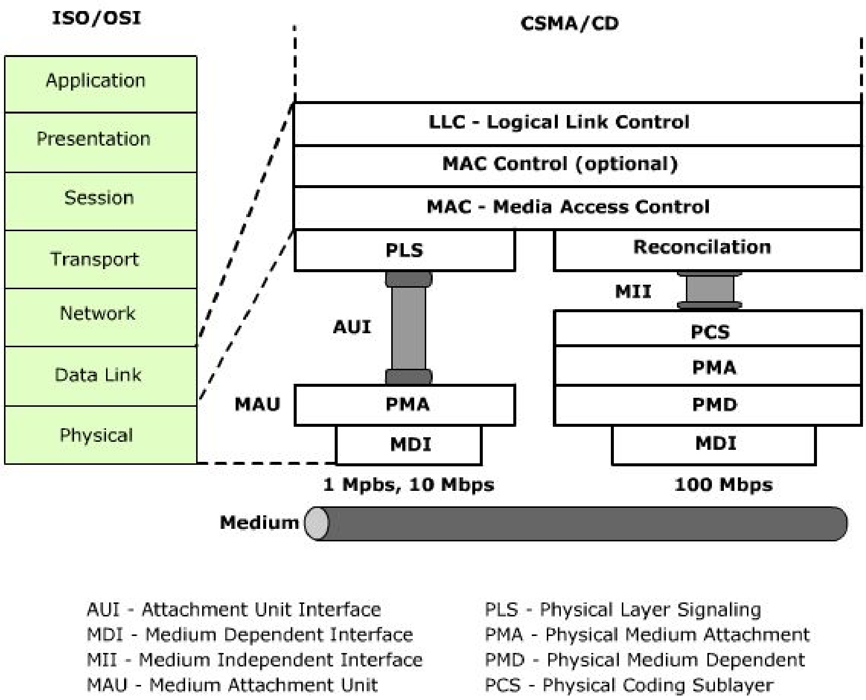


Рисунок 5.9 - Фізичний і канальний рівні в Ethernet

Стандарт Ethernet складається з наступних компонентів:

- MAC (управління доступом до середовища);
- PLS (сигналізація фізичного рівня);
- AUI (інтерфейс модуля прив'язки);
- MAU (Середнє додаток);
- PMA (фізичне додаток);
- MDI (інтерфейс, залежний від середовища).

Методи доступу

Ethernet використовує метод доступу до мережі CSMA/CD для управління доступом до мережі. Пристрої, підключені до мережевого кабелю, прослуховують середу передачі і починають передавати дані, як тільки носій доступний. Якщо носій використовується, пристрої чекають перед передачею. Моніторинг центральної станції або контроль доступу до неї відсутні.

CS (Carrier Sense): пристрої перевіряють, чи доступний носій.

MA (множинний доступ): MA вказує, що багато пристроїв можуть підключатися до однієї і тієї ж мережі або спільно використовувати її. Якщо носій доступний, кожен пристрій може почати передачу даних.

CD (виявлення зіткнень). Якщо два або більше пристрої визначають чистий канал і починають передачу даних одночасно, це викликає так зване зіткнення даних. Якщо відбувається зіткнення, пристрої переривають свою спробу передачі, очікують випадковий проміжок часу і потім повторно передають дані. Цей підхід називається усіченим бінарним алгоритмом експоненційної відстрочки і працює наступним чином (рис.10):

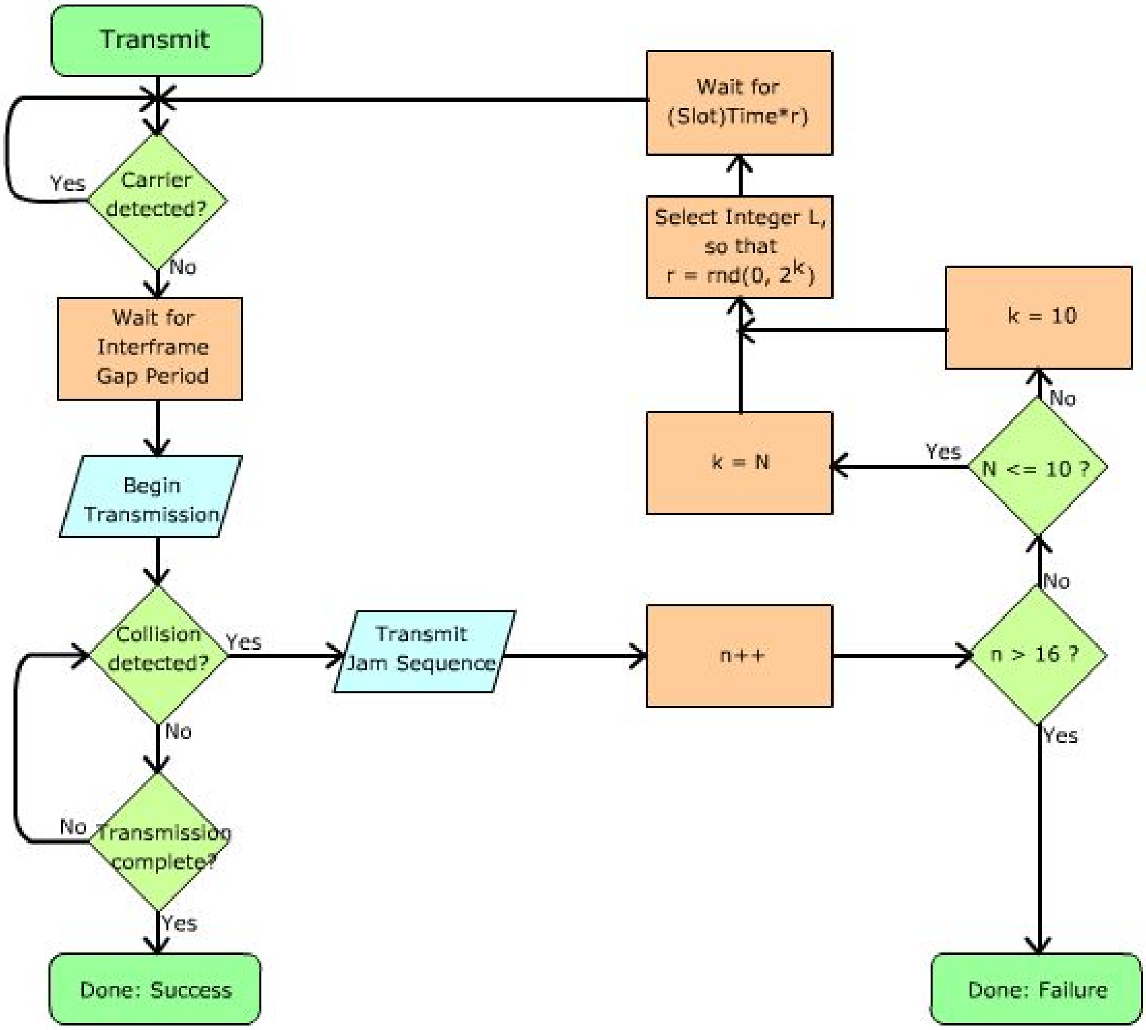


Рисунок 5.10 - CSMA/CD с усіченим бінарним експоненціальним алгоритмом повернення

1. Ініціація: n=0, k=0, r=0

2. Інкрементація n, лічильник передачі, який підраховує число послідовних зіткнень, які долаються вузлом

3. Якщо n> 16, також 16 невдалих послідовних спроб передачі, збій передачі і більш високі рівні повинні бути поінформовані

4. Якщо n <= 16, вибрати номер з набору k = min (n, 10), називається урізанням.

5. Випадкове число r, вибране з набору (0, 1, 2, 4 ... 2 k), також експоненціальна і двійкова.

6. Потім вузол чекає (r x slot_time) перед відновленням спроби передачі.

Стандартний фрейм Ethernet.

Ми визначаємо «кадри» як формат пакетів даних, які передаються від А до Б по дроту або по бездротовому зв'язку. Кадр зазвичай починається з так званої преамбули, яка показує фактичне фізичне обладнання як початкові біти. В кінці кадру є так звана Frame Check Sequence (FCS). FCS потрібно всім фізичним обладнанням для перевірки цілісності даних.

Стандартний кадр IEEE 802.3 має семібайтную преамбулу, за якою слідує одиночний роздільник статичного кадру (SFD). Потім слід четирнадцатібайтовий заголовок MAC, потім фактичні дані, а в кінці - чотирьохбайтового FCS (рис. 5.11).

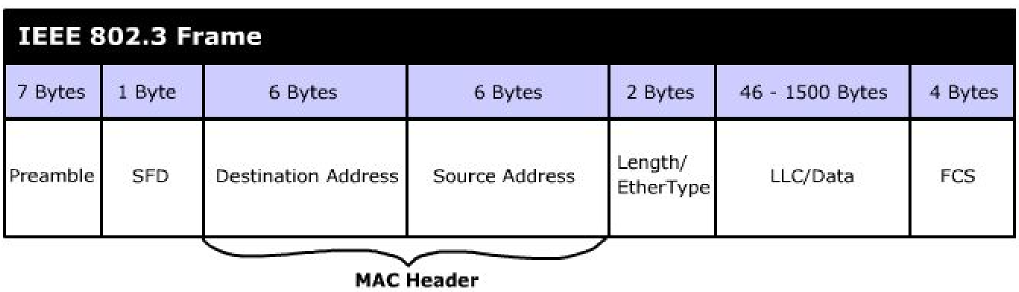


Рисунок 5.11 - Структура фрейма MAC адреси

Преамбула: преамбула складається з 7 байтів що складаються з 1 і 0, також 1010101010 і так далі.

– Початковий роздільник кадрів (SFD): SFD складається з 10101011. Останні два символи «1s» вказують на те, що починається дійсний кадр.

– Адреса призначення: Адреса призначення складається з MAC-адреси одержувача. Перший біт вказує, чи є адреса багатоадресних адресою (1) або одноадресна адресою (0).

– Адреса джерела. Вихідна адреса складається з MAC-адреси відправника.

– Length / EtherType: це поле складається з довжини або EtherType наступного поля даних, відповідно до IEEE 802.3.

– Дані: заповнені дані складаються з призначених для користувача даних, довжина яких може складати до 1500 байт, але не менше 46 байт.

– Frame Check Sequence (FCS): це поле складається з контрольної суми переданих даних, включаючи адресу призначення, адреса джерела, довжину і дані.

З рисунка 11 можна легко обчислити легальну довжину кадру, виключаючи преамбулу. Це від 64 до 1518 байтів.

MAC Адреса.

Якщо ви хочете відправити лист зазначеній особі, ви повинні знати його / її адресу, щоб він отримав ваш лист. У світі Ethernet кожний активний компонент Ethernet повинен мати унікальну адресу для ідентифікації.

MAC-адресу (управління доступом до середовища передачі) - це апаратний адресу пристроїв, що беруть участь в мережі. Існує багато імен для MAC-адрес, таких як Ethernet-адреса, фізичну адресу, адресу станції або адреса адаптера. MAC-адреса присвоюється виробником пристрою і має довжину 6 байт

MAC-адреса складається з двох частин, кожна довжиною 3 байти, заводський і послідовної нумерації. Частина MAC-адреси, яка визначається виробником, може бути отримана в IEEE за адміністративну плату і називається OUI (Організаційно-унікальний ідентифікатор). Послідовна частина може бути отримана від виробника. Кожне виробництво може зареєструвати до 16 777 214 пристроїв.

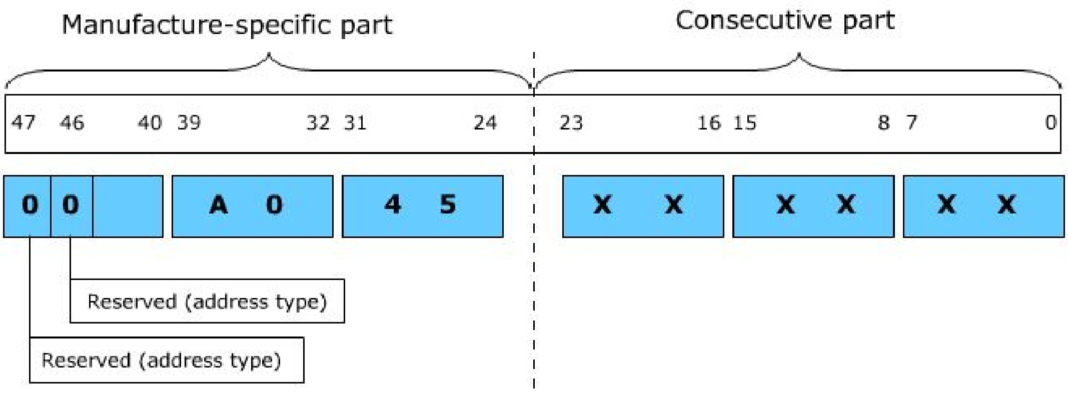


Рисунок 5.12 - MAC адреса

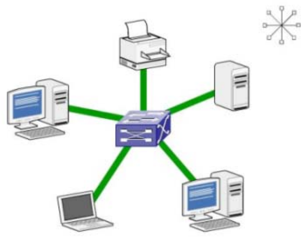
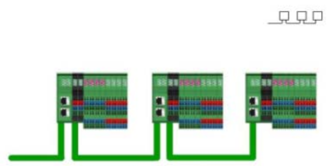
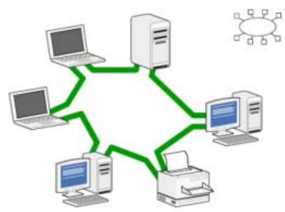
На рисунку 5.12 показана MAC-адреса 00.A0.45.XX.XX.XX, де 00.A0.45 - частина, що відноситься до виробника, і в цьому випадку представляє компанію «Phoenix Contact». XX.XX.XX - це послідовна нумерація, де XX - шістнадцяткові числа, від 0 до F.

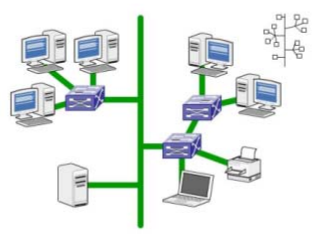
Виробники невеликих пристроїв, які не хочуть застосовувати OUI з IEEE самостійно, можуть придбати частину OUI з PI, яка є 00.0E.CF.XX.XX.XX.

Мережні технології.

Можна підключати Ethernet-пристрої до мережі відповідно до так званої топологією мережі. Існує безліч мережевих топологій, включаючи Star, Bus, Tree і Ring. Яку мережеву топологію слід використовувати, залежить від того, що ми хочемо зробити. Призначення мережі впливає на вимоги топології мережі. На практиці у вас є змішана система, що складається з декількох типів мережевих топологій. Огляд мережевих топологій наведено в таблиці. 3 нижче.

Таблиця 5.3 – Топологія мережі

Description	Structure
Зірка: характеристика топології зірки полягає в тому, що задіяний перемикач. Це означає, що у нас є центральний розподільник сигналів, і пристрої мережі підключаються до цього окремо. Якщо окремий пристрій виходить з ладу, це не призводить до збою всієї мережі. Проте, якщо комутатор виходить з ладу, з'єднання в цій мережі також не спрацьовує. Зіркоподібна топологія використовується в областях з високою щільністю пристроїв і малими відстанями, наприклад, в невеликих виробничих осередках або на однієї виробничі машині.	
Шина: всі мережеві пристрої з'єднані послідовно у вигляді шини. Для з'єднання між мережевими пристроями є або пристрій (комутатор) зв'язку, підключений до мережевого пристрою, або пристрій зв'язку інтегровано в мережевий пристрій. Якщо пристрій зв'язку виходить з ладу, з'єднання після цього пристрою зв'язку також не працює. Така топологія вимагає найменшої кількості проводів. Шинна топологія використовується переважно для підключення віддалених систем, таких як конвеєрні системи або для з'єднання виробничих осередків	
Кільце: якщо ви з'єднаєте два кінця топології шини, у вас буде кільцева топологія. Кільцеві топології є надлишковими, тобто такий тип топології підвищує безпеку відмови мережі. Якщо провід в мережі переривається або будь-який з компонентів мережі виходить з ладу, вся мережа продовжує функціонувати.	

Дерево: якщо ви поєднуєте кілька топологій зірка разом, у вас буде топологія дерева. У топології дерева можливе комбінування волоконно-оптичних і мідних (кручених пар) кабелів. Ця топологія використовується, коли складна система поділяється на кілька сегментів системи, і вони повинні бути взаємопов'язані.	

Деякі мережні компоненти.

Комутатори. У PROFINET промислові комутатори Ethernet використовуються для створення простого шлюзу між мережевими пристроями. Мережа, в якій кожен пристрій призначається виключно на один порт комутатора, називається «Комутований Ethernet». За допомогою комутаторів можна знайти компоненти пристроїв або мережевої інфраструктури. Це призводить до запобігання зіткнення доменів з виділеним з'єднанням «точка-точка». Таким чином, мережа стає більш стійкою під час тимчасового високого трафіку. Крім того, комутатори можуть реалізувати повнодуплексне з'єднання для кожного з'єднання. Комутатори працюють на рівні 2 (канальному рівні) еталонної моделі ISO/OSI.

Існує два режими роботи перемикачів:

– *Наскрізна комутація*: наскрізний перемикач або перемикач «на льоту» пересилає пакет даних відразу після отримання 6-байтового адреси призначення. Таким чином, час затримки між відправником і отримувачем зменшується, і пакети даних не будуть запам'ятовуватись повністю. Час затримки поточного наскрізного перемикача становить від 5 до 60 мікросекунд.

– *Комутація зберігання і пересилки*. Комутатор зберігання та пересилання перевіряє кожен пакет даних. Пакети даних буферизуються, перевіряються на правильність або достовірність, а потім або відхиляються, або пересилаються. Таким чином, збільшується час затримки для пересилки пакетів даних. З іншого боку, ніякі помилкові пакети даних не будуть передані. Час затримки пакета даних залежить від його довжини кадру. У короткій довжини кадру є час затримки між 8-100 мікросекунд, а у довгого - час затримки до 1.5 мілісекунд.

Крім того, комутатори можуть мати такі властивості:

– блокування або прискорення з'єднання (без блокування): якщо комутатор може обслуговувати всі служби одночасно, то він не блокується. Однак, якщо у нього обмежена пропускна здатність, це блокуючий комутатор.

– керований або некерований: якщо комутатор просто використовує таблицю адрес для передачі пакетів даних, він називається «некерованим». Однак, якщо ми можемо вставити додаткові критерії для передачі даних, такі як пріоритет пакета, пріоритет одиночної IP-адреси і т. д., він називається «керованим». Зверніть увагу, що з'єднання між мережами з різними швидкостями передачі може виконуватися тільки з комутаторами store-and-forward.

У PROFINET комутатор повинен пересилати пакет даних менш ніж за 10 мікросекунд після його отримання і підтримувати щонайменше 4 класи пріоритету. Оскільки комутатори є активними мережевими компонентами, і вони приведуть до збою всієї мережі в разі збою, для виявлення цього використовується протокол Rapid Spanning Tree Protocol (RSTP). Ідея цього протоколу полягає в тому, що при вказівці зміни топології структура мережі не буде негайно видалена. Будуть обчислені альтернативні маршрути і буде створено нове дерево. Цей метод дозволяє скоротити час простою мережі з 30 секунд до менше однієї секунди.

Концентратори. Концентратор Ethernet - це пристрій, який використовується для з'єднання двох або більше мереж або пристроїв разом. Також можуть бути підключені декілька кручених пар або волоконно-оптичні Ethernet-пристрої. Це дозволяє їм діяти як один сегмент. Концентратори працюють на рівні 1 (фізичному рівні) еталонної моделі ISO/OSI. Концентратори Ethernet також працюють в якості ретранслятора для мережі. Це означає, що сигнал буде посилений і регенерирован на велику відстань. Вони також відповідають за пересилку сигналу застрягання на всі порти, якщо вони виявляють зіткнення.

За допомогою концентраторів ви можете передавати тільки один пакет даних за раз зі швидкістю 10/100 Мбіт/с для двохшвидкісного концентратора. Крім того, концентратори не знають, яка станція підключена до якого порту, і не можуть дізнатися цю інформацію. Переваги концентраторів - це простота використання, оскільки їх не потрібно налаштовувати, і вони дешевші, ніж комутатори. Концентратори не використовуються в мережах PROFINET.

Маршрутизатор. Маршрутизатор - це комп'ютерно-мережевий пристрій, який використовується для з'єднання двох або більше окремих підмереж (рис. 5.13). Маршрутизатор складається з апаратного та програмного забезпечення, які зазвичай адаптуються до завдань маршрутизації і пересилання пакетів даних.

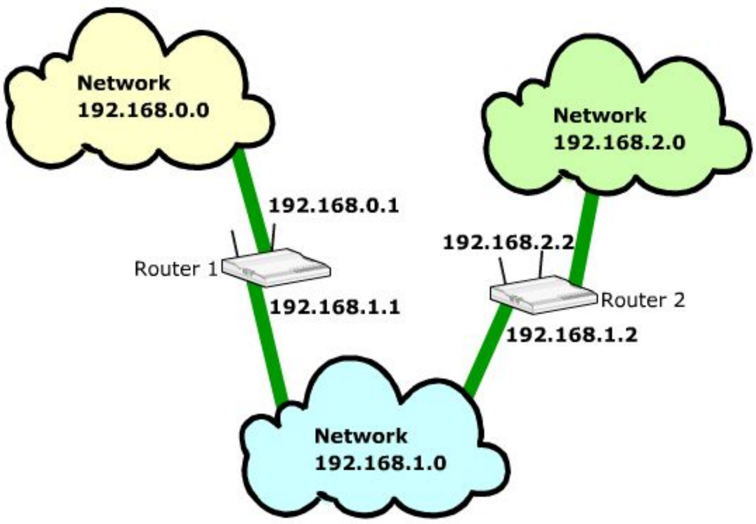


Рисунок 5.13 - Мережа з використанням маршрутизаторов

Маршрутизатор буде вивчати IP-адресу пакета даних і визначати за допомогою підмережі, чи знаходиться він в тій же мережі чи ні. Якщо немає, він відправить пакет даних у відповідну мережу.

Ще однією перевагою маршрутизаторів є можливість підвищення безпеки мережі. Мережа автоматизації може бути захищена строгими і індивідуально адаптованими правилами доступу. Пристрій для мережевої безпеки також називається «брандмауер», яке може бути інтегровано в маршрутизатори або розділене як один пристрій.

Будь ласка, зверніть увагу; Зв'язок в режимі реального часу в PROFINET не функціонує поза межами мережі. Це також можливо тільки для зв'язку з пристроями PROFINET всередині мережі.

Кодування сигналів. Який сигнал кодування для Ethernet буде використовуватися, залежить від швидкості передачі цього Ethernet. У стандарті Ethernet 10 Мбіт / с від IEEE 802.3 використовуватиметься манчестерське кодування. В Fast Ethernet використовується технологія FDDI. Інформація довжиною 5 біт буде зібрана в «символ». Символи будуть передані. У Gigabit Ethernet інформація довжиною 10 біт буде зібрана в символ.

VLAN (Віртуальна локальна мережа) - це віртуальна локальна мережа, VLAN - це метод створення замкнутої мережі. Ця логічна мережа може незалежно існувати у фізичній мережі, тобто ви можете відокремити логічну і фізичну структуру мережі. VLAN створює свій власний широкомовний домен відповідно до зазначених логічними критеріями.

Компоненти мережі, які підключаються до однієї і тієї ж віртуальної локальної мережі, можуть фактично фізично підключатися до різних сегментів локальної мережі. Якщо ви переміщаєте мережевий компонент фізично з А в В, вам не потрібно переконфігурувати апаратне забезпечення. Нижче наведені деякі властивості VLAN:

– зниження мережевого трафіку і підвищення мережевої безпеки;
– скорочення управлінських зусиль для створення підмереж;
– зниження вимог до обладнання та вартості;

– компоненти мережі можна розділити на функціональні групи, наприклад, сервери, IPC і т. Д.;

– фізичне відстань не впливає на розподіл завдань для пристроїв;

– просте зіставлення функціональних блоків з мережевою структурою.

5.4.3 Сімейство Internet Protocol

Коли ви хочете спілкуватися з людьми, ви повинні використовувати одну і ту ж мову. Аналогічно, в світі інформаційних технологій (IT), де протоколи використовуються в якості спільної мови, кожний IT-пристрої використовує для зв'язку однакові протоколи. В IT використовується безліч протоколів. Деякі з них незалежні від виробника, але деякі з них є власністю. Можна подивитись на деякі з необхідних протоколів в промислової автоматизації з використанням Ethernet.

Інтернет-протокол (IP). Протокол Інтернету, ймовірно, є найбільш часто використовуваним протоколом в світі інформаційних технологій. Поточна версія цього протоколу - версія 4 (IPv4), але версія 6 (IPv6) буде повністю використовуватися в найближчому майбутньому, оскільки можна призначити набагато більше адрес. IP забезпечує передачу даних без встановлення з'єднання, що передає і приймає дейтаграми. Датаграма складається з заголовка і частини даних (корисного навантаження).

Термін датаграмма або пакет використовується для опису фрагмента даних IP. Кожна датаграмма IP містить певний набір полів в певному порядку, так що читач знає, як декодувати і читати потік отриманих даних.

Використовується 32-розрядна IP-адреса для адресації польових пристроїв, що позначається десятковими цифрами. IP-адреса складається з двох частин: ідентифікатор вузла та ідентифікатор мережі. IP-адресу можна розділити на 5 класів (A, B, C, D і E), але в практичних додатках використовуються тільки A, B і C (Таблиця 4). Перші біти представляють собою мережевий клас, за яким слідує мережева адреса І адреса користувача. Погляньте на таблицю нижче.

В межах IP-мережі ви можете використовувати маску підмережі для розширення своїх мережевих адрес. У цьому прикладі ми використовуємо 255.255.255.0 в якості маски підмережі. Тому ви можете призначати адреси від 192.168.0.0 до 192.168.0.255, де у останню адресу 192.168.0.255 є спеціальна функція; А саме функції широкомовної передачі.

Маска підмережі. Маска підмережі вказує, скільки мережевих пристроїв може підключитися до цього IP-адресою. В бінарній формі ті, що в масці підмережі, вказують частина номера підмережі, а нулі вказують частина хоста. У таблиці 6 показана маска підмережі 255.255.255.0 для IP-адреси класу C, 192.168.1.1.

Таблиця 5.4 - Класи IP-адрес

	Перві біти	**ID мережі**	**ID абонту**	**Адреса**	**Кіл-ть адрес**	**Абон-ти**
Class A	0	7 bits	24 bits	0.0.0.0 - 127.255.255.255	126	16777216
Class B	10	14 bits	16 bits	128.0.0.0 - 191.255.255.255	16384	65536
Class C	110	21 bits	8 bits	192.0.0.0 - 223.255.255.255	2097152	256
Class D	1110	28 bits, multicast identifier		224.0.0.0 - 239.255.255.255		
Class E	11110	27 bits, reserved		240.0.0.0 - 247.255.255.255		

Приклад в таблиці 5.5 показує IP-адресу класу C.

Таблица 5.5 - Приклад IP адреси класу C

Адреса мережі	**Маска підмережі**	**Адреси в мережі**	**Широковещательный**
192.168.0.0	255.255.255.0	192.168.0.1 - 192.168.0.254	192.168.0.255

Таблиця 5.6 - Маска підмережі IP-адреси класу C

	Сеть	**Абонент**
Decimal	255.255.255.	0
Binary	11111111.11111111.11111111.	00000000

З прикладу в таблиці 6 ми можемо підключити до цієї мережі 256 мережевих пристроїв, і вони будуть мати IP-адреси від 192.168.1.0 до 192.168.1.255. Але, як і раніше, перший (192.168.1.0) і останній (192.168.1.255) адреси зарезервовані для спеціальних призначень.

Протокол призначених для користувача дейтаграм (UDP) - використовується для надання послуг передачі даних без встановлення з'єднання і ненадійність. UDP не буде повідомляти відправнику, чи був отриманий або втрачений пакет даних. Використовуючи UDP, ви можете зіткнутися з проблемами цілісності даних, тому що пакет даних може бути видалений, неправильно впорядкований або подвоєний. Однак UDP має короткий і простий заголовок, довжиною всього 8 байт. UDP вводить концепцію номерів портів, які використовуються рівнем додатки, який знаходиться вище UDP.

UDP забезпечує більш швидкий обмін даними, тому що відправнику буде підтверджено підтвердження. Тому цей протокол використовується для ациклічного обміну даними і запуску системи в PROFINET IO.

Протокол управління передачею (TCP) - використовується для надання послуг передачі даних на основі сполучень і надійних пакетів даних. Можна бути впевненим, що пакет даних доставлений правильно. Крім надійних з'єднань, TCP забезпечує контроль потоку, щоб гарантувати, що станції не будуть затоплені даними.

Тема TCP більше, ніж UDP. У деяких випадках довжина може становити до 40 байтів. TCP використовує ту ж схему призначення портів, що і UDP.

Призначення номера порту. Номер порту вводиться в поєднанні з транспортним протоколом, таким як UDP або TCP. Це 16 бітний ідентифікатор. Номер порту має значення, так як він ідентифікує конкретне застосування на 7-му рівні моделі OSI. Ця властивість номера порту дозволяє одній станції одночасно підтримувати кілька різних застосувань. Використання номера порту разом з IP-адресою створює так званий сокет і виглядає так: 192.168.0.1:8080.

Існує три способи класифікації номерів портів:

– назначені: повноваження інтернет-привласнених номерів (IANA) визначило призначені номери портів в діапазоні від нуля до 1023 для різних додатків, які вважаються частиною набору протоколів TCP/IP. Ці номери портів добре відомі, що означає, що вони використовуються в звичайних застосуваннях і не можуть використовуватися іншими додатками. Ці програми включають TELNET, FTP, SMTP, WWW, POP3 і т. Д.

– зареєстровані: номера портів в діапазоні від 1024 до 49151 можуть бути зареєстровані фірмами або організаціями для власних цілей. Інші фірми або організації більше не можуть використовувати ці номери портів.

– динамічні: номера портів в діапазоні від 49152 до 65535 вважаються приватними або динамічними і можуть використовуватися будь-яким користувачем.

Протокол дозволу адрес (ARP) і Протокол дозволу зворотної адреси (RARP). Ці два протоколи дуже важливі для рівня мережевого доступу. За допомогою цих протоколів можна дозволити логічні адреси фізичним адресами і навпаки. ARP використовується для дозволу Ethernet-адреси з IP-адреси, щоб побудувати пакет Ethernet навколо пакету даних IP. Це повинно відбуватися для відправки будь-яких даних по мережі. Якщо IP хост хоче викликати інший IP хост, знаючи тільки його IP-адресу, ARP-запит через трансляцію LAN відправляється всім локальним IP-хостам. Їй відповідає станція (якщо активна) відповіддю ARP, повідомляючи свій MAC-адресу у відповідь.

RARP використовується для визначення IP-адреси з використанням мережі. Коли мережевий пристрій хоче дізнатися свою IP-адресу, він відправляє запит RARP на сервер RARP. Сервер RARP знайде відповідну IP-адресу в своєму файлі конфігурації і відправить цю IP-адресу назад з відповідним пакетом RARP.

Інші корисні протоколи:

– LLDP: протокол виявлення низького рівня використовується для розпізнавання свого сусіда. Кожна станція відправляє свою власну інформацію, таку як MAC-адресу, ім'я пристрою і т. д. у вигляді кадру спрямованому сусідові.

– DHCP: протокол конфігурації динамічного хоста використовується для призначення IP-адрес і пов'язаних з ними параметрів, якщо відповідна інфраструктура доступна.

– DNS: служба імен доменів використовується для управління логічними іменами.

– SNMP: для моніторингу мережі використовується простий протокол мережевого управління. За допомогою цього протоколу можна зчитувати стан, статистичну інформацію та виявляти помилки зв'язку.

– ICMP: Протокол керуючих повідомлень Інтернету використовується для пересилання інформації про помилку.

5.5. PROFINET IO

5.5.1 Огляд

У попередньому розділі обговорювалися основи і переваги PROFINET. У цьому розділі більш детально розглянуті деталі PROFINET. Почнемо з PROFINET IO.

PROFINET IO - це засіб комунікації для розподілених IO пристроїв. На відміну від моделі комунікації "Ведучий-ведений" (Master-Slave), що використовується в PROFIBUS, концепція з'єднання PROFINET IO використовує так звану модель "Споживач-постачальник" (SOA). Пристрій-одержувач даних називається "Споживач", а пристрій, що відправляє дані - "Постачальник". Існує безліч пристроїв з різними конфігураціями, що використовуються в PROFINET IO. Схожі і прості пристрої групуються і підключаються до розподілених IO пристроїв.

Призначення пристроїв:

– *IO Controller:* IO контролер - це зазвичай ПЛК (Програмований Логічний Контролер), на якому виконується програма автоматизації процесу. Цей контролер використовується для адресації підключених пристроїв, тобто, що IO контролер дозволяє обмінюватися вхідними та вихідними сигналами з присвоєними йому пристроями. Для порівняння, в PROFIBUS пристрій, що виконує цю функцію називається Master class 1.

– *IO Supervisor:* IO керівник - це зазвичай робоча станція. Це може бути програматор, персональний комп'ютер або ЧМІ (людино-машинний інтерфейс) для експлуатації або діагностики. Для порівняння, в PROFIBUS пристрій, що виконує цю функцію називається Master class 2.

– *IO Device:* IO пристрій - це розподілений пристрій, підключений до одного з IO Контролерів. Для порівняння, в PROFIBUS пристрій, що виконує цю функцію називається

Slave.

Підсистема повинна містити як мінімум один IO Контролер і один або більше IO Пристрій. З'єднання між цими пристроями виконується синхронно шляхом передачі даних через систему введення-виведення і асинхронно - по аварійним сигналам і передачі керуючих даних.

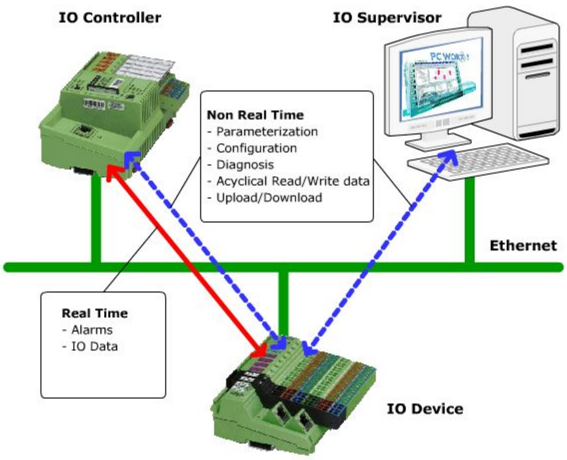


Рисунок 5.14 – Ролі пристроїв в PROFINET

Служби PROFINET IO:

– *об'єкти введення-виведення даних:* об'єкти введення-виведення даних здійснюють синхронне пересилання даних між пристроєм-споживачем і пристроєм-постачальником. Ці передачі даних здійснюються без підтвердження. Інтервал передачі даних налаштовується за використанням спеціального інженерного інструментарію.

– *об'єкти записи даних:* об'єкти записи даних використовуються для установки параметрів IO пристроїв, їх налаштування та зчитування з них поточної інформації. Об'єкти записи даних використовують асинхронну передачу даних за допомогою служб читання і запису. Прикладами таких даних може бути діагностична інформація, параметризаційна інформація, інформація для I&M (ідентифікації та технічного обслуговування), інформація про дані вводу-виводу та інше.

– *об'єкти сигналізують даних:* вони служать для передачі інформації про події, які повинні бути підтверджені для подальшого виконання. Існує два їх різновиди: визначенні системою і визначені користувачем. Події визначаються системою - це додавання чи видалення нових модулів. Події, визначені користувачем - це некоректна напруга або температура. Користувач може по різному розставити пріоритети діагностичних і обробних попереджувальних сигналів.

Модель IO пристроїв. IO пристрій має модульну структуру (рис. 15). Він складається зі слотів. Ви можете підключати модулі/суб-модулі в ці слоти. Модулі/субмодулі мають канали, що використовуються для зчитування і передачі сигналів. Входи і виходи рознесені по цих каналах і встановлюється виробником пристрою. Модулі/субмодулі можуть бути підключені або відключені в процесі роботи пристрою, якщо це необхідно.

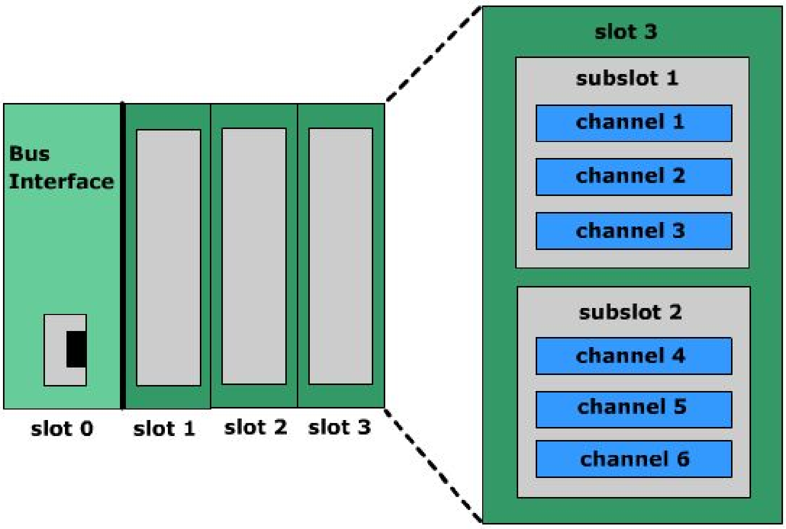


Рисунок 5.15 - Модель IO пристроїв

– **Слот:** Слот - це фізичне місце в IO пристрої в який Ви можете помістити модуль. У різних слотів існує кілька субслотов. Ці субслоти містять дані для синхронної передачі даних. Завод-виробник визначає можливості модуля і визначає технічні дані у файлі GSD.

- **Субслот:** Субслот використовується для адресації. Слот повинен містити як мінімум один субслот, що містить канали. Таким чином, ми можемо згрупувати канали на одному субслоті. Опис міститься в GSD файлі від постачальника.
- **Індекс:** Індекс визначає асинхронні служби читання або запису для зазначеного субслоту. Виробник визначає можливості доступу до слоту і його субслотів. Доступ може бути здійснений через віддалені IO контролери, в той же час, доступ на запис або аварійний сигнал може бути переданий тільки на один IO Контролер. З іншого боку, доступ для читання може бути дозволений для кількох контролерів.

Для простої інтеграції профілів додатки, PROFINET IO забезпечує спеціальні адресації елементів, так звані API (Програмовані інтерфейси додатка). Таким чином, можна використовувати різні інтерфейси API для підтримки декількох профілів додатків в одному пристрої.

Засоби зв'язку. Відповідно до асинхронної або синхронної передачі даних між IO Контроллером/Керівником і IO пристрої, IO контролер повинен створити необхідні комунікації на старті системи. Рисунок 3 показує комунікації між PROFINET IO контролером і IO пристроєм. Щоб це зробити, IO Контролер встановлює з'єднання на основі даних інженерного застосування.

Відносини застосувань (AR) містять всі дані, необхідні для обміну даними. В межах відносин застосувань (AR), є один або кілька API-інтерфейсів, які дозволяють точно налаштувати групи застосувань. Відносини застосувань (AR) можуть містити одне або кілька відносин зв'язків (CR).

Нижче наведені Відносини/Зв'язки, які доступні для кожного API:

– Одне або більше IO Відносин/Зв'язки для синхронної передачі даних.

– Сигналізують Відносини/Зв'язки для подій.

– Відносини/Зв'язки, Записи даних для асинхронної передачі даних.

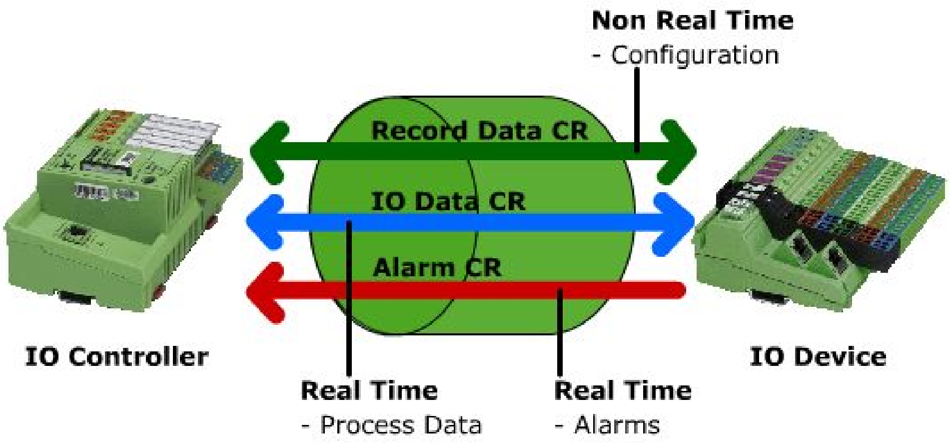


Рисунок 5.16 – Средства связи

5.5.2 Комунікації

У попередньому розділі ми обговорили принципи комунікації в PROFINET в режимі реального часу, в режимі ізохронного реального часу і в некритичному режимі. Крім того, мережа PROFINET заснована на взаємодії через комутатори Ethernet. Це має на увазі, що кожен пристрій PROFINET безпосередньо пов'язане з кожним іншим пристроєм за допомогою комутаторів. Комутатор може бути як стороннім пристроєм, так і інтегрованим в один з пристроїв з PROFINE. Комутатор має кілька портів в які можуть бути приєднані пристрої.

Всі пристрої під'єднані через комутатор, в тому числі інші комутатори PROFINET. Таким чином ми можемо поєднати будь-яку кількість пристроїв PROFINET в єдину велику мережу.

Процес передачі даних може бути поділений на п'ять частин, як ми бачимо на (Рис. 5.17). Спочатку відправник передає дані (T1), вони передаються в комутатор (T2)

Потім дані відправляються до одержувача (T3). Отримувач отримує дані через ланцюг комунікаційних пристроїв (T4), в кінці відбувається процес обробки вхідних даних (T5). Час, що використовується в цих п'яти кроках підсумовується і розглядається як єдине або окремими інтервалами. Цей час може бути визначено як некритичний, реального часу, ізохронного реального часу.

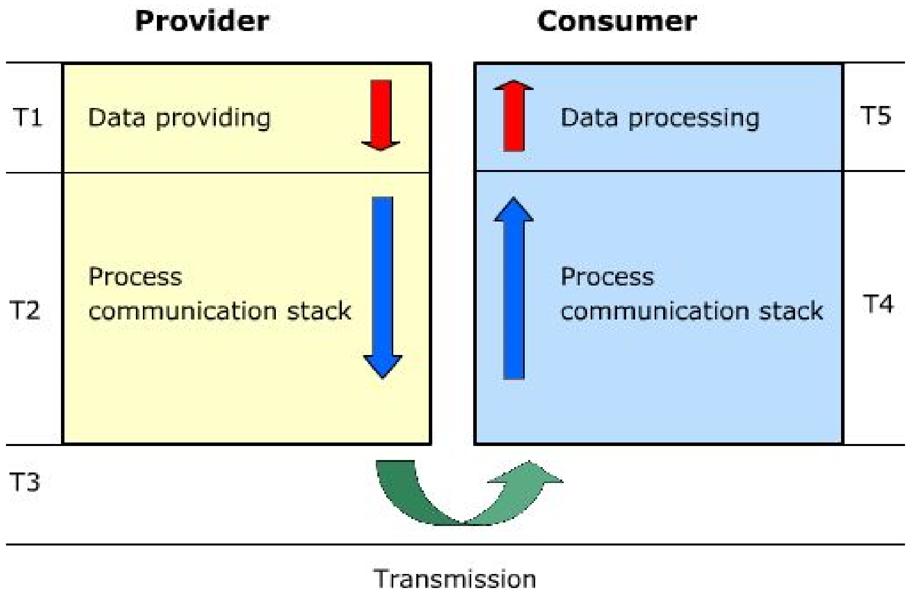


Рисунок 5.17 - Процесс передачи данных

Процеси некритичного часу. У PROFINET, для процесів некритичного часу, таких як параметризація, конфігурація або людино-машинна взаємодія, ми використовуємо комунікацію за допомогою стандартних механізмів Ethernet, таких як TCP/IP або UDP/IP, відповідно до стандарту IEEE 802.3. Цей тип комунікації в PROFINET також має називають НРТ комунікація (комунікація не-реального часу).

Присвоєння адреси виконується за допомогою IP і MAC адрес. IP-адреса застосовується для комунікації між двома і більше мережами. MAC-адресу використовується для однозначної ідентифікації пристрою всередині мережі. Всі пристрої PROFINET повинні підтримувати можливість передачі даних за допомогою UDP / IP. Звичайний час циклу становить 100 мілісекунд (запит-відповідь).

Процеси реального часу. Для передачі даних в реальному часі використовуються RT-комунікації (Рис. 5.18). Таким чином досягається той же рівень продуктивності, як в існуючих шинних системах. Це дозволяє повністю покрити вимоги автоматизації виробничих процесів.

Щоб перейти в режим реального часу, PROFINET відкидає частину пакетів IP, TCP і UDP. Механізм протоколу Ethernet, що працює на 2 рівні моделі OSI дозволяє це здійснити. З'єднання реального часу можуть бути здійснені одночасно з зв'язком нереального часу.

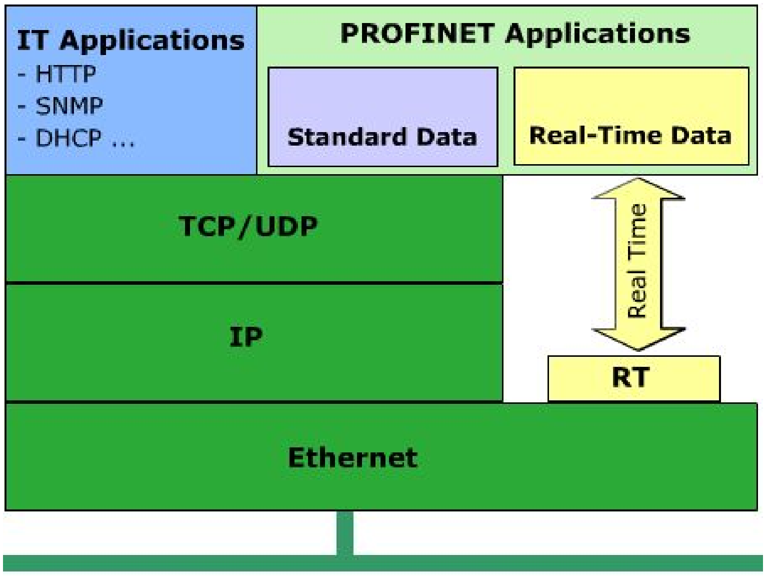


Рисунок 5.18 – PROFINET связь в реальном времени

Існує три варіанти зв'язку в реальному часі:

– RT-комунікація всередині мережі: в цьому випадку видаляється інформація про адресу.

– RT-комунікація між мережами: в цьому випадку виконується передача по UDP.

– Широкомовне передача даних за допомогою РВ-комунікації: в цьому випадку для синхронної передачі даних виконується передача по UDP.

Звичайний час циклу становить 5-10 мілісекунд (запит-відповідь).

Процеси ізохронного реального часу. Для тимчасово синхронізованих додатків, особливо в області застосувань контролю руху і високопродуктивних застосувань автоматизації виробництва, апаратні засоби реалізуються на основі ІРЧ (Ізосінхронного Реального Часу) (Рис. 5.19). ІССН (ASICs) (Інтегральна Схема Спеціального Призначення) з функцією перемикання і синхронізації циклу використовуються для актуалізації цього типу зв'язку.

Phoenix Contact розробив новий Tiger-Chip (Тайгер-Чіп) TPS 1. Він став доступний в 3 кварталі 2011 року. Чіп в межах низьких цін (около.10 €) повністю сумісний з ІССН (ASICs) Ertec 200, Ertec 400 і підходить для РВ, ІРВ також для всіх можливих розширень реального часу.

Цикл зв'язку розділений на визначальну частину і відкриває частину. Визначальний канал переносить цикл ІРЧ пакетів, в той час, як пакети TCP/IP і РВ транспортуються по відкритому каналу. В цьому випадку, обидва типи передачі даних існують разом, не заважаючи один одному. Важливо знати, що якщо необхідно, то для зв'язку типу ІРВ, ширина шини (пропускна здатність) повинна бути розділена на детерміновану і відкриває частину під час стадії проектування.

Типовий час циклу становить менше 1 мс з похибкою менше 1 мкс.

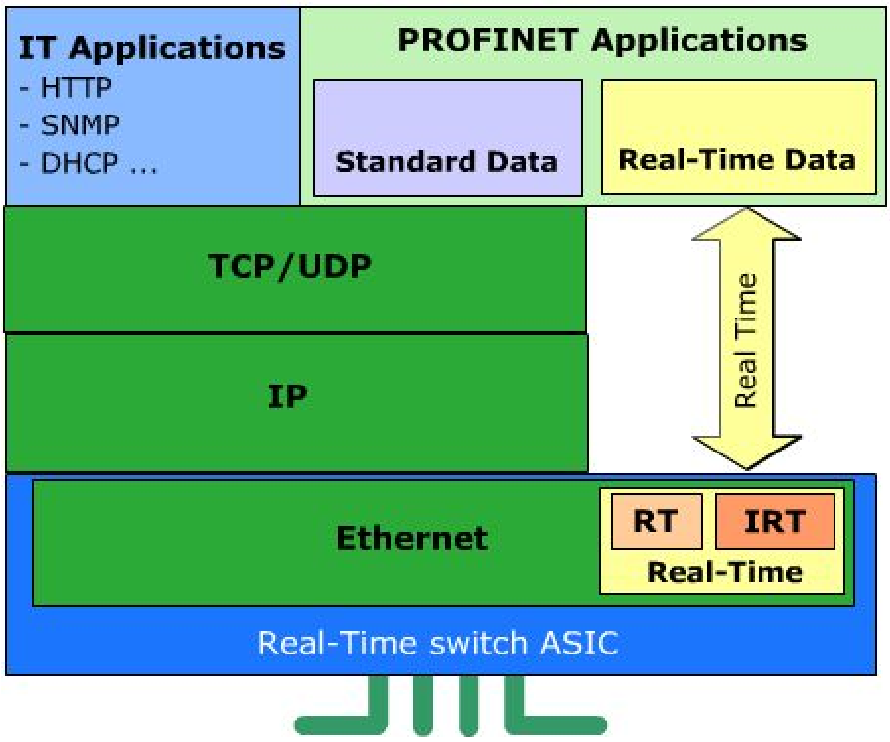


Рисунок 5.19 – З'єднання PROFINET в ізохронному реальному часі

5.5.3 GSDML

Кожне поле вводу/виводу пристрою PROFINET має бути описано з GSD файлом. A GSD (General Station Description (Основний Опис Станцій)) файл XML-подібний і містить всю технічну інформацію і функції пристрою. Ця інформація має важливе значення для проектування і обміну даними. GSD файли надаються виробником пристрою. Файл GSD є багатомовним і може бути відредагований в практично всіх текстових редакторах. Так як файл GSD записується в XML ми називаємо мову опису GSD файлів GSDML (General Station Description Markup Language (Мова Розмітки).

Назва GSD Файлів. Стандарти для призначення імені GSD файлів наступні:

GSDML-[GSD schema version]-[manufacturer name]-[device family name]-[date].xml

Там, де ключові слова в квадратних дужках, вони визначаються в такий спосіб

– GSD версія схеми це версія використовуваної схеми.

– Ім'я виробника це ім'я виробника пристрою. Воно може містити дефіси і прогалини.

– Серійне ім'я пристрою це ім'я сімейства пристроїв. Він може містити дефіси і прогалини.

– Дата має формат гггг.мм.дд. Різні GSD файли з тією ж датою і сімейством пристрою не допускається.

Допустиме ім'я файлу GSD може виглядати наступним чином:

GSDML-V1.0-Phoenix Contact-ILB PN 24 DI16 DIO16-2TX-V1.0-20051206.xml

Унікальний Ідентифікатор Пристрої. Кожне поле вводу/виводу пристрою PROFINET ідентифікується по всьому світу унікальним ідентифікаційним номером пристрою (Рис. 20). Це число містить дві частини, VENDOR_ID і DEVICE_ID. Кожна частина числа має довжину 16 біт.

– Vendor_ID: Використовується для ідентифікації виробника і може бути отриманий з PROFIBUS Nutzerorganisation e.V. А Для виробника пристрою необхідний тільки один Vendor_ID.

– Device_ID: Використовується для ідентифікації поля пристрою і може бути призначений виробником.

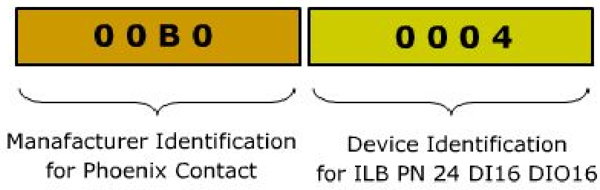


Рисунок 5.20 - Приклад унікального ідентифікаційного номера пристрою (Device_ID)

Структура GSD файлу. GSD файл заснований на стандарті ISO 15745 та складається з двох частин - заголовок профілю і тіло профілю - profile header і profile body. На рисунку 5.21 показана структура GSD файлу

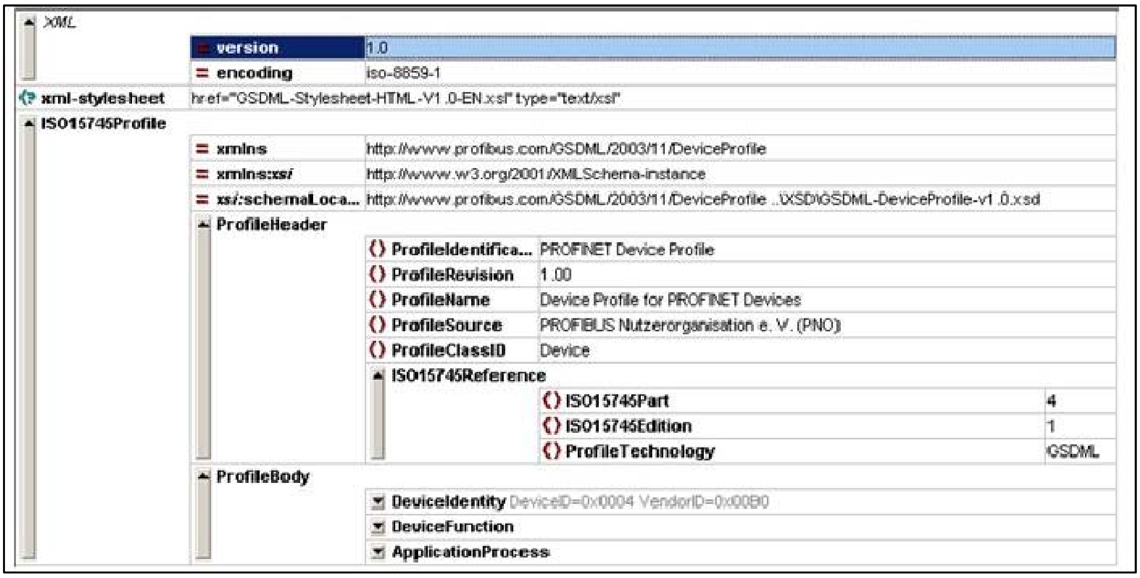


Рисунок 5.21- Структура GSD файлу

Тема профілю. Ця частина містить основні положення специфікації, що описують функціональність пристрою. Також є посилання на міжнародний стандарт ISO 15745.

<ProfileHeader>
<ProfileIdentification>PROFINET Device Profile</ProfileIdentification>
<ProfileRevision>1.00</ProfileRevision>
<ProfileName>Device Profile for PROFINET Devices</ProfileName>
<ProfileSource>PROFIBUS Nutzerorganisation e. V. (PI)</ProfileSource>
<ProfileClassID>Device</ProfileClassID>
<ISO15745Reference>
<ISO15745Part>4</ISO15745Part>
<ISO15745Edition>1</ISO15745Edition>
<ProfileTechnology>GSDML</ProfileTechnology>
</ISO15745Reference>

</ProfileHeader>

Тіло профілю. Інформація в цьому розділі описує технічні можливості пристрою. Всі модулі, підмодулі, вихідні попередні налаштування і засоби діагностики описані тут.

Цей блок може бути поділений на:

– блок ідентифікує пристрій - DeviceIdentity block (VendorID, DeviceID, InfoText, VendorName)

– блок описує функціональність пристрою - DeviceFunction block (Family, MainFamily, ProductFamily)

– блок прикладного процесу - ApplicationProcess block

a) Список пристроїв, що підключаються - DeviceAccessPointList: містить опис розширень окремих інтерфейсних модулів.

b) Вікно діагностики - ChannelDiagList: поля для помилок і їх змісту.

c) Вікно відображення - GraphicsList: містить посилання на графічне представлення пристрою.

d) Список категорій - CategoryList: використовується для створення категорій для інженерного інструментарію.

e) Зовнішній текст - ExternalTextList: містить весь текст, який може посилатися на інші сегменти.

5.5.4 Практична вправа: перші кроки в конфігурації

Знадобляться такі компоненти для того, щоб створити систему:

– Інженерний інструментарій: Це IO Supervisor, такі як PCWorX або STEP 7.

– IB контролер: Нам потрібен принаймні, один IO контролер для системи.

– IB Device: Нам потрібен принаймні один IO Device.

– GSD-файли: Вони описують властивості PROFINET пристроїв, і містять всю інформацію, необхідну для конфігурації.

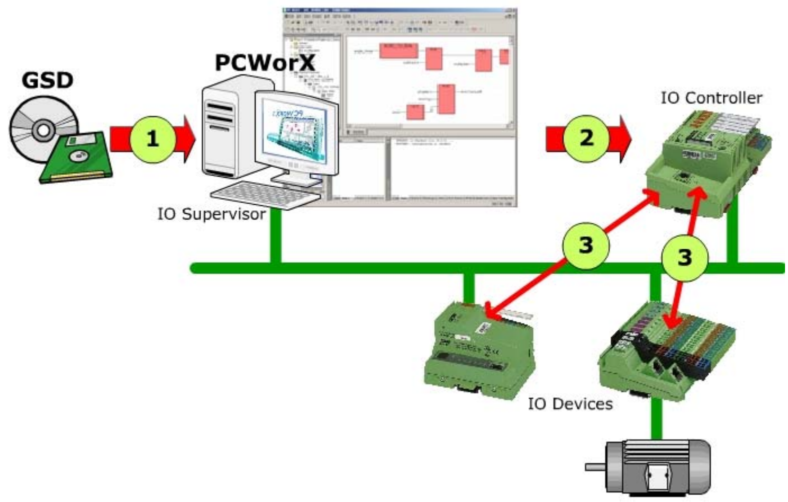


Рисунок 5.22- Шаги для створення PROFINET IO системи

Крок за кроком: Файли GSD імпортуються в інженерні інструменти. В інженерних інструментах, децентралізовані польові пристрої призначені одному або декільком контролерам вводу-виводу. На цьому етапі, пристрої введення-виведення повинні бути налаштовані в фактичну системну мережу, засновану на утриманні GSD файлу. Пристрій вводу-виводу одночасно інтегровані, відповідним чином параметризованих і конфігуровано в топології PROFINET.

Після завершення процесу проектування, відомості про конфігурацію і призначені для користувача програми передаються в контролер введення-виведення.

Контролер введення-виведення самостійно приймає на обмін дані з децентралізованою IO пристрої автоматично.

Призначення адрес.

Адресація польових пристроїв на основі IP-зв'язку може бути зроблено за допомогою IP-адреси. Для присвоєння адрес, PROFINET використовує DCP (Discovery і Configuration Protocol).

Як представлено, кожний польовий пристрій має, крім усього іншого, MAC-адреса і символічне ім'я. Ми можемо використовувати цю інформацію, щоб привласнити ім'я відповідного польового пристрою.

Призначення адрес може бути зроблено з використанням протоколу DCP наступним чином (рис. 23):

– Присвоєння унікального імені для відповідного польового пристрої;

– Присвоєння IP-адреси з контролера вводу-виводу до запуску системи.

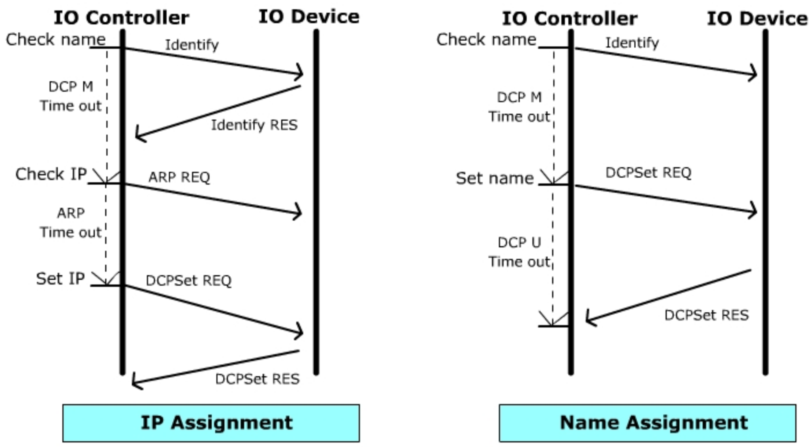


Рисунок 5.23 – Призначення адресів

Заміна пристрою. У разі якщо модуль пристрою або повне пристрій є несправним, воно може бути швидко замінено без необхідності повторної конфігурації запасної частини. Це відбувається тому, що змінні носії, такі як MMC (Multi Media Card) і C-PLUG дозволяють просту заміну польових пристроїв і мережевих компонентів без необхідності влаштування програмування.

Якщо вийняти картку пам'яті з пристрою PROFINET і встановленні її в інший пристрій PROFINET, ім'я пристрою передається. У разі активних компонентів мережі, такі як комутатори, вся сукупність даних конфігурації знаходиться безпосередньо на C-Plug.

Після заміни, контролер введення-виведення автоматично виконує параметризацію і конфігурацію нового пристрою, і призначає новий IP-адресу. Потім, циклічний обмін даними користувачів відновлюється.

Діагностика и аварійні сигнали. Діагностика PROFINET пристроїв проводиться в 3 етапи:

– Діагностика помилок в пристрої, наприклад, виходу з ладу станції.

– Діагностика помилок в розташуванні модуля, наприклад, пошкодження модуля аналогового вводу.

– Діагностика помилок на каналі, наприклад, обрив дроту.

На додаток до пристрою, в мережі PROFINET мережева інфраструктура може бути діагностовано за допомогою протоколу SNMP.

У прикладі, показаному на рис. 5.24, провід між пристроєм введення-виведення і вимикачем був розірваний. Комутатор генерує сигнал тривоги в контролер введення-виведення, який також містить інформацію про пошкодження порту. Крім того, діагностична інформація може також зчитуватися безпосередньо з головного контролера вводу-виводу.

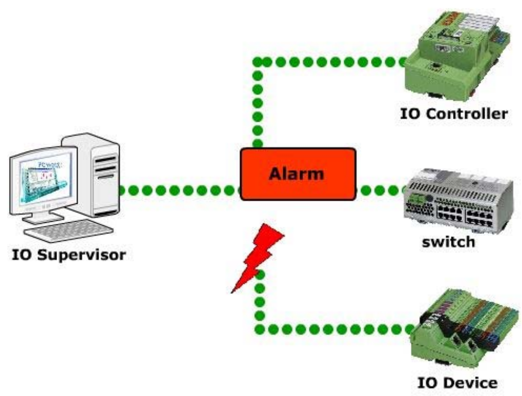


Рисунок 5.24 – Приклад виникнення аварійного сигналу

Користувач має дві можливості доступу до діагностики PROFINET:

– Діагностика через контролер введення-виведення: Пристрій вводу-виводу відправляє інформацію діагностики в контролер введення-виведення, що поміщає її в SZL (списки стану системи). Головний контролер введення-виведення звертається до SZL через контролер введення-виведення. Будь-який дефект після можуть бути візуалізовані в HMI (Human Machine Interface). Крім цього, це може бути відповіддю на помилку в додатку.

– Діагностика безпосередньо з пристрою введення-виведення: технічний інструмент (Головний контролер вводу-виводу) також може зчитувати стан пристрої введення-виведення безпосередньо. Щоб зробити це, пристрої введення-виведення має бути підключений до мережі Ethernet.

PROFINET дозволяє ще один варіант для діагностики. Ви можете прочитати безпосередньо індикатори стану на пристрої: чи можливий зв'язок без перешкод, і чи можуть дані бути відправлені або отримані.

5.6. PROFINET CBA

5.6.1 Огляд

У той час як PROFINET IO забезпечує концепцію модульної автоматизацію в рамках заводу, PROFINET CBA (Component Base Automation) забезпечує модульну концепцію автоматизації між заводами (рис. 25). Різні підсистеми або машини можуть бути розроблені, випробувані і введені в експлуатацію одним або декількома виробниками з одним стандартом. Горизонтальна масштабованість може бути досягнута таким способом.

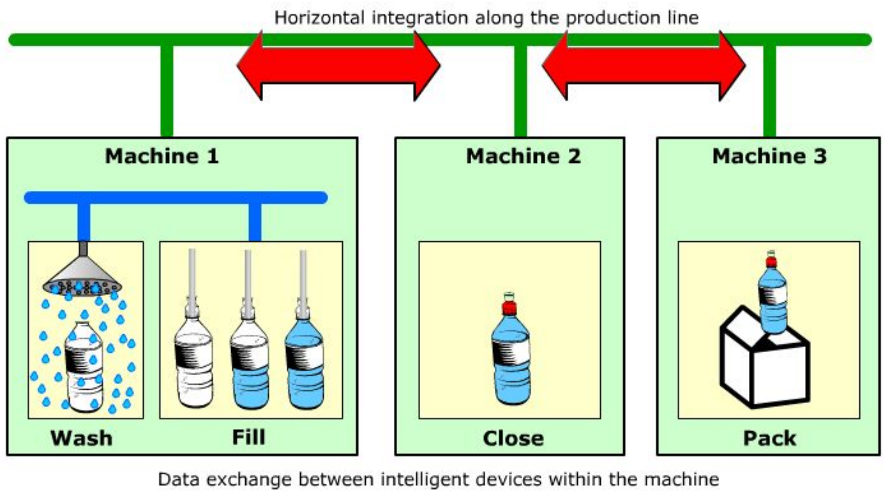


Рисунок 5.25 - Горизонтальне масштабування в системах автоматизації

Технологічні модулі в PROFINET CBA складаються з:

– Механічних компонентів,
– електричних або електронних компонентів,
– контроль логіки або програмного забезпечення.

Три компонента технологічного модуля описують функцію автоматизованого заводу або машини (рис. 26). Вони працюють один з одним в процесі виготовлення продукту. Таким чином, технологічний модуль являє собою повну конкретну систему заводу або машини.

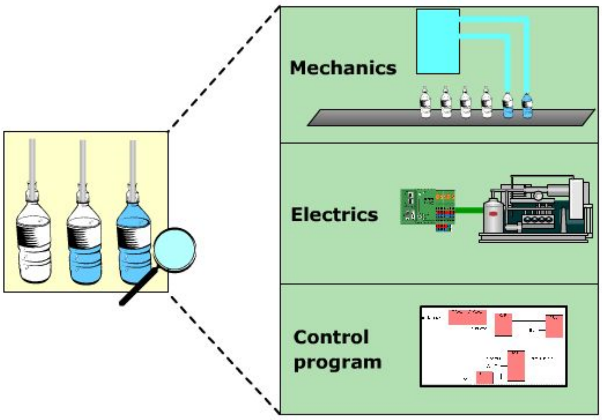


Рисунок 5.26 – Три компонента технологічного модуля

PROFINET CBA заснована на компонентної моделі (рис. 27), тобто вона складається з декількох невеликих підрозділів. Ці підрозділи координують між собою, обмінюючись своєю інформацією, і діють автономно, як технологічні модулі.

Компоненти, такі як машини, або заводи, або їх частини, будуть розглядатися в якості технологічних модулів. Функції цих компонентів інкапсулюються унікальними COM-об'єктами, звані компоненти PROFINET. COM (Component Object Model) була розроблена Microsoft з використанням об'єктно-орієнтованим концептом. З COM, можна легко розробляти компоненти, які базуються на готових компонентах.

Компоненти PROFINET є також COM-об'єктами, але їх функції є специфічними для PROFINET. Компонент PROFINET можуть бути використані як чорний ящик з унікальним певними інтерфейсами. Таким чином, ви можете підключити компоненти PROFINET один до одного в будь-якій комбінації.

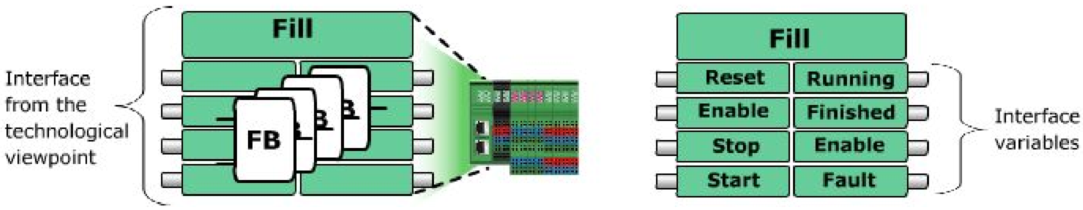


Рисунок 5.27 – Компонентна модель PROFINET CBA

Для конфігурації і взаємозв'язку з іншими пристроями PROFINET, тільки до змінних, яким необхідно взаємодіяти один з одним, можна отримати доступ. Визначення для зв'язку визначається в COM-об'єкти, і включає в себе;

– Комунікація в рамках одного процесу.
– Комунікація між двома процесами в одному пристрої.
– Комунікація між двома процесами в різних пристроях.

Цей вид конструкції розподіленої системи автоматизації (рис. 28) необхідний в разі, якщо ви хочете мати модульну реалізацію ваших систем і машин. Крім того, повторне використання ваших системних або машинних частин можливо, що означає зниження технічних витрат.

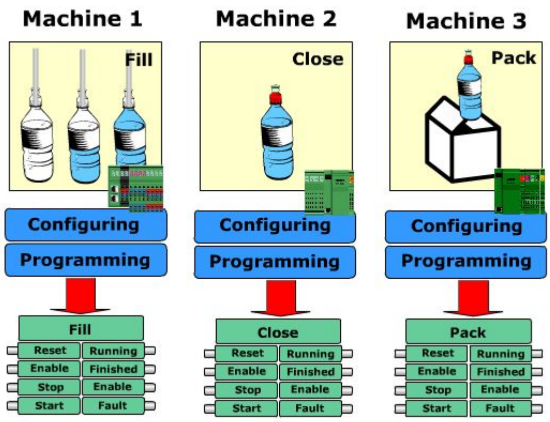


Рисунок 5.28 - Приклад розподіленої автоматизованої системи

Структура автоматизації. Як і PROFINET IO, компоненти PROFINET CBA пов'язані з використанням комунікаційних промислових механізмів Ethernet зв'язку. Існуючі компоненти PROFIBUS можуть бути інтегровані в цю систему, використовуючи підходящий проксі-сервер (Рис. 5.29).

5.6.2 Проектування

Для користувачів, важливо, щоб прикладні інструменти PROFINET були прості у використанні і мали стандартизовані характеристики. Для цієї мети, незалежні постачальники домовляються про визначення прикладних понять. Ця концепція заснована на проектуванні об'єктної моделі. Це означає, що ви можете використовувати компоненти від різних виробників, а також функціональні розширення від конкретних постачальників або від конкретних користувачів з одними і тими ж прикладними інструментами. Продукти різних виробників можуть бути легко продіагностіруємимі і інтегровані в одну установку за допомогою редактора з'єднань. За допомогою редактора з'єднань можна встановити зв'язок відносин між компонентами, малюючи лінії між ними. Потім редактор з'єднання перевірить правдоподібність з'єднань.

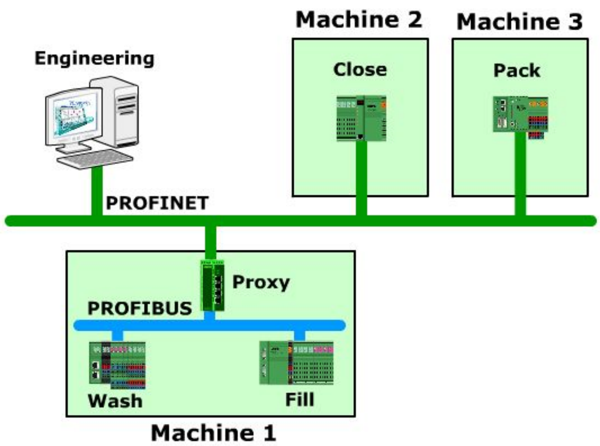


Рисунок 5.29 – Структура автоматизації з PROFINET и PROFIBUS

У PROFINET CBA, обмін даними між вузлами забезпечується сконфігурованими сполуками. Файл на основі XML описує технічні функції пристрою PROFINET CBA. Цей файл називається PCD (PROFINET Component Description - PROFINET компонентний опис) і надається виробником пристрою.

Загальносистемне застосування може бути побудовано в три етапи (рис. 5.19):

– створення компонентів;
– встановлення з'єднань;
– завантаження інформації про з'єднання в PROFINET

пристрою.

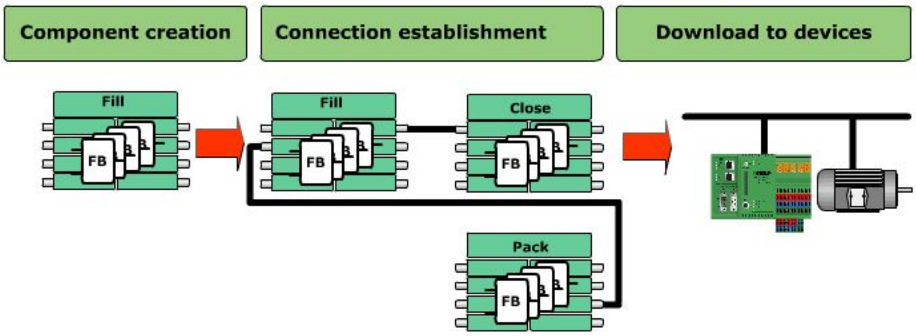


Рисунок 5.30 – Три етапи PROFINET CBA застосувань

Створення компонент. Створення PROFINET компонент, як образів технологічних модулів, може бути зроблено в три кроки:

1. Інструмент конкретного виробника може використовуватися для конфігурації і параметризації пристроїв. Виробник пристрою дозволяється використовувати старий інструмент програмного забезпечення, просто розширивши новий інтерфейсу компонента. Таким чином, як і раніше можуть бути використані існуючі знання програмування.
2. Буде визначено інтерфейс компонента. Інтерфейс складається з декількох змінних, таких як: ім'я, тип, напрям PROFINET і коментар.
3. PROFINET компонент буде створюватися з енкапсуліруемимі змінними. Властивості компонента включають в себе назву компонента, номер версії та зберігання місце розташування.

Після цього, у вас буде PCD файл, який потім може бути імпортований в бібліотеку редактора з'єднань.

Завантаження в PROFINET пристрою. Інформація про підключення буде завантажена в відповідні пристрої PROFINET відповідно до плану підключення компонентів (рисунок 31). Це означає, що кожен компонент знає свого власного комунікаційного партнера, комунікаційні відносини і інформацію, що підлягає обміну.

Встановлення з'єднань зв'язку з партнером і обмін даними відбувається автоматично. Інформація про підключення завантажується в кожного споживача, який потім самостійно створює і контролює комунікаційні з'єднання з беруть участь партнерськими пристроями.

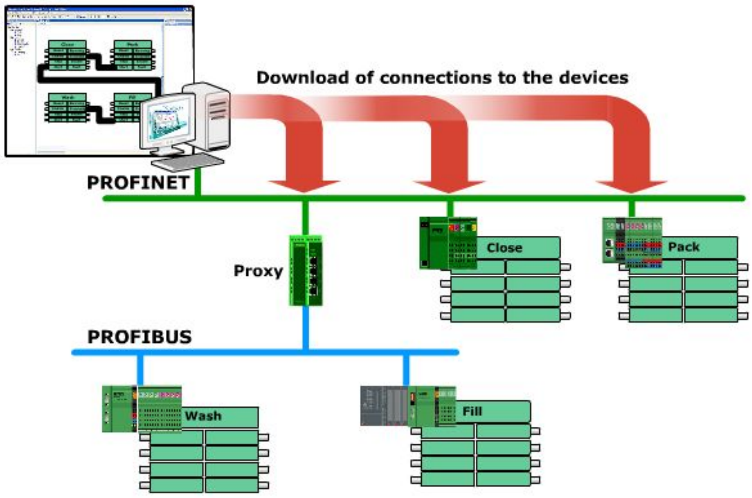


Рисунок 5.31 – Завантаження інформації про підключення

Комунікації PROFINET CBA засновані на TCP / IP АБО UDP / IP для зв'язку в реальному часі и RT для зв'язку в режимі реального часу.

5.6.3 PCD – Опис PROFINET компонент

Являє собою XML-файл, який описує функції і об'єкти компонента PROFINET CBA. Використовуючи XML, ви можете описати дані в форматі незалежним від платформи і постачальника.

Зазвичай створюють файл PCD, після створюється для користувача програмне забезпечення за допомогою інструменту для конкретного виробника (наприклад, STEP 7 від Siemens) для відповідного пристрою. Потім PCD зберігається в інженерній системі і може бути завантажено в технічні інструменти (наприклад, Siemens Imap), якщо ви хочете з'єднати їх. В якості альтернативи, ви можете отримати незалежний від постачальника PROFINET Component Editor від PI, який може бути використаний для створення файлу PCD.

PCD файл містить наступну інформацію:

1. Опис компоненту як елементу бібліотеки:

– ідентифікатор

– ім'я

2. Опис апаратної частини:

– IP адрес запам'ятовуючого пристрою

– Доступ до діагностики даних

– завантаження взаємозв'язку

3. Опис функціонування програмної частини:

– Установка зв'язку між програмної та апаратної частинами

– Інтерфейс компонента

– Властивості змінних (1..n)

Ім'я(e.g. Start)

Тип даних (e.g. Boolean)

Напрям (e.g. Input or Output)

4. Розташування запам'ятовуючого пристрою компоненту проекту

\endash наприклад STEP 7

5.6.4 Вправа: Інтеграція с PROFIBUS

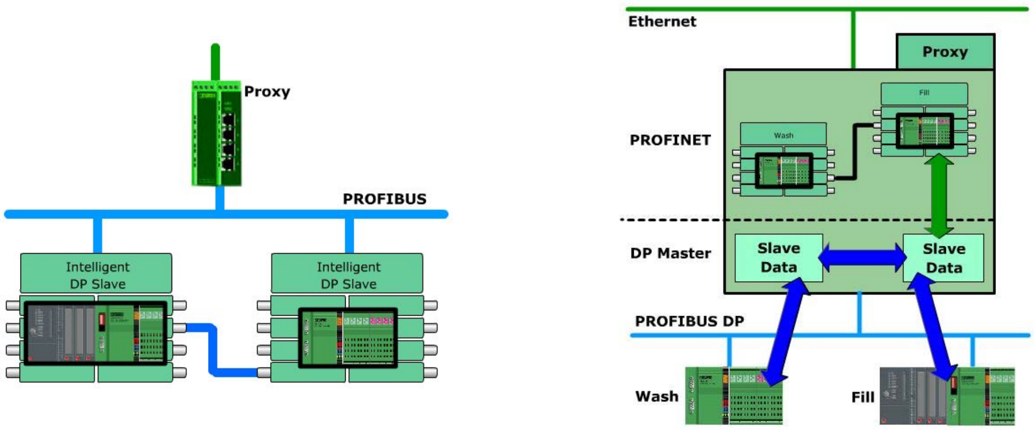
Якщо є існуюча PROFIBUS система, у вас є два способи інтегрувати її в PROFINET систему:

– Використовуючи права адміністратора: Ви можете підключити шину пристроїв через проксі.

– За допомогою інтегрування додатки шини: Ви можете підключиться все шинні додатки.

Підключення з використанням проксі. Концепція проксі в PROFINET CBA аналогічна концепції проксі в PROFINET IO (рис. 32). Проксі-сервер дозволяє здійснювати інтеграцію існуючих систем польових шин в систему PROFINET в простій і зрозумілій чином. Проксі-сервер підтримує зрозуміле перетворення повідомлень (без протоколу тунелювання) між мережами.

З одного боку в PROFIBUS, проксі-сервер є майстром PROFIBUS, координуючи обмін даних між вузлами PROFIBUS, а з іншого боку він є вузлом мережі Ethernet з PROFINET комунікацій (рис. 32). Проксі-сервери можуть бути реалізовані у вигляді ПЛК, контролери на базі ПК, або чисті шлюзи. У середовищі компонентів, інтелектуальні частини використовуються в якості незалежних компонентів PROFINET. У редакторі з'єднань PROFINET, такі компоненти PROFIBUS не відрізняються від компонентів Ethernet. Проксі-сервер дає змогу пристроям обмінюватися на різних шинних пристроях, можливими.



a) з'єднання PROFIBUS в системі PROFINET CBA через проксі-сервер

b) Компонентна структура проксі-сервера PROFINET в PROFIBUS

Рисунок 5.32

Підключення з використанням інтеграції Fieldbus додатків.

Ціле додаток польовий шини може бути змодельоване як PROFINET компоненти в рамках моделі компонентів (рис. 5.33). Це важливо, коли вже існуюча система повинна бути розширена за допомогою PROFINET. Це не важливо, яка польова шина використовується для автоматизації сегмента.

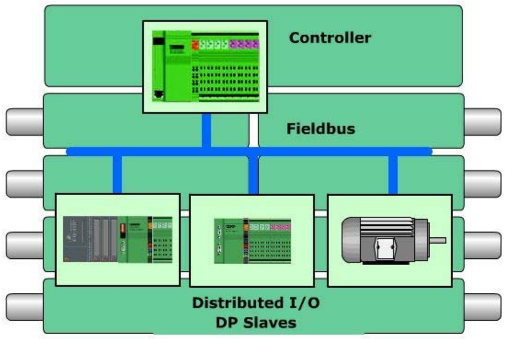


Рисунок 5.33 - Підключення з використанням інтеграції з Fieldbus застосувань

Для зв'язку між існуючою системою і PROFINET, майстер створення польової шини в PROFINET компоненті повинен бути PROFINET здатним. Таким чином, існуючі механізми польовий шини використовуються в компонентах, і в PROFINET механізмом поза компонентів.

Ця опція переміщення гарантує, що користувач (системний оператор або розробник) може захистити інвестиції, вкладені в систему і механізми. Крім того, існуючі нововведення призначених для користувача програм зберігається. Це дозволяє плавно перейти до нових сегментах системи з PROFINET.

5.7. Кабельні системи та кабельні роз'єми PROFINET

Встановлення через мережу. Компоненти мережі для використання в промисловій автоматизації повинні бути розроблені для підтримки екстремальних умов у виробничій зоні. Крайніми умовами є, наприклад, пил, електромагнітна сумісність, температура, вологість, вібрація і т. Д. Визначення і опису галузевого стандарту для Fast Ethernet можна знайти в «Керівництві по установці PROFINET» з PI, які засновані на IEC 11801. У таблиці 7 порівнюються деякі характеристики офісних і автоматичних областей.

Таблица 5.7 порівняння: Офіс и автоматизація

Офісна площа	Область автоматизації
Виправлена базова установка в будівлі	Великі системні кабелі
Викладений під фальшпол	Системна маршрутизація кабелів
Підключення змінного пристрою на робочому місці	Точки підключення рідко змінюються
Кабель збірного пристрій	Зєднальне польове з'єднання
Топологія мережі дерева	Часто виходите: топологія ліній і кільцевих мереж
Великі пакети даних (наприклад, зображення)	Малі пакети даних (наприклад, дані процесу)
Середня доступність мережі	Висока доступність мережі
Помірна температура (от 0 до +50 ° С)	Крайня температура (от -20 до +70 ° С)
Відсутність вологи	Можлива волога
Мінімальні коливання	Вібраційні машини
Низька ЭМС	Висока ЭМС
Низька механічна небезпека	Висока механічна небезпека
Низька хімічна небезпека	Висока хімічна небезпека, наприклад. Масла або агресивних середовищ

В основному, розрізняють установку мережі в промисловій зоні між внутрішньою і зовнішньою панеллю керування. У таблиці 8 показані відповідні умови навколишнього середовища.

Таблиця 5.8 - Зовнішні умови всередині і зовні шафи управління

Всередині шафи управління	Всередині шафи управління Поза шафи управління	Всередині шафи управління Поза шафи управління
Клас захисту IP20 IP65 и IP67	Клас захисту IP20 IP65 и IP67	Клас захисту IP20 IP65 и IP67
Вторгнення 12,5 мм	Вторгнення 12,5 мм	Вторгнення 12,5 мм 50 мкм

50 мкм	50 мкм	
Рідини немає Водоструминне нарізання (12,5 *л* / мин с 2,5 м)	Рідини немає Водоструминне нарізання (12,5 *л* / мин с 2,5 м)	Рідини немає Водоструминне нарізання (12,5 *л* / мин с 2,5 м)
Водонепроникність (1 м в протягом 30 мінут)	Водонепроникність (1 м в протягом 30 мінут)	Водонепроникність (1 м в протягом 30 мінут)

Кабелі. Як і інші мережеві компоненти, описані вище, кабелі промислового стандарту повинні витримувати екстремальні механічні навантаження у виробничій зоні.

В принципі, у вас є 2 типу кабелю, кабелі з міддю і кабелі з оптоволоконним кабелем. Крім того, у вас є так званий гібридний кабель, який передає не тільки дані процесу, а й потужність. Крім того, можна передавати дані процесу по бездротовій мережі.

Кабелі кручений пари (з міді). В даний час використовується так званий симетричний кабель. Це також називається симетричним кабелем, тому що різниця потенціалів щодо опорного заземлення становить 0 В в ідеальному випадку. Щоб передати сигнал, два дроти скручені разом як пара. Для повного дуплексу вам потрібно чотири дроти. Тому кабель промислової мережі повинен мати не менше 4 проводів (чотирьохжильний кабель). На рисунку 34 показаний кабель витої пари, який використовує мідь.

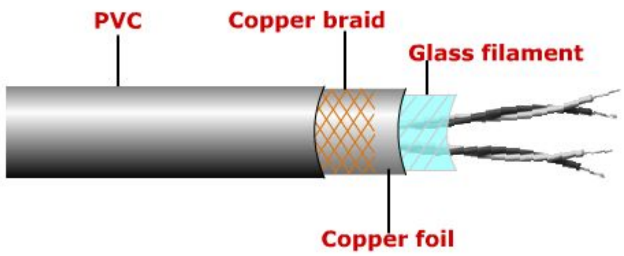


Рисунок 5.34 – Кабель «витая пара»

На практиці використовується кабель 100BASE-TX для Fast

Ethernet з 4-ма проводами. Кабель визначається як STP (кручена пара), що означає, що два екранованих мідних дроти скручені в пару. Кабелі відповідають вимогам CAT5 з кабелем класу D IEC 11801.

CAT5 призначений для передачі даних до 100 МГц, ослаблення 24 дБ і NEXT (перехресні перешкоди ближнього кінця) 27 дБ. Клас кабелю D означає, що смуга пропускання становить до 100 МГц. Кожен провід має поперечний переріз AWG 22 (американський калібр проводу), що означає, що діаметр не перевищує 0,64 міліметра. Ця специфікація мінімізує загасання і забезпечує складну структуру проводів. Максимальна довжина сегмента кабелю становить 100 метрів. Як роз'єму використовується вилка RJ45 або M12.

2. Волоконно-оптичні кабелі. Оптичний кабель містить сердечник і екран. Сердечник є високо стерильну, надзвичайно тонку кварцову або кварцову скляну різьблення, екрановану склом з низькою оптичною щільністю.

Світлові імпульси повністю відображаються і направляються уздовж ядра. Щоб забезпечити можливість передачі даних, сигнали даних перетворюються в світлові імпульси з використанням світлодіодів або лазерних діодів. Малюнок 35 ілюструє волоконно-оптичний кабель.

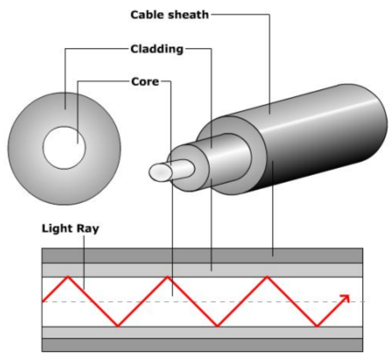


Рисунок 5.35 - Волоконно-оптичний кабель

Існує два типи волоконно-оптичних кабелів, одномодовий і багатомодовий. PROFINET може працювати з обома з них. Зверніть увагу, що для правильної роботи оптичне волокно і оптичний інтерфейс повинні мати однаковий режим.

Для Fast Ethernet 100BASE-FX використовуються два оптоволоконних кабелі. Оптичні інтерфейси відповідають специфікаціям ISO / IEC 9314-3 (багатомодовий) і ISO / IEC 9314-4 (одномодовий режим). Для багатомодових максимальна довжина сегмента складає 2 кілометри, а для одиночного режиму - 14 кілометрів. Для установки за межами шафи управління властивості екрану, який оточує ядро, повинні відповідати вимогам, наприклад, механічних, хімічних, термічних і т. д. для місця установки.

Є багато переваг оптичних кабелів в порівнянні з мідними кабелями:

– Оптичні кабелі нечутливі до електромагнітних впливів;
– малу вагу кабелю на метр;
– Відсутність електромагнітного випромінювання;
– Може використовуватися для максимального відстані;
– Стійкість до постукування;
– Велика пропускна здатність.

Недоліки в порівнянні з кабелями на основі міді - більш висока ціна, а збірка з'єднання складніша.

3. Гібридні кабелі. У виробничій зоні деякі пристрої повинні бути підключені до джерела живлення. Тому дуже корисною є гібридна кабельна структура (рис. 5.36). Напруга становить 24 В. Гібридні кабелі містять дроти (оптичні або мідні) для передачі даних і передачі енергії. Для оптичних кабелів вам необхідні 2 оптичних дроти для передачі даних і 4 мідних дроти для передачі енергії. Для кабелів на основі міді необхідно 4 мідних дроти для передачі даних і 4 мідних дроти для передачі енергії.

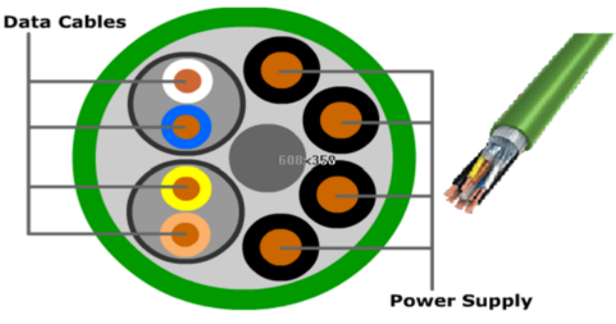


Рисунок 5.36 – Гібридний кабель

Бездротова система. Якщо подивитися на системи з'єднання, використовувані в даний час в промисловій автоматизації, ви побачите, що бездротова система стає все більш поширеною. Найбільші переваги - це економія на вартості кабельних систем і витрат на установку. Завдяки своїй мобільності та гнучкості, ми можемо встановити пристрої, де лінії електропередачі можуть використовуватися тільки з обмеженнями.

Комунікації PROFINET також можливі в цих мережах бездротового зв'язку. Ви можете використовувати польові пристрої PROFINET з різними радіотехнологіями для різних областей застосування, з певними параметрами щодо швидкостей передачі, діапазону, кількості вузлів і т.д. Таким чином, профілі вказуються для кожної технології. Вони визначають, як інтегруватися в PROFINET, які топології і значення продуктивності можуть бути досягнуті, а також умови для вимог безпеки.

PROFINET підтримує стандарт WLAN (бездротова локальна мережа) IEEE 802.11b / g, стандарт Bluetooth IEEE 802.15.1 і стандарт ZigBee IEEE 802.15.4. Вони використовують частотний діапазон 2,4 ГГц в частотному діапазоні ISM (Industrial, Scientific and Medical Band), який можна використовувати безкоштовно і без ліцензії.

РОЗДІЛ 6
БЕЗПРОВІДНІ ЛОКАЛЬНІ МЕРЕЖІ

Існує багато об'єктів автоматизації, де складно обійтися без безпровідних мереж або де їх застосування явно бажано:

- датчики і виконавчі пристрої на рухливих частинах конвеєрів, вітряних млинів, ліфтів, міксерів, візків для переміщення вантажів по цеху, на крилах і лопатях літаків, на підшипниках двигунів, на роботах, в пересувних лабораторіях, датчики на тілі людини і тварин; датчики вібрації на контейнерах для перевезення вантажів;
- об'єкти, в яких небажано свердлити стіни або псувати дизайн : офісні будівлі, в яких встановлюється пожежна і охоронна сигналізація, датчики для систем обігріву і кондиціонування повітря, для моніторингу механічної напруги в конструкціях будівель; у системах "Розумного будинку" (управління освітленням, кондиціонуванням-обігрівом, охоронними датчиками, побутовими приладами та ін.);
- епізодичне програмування і діагноста ПЛК, коли прокладати постійні кабелі не вигідно; дистанційне прочитування свідчень лічильників, самописців;
- об'єкти з агресивними середовищами, вібрацією; об'єкти, що знаходяться під високою напругою або в місцях, не зручних для прокладення кабелю;
- відстежування траєкторії руху транспорту, охорона кордонів держави, моніторинг напруженості автомобільного трафіку в містах і умов на дорогах, моніторинг лісу, моря, сільськогосподарських культур, моніторинг шкідливих викидів в екології;
- будь-які об'єкти, для яких відомо, що вартість кабелів, кабельних каналів, опор або траншей, а також робіт по монтажу і обслуговуванню істотно перевищує вартість замінюючої безпровідної системи, за умови відсутності жорстких вимог до надійності доставки повідомлень вреальному часі;
- об'єкти у вибухонебезпечних зонах.

6.1 Особливості і проблеми безпровідних мереж

У більшості застосувань безпровідні мережі дозволяють досягти наступних переваг в порівнянні з дротяними мережами:

- істотно понизити вартість установки датчиків;
- виключити необхідність профілактичного обслуговування кабелів;
- виключити дорогі місця розгалужень кабелю;
- зменшити кількість кабелів;
- зменшити трудовитрати і час на монтаж і обслуговування системи;
- понизити вартість системи за рахунок виключення кабелів;
- понизити вимоги до навчання персоналу монтажної організації;
- прискорити відладку системи і пошук несправностей;
- забезпечити зручну модернізацію системи.

Оскільки реконфігурація системи і її монтаж стають набагато простішими, безпровідні мережі можна використати і в традиційних сферах застосування кабельних зв'язків, коли вартість кабелю і монтажу виявляється вище, ніж установка безпровідної системи.

Безпровідні мережі діляться на наступні класи:

- стільникові мережі WWAN (Wireless Wide Area Network);
- безпровідні LAN (WLAN - Wireless LAN);
- безпровідні мережі датчиків.

У промисловій автоматизації найбільше поширення отримали три типи безпровідних мереж : Bluetooth на основі стандарту IEEE 802.15.1, ZigBee на основі IEEE 802.15.4 [IEEE] і Wi - Fi на основі IEEE 802.11. Фізичні рівні моделі OSI для цих мереж засновані на відповідних стандартах IEEE, а протоколи верхніх рівнів розроблені і підтримуються організаціями Bluetooth, ZigBee і Wi - Fi відповідно. Тому в назві мереж зазвичай вказують посилання на стандарт. Усі три мережі використовують ISM (Industrial, Scientific, and Medical) діапазон, що не ліцензується, 2,4 ГГц.

TSL дозволяє вивчити ряд сучасних безпровідних технологій для побудови систем автоматизації. До таких технологій відноситься:

- промисловий Wi - Fi;
-Trusted Wireless.

З точки зору вимог до промислових мереж безпровідні мережі поступаються дротяним за наступними характеристиками:

- час доставки повідомлень: використовуваний механізм випадкового доступу до каналу CSMA/CA не гарантує доставку в заздалегідь відомий час і цю проблему не можна вирішити за допомогою комутаторів, як в дротяних мережах;
- завадозахищеність: безпровідні мережі схильні до впливу електромагнітних завад значно сильніше, ніж дротяні;
- надійність зв'язку: зв'язок може зникнути при несвоєчасній змінібатарей живлення, зміні розташування вузлів мережі або появі об'єктів, які викликають загасання, відображення, заломлення або розсіяння радіохвиль;
- обмежена дальність зв'язку без використання ретрансляторів (звичайний не більше 100 м усередині приміщень);
- різке падіння пропускної спроможності мережі при збільшенні кількості одночасно працюючих станцій і коефіцієнт використання каналу;
- безпека: можливість просочування інформації, незахищеність від штучно створених перешкод, можливість непомітного управління технологічним процесом ворожими особами.

Унікальною гідністю безпровідних мереж є відсутність кабелів, що і визначає вибір сфер їх застосування в системах промислової автоматизації.

Розглянемо фізичні причини виникнення перерахованих проблем і методи боротьби з ними. Головними причинами є інтерференція, дифракція, заломлення, відображення, розсіяння (перевипромінювання) і зниження щільності потужності випромінювання при збільшенні відстані від джерела, а також неможливість локалізації радіохвиль в обмеженому просторі.

Залежність щільності потужності від відстані

Відомо, що щільність потужності радіохвилі зменшується в міру віддалення від антени внаслідок розбіжності пучка, розсіювання і поглинання хвиль перешкодами на шляху їх поширення. Щільність потужності $P(d)$ хвилі на відстані d від джерела наближено описується залежністю $P(d) \approx P_t(d_o/d)^{\gamma}$ (1)

де d_o - деяка константа, яка визначається експериментально;

параметр $\gamma \approx 2 \ldots 6$ залежить від конструкції антени, діапазону частот, наявності перешкод на шляху поширення електромагнітної хвилі. В умовах промислового підприємства $\gamma \approx 2 \ldots 3$;

P_t - щільність потужності на відстані d_o від джерела.Формула (1) справедлива при $d > d_o$.

Кожен учасник бездротової мережі має обмежену зону впевненого прийому, яка представляється приблизно в формі сфери. Це призводить до необхідності планування розташування бездротових станцій таким чином, щоб зони впевненого прийому безпосередньо зв'язуються станцій перекривалися. Якщо станції розташовані на відстані невпевненого прийому, то невеликі зміни навколишнього оточення можуть призвести до втрати повідомлень або зниження швидкості передачі.

Обмеженість радіусу дії передавачів призвела до виникнення пористих мереж, в яких інформація передається не через загальний канал зв'язку, як в провідних мережах, а від вузла до вузла, використовуючи проміжні вузли мережі в якості ретрансляторів і маршрутизаторів. Привиході з ладу або видаленні з мережі деяких вузлів мережі пристроєм знаходить новий маршрут, щоб доставити дані адресату. Додавання до мережі нового пристрою також може відбуватися автоматично, тобто ніздрюваті мережі мають властивість самоорганізації.

Вплив інтерференції хвиль

Електромагнітна хвиля передавальної станції на шляху прямування відчуває інтерференцію, дифракцію, відображення, заломлення і розсіювання. Тому в точці прийому хвиля є суперпозицією безлічі хвиль, що мають різні фази і напрямки хвильового вектора. Накладення хвиль призводить до інтерференції, яка може бути конструктивною (коли сигнал в точці прийому посилюється) або деструктивною (якщо сигнал послаблюється - ефект "завмирання").

Деструктивна інтерференція призводить до виникнення кількох негативних наслідків. По-перше, сигнал в точці прийому може виявитися нижче порога чутливості приймача, що призведе до втрати зв'язку. По-друге, при рухомому джерелі або приймачі в точці прийому можуть бути багаторазові зміни сильного і слабкого сигналу, що може привести до втрати декількох біт інформації абозменшення швидкості передачі за рахунок повторних передач кадрів з помилкою. По-третє, якщо різниця часу затримки хвиль, що пройшли різними шляхами, перевищить тривалість символу, сусідні символи в повідомленні можуть накладатися один на одного, викликаючи ефект межсимвольной інтерференції.

Джерела перешкод

Існують також інші причини спотворень переданого сигналу: паразитное взаємовплив сусідніх каналів; ефект Доплера, перешкоди від працюючих двигунів, розряди статичної електрики, і ін. Це може привести до втрати пакету, повторної передачі і, як наслідок, непередбачена затримка в каналі. Інтенсивність потоку помилок залежить від потужності джерел перешкод, типу модуляції і потужності передавача, від частотного діапазону, інших причин і зазвичай змінюється з плином часу.

Вимірювання, виконані в роботі показали, що чіпсет, що відповідає стандарту IEEE 803.11b, в індустріальному оточенні дає потік короткочасних помилок $10^{-4}...10^{-2}$ величиною при швидкості передачі 2 Мбіт / с і використанні квадратурної фазової модуляції QPSK (Quaternary Phase Shift Keying). Крім того, в процесі вимірювань епізодично виникали періоди тривалістю до 1 хв. коли втрати даних доходили до 10% і навіть 80%. Аналогічні результати спостерігалися і в інших експериментах.

Наслідком перешкод в каналі може бути не тільки втрата даних або уповільнення швидкості передачі, але і "проблема просторової несуперечності". Вона полягає в наступному. Коли система використовує широкомовний режим передачі без повідомлення про отримання, передбачається, що всі приймачі повинні отримати одні й ті ж дані одночасно. Однак внаслідок помилок в каналі деякі споживачі можуть отримати помилкові дані. Така помилка особливо небажана, якщо широкомовний режим використовується для забезпечення синхронної роботи декількох контролерів в одному і тому ж технологічному процесі, оскільки вона приведений до рассинхронизации процесу.

Особливістю розглянутого випадку є те, що ймовірність помилки в системі різко зростає в порівнянні з імовірністю помилки в одному каналі p

$p = 0.1$. Оскільки ймовірність безпомилкової передачі в системі є твором ймовірностей безпомилкової передачі в кожному з каналів, то при кількостіоднакових k каналів ймовірність відсутності збоїв в системі буде дорівнює $(1-p)^k$. Наприклад, в системі з 8 каналів при ймовірності помилки в каналі ймовірність безпомилкової передачі складе всього 43%.

Одним із прикладів, де описана ситуація може грати роль, є режим одночасного введення декількома пристроями сигналів датчиків. У провідних мережах для цього використовують широкомовні команди, які доходять до всіх пристроїв одночасно (в мережах Modbus це команда з адресою "0"). Якщо аналогічний режим використовувати в бездротовій мережі, то ймовірність того, що всі k датчиків введуть відліки одночасно, буде також дорівнює $(1-p)^k$.

У мережах з передачею маркера перешкоди можуть призвести до втрати маркера і відключення пристроїв з втраченим маркером на кілька періодів звернення маркера по логічному кільцю.

Широкосмугова передача

Одним з методів усунення впливу інтерференції хвиль і вузькосмугових перешкод є застосування широкосмугової модуляції. У бездротових мережах використовуються два методу: широкої смуги з прямим розширенням спектра (DSSS - Direct Sequence Spread Spectrum) і з перескоком з однієї несучої на іншу (FHSS - Frequency Hopping Spread Spectrum).

Метод DSSS полягає в наступному. Якщо один біт інформації уявити прямокутним імпульсом, то ефективна ширина спектру імпульсу буде обернено пропорційна його тривалості. У методі DSSS один прямокутний імпульс замінюють послідовністю з 11 імпульсів, які в 11 разів коротшевихідного. При цьому ефективна ширина спектру такій послідовності імпульсів виявляється в 11 разів ширше, ніж у вихідного одиночного імпульсу (біта) і для Wi-Fi мереж становить 22 МГц. Оскільки енергія сигналу виявляється "розмазаний" по всьому спектру, то спектральна щільність потужності сигналу виявляється в 11 разів менше, якщо її вимірювати в тій же смузі частот, яку займав початковий прямокутнийімпульс. Практично потужність передавача (близько 1 мВт) для діапазону2,4 ГГц вибирають таким чином, щоб спектральна щільність корисного сигналу була порівнянна або навіть менше спектральної щільності шуму.

Для ще більшого зменшення спектральної щільності потужності сигналу його спектральна характеристика повинна бути близька до прямої лінії, паралельної осі абсцис, тобто сигнал повинен бути подібний до білого шуму. Для цього послідовність коротких імпульсів не повинна бути періодичною, вона повинна бути шумоподобного (псевдослучайной), з малим часом автокорреляции. Процес перетворення спектра сигналу дозазначеного виду називають процесом "обілення" ("відбілювання") спектра. Крім того, для полегшення виявлення сигналу в приймачі псевдослучайная послідовність, обрана для кодування, повинна бути такою, щоб її автокореляційна функція мала тільки один яскраво виражений максимум. Такому вимозі задовольняють, зокрема, послідовності Баркера. Послідовність (код) Баркера завдовжки 11 імпульсів для кодування логічного одиниці використовується в мережах Wi-Fi і має вигляд 11100010010. Логічний нуль кодується инверсной послідовністю Баркера.

Для виділення корисного сигналу з такою малою потужністю на тлі шуму в приймачі повинна зберігатися копія переданого сигналу (код Баркера). Це дозволяє використовувати дуже ефективні методи оптимальної фільтрації. Знаючи, що корисний сигнал являє собою послідовність Баркера, в приймальнику будують оптимальний фільтр з імпульсною характеристикою, яка представляє собою масштабну копію вхідного сигналу, розташовану дзеркально по осі часу щодо вхідного сигналу і зрушену в сторону запізнювання на величину не менше тривалостівиділяється імпульсу.

Ширина спектра сигналу в методі DSSS при швидкості передачі 1 Мбіт/с становить 22 МГц, а ширина виділеного для Wi-Fi частотного діапазону - 83,5 МГц, тобто у всьому діапазоні можна розмістити тільки 3 неперекриваающіхся каналу. Однак стандарт ділить весь діапазон на 11 перекриваються каналів, з яких тільки три (1-й, 6-й і 11-й) можуть працювати, не впливаючи один на одного.

Перевагами методу DSSS є:

- висока стійкість до вузькосмуговим перешкод;
- можливість відновлення інформації при втраті під час передачі декількох біт в коді Баркера.

Другим методом широкосмугової модуляції є FHSS - метод стрибкоподібної зміни несучої частоти. Він використовує той же діапазон 2,4 ГГц шириною 83,5 МГц, в якому виділяється 79 неперекриваающіхся частотних смуг по 1 МГц кожна. В процесі передачі частота несучої змінюється стрибкоподібно. Частота переходів з однієї несучої на іншу для мережі Wi-Fi повинна бути не менше 4 Гц в Wi- Fi і 1,6 КГц в мережі Bluetooth. Для прийому такого сигналу приймач і передавач містять таблиці, в яких занесена одна і та ж послідовність зміни каналів. При такому способі передачі вузькосмугові завади призводять до втрати тільки тих фрагментів повідомлень, які передаються на частоті перешкоди, тобто фактично тільки до збільшення часу передачі за рахунок повторної передачі зіпсованих фрагментів.

Модифікацією FHSS є адаптивний метод FHSS (Adaptive Frequency Hopping - AFH), в якому під час передачі виявляються і запам'ятовуються частоти, на яких передача виконувалася з помилками контрольної суми. Ці частоти виключаються з таблиці використовуваних частот.

Перехід з однієї частоти на іншу зменшує ймовірність взаємного впливу при спільній роботі декількох передавачів в мережі, оскільки при 79 частотах ймовірність збігу частот двох працюючих станцій дуже низька (приблизно $(1/79)^2 = 1{,}6 \cdot 10^{-4}$). Тому метод FHSS дозволяє використовувати більшу кількість одночасно працюючих станцій в мережі. Практично наодній і тій же території можуть працювати до 15 передавачів.

FHSS забезпечує швидкість передачі 1 і 2 Мбіт / с. Використовується частотна модуляція з двома дискретними значеннями частот і f_1, f_2 які дозволяють зробити чотири комбінації модульованих сигналів: $f_0 + f_1, f_0 - f_1, f_0 + f_2, f_0 - f_2$ і закодувати таким чином 4 біти інформації.

На практиці системи з FHSS здатні працювати при більш високому рівні шуму, ніж DSSS завдяки тому, що вони займають більш широку смугу частот (83,5 МГц у порівнянні з 22 МГц для DSSS), а ймовірність того, що перешкода буде займати смугу 83, 5 МГц нижче, ніж смугу 22 МГц. Однак інтерференція, яка веде до завмирання сигналу, більш позначається на FHSS, оскільки в DSSS завмирання відбуваються тільки у вузькій смузі частот, що призводить до випадання кількох біт з 11 переданих, а що залишилися біт досить для безпомилкового розпізнавання закодованого значення "0" або "1" .

Розширення спектра мають такі переваги:

– висока стійкість завдяки великій надмірності коду і можливості застосування оптимальної фільтрації;

– можливість уникнути впливу інтерференції, оскільки вона відбувається тільки в частині широкосмугового діапазону. У методі DSSS вона призводить тільки до втрати декількох бітів, які можна відновити, а в методі FHSS - до втрати окремих фрагментів повідомлень, які відновлюються шляхом повторної передачі або губляться тільки один раз (в методі AFH), до того, як система виключить цю частоту зі списку використовуваних;

– широкосмуговий сигнал складніше перехопити, ніж вузькосмуговий. FHSS виглядає як шум, якщо в приймачі не використана таж черговість зміни частот, що і в передавачі;

– широкосмуговий передавач може використовувати один і той же діапазон частот спільно з іншими типами передавачів з мінімальним взаємовпливом. Зокрема, він практично не вносить перешкод в вузькосмугові системи завдяки дуже малої потужності;

– робота при спектральної щільності сигналу на рівні і нижче рівня шуму дозволяє виключити необхідність отримання ліцензії на використання таких передавачів.

Методи модуляції несучої

Ідея модуляції полягає в тому, щоб перенести спектр інформаційного сигналу в область високих частот, в нашому випадку в діапазон 2,4 ГГц, що дозволить передати його за допомогою електромагнітної хвилі. Електромагнітні хвилі збуджуються в антені струмом синусоїдальної форми

$i = A\sin(\omega t + \varphi)$, який називається несучим коливанням або просто несучою.

Принаймні один з параметрів A, φ, ω несучої може залежати від часу: $A = A(t)$, $\omega = \omega(t)$, $\varphi = \varphi(t)$. Форма цієї залежності відповідає формі сигналу, який потрібно передати за допомогою радіоканалу. Процес управління параметрами несучої називається модуляцією. Окремим випадком модуляції є маніпуляція, коли модульовані параметри змінюються стрибкоподібно між двома їхніми значеннями. Залежно від того, який параметр стає залежним від часу, модуляція називається амплітудною, фазовою або частотної. Можливі також комбіновані способи модуляції: амплітудно-фазова, фазо-частотна і т. П.

φ Кількість інформації, яке може бути внесено в сигнал, можна збільшити, використовуючи кілька одночасно змінюваних параметрів. У цифрових системах передачі модульовані параметри змінюються дискретно. Тому кількість інформації, що припадає на Бодов інтервал, можна збільшити, збільшуючи кількість дискретних рівнів. Бодов інтервалом називають часовий інтервал, протягом якого параметри A, ω, залишаються постійними.

Оскільки $\sin(\omega t + \varphi) = \cos\varphi \sin\omega t + \sin\varphi \cos\omega t$, т. е. зміна фази можна уявити за допомогою змін амплітуди синусоїдальної і косинусоидальной компоненти,то параметри вихідного синусоїдального коливання можна уявити на площині за допомогою графіка (рис. 6.1), у якого по осі абсцис відкладена амплітуда синусоїдальної компоненти (т. е. величина $\cos\varphi$, її називають синфазной компонентою і позначають на графіку буквою I, від слова In- phase), по осі ординат - амплітуда косинусоидальной компоненти, т. е. $\sin\varphi$, яку називають квадратурної і позначають буквою Q, від слова Quadrature. Отриманий таким способом графік називається сигнальним сузір'ям (рис. 6.1). Він збігається з графіком, що зображує синусоїдальне коливання на комплексній площині.

При амплітудної модуляції фаза не змінюється, тому всі точки графіка розташовуються на осі абсцис. При фазової модуляції амплітуда постійна, тому всі точки графіка лежать на окружності, радіус якої дорівнюєамплітуді коливання.

При двійковій фазової модуляції (BPSK - "Binary Phase Shift Keying") фаза приймає тільки два дискретних значення: 0 і π, тому сигнальне сузір'я складається з двох точок, розташованих на осі абсцис. Цей різновид фазової маніпуляції є найбільш перешкодостійкою.

Модифікацією цього методу є диференціальна двоичная фазова маніпуляція (DBPSK - Differential BPSK), коли логічним значенням "0" або "1" відповідають чи не абсолютні значення фази, а зміна фази щодо попереднього її значення. Наприклад, якщо фаза сигналу дорівнювала 0, то для кодування значення "1" її змінюють на 180°, а для кодування логічного "0" фазу залишають незмінною. Аналогічна ідея використовується в методі NRZI-кодування, коли логічній одиниці відповідає зміна рівня сигналу, а логічному нулю - її відсутність.

Якщо $\cos\varphi$ приймає значення 0 або 1 і при цьому $\sin\varphi$ приймає значення 1 і 0, то така модуляція називається квадратурної фазової маніпуляцією QPSK (Quadrature Phase-Shift Keying) і дозволяє отримати 4 стану переданого сигналу в межах Бодов інтервалу. Сигнальне сузір'я QPSK показано на рис.6.1.

Модифікацією QPSK є DQPSK-модуляція (Differential QPSK), при якій, аналогічно DBPSK, кодується не величиною фази, а її зміна відносно попереднього значення. Зміна фази на 0° кодується як "00", зміна на 90° кодується як "01", на 180° - як "11", на 360° як "10".

Перешкодостійкість методу модуляції можна оцінити по відстані між точками сигнального сузір'я - це відстань характеризує амплітуду і фазу перешкоди, яка достатня, щоб був прийнятий помилковий сигнал. Тому при проектуванні схем модуляції точки сигнального сузір'я вибирають таким чином, щоб відстань від будь-якої точки до її сусідів було однаковим для всіх точок сузір'я. При цьому досягається однакова стійкість для будь- яких переданих чисел.

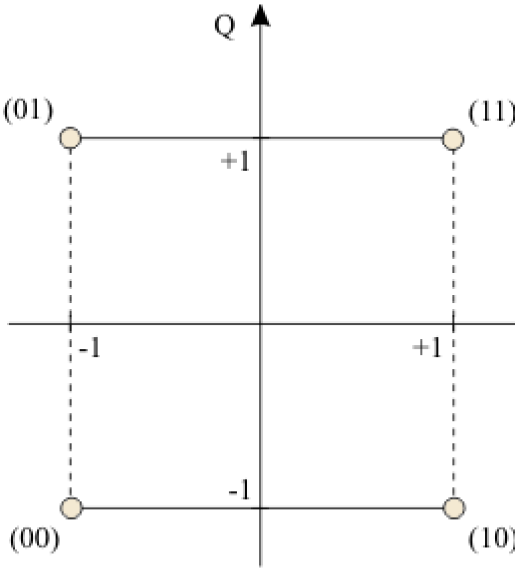


Рисунок 6.1 – Сигнальне сузір'я

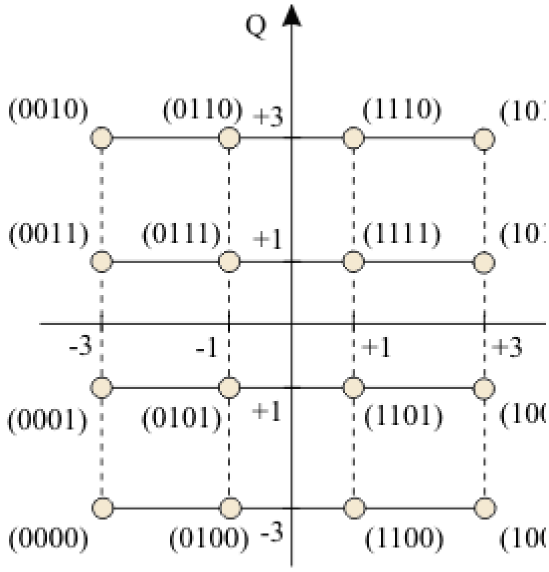


Рисунок 6.2 – Сигнальне сузір'я

Бездротові мережі використовують також амплітудно-фазову модуляцію 16-QAM (рис. 6.2) і 64-QAM, коли змінюється не тільки фаза, а й амплітуда коливання. Сигнал може приймати відповідно 16 і 64 біта інформації на Бодов інтервал, що збільшує швидкість передачі, але за рахунок зниження завадостійкості.

Інші особливості бездротових каналів

Ряд особливостей бездротової передачі даних не дозволяє використовувати багато методів, характерні для провідних промислових мереж.

Бездротові трансивери не можуть передавати і приймати сигнал на одному і тому ж каналі. Це пов'язано з швидким зменшенням щільності потужності випромінювання від відстані. Сигнал власного передавачавиявляється на порядки сильніше сигналу і заглушає його. На відміну від цього, в дротових каналах обидва сигналу мають приблизно однакову потужність. Тому бездротові трансивери в принципі не можуть прослуховувати лінію під час передачі, як це робиться, наприклад, в CAN або Ethernet. Описане властивість робить неможливим застосування методівдоступу до мережі, заснованих на виявленні колізій.

Виявлення несучої чужій станції навіть при непрацюючому власному передавачі також виявляється проблематичним, см. рис. 6.3. Тут три кола показують межі прийому сигналів трьома станціями А, В і С. Припустимо, що станція А передає повідомлення станції В. У цей час станція З прослуховує ефір і не чує несучу, оскільки знаходиться поза зоною дії станції А. Виявивши відсутність несучої, станція З починає передачу одночасно зі станцією А, що призводить до втрати інформації, оскільки станція В може приймати тільки один сигнал ("проблема прихованого вузла").

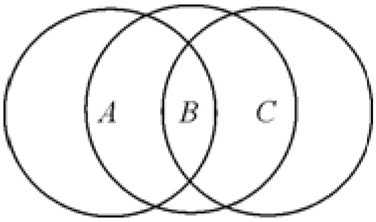


Рисунок 6.3 – Ілюстрація"проблеми прихованого вузла"

Для вирішення цієї проблеми можна використовувати сигнал "зайнято", що подається станцією В. Однак найбільш загальне рішення проблеми запропоновано в стандарті IEEE 802.11. Воно полягає в тому, що станція А починає сеанс зв'язку з обміну пакетами запиту на передачу RTS (Request To Send). Станція В може відповісти пакетом CTS ("Clear To Send" - "вільно"). Тільки при отриманні цього повідомлення станція А починає передачу пакета даних. Будь-яка інша станція, що отримала пакет RTS або CTS, призначений не їй, буде залишатися в стані очікування. Недоліком цього методу є те, що сигнали RTS/CTS істотно погіршують швидкість обміну між пристроями, оскільки розміри їх пакетів можна порівняти з розмірами корисних даних.

Вищевикладене показує, що бездротові канали не можуть використовувати метод доступу до каналу типу CSMA/CD. Для них застосовується метод CSMA/CA - ("Carrier Sense Multiple Access with Collision Avoidance" - "множинний доступ з контролем несучої і предотвращеніемколлізій"). Від CSMA/CD він відрізняється тим, що колізії в ньому не виявляються, в той час як в CSMA/CD колізії виявляються і приймаються заходи для їх вирішення. Оскільки в CSMA/CA колізії не можуть побут виявлені, т. К. Приймач завжди не чути сигналом свого передавача, то приймаються спеціальні заходи для зниження ймовірності виникнення колізій. Зокрема, використовують сигнали резервування каналузв'язку, завдяки чому колізії виникають між короткими сигналами резервування, а не між довгими пакетами даних. Запобігання колізій виконується завдяки тому, що станція, яка збирається почати передачу, інформує всіх учасників мережі про це, резервуючи для себе певний час. І тільки після того, як всі станції взяли цей сигнал, вона починає передавати. Використовують також випадкову затримку після звільнення каналу (в методі CSMA/CD передача починається відразу після звільнення каналу), щоб зменшити ймовірність колізії, оскільки дуже ймовірно, що багато станції чекають звільнення каналу і можуть почати передачу відразу і одночасно, як тільки він звільниться.

Наступною проблемою є електроживлення бездротових мереж, оскільки бездротові пристрої (в основному, датчики) не повинні мати проводів (підключатися куди-небудь за допомогою проводів). Тому дуже актуальна проблема економії енергії батарей, пошук простих способів їх заміни, виключення відмов через розряду, пошук альтернативних джерел енергії. У літературі розглядаються варіанти передачі енергії харчування електромагнітними хвилями [Hirai], трансформаторами з великим повітряним зазором (на відстань до декількох метрів) [O'Brien], витяг енергії згоряння палива, застосування сонячних батарей.

Методи зменшення кількості помилок в каналі

Вищевикладене показує, що помилки в радіоканалі з'являються частіше, ніж в екранованому кабелі, і зазвичай носять характер збоїв, в той час як в провідних системах, навпаки, помилки частіше бувають катастрофічними, пов'язаними з виходом з ладу (відмовою) канального устаткування. Проте, після прийняття всіх викладених заходів потік помилок в радіоканалі може бути знижений до необхідного рівня. Наприклад, в авіації ймовірність помилок в бездротових каналах становить менше. Однак такі значення ймовірності досягаються дуже великими зусиллями. Тому найбільш перспективною сферою застосування бездротових мереж є системи, в яких допускається деякий відсоток помилок. Імовірність помилки може бути використана як компонента цільової функції при проектуванні бездротової системи.

У промислових мережах часто використовується режим широкомовної передачі, коли повідомлення одночасно повинні прийняти всі учасники мережі. Його особливістю є відсутність підтвердження про отримання повідомлення. З огляду на низьку ймовірність безпомилкової передачі по радіоканалу для реалізації широкомовної передачі необхідно вжити заходів для збільшення ймовірності доставки повідомлень в бездротовому каналі. Одним з можливих методів є кодування широкомовного повідомлення з великою надмірністю, при якій приймач може відновити загублені під час передачі біти. Незважаючи на зниження пропускної здатності каналу, такий метод може бути дуже ефективний.

Для збільшення вірогідності передачі використовують метод ARQ (Automatic Repeat reQuest - "автоматичний повтор у відповідь на запит"). Метод ARQ може використовувати, наприклад, такі принципи:

- передача додатково до повідомлення коригуючого коду з великою надмірністю;
- відправлення одночасно кількох однакових пакетів; приймач робить повторний запит тільки якщо жоден з пакетів не був прийнятий без помилок;
- використання кількох антен для повторної передачі повідомлень.

Для збільшення вірогідності передачі використовують також чергування. Методи надлишкового кодування і корекції помилок зазвичай засновані на припущенні про випадковий характер впливів, що призводять до появи помилок. Однак на практиці помилки можуть бути корельовані. Це може бути, наприклад, в разі, коли період основної гармоніки перешкоди дорівнює тривалості передачі декількох бітів. Щоб зробити помилки більш схожими на некорельовані, використовують процедуру чергування - перестановку бітів за певним законом, одному і тому ж в передавачі для виконання чергування і в приймачі, для виконання відновлення початкового порядку проходження бітів. Одним з методів чергування є запис переданого фрейму в клітинки матриці, наприклад, по три біта в рядку, а потім зчитування бітів з матриці не по рядках, а по стовпцях.

Передача повідомлень без підтвердження про отримання

Існують також інші методи збільшення вірогідності передачі широкомовних повідомлень без зворотного зв'язку від одержувача: методи модуляції, стійкі до інтерференції радіохвиль (OFDM - "Orthogonal Frequency-Division Multiplexing") - модуляція з застосуванням декількох несучих частот, яка використовує велику кількість близько розташованих ортогональних піднесучих; передача одного і того ж пакету кілька разівпоспіль; оптимізація просторового розміщення станцій та застосування додаткової інфраструктури (ретрансляторів і вузлів доступу).

Системи зв'язку зі зворотним зв'язком отримують від приймаючої станції повторний запит в разі, якщо повідомлення було прийнято з помилками. Такий спосіб використовується, коли пред'являються високі вимоги до достовірності передачі, наприклад, при передачі сигналів про аварію. Однак кількість повторних запитів має природний межа, який визначається граничним часом, після закінчення якого передається інформація застаріває і тому стає марною.

Використовують також гібридний ARQ-метод HARQ (Hybrid Automatic Repeat- reQuest), в якому поєднуються повторна передача, тайм- аути і надлишкові коригувальні коди. Якщо приймач передавальної станції не отримав підтвердження від приймаючої станції, то після закінчення часу очікування виконується автоматична повторна передача. Додатково використовується надлишкове кодування, яке дозволяє відновити втрачені при передачі біти.

Приймач може також використовувати кілька прийнятих помилкових пакетів для того, щоб шляхом голосування вибрати з них біти, які мають найбільшу ймовірністю того, що вони правильні.

Оскільки рівень перешкод в бездротовому каналі набагато вище, ніж вдротовому, більшість систем використовують на початку фрейму преамбулу збільшеною тривалістю в порівнянні з дротяними системами, що збільшує частку "накладних витрат". Наприклад, фізичний рівень стандарту IEEE 802.11 (Wi-Fi) з режимом DSSS використовує преамбулу довжиною 128 мкс, яка передається в кожному пакеті і займає значну його частину.

Використання просторового рознесення антен

Внаслідок замираний радіохвиль напруженість поля в точці прийому буде різною для різних положень прийомної антени. Якщо два приймача, r и s , розташовані на однаковій відстані від передавача, знаходяться близько один до одного, то ймовірність того, що вони обидва знаходяться в зоні завмирання, вище, ніж коли вони рознесені на велику відстань, точніше, на відстань, при якому ефекти, пов'язані з завмираннями, стають некорреліpованнимі. Ця властивість може бути використано кількома шляхами.

Один з варіантів полягає в застосуванні декількох антен для одного приймача. Відстань між антенами вибирають таким чином, щоб при завмиранні в зоні розташування однієї антени в зоні іншої був хороший прийом. Приймач повинен бути здатний відрізнити хороший сигнал від поганого і вибрати найкращий.

Аналогічний варіант з декількома антенами може бути використаний для передавача. У методі передачі зі зворотним зв'язком передавальні антени перебираються по черзі, поки від приймача не прийде сигнал проте, що повідомлення прийнято. Якщо застосування декількох антен неможливо, замість додаткових антен можна використовувати інші станції вякості ретрансляторів.

Питання безпеки

Проблеми безпеки поділяються на завдання аутентифікації (встановлення автентичності особистості), яка виконується зазвичай за допомогою ідентифікації імені користувача та пароля, завдання розмежування прав доступу до системи і завдання захисту інформації за допомогою методів шифрування.

Механізми шифрування засновані на алгоритмах, які перетворять сигнали, що несуть інформацію, в шумоподібні (псевдовипадкові) сигнали. Використовуються два види шифрів: потоковий (груповий) і блоковий шифр.

Шифри обох типів генерують ключовий потік, який визначається значенням секретного ключа. Ключовий потік змішується з кодованими даними за схемою "Що виключає АБО", в результаті чого виходить закодований текст.

6.2 Технологія Bluetooth

Нині існує три широко поширених стандарти на безпровідні мережі: Bluethooth (IEEE 802.15.1), ZigBee (IEEE 802.15.4) і Wi - Fi (IEEE 802.11).

Устаткування для цих мереж не вимагає отримання ліцензії (що у багатьох випадках найбільш важливе), хоча і вимагає реєстрації.

Bluetooth був спроектований на базі стандарту IEEE 802.15.1 спеціально для заміни кабелю при з'єднанні різних облаштувань офісної і побутової техніки з використанням частотного ISM діапазону 2,4 ГГц. Специфікація Bluetooth підтримується організацією SIG (Bluetooth Special Interest Group), утвореною в 1998 р. і включаючою 1900 членів. У системах автоматизації Bluetooth зручний для запису програм в ПЛК, дистанційного прочитування показників з накопичувачів інформації. Він організований у вигляді "пикосетей" (piconet), в яких один провідний пристрій здійснює взаємодію не більше ніж з сім'ю веденими. Ведені пристрої можуть взаємодіяти один з одним тільки через те, що веде. Кожен пристрій може бути членом чотирьох пикосетей одночасно, але головним може бути тільки в одній з них. Такий пристрій виконує роль моста між пикосетями. Декілька взаємодіючих пикосетей утворюють так звану scatternet ("розкидану мережу").

Трафік в мережі організований з тимчасовим розділенням каналів і дуплексною передачею. Тимчасове розділення здійснюється інтервалами (тимчасовими слотами) завдовжки в 625 мкс (рис. 6.4). Провідні пристрої можуть починати передачу тільки впродовж інтервалів з непарними номерами, ведені відповідати впродовж парних інтервалів. Впродовжкожного інтервалу можна передати 366 біт.

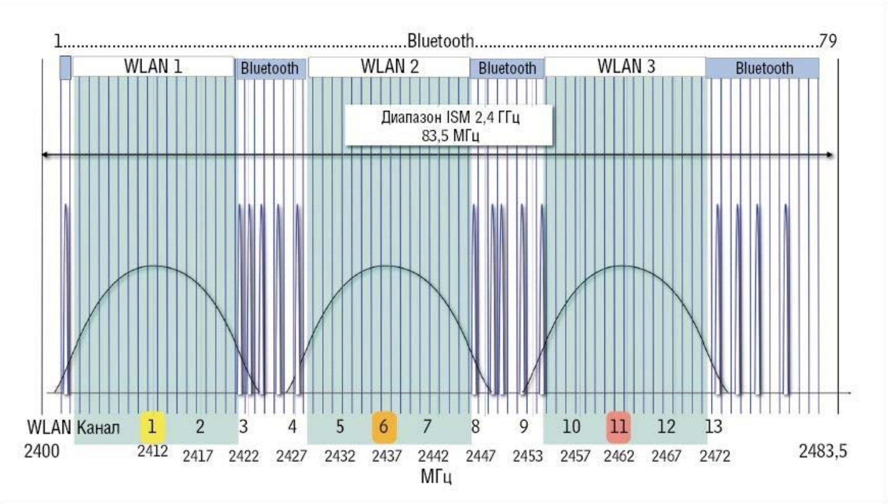


Рисунок 6.4 – Тимчасове розділення каналів Bluetooth

У Bluetooth використовується широкосмугова модуляція типу FHSS. Перехід з однієї частоти на іншу виконується за випадковим законом, який встановлюється для кожного з'єднання індивідуально. Це підвищує міру захисту інформації. Частота, що несе, змінюється 1600 разів в секунду. Швидкість передачі дорівнює 433,9 Кбит/с.

Якщо пикосети розташовані близько одна від одної, то вони можуть впливати один на одного, оскільки між ними немає ніякої синхронізації. Щоб зменшити вірогідність взаємовпливу, використовується адаптивний метод стрибкоподібної зміни частоти AFH.

На канальному рівні використовуються два типи пакетів даних : ACL (Asynchronous ConnectionLess - "асинхронний без прямого з'єднання каналів") і SCO (synchronous connection - oriented - "синхронний з прямим з'єднанням"). ACL -пакеты використовуються спільно з перевіркою контрольної суми (CRC). Якщо контрольні суми приймача і передавача не співпадають, проситься повторна передача пакету. Використовується шість різних ACL пакетів, що охоплюють різну кількість тимчасових слотів. ACL пакети використовуються у тому випадку, коли цілісність даних важливіша за швидкість їх доставки.

Пакети SCO підтримують трафік реального часу шляхом резервування тимчасових слотів. Повторна передача тут не допускається, хоча є "розширений" варіант SCO, в якому допускається обмежена кількість повторних передач. Існує три типи SCO пакетів однакової довжини (HV3, HV2, HV1), по 366 мкс, які дозволяють передавати дані із швидкістю 64кбит/с.

Кожне Bluetooth пристрій має 48-бітову адресу.

Більшість Bluetooth пристроїв мають потужність передавача 1 мВт, проте дозволений наступний ряд потужностей, що ділить усі пристрої на три класи :

- клас 1 - до 100 мВт (максимальна дальність на відкритомупросторі до 100 м);
- клас 2 - до 2,5 мВт (максимальна дальність на відкритомупросторі до 15 м);
- клас 3 - до 1 мВт (максимальна дальність на відкритому просторідо 5 м).

Можна назвати наступні достоїнства технології Bluetooth : малі розміри устаткування, простота використання, безпека передачі інформації (завдяки аутентифікації і кодуванню), хороша підтримка стандартів. До недоліком можна віднести відносно велике споживання енергії і неможливість побудови мереж складної конфігурації.

Ці особливості пов'язані з тим, що Bluetooth вирішує проблему заміни кабелів для пристроїв, що підключаються до комп'ютера, а не проблему створення безпровідної LAN.

6.3 Технології ZigBee и IEEE 802.15.4

Для вирішення завдань моніторингу в різних сферах в даний час все більше застосування знаходять бездротові сенсорні мережі (БСМ). БСМ – це розподілена самоорганізована бездротова мережа, що складається з вузлів мережі – мініатюрних обчислювально-комунікаційних пристроїв, що називаються сенсорами або вузлами БСМ. Вона будується як сукупність підмереж (кластерів), пов'язаних шлюзами. Вузол мережі містить датчик, який сприймає дані від зовнішнього середовища, мікроконтролер, пам'ять, радіо-приймач, автономне джерело живлення (рис. 6.5).

Рисунок 6.5 – Узагальнена схема вузла бездротової сенсорної мережі

БСМ складається з вузлів чотирьох видів:

1) пристрою з обмеженими функціями RFD (reduced-function device), що здійснюють збір інформації в деякій околиці певної точки зовнішнього середовища;

2) повнофункціональні пристрої FFD (full-function device), що виконують як збір інформації, так і ретрансляцію пакетів;

3) маршрутизаторів, що передають повідомлення між кінцевими пристроями і координатором або іншими маршрутизаторами;

4) координатора, який здійснює глобальну координацію, організацію та установку параметрів мережі.

При проектуванні бездротових сенсорних мереж неможливо перенести програмне забезпечення, яке добре протестоване і налагоджене влабораторних умовах для невеликої мережі, на реальну мережу. Цепов'язано з тим, що динамічні процеси, що протікають в бездротових сенсорних мережах, істотно змінюються при масштабуванні систем. Наявність великої кількості транзитних вузлів, які ретранслюють повідомлення (пакети), істотно впливає на оптимальність вибраних параметрів протоколів передачі даних. Тому, для аналізу і розуміння динамічних процесів, що протікають в бездротових сенсорних мережах, також необхідні їх математичні моделі.

Одним із перших прототипів бездротової сенсорної мережі можна вважати систему СОСУС, призначену для виявлення та ідентифікації підводних човнів. Технології бездротових сенсорних мереж стали активно розвиватися в середині 1990-х років, на початку 2000-х років розвиток мікроелектроніки дозволив виробляти для таких пристроїв досить дешеву елементну базу. Бездротові мережі початку 2010-х років переважнобазуються на стандарті ZigBee.

Розглянемо вимоги до сучасних бездротових сенсорних мереж і проаналізуємо основні телекомунікаційні технології, які можуть забезпечити вирішення поставлених завдань.

Інфотелекомунікаційна бездротова сенсорна мережа, що охоплює велику кількість точок моніторингу, будується як сукупність підмереж, кожна з яких представляє собою групу функціональних і транзитних вузлів, що виконують збір інформації та її доставку в один спеціальний вузол (шлюз, маршрутизатор), який здійснює управління вузлами кластера, обробку отриманої інформації та її передачу в базову станцію (сервер, координатор), керуючий роботою всієї БСМ, рисунок 6.6.

За оцінками експертів, тільки збір даних в реальному часі про різні аспекти виробничих процесів призведе в найближчі роки до багаторазового збільшення трафіку в розподілених системах промислового управління. По мірі зростання БСМ можливо, що її продуктивність буде знижуватися. Причина полягає в наступному: в типовій бездротовій сенсорній мережі дані, що збираються усіма вузлами, зберігаються на єдиному сервері (координаторі), який виступає в якості шлюзу з мережею IP або іншими мережами. Аби досягти шлюзу, дані переміщуються по мережі від вузла до вузла, при цьому існує ймовірність втрати деякої кількості даних, що зростає зі збільшенням розміру мережі.

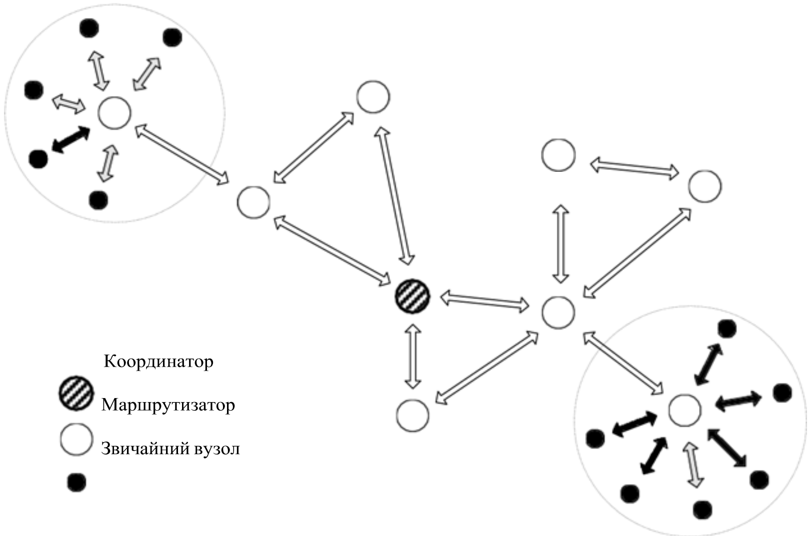


Рисунок 6.6 – Структура бездротової сенсорної мережі

Крім того, коли вузол передає дані сусіднього вузла, а той передає їх далі, витрачається енергія. БСМ великого обсягу з безліччю вузлів витрачають для передачі даних значно більше електроенергії. Тому по мірі зростання сенсорної мережі її продуктивність знижується. Щоб вирішити проблеми з продуктивністю, необхідно використовувати спеціальні технології та протоколи для підвищення рівня гетерогенності в бездротових сенсорних мережах.

Все це свідчить про те, що роботи, націлені на створення моделей бездротових сенсорних мереж, які давали б можливість з високою точністю моделювати поведінку мережі, є актуальними.

Системи моделювання бездротових сенсорних мереж дозволяють:

- підвищити ефективність розробки сенсорних мереж;
- зрозуміти обґрунтованість реального розгортання сенсорної мережі;
- проводити наукові дослідження в галузі сенсорних мереж;
- істотно скоротити витрати на розробку і розгортання мережі.

На сьогодні для моделювання бездротових сенсорних мереж використовуються системи NS-2, NS-3 [9] і TOSSIM [10]. Але ці системи моделювання мають недоліки, які не дозволяють або роблять складним їх використання для моделювання великих сенсорних мереж.

Об'єднані в бездротову мережу сенсори можуть відслідковувати параметри навколишнього середовища: рух, світло, температуру, тиск, вологість і т.і. Моніторинг може здійснюватися на великій території, тому що сенсори передають інформацію по ланцюжку від сусіда до сусіда. Технологія БСМ дозволяє роками працювати без зміни батарей. У перспективі кожен з мільярдів сенсорів отримає IP-адресу, і вони навіть можуть сформувати щось на кшталт Глобальної сенсорної мережі.

З точки зору апарату, очевидних перешкод до подальшого розвитку бездротових мережевих технологій не спостерігається. Багато в чому розвиток БСМ стримується відставанням у сфері програмного забезпечення, а саме в області алгоритмів маршрутизації і моделей збору інформації, застосовних для мереж із змінною топологією, які повинні реалізовувати управління інформаційними потоками в таких мережах з метою їх ефективного використання. За останнє десятиріччя, створені десятки і сотні алгоритмів маршрутизації для звичайних мереж, а багато в чому і відповідні математичні методи і моделі, що лежать в їх основі, виявляються зовсім незастосовні, або, принаймні, не застосовні в початковому вигляді, без доопрацювання, для використання в мережах зі змінної, особливо з швидко змінюваною топологією. Такі алгоритми повинні бути здатні вирішувати різні, в тому числі і специфічні (створення стійкого каналу зв'язку, маршрутизація в неорганізованому середовищі, доставка повільного трафіку) завдання маршрутизації в існуючих і перспективних бездротових мережах різних типів. У першу чергу мова йде про мережі з обмеженою мобільністю.

У конструктивному плані основною відмінністю сенсорних мереж від класичних телекомунікаційних радіомереж є використання як об'єктів мережі великого числа надмініатюрних і дешевих пристроїв. Кожен вузол бездротової сенсорної мережі може містити різні датчики фізичних параметрів навколишнього середовища (освітленість, температура, вологість, тиск, датчики вібрацій, датчики руху, датчики диму і вогню, датчики води, та ін.), а також засоби для первинної обробки та зберігання отриманих даних. Кількість об'єктів у такій БСМ теоретично визначається тільки областю застосування і бюджетом, а їх кількість, завдяки низькій ціні окремих пристроїв, може бути вельми велика, порядку тисяч.

У промисловості сенсорні мережі можна використовувати для безперервного моніторингу обладнання та передбачення його відмов або необхідності профілактичного обслуговування з набагато більшою точністю, що дозволить компаніям уникнути дорогих аварій або простоювання виробничих ліній.

Операційні системи для бездротових сенсорних мереж менш складні, ніж універсальні операційні системи в силу обмеженості ресурсів в апаратному забезпеченні сенсорної мережі. Завдяки цьому, операційній системі не потрібно включати підтримку користувацьких інтерфейсів. Устаткування бездротових сенсорних мереж не відрізняється від традиційних вбудованих систем, і тому для сенсорних мереж можна використовувати вбудовану операційну систему. TinyOS – перша операційна система, спеціально призначена для бездротових сенсорних мереж. Потоки в TinyOS є атомарними і, на відміну від потоків в інших ОС, виконуються аж до свого завершення, хоча вони і можуть бути витіснені подіями. TinyOS і програми для TinyOS написані на спеціальній мові програмування nesC, яка є розширенням мови Сі.

При проектування та реалізації бездротових сенсорних мереж необхідно вирішити безліч складних проблем, що відносяться до різних областей досліджень. До однієї з основних проблем відносять забезпечення високої відмовостійкості БСМ. Важливість цієї проблеми для БСМ визначається їх застосуванням, а також високою ймовірністю порушення роботи мережі в результаті відмов вузлів і каналів зв'язку, що обумовлено великою кількістю бездротових сенсорних вузлів, можливістю впливу зовнішніх несприятливих факторів, а також обмеженнями в енергоспоживанні вузлів. Процес проектування відмовостійкої бездротової сенсорної мережі є досить складним. Так він не формалізований у вигляді жорсткого набору правил, алгоритмів і стандартів, виконання яких гарантує побудову БСМ, що задовольняє вимоги до її структурної надійності, енергоефективності та вартості. На поточний момент часу розроблено ряд методів і алгоритмів вирішення окремих завдань, що виникають у процесі проектування відмовостійких БСМ, проте ці методи мають істотні обмеження, що перешкоджають їх застосуванню при побудові складних БСМ. Вони не враховують деякі важливі особливості БСМ, такі як можливість відмови мережі внаслідок виходу з ладу її компонентів, вичерпання енергоресурсів автономних джерел живлення вузлів мережі.

Для реалізації бездротової сенсорної мережі для моніторингу приміщень як системі інтелектуального будинку необхідно, щоб мережа мала:

- тривалий термін автономної роботи вузлів, для зменшення витрат напоточне обслуговування;
- високу надійність, мережа повинна функціонувати навіть у разі виходу з ладу деяких вузлів;
- масштабованість мережі, з можливістю підключення нових типіввузлів і виконавчих вузлів;
- гнучкість протоколів, можливість розширення протоколу передачі повідомлень, можливість передачі керуючих команд і команд моніторингу стану мережі ;
- переносимість програмного забезпечення на інші платформи .

Також до основних проблем, що вимагають рішення при побудові відмовостійких БСМ відносяться:

- розробка алгоритмів синтезу базової і надлишкової структури БСМ;
- розробка алгоритмів оцінки структурної надійності БСМ;
- розробка алгоритмів оцінки загального часу роботи БСМ до моментуїї відмови.

Задачу забезпечення відмовостійкості БСМ можна сформулювати як класичну постановку задачі оптимізації: спроектувати структуру БСМ, яка забезпечує надійність (тобто ймовірність зв'язності) Q ≥ Q * та енергоефективність * E ≥ E , де значення * Q і * E задані спочатку, при мінімізації вартості C всієї структури CC, тобто при C → Cmin.

Бездротова сенсорна мережа повинна володіти такими особливостями:

- для передачі інформації використовується електромагнітневипромінювання;
- потужність передавачів вузлів БСМ не повинна перевищувати 1 мВт;
- швидкість передачі даних в БСМ до 250000 біт в секунду;

- вузли БСМ оснащені датчиками для вимірювання значень полівфізичних величин.

Таким чином, до складу вузла будь-якої проектованої БСМ повинні входити мікропроцесор, приймач - передавач, датчики, джерело живлення. За своїм призначенням в БСМ вузол може бути вимірником, маршрутизатором або шлюзом-координатором. З чого випливає дві групи проблем.

Перша полягає в тому, що при використанні радіоканалу для передачі результатів вимірювань в умовах малої потужності вузлів, складноїрадіоелектронної обстановки і змінюваної конфігурації об'єкта обслуговування можлива втрата результатів вимірювань того чи іншого вузла в процесі їх доставки до шлюзу (координатора). У цьому випадку необхідно скоротити втрати інформації при передачі результатів вимірювань від кожного вузла БСМ координатору мережі. При цьому кількісна характеристика втрат інформації є основним критерієм оцінки системи моніторингу.

Друга проблема пов'язана з забезпеченням вузлів мережі електроживленням. Вона обумовлена необхідністю використання в БСМ автономних джерел живлення, у зв'язку з тим, що використання мережевих джерел живлення вимагає прокладання проводів для живлення сенсорних

вузлів і робить використання бездротової технології для передачі даних безглуздим. У свою чергу необхідність використання автономних джерел живлення, у яких термін служби без прийняття спеціальних заходів може виявитися неприйнятно коротким, породжує проблему забезпечення довговічності роботи БСМ, кількісна характеристика якої визначається часом вичерпання заряду джерел живлення вузлів БСМ і такожрозглядається як критерій оцінки БСМ.

Бездротові сенсорні мережі, побудовані на основі стандарту IEEE 802.15.4, є єдиною бездротовою технологією, не критичною до часу автономної роботи пристроїв. Основною можливістю для економії енергії вузлами мережі є розрідження інтервалів активності їх пристроїв, і в першу чергу прийомопередавача, як основного споживача. Для забезпечення розумного часу життя мережі, частка інтервалів активності радіопередачі повинна складати близько 0.1% від загального часу роботи.

Визначення часу включення передачі – завдання алгоритмів рівня доступу до середовища стека мережевих протоколів. Існуючі алгоритми відрізняються один від одного підходом до упорядкування періодів активності радіо між вузлами мережі, способом визначення готовності приймаючої сторони, способом визначення зайнятості середовища передачі.

Другим способом економії на алгоритмічному рівні є багатоланкова передача, тобто відправлення повідомлень з однієї точки мережі в іншу по ланцюжку проміжних вузлів замість прямої далекої радіопередачі.

Третій спосіб – по можливості обробляти дані вимірювань на вузлах, тому що в ряді завдань попередні обчислення на місці можуть бути ефективніше передачі «сирих» даних до центру обробки. Це дозволяє скоротити обсяг даних, які необхідно передати. А значить, зменшити час роботи сенсора в режимі передачі.

Для БСМ характерні такі аспекти, як зменшення енергоспоживання, збільшення мобільності, повідомлення вузлів мережі на далеких відстанях. Порівняння протоколів для побудови БСМ, реалізованих на основі стандарту IEEE 802.15.4 і не на його основі наведено в таблиці 6.1

Таблиця 6.1 – Порівняння протоколів, реалізованих на основі стандарту IEEE 802.15.4 і не на його основі для побудови БСМ

Параметр	IEEE 802.15.4/ZigBee			Z-Wave	
Робоча частота, МГц	860	908	2400	860	908
Швидкість передачі даних, Кб	20	40	250	40	
Кількість каналів	1	10	16	1	
Кількість вузлів	65535			До 232	
Доступ до середовища	CSMA-CA			випадковий	

Топологія	“зірка” “багатокоміркова” “дерево”	“багатокоміркова”
Максимальна кількість стрибків	не більше 32	не більше 4
Кількість виробників	Chipcon(TI), Freescale, Ember, Atmel, ZMD и т.д.	Zensys

Як можна побачити з таблиці 6.1, поєднання стандарту IEEE 802.15.4 та протоколу ZigBee володіє значно більшою гнучкістю в порівнянні з Z- Wave. Потрібно також відзначити, що для реалізації БСМ є стандарт фізичного рівня, широкий вибір елементної бази, але програмну частину необхідно реалізовувати самостійно під вузькі специфічні завдання. Прийнятий стандарт IEEE 802.15.4 описує контроль доступу до бездротового каналу і фізичний рівень для низькошвидкісних бездротових персональних мереж, тобто два нижніх рівні згідно мережевої моделі OSI.

6.4 Промисловий Wi – Fi

Головне призначення технології Wi-Fi (Wireless Fidelity - "бездротова точність") – бездротове розширення мереж Ethernet. Вона використовується там, де неможливе або небажано використовувати провідні мережі. Наприклад, для передачі інформації від частин механізмів що рухаються, коли не можна використовувати стіни, на великих площах – складах, де потрібно ПК завжди мати із собою.

Wi - Fi розроблений консорціумом Wi - Fi на базі серії стандартів IEEE

802.11 (1997 р.) і забезпечує швидкість передачі від 1...2 до 54 Мбіт/с. Wi - Fi

консорціум розробляє прикладні специфікації для втілення стандарту Wi - Fi в життя, виконує тестування і сертифікацію продукції інших фірм на відповідність стандарту, організовує виставки, забезпечує необхідною інформацією розробників Wi - Fi устаткування.

Попри те, що стандарт IEEE 802.11 був ратифікований ще в 1997 р., мережі Wi - Fi отримали широке поширення тільки останніми роками, коли істотно знизилися ціни на серійне мережеве устаткування. У промисловій автоматизації з безлічі стандартів серії 802.11 використовуються тільки два: 802.11b із швидкістю передачі до 11 Мбіт/с і 802.11g (до 54 Мбіт/с).

Архітектура мережі Wi-Fi

Стандарт IEEE 802.11 становить три варіанті топологій мереж:

– незалежні базові зони обслуговування (Independent Basic Service Sets, IBSS);
– базові зони обслуговування (Basic Service Sets, BSS);
– розширені зони обслуговування (Extended Service Sets, ESS).

При створенні проекту по розгортанню Wi - Fi мережі на промисловому підприємстві робиться комп'ютерне моделювання, що дозволяє оптимально розташувати точки доступу з найкращим покриттям зони роумінгу при найменших витратах на устаткування.

Рисунок 6.7 – Планування розміщення точок доступу на підприємстві

Стандарт IEEE 802.11 встановлює три варіанти топології мереж:

- незалежні базові зони обслуговування (Independent Basic ServiceSets, IBSS);
– базові зони обслуговування (Basic Service Sets, BSS);
– розширені зони обслуговування (Extended Service Sets, ESS).

Під зоною обслуговування тут розуміється набір логічно згрупованих пристроїв. Кожна зона обслуговування має свій ідентифікатор (Service Set Identifier, SSID). Станція-приймач використовує SSID для визначення того, з якої зони обслуговування приходить сигнал.

У архітектурі IBSS станції зв'язуються безпосередньо одна з іншою, без використання точки доступу і без можливості під'єднування до дротяної локальної мережі. Зона обслуговування SSID використовується зазвичай дляоб'єднання в мережу малої кількості станцій, оскільки в ній не передбачена можливість ретрансляції сигналу для збільшення дальності зв'язку і механізми для вирішення проблеми прихованого вузла.

При використанні BSS станції спілкуються другом з другом через загальний центральний вузол зв'язку, що називається точкою доступу. Точка доступу зазвичай підключається до дротяної локальної мережі Ethernet.

Розширена зона обслуговування виходить при об'єднанні декількох BSS в єдину систему за допомогою розподільної системи, якою може виступати дротяна мережа Ethernet.

Модель OSI в промисловій бездротовій мережіФізичний і канальний рівень

Таблиця 6.2 – Рівні моделі OSI для Wi-Fi/IEEE 802.11

Номер рівня	OSI модель	Мережа	Функції
2	Канальний (передачи данных)	Підрівень LLC	-
		Підрівень MAC	
1	Фізичний	Підрівень PLCP	Бездротовапередача, оцінка стану ефиру
		Підрівень PMD	

Модель OSI для стандартів Wi-Fi і IEEE 802.11 показана в табл. 6.2.

Основне призначення фізичних рівнів - забезпечення інтерфейсу з бездротовою середовищем передачі (з ефіром), а також оцінка стану ефіру і взаємодія з рівнем MAC.

Фізичний рівень складається з двох підрівнів:

– PLCP (Physical Layer Convergence Protocol) - виконує процедуру відображення PDU рівня MAC у фрейм формату FHSS або DSSS. Ця процедура виконує передачу, виявлення несучої і прийом сигналу;

– PMD (Physical Medium Dependent) - "підрівень, що залежить від середовища передачі". Цей рівень буде різним для різних швидкостей передачі і різних стандартів із серії 802.11. Підрівень PMD забезпечує дані і сервіс для підрівня PLCP і функції радіопередачі і прийому, результатом яких є потік даних, інформація про час, параметри прийому.

Основним робочим станом рівнів PLCP є виявлення несучої і оцінка незайнятості каналу. Для виконання передачі PLCP перемикає PMD з режиму "прийом" в режим "передача" і посилає елемент даних PPDU (PLCPData Unit).

Фізичний рівень виконує скремблирование, кодування і чергування.

Передача сигналів по радіоканалу виконується двома методами: FHSS і DSSS. При цьому використовується диференціальна фазова модуляція DBPSK і DQPSK із застосуванням кодів Баркера, комплементарних кодів(CCK - Complementary Code Keying) і технології подвійного сверточногокодування (PBCC).

Wi-Fi 802.11g на швидкості 1 і 2 Мбіт/с використовує модуляцію DBPSK. При швидкості передачі 2 Мбіт/с використовуються ті ж метод, що і при швидкості 1 Мбіт/с, проте для збільшення пропускної здатності каналу використовується 4 різних значення фази (0, $\pi/2$, $(3/4)\pi$, π) для фазовоїмодуляції несучої.

Протокол 802.11b, використовує додатково швидкості передачі 5,5 і 11 Мбіт / с. На цих швидкостях передачі замість кодів Баркера використовуються комплементарні коди (ССК).

Wi-Fi використовує метод доступу до мережі CSMA/CA в якому для зниження ймовірність колізій використані наступні принципи:

– перш, ніж станція почне передачу, вона повідомляє, як довго вона буде займати канал зв'язку;

– наступна станція не може почати передачу, поки не закінчиться зарезервоване раніше час;

– учасники мережі не знають, чи прийнятий їх сигнал, поки не отримають підтвердження про це;

– якщо дві станції почали працювати одночасно, вони зможуть дізнатися про це тільки по тому факту, що не отримають підтвердження про прийом;

– якщо підтвердження не отримане, учасники мережі вичікують випадковий проміжок часу, щоб почати повторну передачу.

Запобігання, а не виявлення колізій, є основним в бездротових мережах, оскільки в них, на відміну від провідних мереж, передавач трансивера заглушає сигнал, що приймається.

Формат фрейму на рівні PLCP моделі OSI в режимі FHSS показанийна рис. 6.8. Він складається з наступних полів:

– «Синхронізація." - містить чергуються нулі і одиниці. Служить для підстроювання частоти на приймаючої станції, синхронізує розподіл пакетів і дозволяє вибрати антену (при наявності декількох антен);

– "Старт" - прапор початку фрейма. Складається з рядка 0000 1100 1011 1101, яка служить для синхронізації фреймів на приймаючої станції;

– "PLW" - "Psdu Length Word" - "слово довжини службового елемента даних PLCP", PSDU - "PLCP Service Data Unit" - елемент даних підрівня PLCP; вказує розмір фрейма, що надійшов з рівня MAC, в октетах;

– "Швидкість" - вказує швидкість передачі даних фрейму;

– "КС" - контрольна сума;

– "MAC-фрейм" - фрейм, що надійшов з MAC-рівня моделі OSI і містить PSDU;

– "Тема PLCP" - поля, додані на підрівні PLCP.

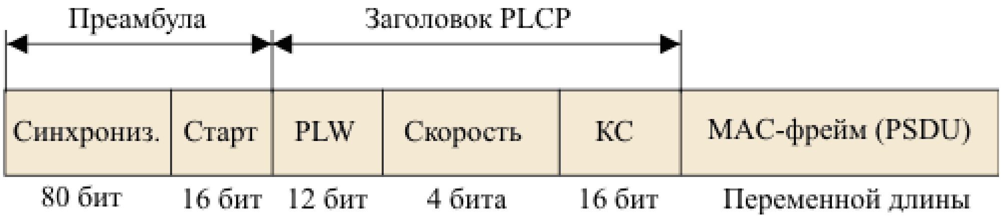


Рисунок 6.8 – Формат фрейму PLCP для режиму FHSS

Формат фрейму на рівні PLCP моделі OSI в режимі DSSS показаний на рис. 6.9. У ньому поля мають такий зміст:

– «Синхронізація." - містить тільки одиниці і забезпечуєсинхронізацію в приймальні станції;
– "Старт" - прапор початку фрейма. Містить рядок 0 xF3A0, яквказує початок передачі параметрів, що залежать від фізичного рівня;
– "Сигнал" - вказує тип модуляції і швидкість передачі данногофрейму;
– "Сервіс" - зарезервовано для майбутніх модифікацій стандарту;
– "Довжина" - вказує час в мікросекундах, необхідне для передачіMAC-фрейма;
– "КС" - контрольна сума;
– "MAC-фрейм" - фрейм, що надійшов з MAC-рівня моделі OSI імістить PSDU;
– "Тема PLCP" - поля, додані на підрівні PLCP.

Рисунок 6.9 – Формат фрейма PLCP для режиму DSSS

Дальність зв'язку засобами Wi-Fi сильно залежить від умов поширення електромагнітних хвиль, типу антени і потужності передавача. Типові значення, що вказуються виробниками Wi-Fi обладнання, складають 100-200 м в приміщенні і до декількох кілометрів на відкритій місцевості із застосуванням зовнішньої антени і при потужності передавача 50 ... 100 мВт. Разом з тим, за повідомленням німецького тижневика "Computerwoche" під час змагань по дальності зв'язку була зафіксована зв'язок на відстані 89 км із застосуванням стандартного устаткування Wi-Fi стандарту IEEE 802.11b (2,4 ГГц) і супутникових антен ("тарілок"). У книзі рекордів Гіннеса зафіксована також Wi-Fi зв'язок на відстані 310 км із застосуванням антен, піднятих на велику висоту за допомогою повітряних куль.

Стандарт Wi-Fi дає клієнту повну свободу при виборі критеріїв для з'єднання, однак не описує всі аспекти побудови бездротових локальних мереж. Виходячи з цього кожен виробник устаткування вирішує цю задачупо своєму, застосовуючи ті підходи, які він вважає за якнайкращі з тієї або іншої точки зору. Ось чому виникає необхідність класифікації способів побудови бездротової локальної мережі.

За способом об'єднання точок доступу в єдину систему виділяють:

- автономні точки доступу;
- точки доступу, що працюють під управлінням контролера;
- безконтролерні, але не автономні.

За способом організації і управління радіоканалами виділяють:

- із статичними налаштуваннями радіоканалів;
- з динамічними (адаптивними) налаштуваннями радіоканалів;
- з «шаруватою» або багатошаровою структурою радіоканалів.

Всі стандарти Wi-Fi 802.11 працюють в ISM (промислових, наукових і медичних) частотних діапазонах. Ці частоти є відкритими для загального кола користувачів тому для роботи в них не потрібно ліцензій.

Є цілий ряд стандартів які знаходяться в загальному користуванні. Це стандарти 802.11a, 802.11b, 802.11g та стандарт 802.11n що забезпечує швидкість передачі даних до 600 Мбіт/с.

Кожен з цих стандартів має різні функції, та були запущені в різні моменти часу. Порівняльна характеристика стандартів 802.11 приведена в таблиці 6.3.

Таблиця 6.3 – Порівняльна характеристика стандартів 802.11

	802.11a	802.11b	802.11n
Дата затвердження стандарту	Липень 1999	Липень 1999	Жовтень 2009
Максимальна швидкість передачі даних, Мбіт/с	54	11	~ 600
Модуляція	OFDM	CCK або DSSS	CCK, DSSS або OFDM
Частотна група, ГГц	5	2,4	2,4 або 5
Кількість просторових потоків	1	1	1, 2, 3 або 4
Ширина каналу, МГц	20	20	20 або 40

Для використання в промисловості технології Wi-Fi пропонуються поки обмеженим числом постачальників. Дані технології застосовуються для управління рухомими об'єктами і в складській логістиці, а також у тих випадках, коли з якої-небудь причини неможливо прокладати провідні мережі Ethernet. Використання Wi-Fi пристроїв на підприємствах обумовлено високою завадостійкістю, що робить їх застосовними на підприємствах з безліччю металевих конструкцій. У свою чергу Wi-Fi прилади не створюють істотних перешкод для вузькосмугових радіосигналів. В даний час технологія знаходить широке застосування на віддаленому або небезпечному виробництві, там де знаходження оперативного персоналу пов'язано з підвищеною небезпекою.

Виходячи з вищеописаного можна виділити переваги та недоліки використання стандарту зв'язку IEEE 802.11.

Перевагами є:

- розгортання мережі без прокладки кабелю;
- дозволяє мати доступ до мережі мобільним пристроям;
- широке поширення на ринку;
- гарантується сумісність устаткування завдяки обов'язковійсертифікації;
- низьке випромінювання у момент передачі даних;

- Wi-Fi устаткування може працювати в різних країнах по всьому світу.

Недоліками можуть бути:

- частотний діапазон і експлуатаційні обмеження в різних країнах неоднакові;
- реальна швидкість залежить від наявності між пристроями фізичних перешкод, перешкод від інших бездротових пристроїв або електронної апаратури;
- точки безпровідного доступу що перевищують 100 мВт, підлягають обов'язковій реєстрації;
- найпопулярніший стандарт шифрування WEP може бути відносно легко зламаний навіть при правильній конфігурації (через слабку стійкість алгоритму).

Модуль працюючий по технології Wi-Fi, другорядний приймальний пристрій, складається з точки доступу WLAN 5100 FL (Рис. 6.11) та адаптеру порту Ethernet WLAN для бездротового підключення інтерфейсу Ethernet до

точки доступу WLAN – WLAN EPA FL.

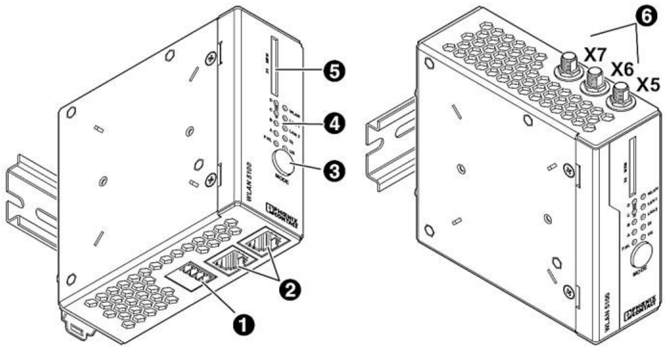


Рисунок 6.11 – Точка доступу WLAN 5100 FL

1 – підключення напруги живлення і цифрового входу або виходу; 2 – два роз'єми Ethernet; 3 – кнопка «Mode»; 4 – світлодіодні індикатори статусута діагностики; 5 – слот для додаткової карти пам'яті, SD; 6 – антенні входи.

6.5 Технологія Trusted Wireless

Radioline - це нова система радіозв'язку для протяжних систем. Ви можете легко привласнювати введення і виводи одним поворотом ручки налаштування, не прибігаючи до програмування.

Завдяки використанню технології радіозв'язку Trusted Wireless 2.0 система Radioline є найкращим вибором надійного застосування в промислових умовах.

Рисунок 6.12 – Зовнішній вигляд модульної структури системи наоснові Trusted Wireless

Можливість безпровідної передачі більшої кількості сигналів введення/виводу — завдяки модульній конструкції.

Рисунок 6.13 – Подлючение накопичувача Conf - Stick для однозначної мережевої адресації

Вставний накопичувач Conf - Stick забезпечує можливість однозначноїмережевої адресації

Trusted Wireless 2.0 – бездротова технологія, розроблена спеціально для промислового використання. Вона особливо підходить для передачі інформації з датчиків виконаних по безкабельній інфраструктурі або для передачі невеликих і середніх обсягів даних – навіть на великих відстанях від кількох сотень метрів до кількох кілометрів.

Основними особливостями Trusted Wireless 2.0 є:

- міцні зв'язки завдяки FHSS (поширення стрибкоподібної частоти спектру);
- автоматичний і ручний механізми налаштувань;
- безпечне з'єднання завдяки 128-бітному шифруванню AES та аутентифікації;
- велика дальність через високу чутливість приймача, зміни швидкості передачі даних і високої потужності передачі (100 мВт для 2,4 ГГц, 1 Вт для 900 МГц);
- гнучкість мережевої структури: точка-точка, зірка, повторювач;
- широкі діагностичні можливості.

У діапазоні 2,4 ГГц, Trusted Wireless 2.0 завдяки FHSS використовує до 440 можливих окремих частот, з пристроями, що використовують вибір до 127 каналів. Число частот, що використовуються залежить від подальших налаштувань і механізмів, таких, як виключення деяких частотних діапазонів, або використання декількох груп частот для оптимізації паралельної роботи.

Trusted Wireless 2.0 пропонує різні, регульовані швидкість передачі даних, в залежності від вимог застосування. Таким чином діапазон може бути більшим в декілька разів ніж діапазони загальних Bluetooth і WLAN систем.

Завдяки своїй замкнутій технології, промислові бездротові мережі з Trusted Wireless 2.0 набагато краще захищені від можливих атак. Крім того, метод стрибкоподібної перебудови частоти набагато складніше піддається злому для перехоплення даних.

Крім того, Trusted Wireless 2.0 має два реальні механізми безпеки: шифрування всіх переданих даних відповідно до AES, а також аутентифікації даних відповідно до RFC3610. В якості радіопередавача використовується бездротовий модуль RAD-2400-IFS (Рис. 6.14) працюючий на частоті 2,4 ГГц з інтерфейсом RS-232/RS-485, розширюваний модулями вводу-виводу.

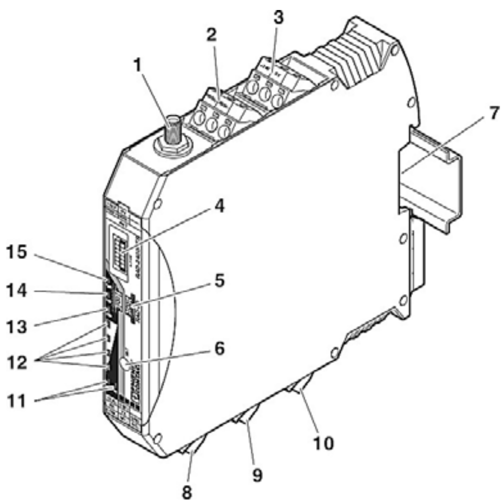


Рисунок 6.14 – Подлючение накопичувачаConf - Stick для однозначної мережевої адресації
1 – антенний роз'єм; 2 – вихід для оцінкисили радіосигналу; 3 – живлення пристрою; 4 – 12-контактнийпрограмний інтерфейс; 5 – RAD-ID; 6 – кнопка SET; 7 – підключенняз'єднувача для монтажної рейки; 8 – інтерфейс RS-485; 9 – інтерфейс RS-232; 10 – релейний вихід з перемикаючим контактом; 11-15 – індикатори стану та діагностики.

Приймач можна використовуватись в якості ведучого пристрою, відомого пристрою або повторювача/веденого пристрою. Застосовується для бездротових мереж (зі структурою «точка-точка», «зірка», зі змішаною топологією), що допускають підключення до 250 кінцевих пристроїв.

Бездротовий радіоприймач характеризується такими параметрами:

- не вимагає ліцензії діапазон частот 2,4 ГГц;
- розширений діапазон температур від -40 °C до +70 °C;
- дальність передачі вимірюється в кілометрах завдяки регульованій швидкості передачі даних бездротового інтерфейсу (від 16 до 500 кбіт/с);
- вбудовані інтерфейси RS-232/RS-485;
- швидкий і простий ввід в експлуатацію без програмування;
- висока надійність завдяки технології Trusted Wireless 2.0

(шифрування AES, стрибкоподібне зміна частоти і паралельне управління);

- має допуск ATEX для роботи у вибухонебезпечній зоні 2;
- багатоконтурні мережі до 250 кінцевих пристроїв.

6.6 Порівняння безпровідних мереж

В таблице зібрані основні параметри трьох розглянутих бездротових технологій. В таблиці відсутні дані о стандартах WiMAX, EDGE, UWB та багатьох, інших, котрі не знайшли широкого використання в промисловой автоматизації.

Таблиця 6.4 – Порівняння бездротових технологій

параметр	*Bluetooth/IEEE 802.15.1*	*ZigBee/IEEE 802.15.4*	*Wi-Fi/IEEE 802.11*
дальність	~10 (50-100) м	10-100 м	~100 м
Швидкість передачі	723 Кбіт/с	250 Кбіт/с	1…2 Мбіт/с, до 54 Мбіт/с
Макс. кількість учасників мережі	8	64 000	НЕ обмежено
споживана потужність	10 мВт	1 мВт	50 мВт
Тривалість роботи від двох батарей розміру АА	-	6 міс. в режимі очікування	-
Ціна / Складність (умовні одиниці)	10	1	20
повторна передача	Є	Є	DCF - нема; PCF і HCF - є,
Основне призначення	Зв'язок периферії з комп'ютером	Бездротові мережі датчиків	Бездротове розширення Ethernet

РОЗДІЛ 7
КОМПОНЕНТИ СИСТЕМ КЕРУВАННЯ З РОЗШИРЕННЯМ РЕАЛЬНОГО ЧАСУ

До основних компонентів систем керування реального часу відносяться:

- операційні системи (OS) з розширенням реального часу (RTX – Run Time Extended);
- PLC реального часу (RTX PLC);
- SCADA-системи;
- комунікаційне обладнання для забезпечення режимів реальног часу.

7.1 Операційні системи з розширенням реального часу

Розширення реального часу модифікують ядро операційної системи і забезпечують детерміновані інтервали часу обробки запитів. В системах керування технологічними процесами раального часу найбільш поширеними є RTX для OS Windows.

7.2 Розширення реального часу IntervalZero RTX

Розширення IntervalZero RTX корпорації «IntervalZero, Inc.» - програмний пакет, призначений для інтеграції функціональності «жорсткого» реального часу в системи під управлінням операційних систем Microsoft Windows. Програмний продукт RTX був з успіхом випробуваний в тисячах різних автоматизованих систем управління, роботах та інш. Це дозволило добитися підвищення їх ефективності, можливостей, ступеня масштабованості і надійності функціонування при одночасному скороченні термінів і вартості розробки нової продукції.

Розширення RTX IntervalZero спочатку розроблялося саме як розширення, доповнення OS Windows, що забезпечувало режими реального часу, а не як операційна система реального часу, перенесена в середовище Windows. Воно надає широкі можливості по управлінню перериваннями IRQ, пристроями вводу-виводу, оперативною пам'яттю і гарантує надійне виконання задач з потрібним пріоритетом.

Розширення RTX підтримує режим «симетричної мультипроцесності» (SMP - Symmetric Multiprocessing) і дозволяє в багатоядерних (багатопроцесорних) системах виділяти під завдання RTX до 31-го процесорного ядра. Крім цього, доступний також і режим поділу процесорних ресурсів між OS Windows і RTX, коли планувальник сам визначає, на якому ядрі виконувати конкретну задачу.

IntervalZero RTX є справжнім розширенням OS без інкапсуляції Windows і будь-якої взаємодії з інфраструктурою для Windows або її зміни (рис. 1.1). Завдяки подібному поділу, додатки підсистеми реального часу RTX (real-time subsystem, RTSS) будуть виконуватися навіть при повному збої для Windows (коли на моніторі відображається «синій екран»). Підсистема RTSS побудована на основі детермінованого планувальника реального часу, в якому реалізовані як алгоритм планування на основі пріоритетів, так і алгоритм планування з витісненням. Планувальник здійснює перемикання контексту всіх найважливіших потоків системи - при використанні сучасних процесорів передача управління потоку з більш високим пріоритетом відбувається за долі мікросекунди.

Як показано на рис. 7.1, в складі RTX присутні компоненти, що організують для додатків RTSS мережу реального часу і роботу в реальному часі з пристроями USB. Це стек протоколів RT-TCP/IP і стек RTX USB. Мережевий стек забезпечує роботу всіх мережевих протоколів і фізичного рівня в детермінованому режимі. Аналогічно здійснюється підтримка USB - всі компоненти працюють з повністю передбачуваними затримками часу.

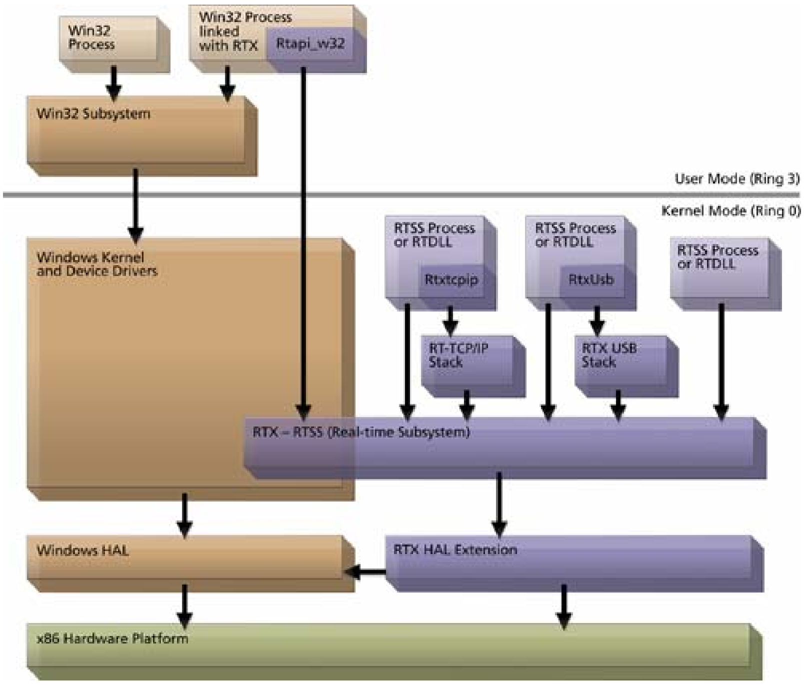


Рисунок 7.1 – Архітектура розширення реального часу RTX

Фактично наявність стеку реального часу означає роботу всіх функцій і драйверів з гарантованими часом. Таким чином, розробник може бути впевнений, що вхідний пакет буде оброблений, а вихідні дані будуть доставлені до фізичного рівня за обмежений проміжок часу. При цьому наявність стека RT-TCP/IP не означає, що вся мережа буде працювати в реальному часі, навіть якщо даний стек буде стояти на всіх вузлах мережі. Мережа складається з багатьох елементів, зокрема в ній є обладнання, яке реалізує її топологію.

На фізичному рівні мережі можуть відбуватися різні процеси, додаткове обладнання може вносити затримки часу. Тому мережевий стек реального часу дозволяє отримати достатню продуктивність, але не гарантує роботу мережі без збоїв і затримок часу. Стек RT-TCP/IP призначений тільки для досягнення максимуму продуктивності в мережевих задачах. Найбільш наближену до реального часу роботу мережі можна отримати, якщо виділити всі мережеві адаптери для RTX в один мережевий сегмент.

Також, RTX дозволяє виділяти до 31-го процесорного ядра тільки для виконання завдань реального часу. Також RTX працює і на одноядерних конфігураціях (для них доступний лише режим поділу ресурсів процесора). Кількість використовуваних ядер задається розробником. Крім того, розробник може управляти завантаженням і розподілом завдань по ядрах.

Управління додатками RTX здійснюється на основі схеми з 128-ма рівнями пріоритетів потоків. При цьому час запуску потоку обслуговування переривання не перевищує 10 мкс (на окремих платформах менше 1 мкс).

Розширення RTX IntervalZero лежить в основі нового підходу до проектування систем реального часу, що отримав назву Архітектура Soft-Control (архітектура програмного управління). Його застосування здатне змінити принципи конкуренції в таких областях, як виробництво засобів промислової автоматизації, медичних систем, контрольно-вимірювальної техніки та інших систем завдяки підвищенню вимог до можливостей і точності управління.

Основа архітектури програмного регулювання для систем «жорсткого» реального часу - інтегрована інструментальна платформа на базі багатоядерної архітектури x86, що працює під управлінням операційної системи Microsoft Windows і розширення IntervalZero RTX. Soft-Control Architecture забезпечує OEM-виробникам значну конкурентну перевагу. Soft-Control Architecture дозволить розробляти більш компактні системи, підвищувати обсяги, темпи і якість виробництва при одночасному зниженні його собівартості.

Soft-Control Architecture дозволяє виробникам відмовитися від застосування спеціалізованих апаратних засобів на основі PLC або додаткових пристроїв управління на користь програмних систем. Використання сучасних платформ спільно з RTX - економічне і багатофункціональне рішення для побудови систем реального часу на базі Windows, з архітектурою x86.

Застосування комплексного пакету засобів, що інтегрується з широко відомим інструментальним пакетом Microsoft Visual Studio, істотно скорочує терміни розробки і налагодження додатків.

7.3 Розширення реального часу INtime

Система INtime є розширенням реального часу Windows, яке було розроблено корпорацією «Radisys Corporation», а в даний час підтримується корпорацією «TenAsys».

INtime комбінує можливості OS реального часу зі стандартними OS Windows, включаючи версії Embedded, без додаткових апаратних засобів. INtime спеціально розроблена під архітектуру процесора x86. Додатки реального часу і не реального часу виконуються на різних віртуальних машинах на єдиному комп'ютері (рис. 1.2).

INtime, на відміну від RTX, слабо пов'язана з NT. Архітектура INtime базується на механізмі апаратного обслуговування задач (hardware tasking), яке забезпечується процесором Intel. В результаті, два ядра виконуються на одних апаратних засобах. Враховуючи, що вони розділяють одні апаратні засоби, стали потрібними деякі модифікації NT HAL. Такий підхід дозволяє захистити і відокремити середовище виконання і область пам'яті від Windows. Всередині INtime кожний процес додатку має свій власний адресний простір. Крім того, ядро і додатки виконуються на різних пріоритетних рівнях, що дозволяє захистити їх одне від одного.

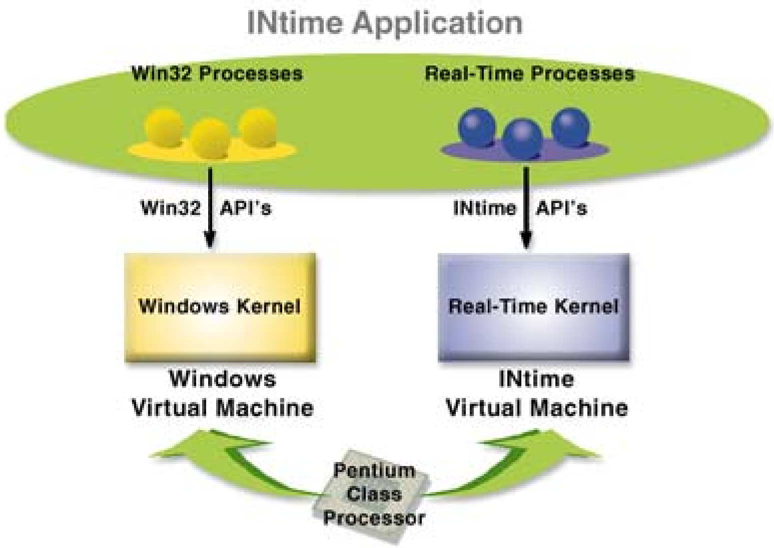


Рисунок 7.2 – Архітектура розширення реального часу INtime

INtime показує передбачувану поведінку, але його складна архітектура не дозволяє системі досягнути значної продуктивності. Внаслідок сегментаційних обмежень INtime підходить не для всіх систем реального часу.

Контрольні запитання

1) Основні компоненти систем керування реального часу;

2) Характеристика операційних систем з розширенням реального часу;

3) PLC реального часу (RTX PLC), принцип функціонування і особливості;

4) SCADA-системи, їх структура, призначення і особливості функціонування в реальному часі;

5) Комунікаційне обладнання для забезпечення режимів реального часу, типи, основні характеристики і особливості застосування;

6) Призначення і функціональність розширення реального часу для операційних систем «IntervalZero RTX»;

7) Основні компоненти і архітектура розширення реального часу «IntervalZero RTX»;

8) Переваги програмного управління на основі «Soft-Control Architecture»;

9) Основні компоненти і архітектура розширення реального часу «INtime»;

10) Класифікація, організація і основні технічні характеристики «Software PLC».

РОЗДІЛ 8
СИСТЕМИ УПРАВЛІННЯ РЕАЛЬНОГО ЧАСУ НА БАЗІ PC WORX RT BASIC Software PLC «PHOENIX CONTACT»

8.1 Функціональність систем управління реального часу

PC WORX RT BASIC Software PLC (програмний PLC) - це PLC з вбудованими шинами PROFINET, Modbus і підтримкою INTERBUS, на основі промислового або персонального комп'ютера.

Обладнання, що підтримується. Поточна версія програмного контролера (Software PLC) є оптимізованою для промислових комп'ютерів від фірми «Phoenix Contact». Найбільш оптимальні умови для експлуатації Software PLC забезпечує промисловий PC з багатоядерним процесором. У такому випадку, одне ядро процесора може бути використане для операційної системи INtime, що працює в режимі реального часу та використовує Software PLC, а інше може використовуватися для стандартної операційної системи (наприклад, Windows 8.1).

Якщо промисловий комп'ютер має два мережевих адаптери, один мережевий адаптер на постійній основі призначається Software PLC для операційної системи INtime, що працює в режимі реального часу, а інший - для операційної системи Windows. Це забезпечує надійну і високопродуктивну роботу програмного контролера Software PLC.

Контроль продуктивності IEC 61131. PC WORX RT BASIC Software PLC послідовно конфігурується та програмується відповідно до стандарту IEC 61131 з використанням програмного забезпечення для автоматизації PC WORX. PC WORX можна встановити локально на тому ж комп'ютері, на якому встановлений і Software PLC, або на окремому комп'ютері. Слід відмітити, що підтримується програмування на всіх п'яти мовах IEC 61131.

Software PLC може використовуватись незалежно від програмного забезпечення PC WORX. Так, наприклад, Sofware PLC і PC WORX можуть бути встановлені на різних промислових комп'ютерах. В такому випадку, програмування здійснюється через мережевий адаптер, що призначений для Software PLC. Якщо Software PLC і PC WORX встановлені на одному PC, програмування здійснюється через внутрішній віртуальний інтерфейс Ethernet. Зв'язок між операційною системою Windows та середовищем виконання Software PLC відбувається також через цей інтерфейс.

Вбудовані функціональні можливості Ethernet/PROFINET

Мережевий адаптер, що призначений для промислового PC, використовується в якості зовнішнього інтерфейсу Ethernet. Використовуючи цей інтерфейс, програмний контролер Software PLC може бути запрограмований через Ethernet. Software PLC також може бути використаний в якості PROFINET-контролера або PROFINET-пристрою.

При використанні сервера AX OPC, представленого в AUTOMATIONWORX Software Suite (версії 1.60 або наступних), дані програмного контролера Software PLC є доступними в мережі Ethernet в стандартному форматі і можуть бути використані для різних пакетів візуалізації.

Функціональність Modbus. Протоколи зв'язку Modbus/TCP і Modbus/UDP можуть бути використані через мережевий адаптер, що призначений для промислового PC. В цьому випадку, Software PLC може бути використаний в якості клієнта Modbus. Також Software PLC використовується в якості сервера Modbus за допомогою функціональних блоків.

INTERBUS. Функціонування INTERBUS стає можливим після налаштування обладнання. Для його використання можна застосувати слот PCI промислового PC від фірми «Phoenix Contact», встановивши плату контролера INTERBUS в роз'єм PCI.

USB-ключ. Для запуску PC WORX RT BASIC Software PLC потрібен ключ у вигляді USB-флеш. Цей USB-ключ використовується в якості ключа ліцензійного програмного забезпечення і він повинен завжди бути підключеним під час наступних інсталяцій Software PLC для можливості його запуску. USB-ключ стандартно поставляється разом із Software PLC. Якщо USB-ключ не вставлений в роз'єм, буде функціонувати демо-версія Software PLC.

Монітор стану Software PLC. Монітор стану Software PLC відображається у вигляді значка в інформаційній області панелі завдань Windows. Цей монітор відображає робочий стан Software PLC.

Монітор стану (індикатор стану) програмного контролера Software PLC запускається автоматично, як тільки Software PLC був встановлений, а операційна система Windows запущена. Даний індикатор з'являється у вигляді значка на панелі завдань Windows (рис. 8.1).

Рисунок 8.1 - Індикатор стану Software PLC

Якщо даний монітор стану не відображається після ввімкнення промислового PC, його можна запустити використавши такий шлях (рис. 8.2):

«Пуск -> All Programs -> Phoenix Contact -> PC WORX RT BASIC -> PC WORX RT BASIC Status Monitor».

Інтерфейси Ethernet програмного контролера Software PLC. Один із мережевих адаптерів промислових PC на постійній основі призначається Software PLC і використовується в якості зовнішнього інтерфейсу для зв'язку через мережу Ethernet. Після інсталяції та призначення, цей інтерфейс доступний тільки для Software PLC. Під час інсталяції, віртуальний інтерфейс Ethernet також встановлюється і використовується для зв'язку між операційною системою INtime та Windows.

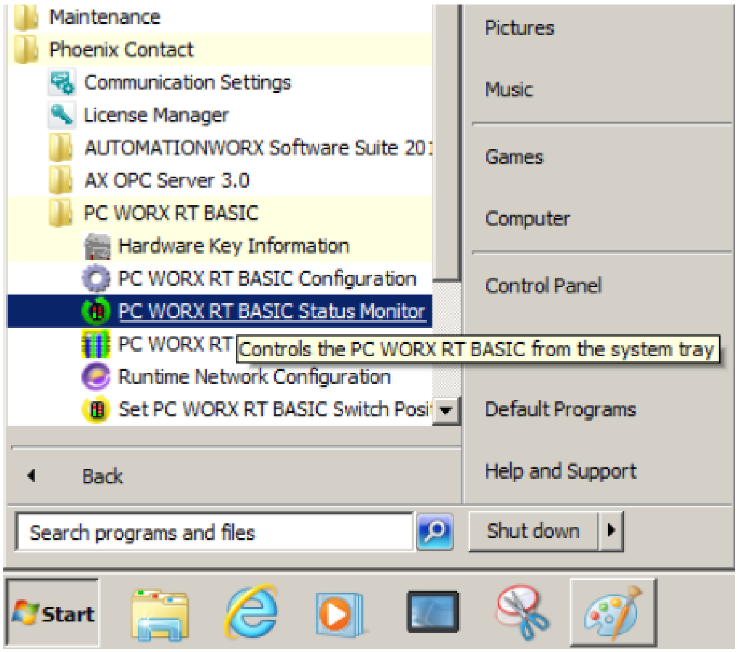
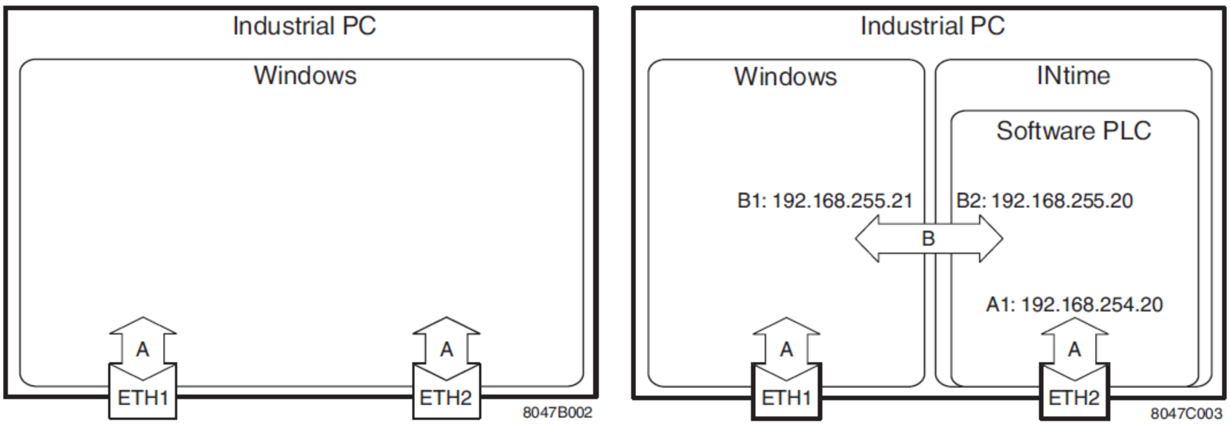


Рисунок 8.2 – Процедура запуску індикатора стану Software PLC

На рис. 8.3 наведено функціональну схему промислового PC із встановленою операційною системою Windows (ліворуч) та промисловий PC із встановленим Software PLC та операційною системою Windows (праворуч).



А - інтерфейси Ethernet промислового PC (A1 - фізична мережева карта); B - віртуальні інтерфейси Ethernet: B1 - віртуальна мережева карта / Windows, B2 - віртуальна мережева карта / Software PLC).

Рисунок 8.3 - Промисловий PC перед (зліва) та після (справа) інсталяції Software PLC

Ліва частина на рис. 8.3 показує, що перед інсталяцією Software PLC, обидва інтерфейси Ethernet промислового PC доступні в операційній системі Windows, як канали зв'язку. Права частина на рис. 2.3 показує, що після встановлення Software PLC:

- на промисловому PC паралельно до операційної системи Windows запускається операційна система реального часу INtime;

- Software PLC працює в операційній системі реального часу INtime;

- віртуальний інтерфейс Ethernet (інтерфейс B на рис. 2.3) був створений між Software PLC і операційною системою Windows.

- цей інтерфейс доступний тільки для внутрішнього зв'язку між Software PLC і Windows;

- цьому інтерфейсу за допомогою майстра установки, для прикладу, присвоюється IP адреса 192.168.255.20;

- програмне забезпечення PC WORX використовує цей інтерфейс, наприклад, для обміну даними з Software PLC, якщо на промисловому PC було встановлене програмне забезпечення PCWORX.

- один з інтерфейсів Ethernet призначається програмному контролеру Software PLC. Цей інтерфейс доступний тільки для Software PLC і не може використовуватись Windows. Під час інсталяції, IP адреса 192.168.254.20 (A1 на рис. 2.3) призначається майстром установки.

Пам'ять параметрування. Програмний контролер Software PLC використовує деяку пам'ять із промислових PC, як пам'ять параметрування (пам'ять програм, пам'ять для збереження даних). Ця пам'ять може бути використана для збереження програм і конфігурацій, що належать до створених проектів. Для визначення наявного обсягу пам'яті параметрування слід звернутися до технічних даних.

Пам'ять для проектних файлів. Залежно від використовуваного промислового PC, в процесі інсталяції можливо вказати де зберігати файли проекту - на внутрішньому жорсткому диску (HDD або SSD), чи на карті флеш-пам'яті CFast. В даний час, тільки один каталог під директорією "C:\\" може бути обраний у вікні вибору.

Інтерфейси, елементи індикації та елементи управління. Для отримання інформації про інтерфейси, елементи індикації та елементи управління, необхідно звертатися до документації конкретного пристрою промислового PC (рис. 8.4).

Description	Type	Order No.	Pcs./Pkt.
Industrial PC			
Data sheets for configurable industrial PCs, for example: VL2 BPC 2000 (order number: 2400332)	DB EN VL2 BPC 2000	–	1
PROFINET			
Quick start guide Installing and starting up the AXC 1050 PN starter kit	UM QS EN AXC 1050 PN STARTERKIT	–	1
User manual PROFINET basics	UM EN PROFINET SYS	–	1
PROFINET Installation Guideline for Cabling and Assembly, Version 1.0, January 2009, Order No.: 8.072 "PROFINET_Guideline_Assembly_8072_V10_Jan09.pdf"	For the latest versions of the document visit www.profinet.com or you can contact your nearest Phoenix Contact representative regarding the document		
Modbus			
Application note Modbus TCP with Phoenix Contact controllers	AH EN MODBUS TCP	–	1
INTERBUS (general)			
Data sheet Controller board for PC systems with PCI bus	DB GB IBS PCI SC/I-T	–	1
Driver Reference Manual for G4-based controller boards using PC bus and Ethernet	IBS PC SC SWD UM E	–	1
Quick start guide Installing and starting up the controller board for PC systems	IBS PCI SC QS UM E	–	1
User manual General introduction to the INTERBUS system	IBS SYS INTRO G4 UM E	–	1
User manual Configuring and installing INTERBUS	IBS SYS PRO INST UM E	–	1
User manual INTERBUS Diagnostics Guide	IBS SYS DIAG DSC UM E	–	1
User manual Firmware services and error messages	IBS SYS FW G4 UM E	–	1
User manual Peripherals Communication Protocol (PCP)	IBS SYS PCP G4 UM E	–	1
Documentation for software			
Quick start guide PC Worx	UM QS EN PC WORX	–	1

Рисунок 8.4 – Приклад документації на пристрої промислового PC

Інсталяція PC WORX RT BASIC Software PLC. «Phoenix Contact» рекомендує інсталювати і запускати PC WORX RT BASIC Software PLC на одному з промислових PC від фірми «Phoenix Contact». Крім того PC WORX RT BASIC Software PLC може встановлюватись і функціонувати на базі стандартних PC.

8.2 Програмний пакет PC WORX RT і установка його PC

Інсталяція PC WORX RT на PC. «Phoenix Contact» PC WORX RT – це програмний пакет, який дозволяє PC під керуванням операційної системи Windows виконувати функції контролера в реальному часі. Підключення сигнальних модулів відбувається з використанням систем PROFINET та INTERBUS. Мінімальними системними вимогами для установки даного програмного забезпечення є:

- процесор: Intel Core 2 Duo;
- оперативна пам'ять: 1 Гбайт;
- монітор з розширенням XGA (1024*768 пікселів);
- порти: Ethernet, USB;
- операційна система: Windows XP SP3, Windows 7, Windows 8.1;
- встановлене програмне забезпечення: (від PC Worx V6.30).

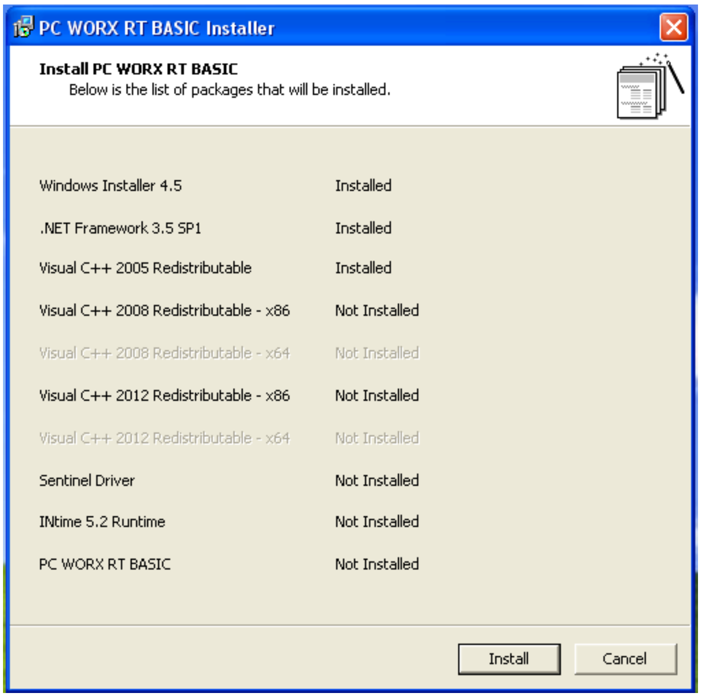


Рисунок 8.5 – Установка додаткових пакетів

Якщо є необхідність підключення комп'ютера, який виконуватиме функції контролера до локальної мережі або мережі

Internet – потрібно встановити додаткову мережеву плату, так як одну для своєї роботи вимагає пакет PC Worx RT. В разі необхідності підключення сигнальних модулів через PROFINET – потрібен комутатор, до якого підключатиметься мережева карта, яку використовує PC Worx RT.

Після завантаження пакету PC Worx RT з офіційного сайту «Phoenix Contact», необхідно запустити установку. Спочатку, програма установки запропонує вибрати мову інтерфейсу.

Далі програма установки запропонує встановити додаткові пакети, якщо вони відсутні на комп'ютері (рис. 8.5). Після активації кнопки «Install» почнеться установка.

Після завершення установки додаткових програм, необідно обрати шлях для установки і для даних.

Далі необхідно задати мережеві параметри. В списку мережевих карт обирається карта, яку виділяється для PC Worx RT і задається IP-адреса, маска підмережі, шлюз (рис. 8.6).

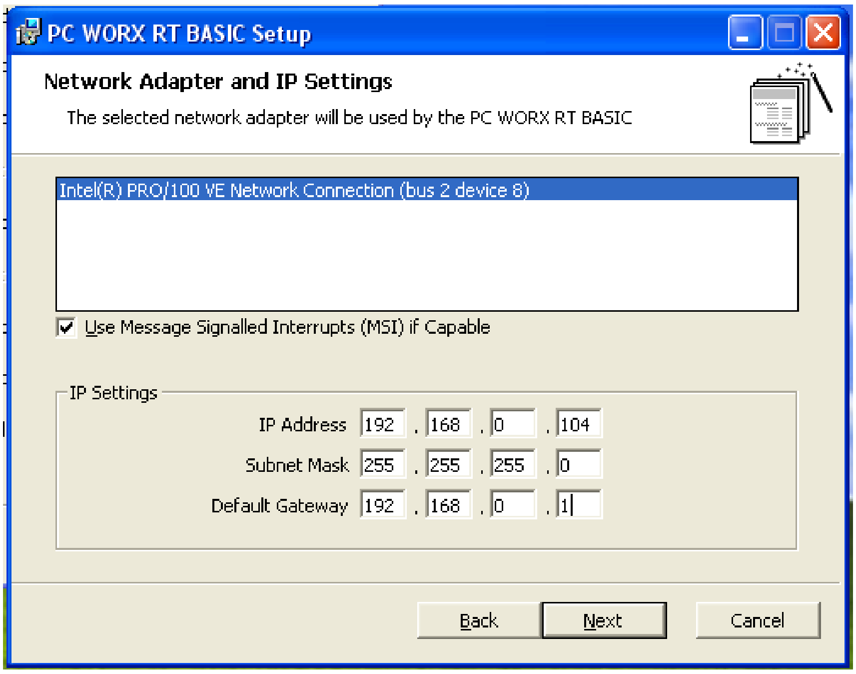


Рисунок 8.6 – Мережеві параметри Software PLC

В наступному вікні будуть відображені задані параметри (рис. 8.7). Далі активується кнопка «Install».

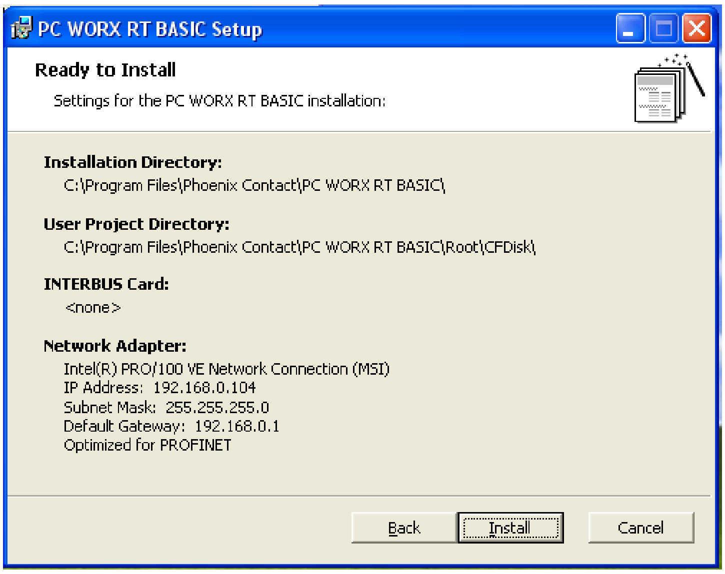


Рисунок 8.7 – Вікно із заданими параметрами інсталяції Software PLC

Після завершення установки виводиться запит на перезавантаження операційної системи (рис. 8.8).

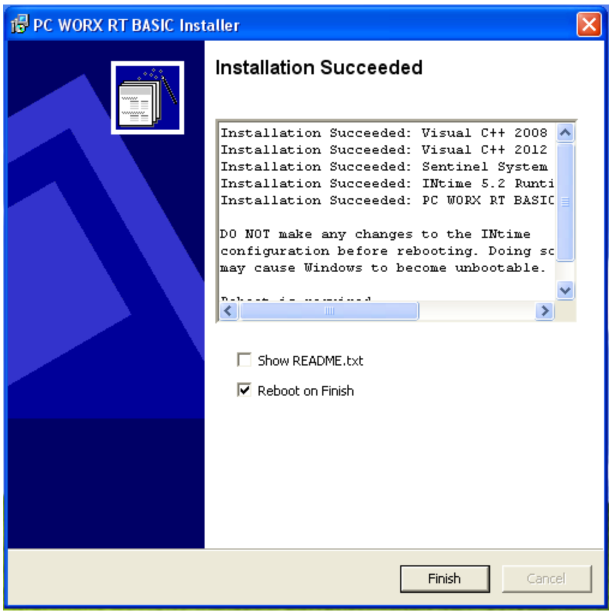


Рисунок 8.8 – Завершення установки Software PLC і додаткових програм

Після перезавантаження, на панелі задач виводяться піктограми, які пов'язані з роботою PC Worx RT. При активації правої кнопки маніпулятора «миш» виводиться контекстне меню (рис. 8.9).

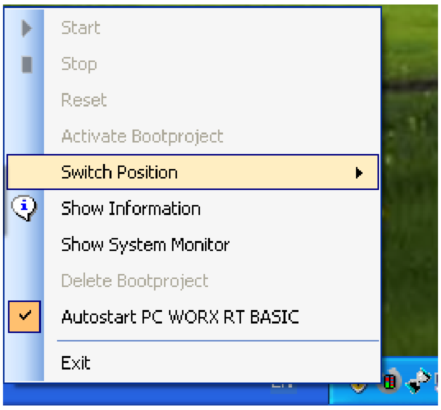


Рисунок 8.9 – Контекстне меню PC Worx RT Software PLC

Для роботи, необхідно в меню «Switch Position» вибрати пункт RUN/PROG. Після цього, якщо в контролер завантажено проект він почне виконуватись.

8.3 Створення проекту і запуск в PC WORX RT BASIC

Для створення проекту необхідно знати MAC-адресу мережевої карти і DNS-ім'я контролера. Для цього в контекстному меню PC Worx RT необхідно вибрати пункт «Show Information». У вікні будуть відображені всі необхідні для проекту дані (рис. 8.10)

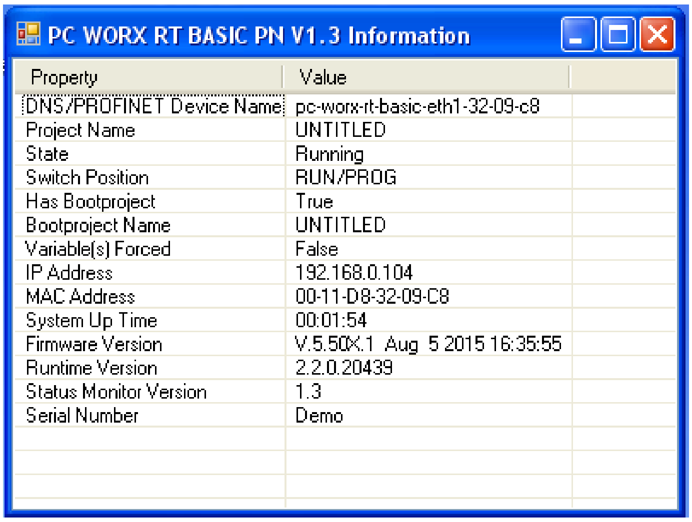


Рисунок 8.10 – PC Worx RT «Show Information»

Далі необхідно створити проект. Для цього, на іншому комп'ютері з установленим PC Worx V6.30 створюється новий проект. Для цього в меню «File» вибирати пункт «New». Далі вибирається тип контролера (рис. 8.11). В даному випадку це PC WORX RT BASIC PN V1.3 Rev.> 00/5.50.

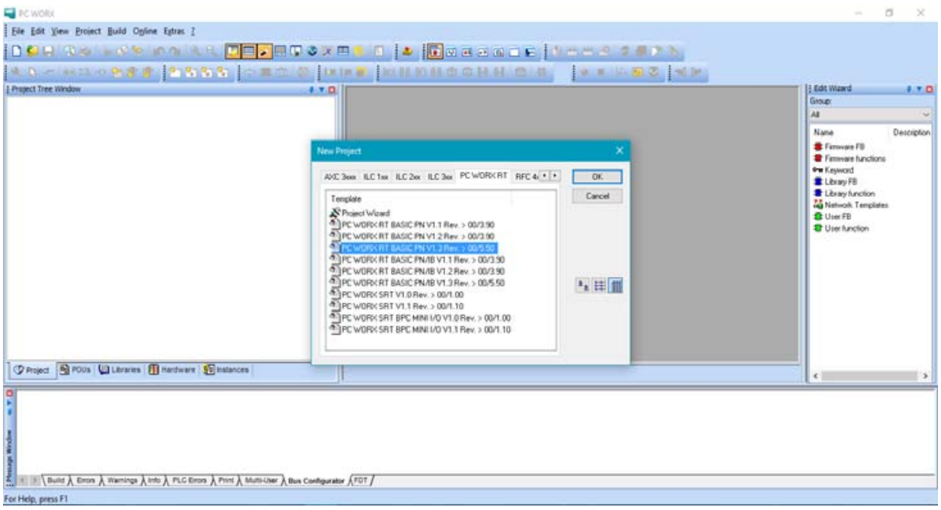


Рисунок 8.11 – Процедура створення нового проекту

Після цього, може виводитись повідомлення про необхідність оновити файли (рис. 8.12). Тому для створення проекту необхідно підтвердити.

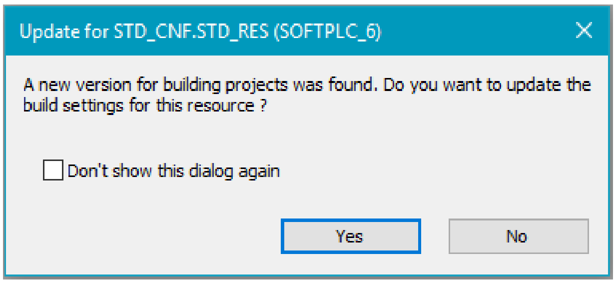


Рисунок 8.12 – Повідомлення про необхідність оновлення файлів

Далі задаються параметри з'єднання. Для цього в меню «View» вибирається пункт «Bus Configuration». У вікні в полях «First IP-address» і «Last IP-address» задається діапазон IP-адрес, в якому знаходиться Software PLC (рис. 8.13).

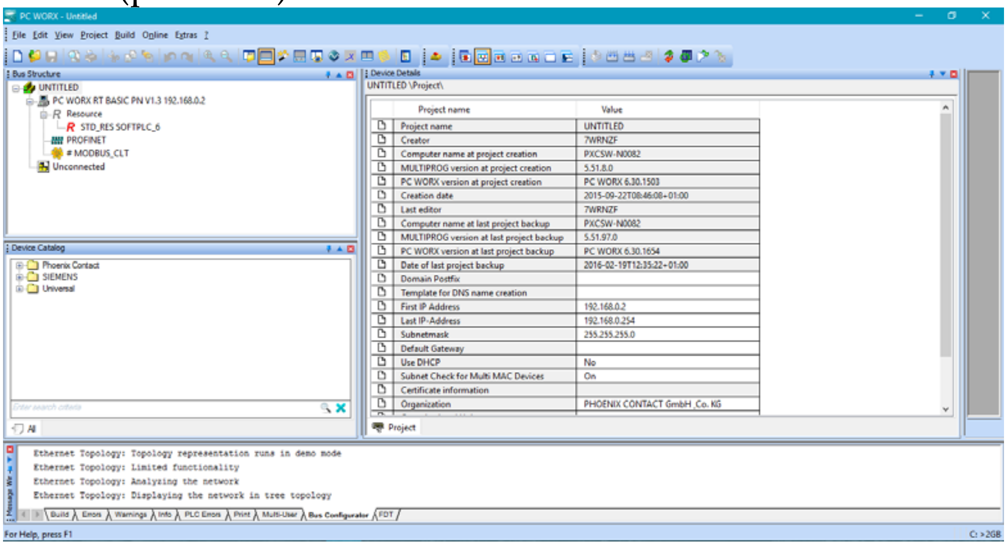


Рисунок 8.13 – Процедура задання діапазону IP-адрес для проекту

Далі виконується активація пункту з назвою контролера і вводиться IP-адреса Software PLC, MAC-адреса мережевої карти і DNS контролера (рис. 8.14).

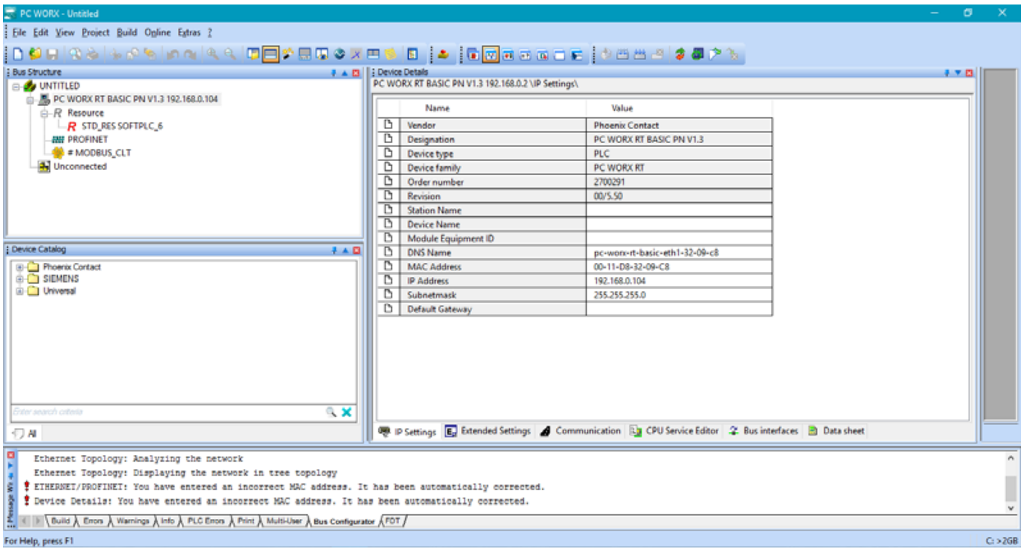


Рисунок 8.14 – Процедура параметрування Software PLC

Далі необхідно перейти на вкладку «Communication», вибирати інтерфейс «LAN1» і активувати кнопку «Test» (рис. 8.15). Після встановлення з'єднання поле над кнопками стане зеленим і виведеться тип Software PLC.

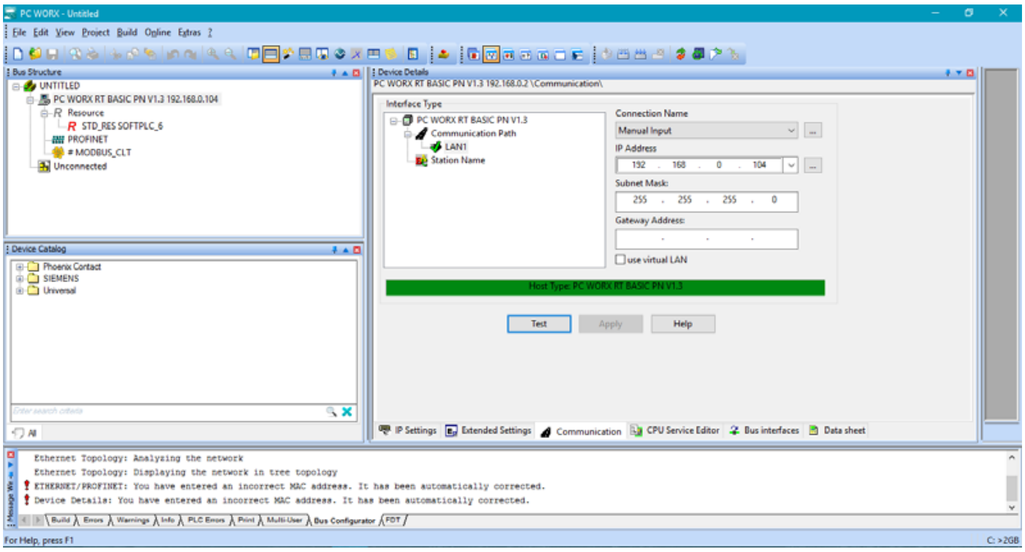


Рисунок 8.15 – Процедура параметрування комунікаційних параметрів Software PLC

Після виконання вищенаведених процедур є можливим створення управляючої програми для Software PLC. В меню «View» вибирається «IEC Programming». В панелі зліва відкрити папку

«Logical POUs», далі відкрити вкладку «Main» і в ній відкрити пункт «Main». Для створення тестової програми обрано блоки «TON» і «AND» (рис. 8.16). Далі виконуєтьмя редагування з'єднань входів і виходів програмних блоків та їх ініціалізації.

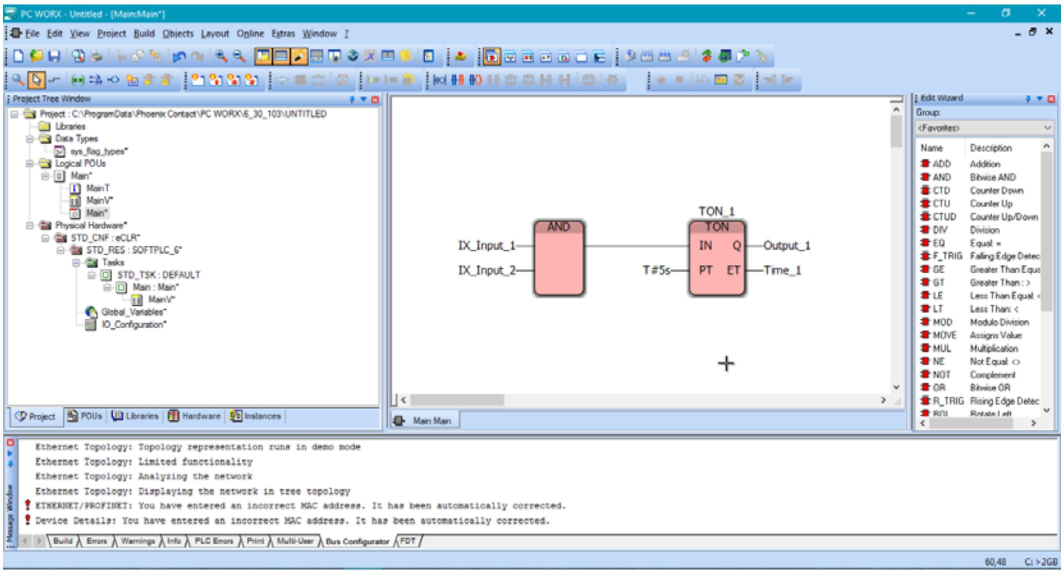


Рисунок 8.16 – Тестова програма для Software PLC

Назви входів у блоці «AND» і виходів «TON_1» є назвами змінних, які будуть задавати значення на виході. На вході «PT» блоку «TON_1» задано час 5 секунд, які будуть відраховуватись після появи двох одиниць на входах «AND». Після створення програми проект необхідно компілювати. Для цього, в меню «Build» вибирається пункт «Rebuild Project» (рис. 8.17).

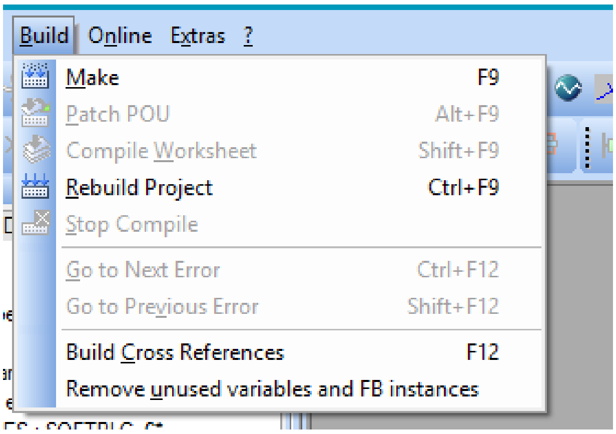


Рисунок 8.17 – Процедура компіляції проекту

Після компіляції, в меню «Online» вибирається пункт «Project Control». У вікні, що виводиться (рис. 8.18), активується кнопка «Download» для завантаження проекту в Software PLC. Після завантаження проекту активувати кнопку «Warm» для перезапуску Software PLC. Після цього почнеться виконання програми.

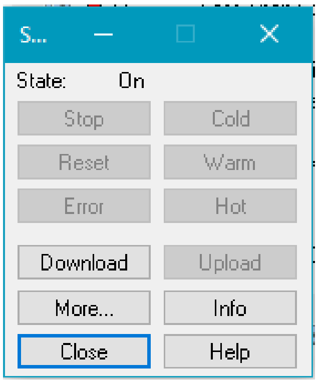


Рисунок 8.18 – Вікно керування Software PLC

Для переходу в режим «on-line» в меню «Online» необхідно вибрати пункт «Debug» для відображення значення змінних (рис. 8.19).

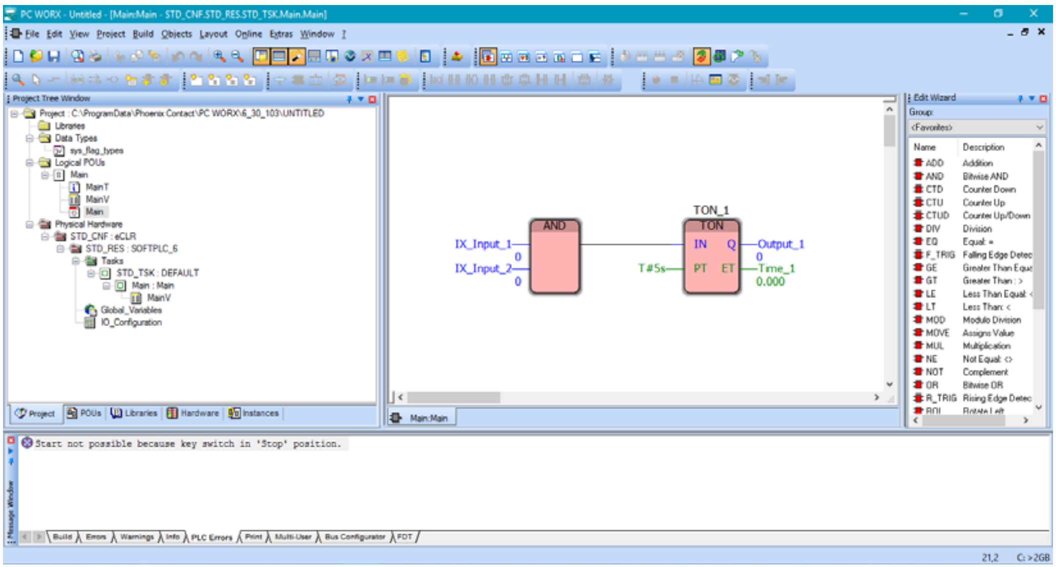


Рисунок 8.19 – Відображення значень змінних в режимі «on-line»

Для тестування програми активувати «IX_Input_1». Вибрати пункт «TRUE» і активувати кнопку «Overwrite» (рис. 8.20). Аналогічно повторити для входу «IX_Input_2».

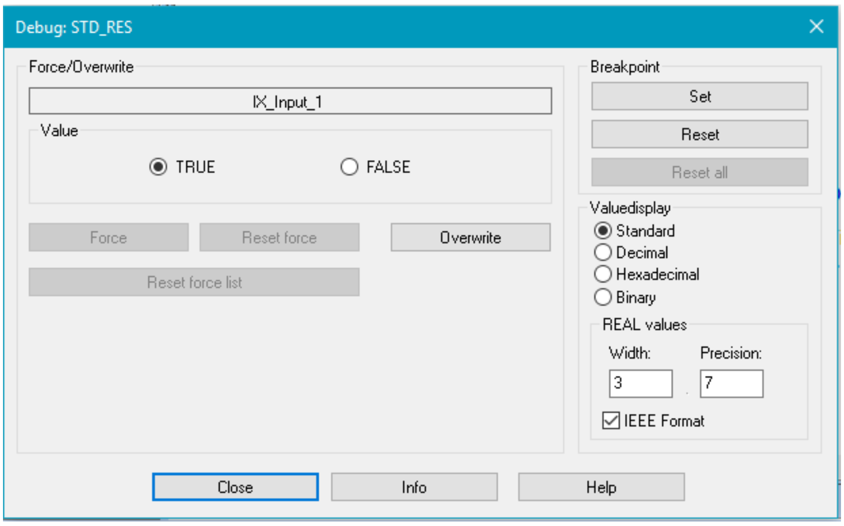


Рисунок 8.20 – Вікно параметрування входу програмного блоку

Після появи одиниць на входах програмного блоку «AND» таймер відрахує 5 секунд і встановлює на виході логічну одиницю. На виході «ET» буде виведено відрахований час (рис. 8.21).

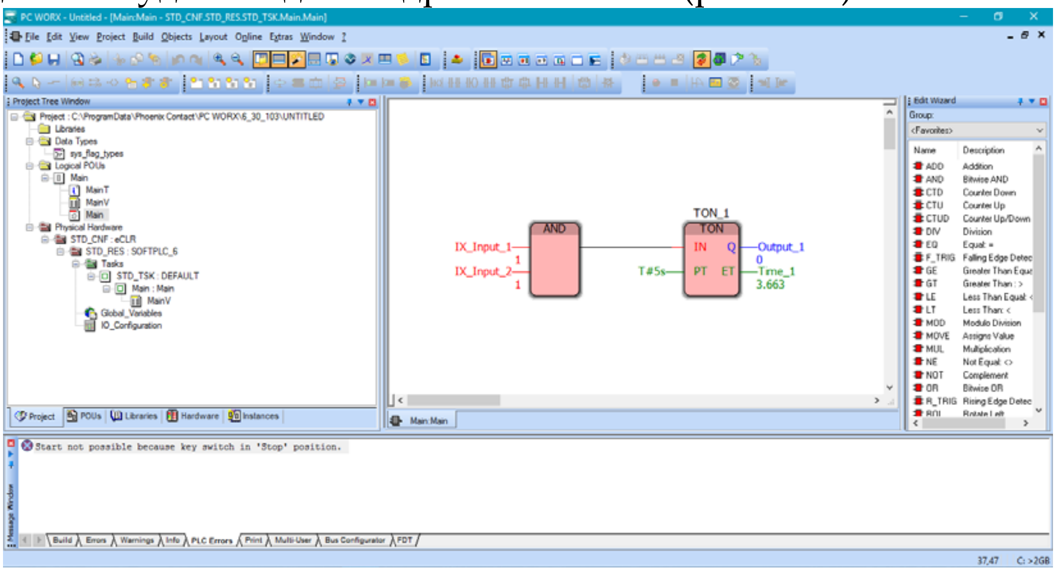


Рисунок 8.21 – Результат роботи тестової програми Software PLC

Таким чином, було створено проект, завантажено його в Software PLC і запущено в режимі «on-line» тестову програму.

8.4 Підключення сигнальних модулів до SOFTWARE PLC

Для роботи із сигнальними модулями разом з Software PLC, їх необхідно підключити фізично і в проекті. В меню «Extras» вибирається пункт «PROFINET Configuration». У вікні (рис. 8.22) вибирається мережева карта, через яку здійснюється зв'язок із Software PLC.

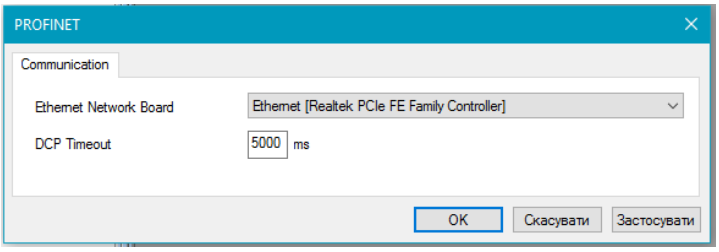


Рисунок 8.22 – Налагодження параметрів комунікації PROFINET

У вікні «Bus Configuration», активувати пункт «PROFINET» і вибирати пункт «Read PROFINET». Після сканування локальної мережі буде виведено список знайдених пристроїв (рис. 8.23). Можна виділити switch (комутатор) і активувати кнопку «Insert».

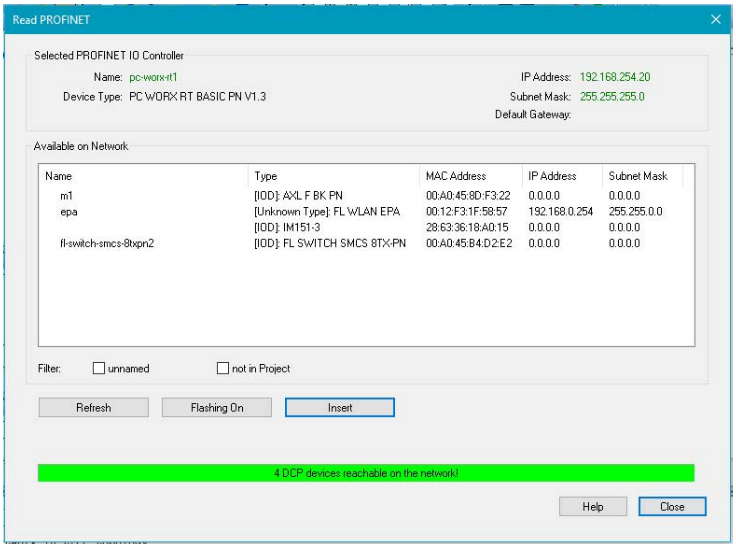


Рисунок 8.23 – Процедура пошуку пристроїв на мережі PROFINET

В списку пристроїв в підменю «PROFINET» відобразиться switch (комутатор) (рис. 8.24). В разі необхідності можна змінити його параметри.

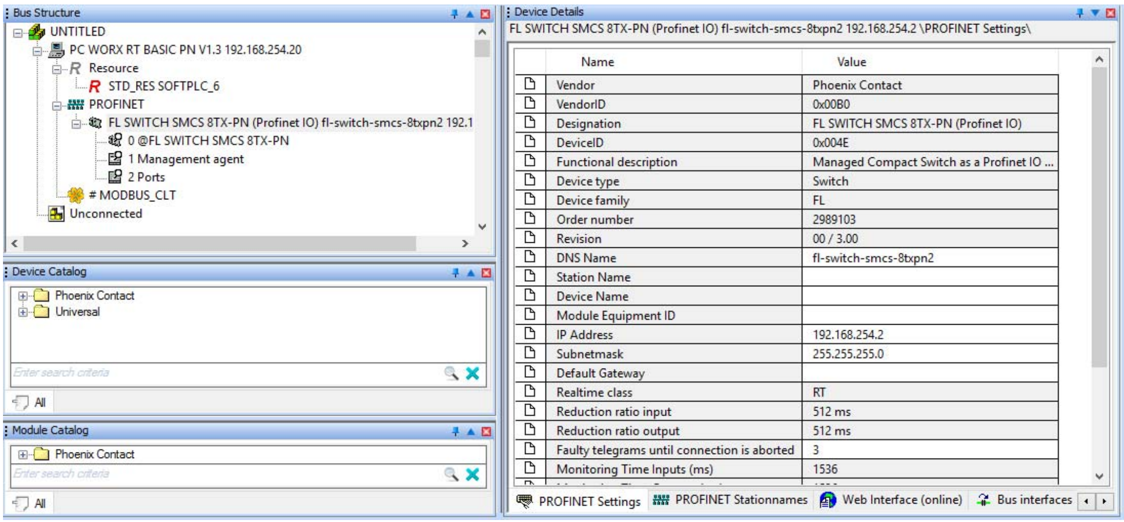


Рисунок 8.24 – Список підключених пристроїв в проекті з Software PLC

Далі необхідно в меню «View» вибирати пункт «Device Catalog». Виведеться додаткова панель з каталогом модулів (рис. 8.25).

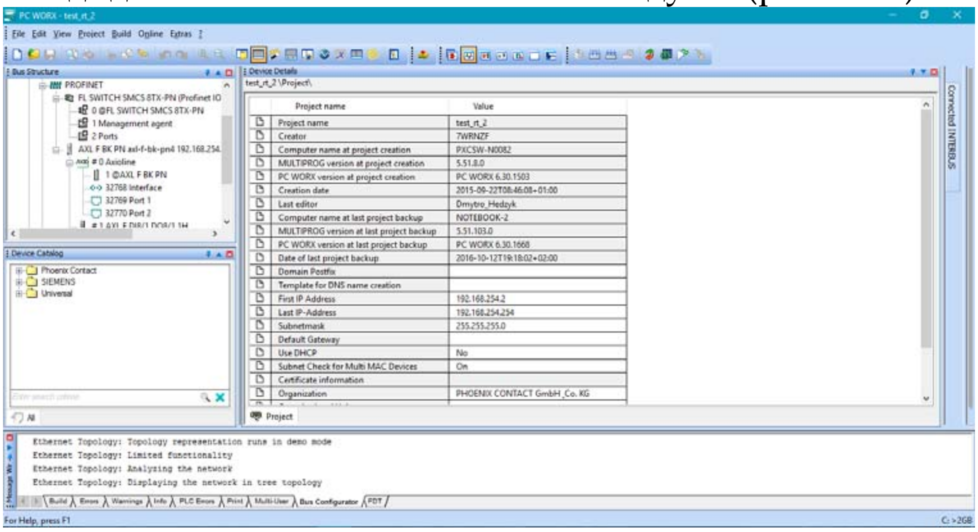


Рисунок 8.25 – Процедура підключення периферійних інтерфейсних і сигнальних модулів в проект Software PLC

Вибирається інтерфейсний модуль периферії (AXL F BK PN) в підменю «PROFINET». Далі вибираються сигнальні модулі (AXL F DI8/1 DO8/1 1H, AXL F AI4 U 1H і AXL F AO4 1H) в підменю «AXL F BK PN».

В разі необхідності підключення модулів інших виробників необхідно активувати панель «Device Catalog» і вибрати пункт «Import GSD file» (рис. 8.26). Потім необхідно вибрати файл, після цього почнеться його обробка, яка може зайняти тривалий час. Після завершення обробки в каталозі виводяться нові модулі.

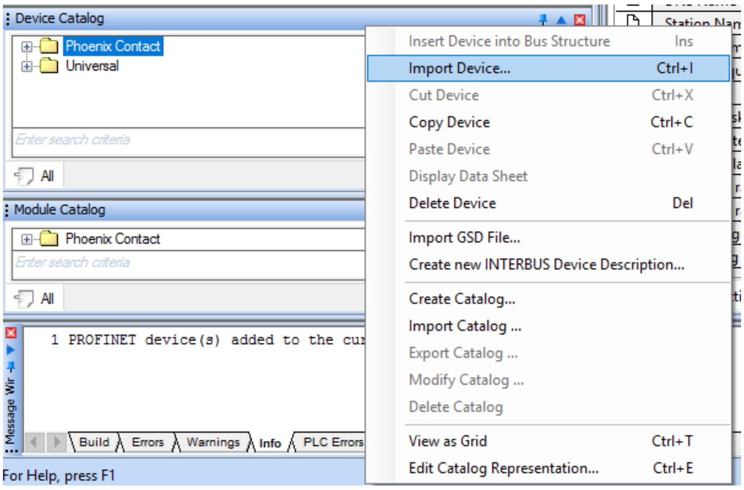


Рисунок 8.26 – Процедура імпорту GSD-файлу

Для роботи з фізичними сигнальними модулями, спочатку необхідно створити змінні для сигнальних модулів. Для цього виконується перехід на вкладку «Process Data Workspace» (рис. 8.27).

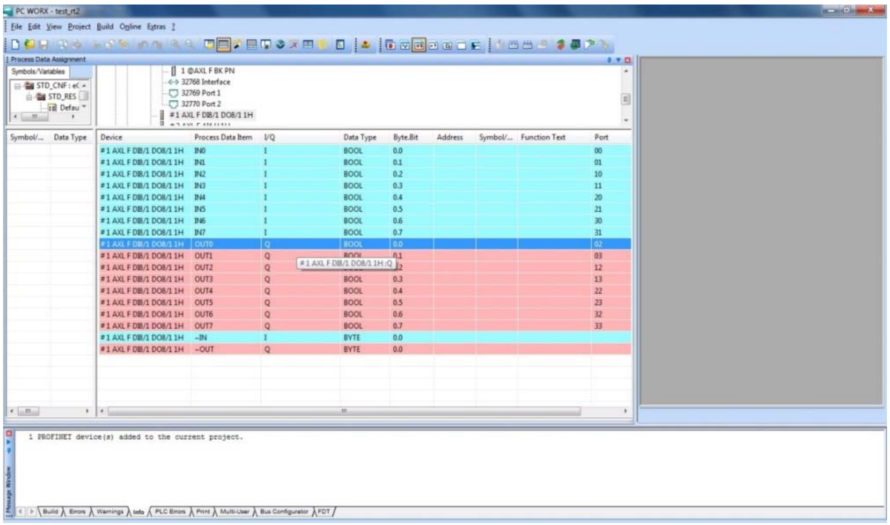


Рисунок 8.27 – Вкладка «Process Data Workspace»

На даній вкладці показані входи і виходи модулів. Створимо змінні активувавши правою кнопкою маніпулятора «миш» по рядку з входом (IN0-IN7) або виходом (Q0-Q7) і оберемо пункт меню «Create Variable». Після цього в колонці «Symbol/Variable» виведеться назва змінної (рис. 8.28).

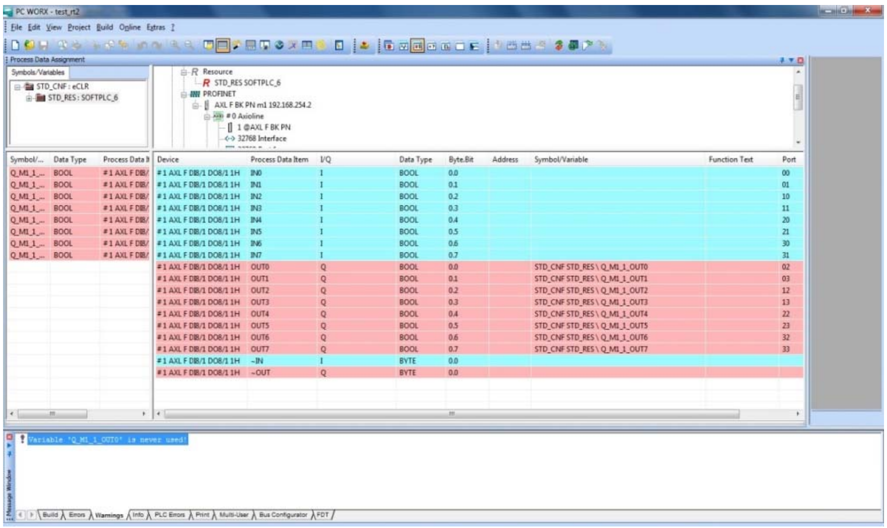


Рисунок 8.28 – Вікно «Process Data Workspace» після створення змінних.

Далі, для демонстрації, створено просту програму з одним блоком «MOVE», який пересилає значення змінної на вході на вихід.

Для виходу даного блоку можна обрати одну зі створених змінних, наприклад змінну «Q_M1_1_OUT0» (рис. 8.29).

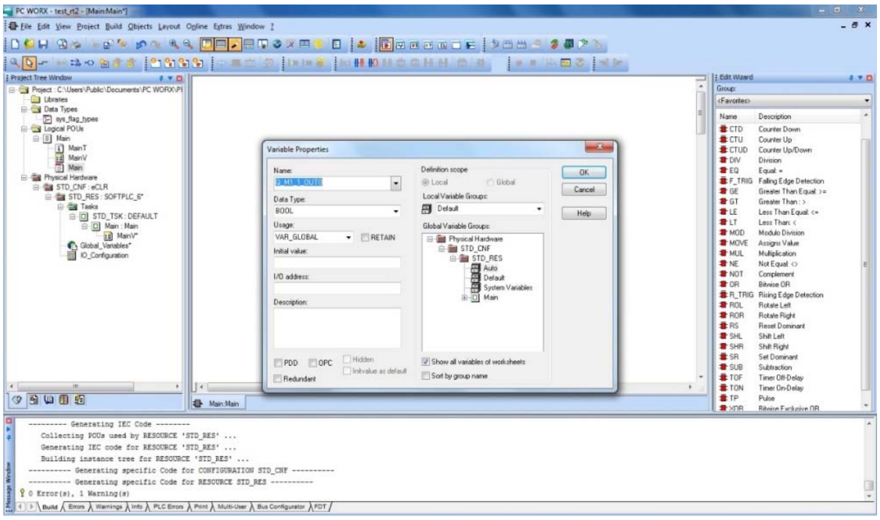


Рисунок 8.29 – Вибір змінної «Q_M1_1_OUT0» для роботи з виходом фізичного сигнального модуля

Приклад програми наведений на рис. 8.30.

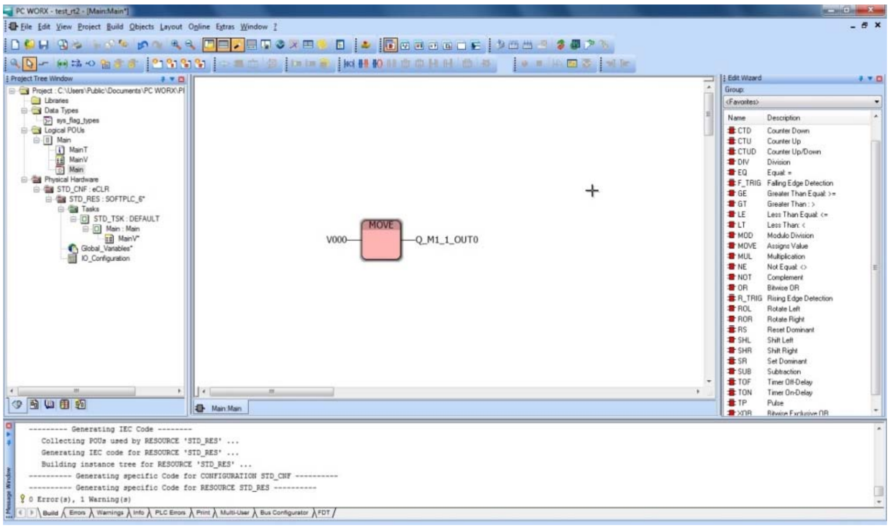


Рисунок 8.30 – Приклад програми для роботи з виходом фізичного сигнального модуля

Кожен вхід і вихід сигнального модуля має свою адресу. Адресу можна подивитись у властивостях відповідних змінних. Далі необхідно скомпілювати і завантажити програму в Software PLC. Для тестування програми призначити вхідній змінній значення «TRUE». На виході в режимі «on-line» виведеться логічна одиниця червоним кольором (рис. 8.31), а також відповідний індикатор на сигнальному модулі (рис. 8.32).

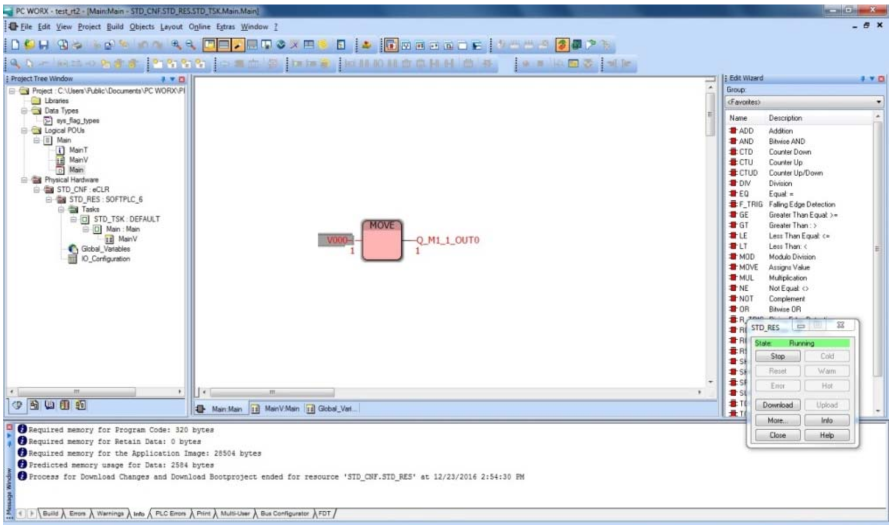


Рисунок 8.31 – Виконання програми в режимі «on-line» - логічна одиниця червоним кольором на виході блока «MOVE»

Рисунок 8.32 – Індикація логічної одиниці, на виході сигнального модуля (оранжевий індикатор), що відповідає змінній «Q_M1_1_OUT0»

Для більш наглядної демонстрації, створено програму, яка буде з заданим періодом послідовно змінювати стан кожного виходу фізичного сигнального модуля з логічного нуля на логічну одиницю і навпаки. Для цього використано додаткову бібліотеку «Moving_Light», яку підключено до проекту (рис. 8.33).

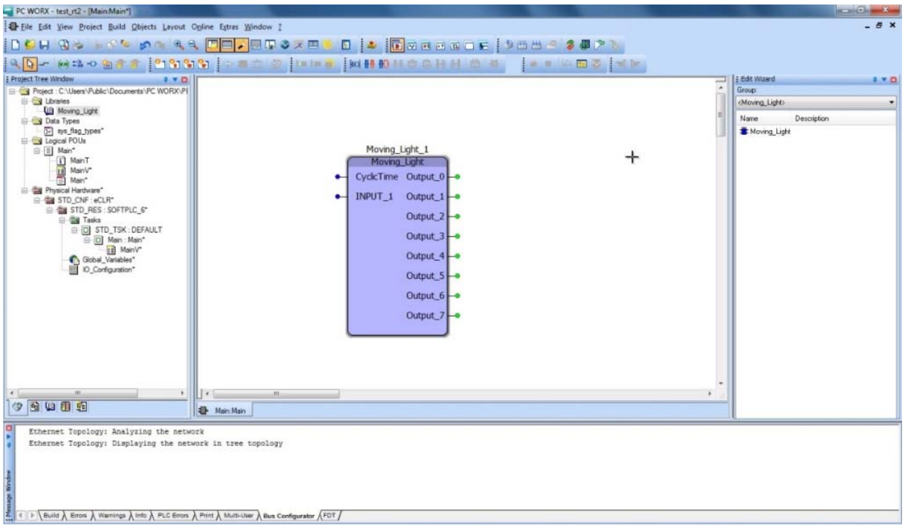


Рисунок 8.33 – Програма з використанням бібліотеки «Moving_Light»

Необхідно задати період і визначити вихідні змінні – виходи сигнального модуля, які виведені в списку змінних (рис. 8.34).

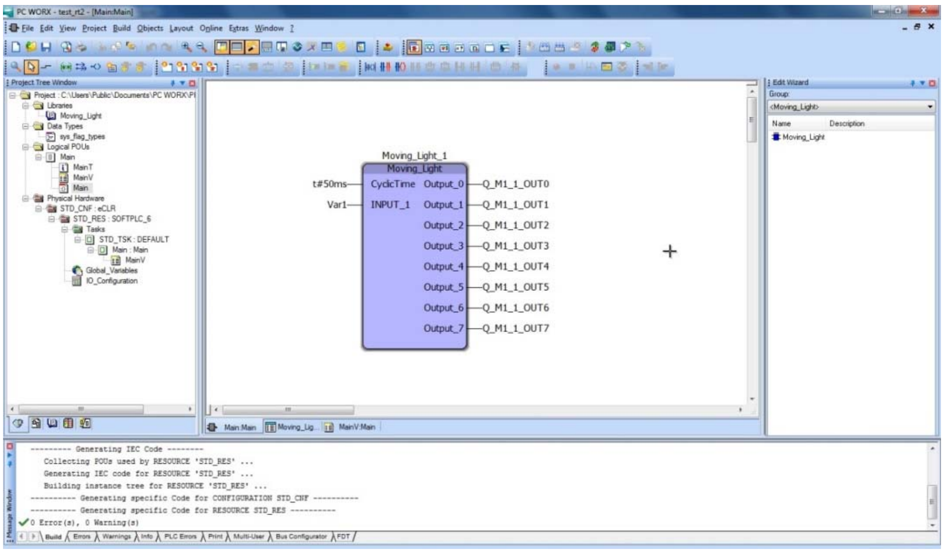


Рисунок 8.34 – Програма з використанням бібліотеки «Moving_Light» і вихідними змінними, асоційованими з фізичними виходами

Далі, для змінної «INPUT_1» встановлюється значення «TRUE». Відповідно на виходах, буде послідовно встановлюватись логічна одиниця (рис. 8.35).

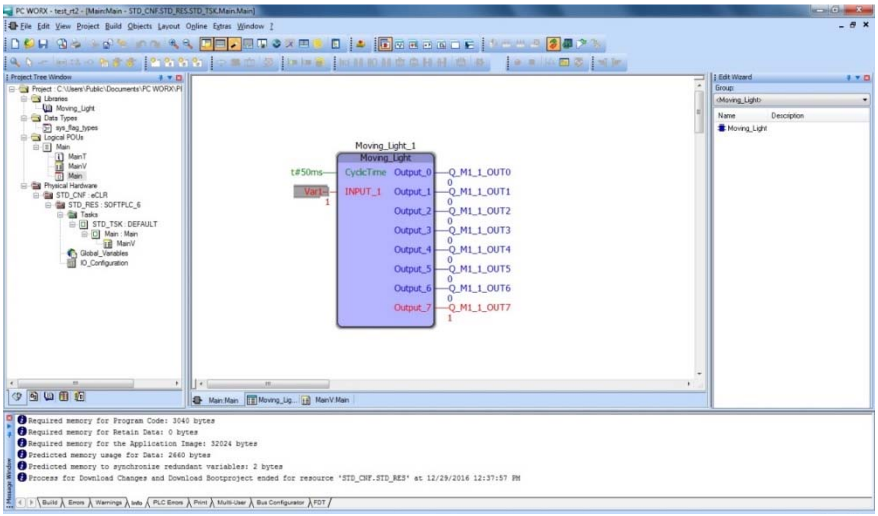


Рисунок 8.35 – Виконання програми «Moving_Light» в режимі «on-line»

Далі програму модифіковано для одночасної зміни станів двох виходів (парні в порядку зростання, непарні – в порядку спадання) (рис. 8.36).

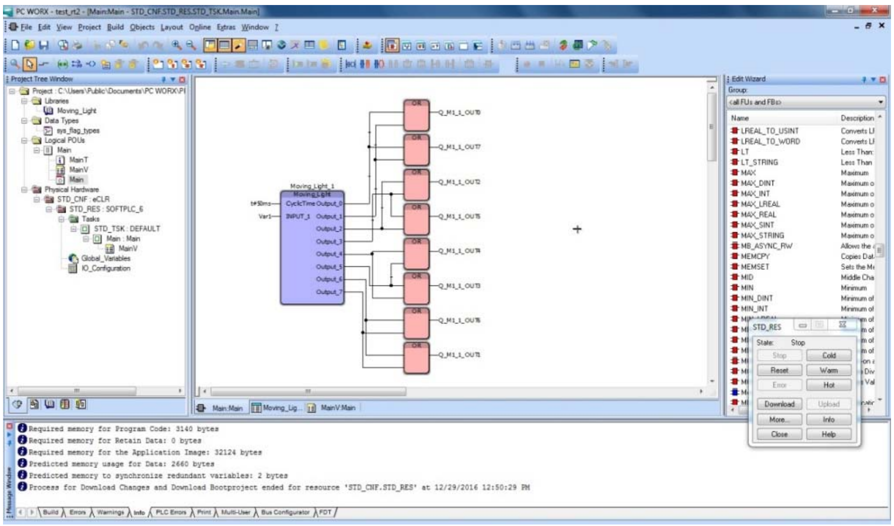


Рисунок 8.36 – Модифікована програма «Moving_Light»

Після завантаження модифікованої програми «Moving_Light» в Software PLC і її запуску, стани виходів в режимі «on-line» наведені на рис. 8.37.

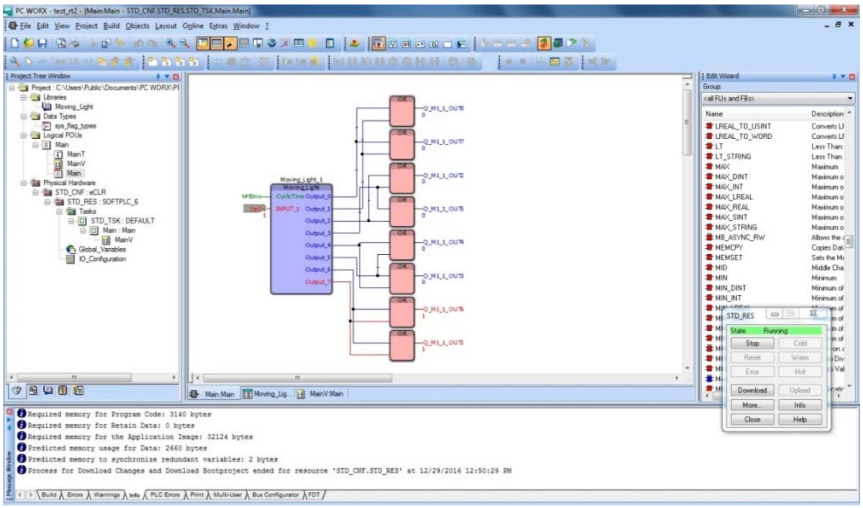


Рисунок 8.37 – Виконання модифікованої програми «Moving_Light» в режимі «on-line»

На рис. 8.38 наведено результа функціонування модифікованої програми «Moving_Light» на фізичних виходах сигнального модуля.

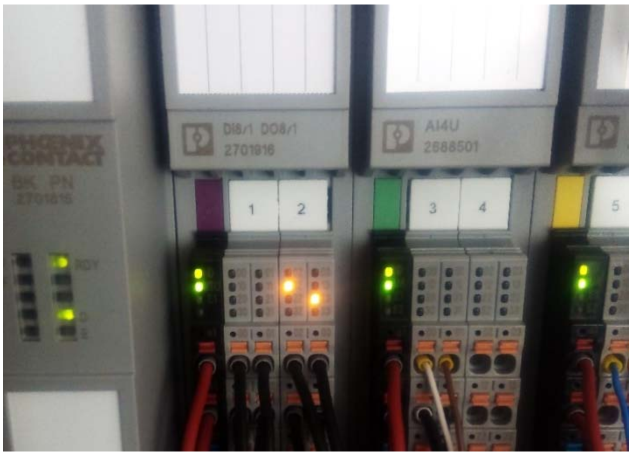


Рисунок 8.38 – Індикація логічних одиниць на виходах сигнального модуля (оранжеві індикатори) при виконанні модифікованої програми «Moving_Light» в Software PLC (PC WORX RT BASIC)

Таким чином створена методика і проведено тестування функціонування компонентів систем автоматизованого управління реального часу на базі PC WORX RT BASIC Software PLC.

Перевагами таких топологій систем управління є:

- можливість побудови систем управління без апаратних PLC на основі Software PLC, що функціонують на промислових і стандартних PC;

- можливість інтеграції даних системи управління в інші програмні пакети і бази даних на основі OS Windows;

- функціонування таких систем в реальному часі за рахунок розділення ресурсів операційної системи і апаратних засобів PC;

- доступність навчання програмування PLC поза межами спеціалізованих лабораторій за рахунок відсутності необхідності в апаратних PLC (PC WORX RT BASIC встановлюється і функціонує на стандартних PC).

Контрольні запитання:

1) Software PLC «PC WORX RT BASIC», його організація і основні технічні характеристики;

2) Комунікації, що підтримуються Software PLC «PC WORX RT BASIC»;

3) Організація і параметрування апаратно-програмних засобів Personal Computer (PC) для функціонування Software PLC «PC WORX RT BASIC»;

4) Мови програмування Software PLC «PC WORX RT BASIC» стандарту IEC 61131;

5) Основні параметри і процедура запуску «монітору стану» Software PLC «PC WORX RT BASIC»;

6) апаратні і віртуальні інтерфейси Software PLC «PC WORX RT BASIC»;

7) організація і параметрування ресурсів пам'яті Software PLC «PC WORX RT BASIC»;

8) Підключення сигнальних модулів «Phoenix Contact» до Software PLC «PC WORX RT BASIC»;

9) Підключення сигнальних модулів «Siemens» до Software PLC «PC WORX RT BASIC»;

10) Процедури створення змінних для сигнальних модулів у складі Software PLC «PC WORX RT BASIC»;

11) Призначення, імпорт та інсталяція GSD-файлів;

12) Процедура тестування управляючої програми в режимі «on-line».

РОЗДІЛ 9
СИСТЕМИ УПРАВЛІННЯ РЕАЛЬНОГО ЧАСУ НА БАЗІ Software PLC WinAC RTX «SIEMENS»

9.1 Компоненти, функціональне призначення та інструменти управління програмним контролером WinAC RTX

Simatic WinAC (Windows Automation Center) - це програмне забезпечення з підтримкою функцій S7-сумісних PLC в середовищі операційних систем Windows. Програмне забезпечення WinAC є додатком Windows і може використовуватися на комп'ютері оператора паралельно з іншими програмами. Ця особливість дозволяє одночасне вирішення завдань автоматизованого управління, візуалізації, комп'ютерної обробки та архівування даних, організації промислового зв'язку та ынш.

Усі контролери Simatic WinAC характеризуються:

1) Повною програмною сумісністю з PLC Simatic S7, можливістю використання єдиного набору інструментальних засобів проектування з можливістю завантаження програм WinAC в S7 і навпаки;

2) використанням стандартних інтерфейсів для інтеграції в різні комп'ютерні додатки;

3) наявністю відкритих інтерфейсів для підтримки спеціалізованої технологічної апаратури та програмного забезпечення;

4) наявність розширень для операційної системи Windows, що забезпечують підтримку роботи PLC d реальному масштабі часу;

5) використанням систем розподіленого вводу-виводу на основі мереж PROFINET IO і PROFIBUS DP для організації взаємодії з об'єктами керування.

Після запуску програми відкриється панель CPU WinLC, як показано на рис. 9.1 ця панель використовується для управління роботою WinLC. Для встановленя зв'язку STEP 7 з WinLC на тому ж самому комп'ютері потрібно відкрити інструментарій інтерфейсу «CPU> Setting the PG/PC Interface».

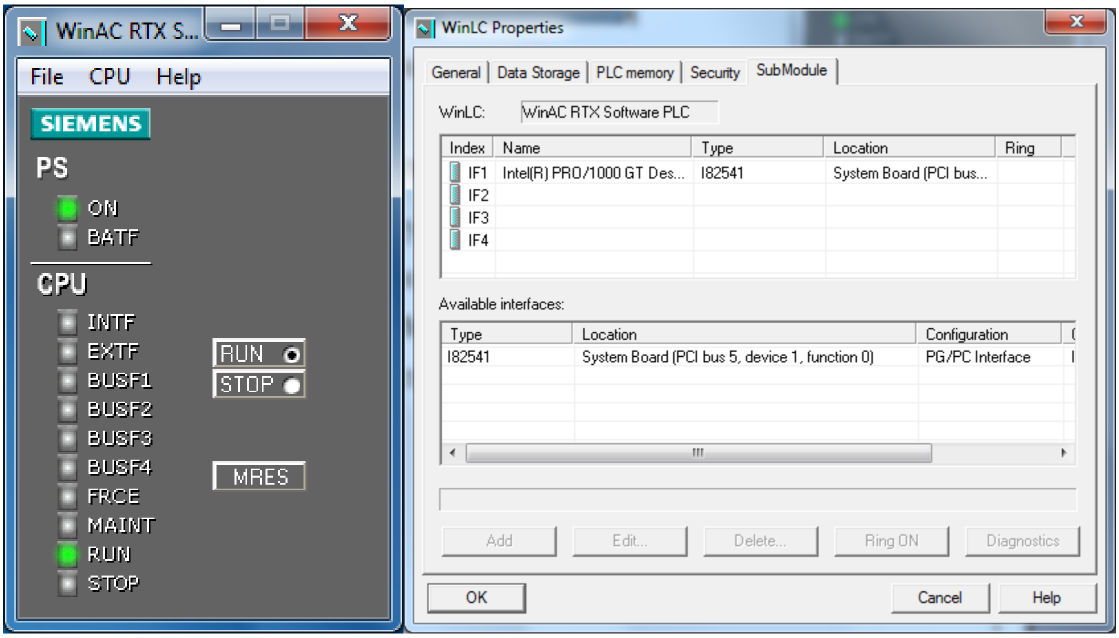


Рисунок 9.1 - Вікно налаштувань програмного контролера WinAC RTX

Запуск та зупинка WinAC RTX. Панель контролера є видимим інтерфейсом для контролера. Контролер може функціонувати, або бути вимкненим:

1) відкриття панелі запускає контролер;

2) вимкнення панелі (командне меню File > Exit) не вимикає контролер;

3) вимкнення контролера не закриває панель.

Коли контролер знаходиться в роботі, на панелі завдань Windows відображається піктограма WinLC RTX  незалежно від того, чи відкрита панель контролера. Піктограма буде підсвічуватися жовтим кольором, якщо контролер знаходиться в режимі «STOP» та зелений кольором в режимі «RUN».

Наступні параметри впливають на запуск або вимкнення контролера:

- вибір функції автозапуску;

- налаштування контролера для запуску після завантаження OS PC.

Якщо панель контролера не відкрита, необхідно застосувати один із наступних способів, для запуску WinLC RTX:

- необхідно вибрати Start > Simatic > PC Based Control. Потім необхідно обрати назву контролера WinLC. (Після завантаження STEP 7 ім'я користувача програми WinLC, збігається у STEP 7);

- необхідно активувати піктограму програми WinLC RTX для її запуску: .

Управління режимами роботи контролера WinLC RTX. Панель контролера забезпечує селекторний режим, що дозволяє змінювати режим роботи контролера. Режим позиції на контрольній панелі відповідає режиму позиції S7 апаратного контролера:

- «RUN»: контролер виконує програму STEP 7;

- «STOP»: контролер не виконує програму STEP 7. Виходи встановлюються в «захищені» стани.

Конкретні дії контролера дозволені або заборонені залежно від режиму роботи.

Для зміни режиму роботи контролера, необхідно застосувати один з наступних способів:

- переключення у режим «RUN» або «STOP»;

- вибір у командному меню CPU > RUN або CPU > STOP.

Режими роботи (RUN/STOP) та індикатори стану. Індикатори «RUN» і «STOP» показують поточний режим роботи контролера.

Якщо індикатор показує інший режим роботи, ніж положення селектора статусу, то це може бути викликано однієї з наступних дій:

- в контролері може бути змінений режим роботи через помилки в програмі;

- змінений режим роботи в STEP 7.

Режими роботи дозволяють або забороняють доступ до контролера для деяких типів операцій які показані у табл. 9.1.

Таблиця 9.1 - Режими роботи контролера WinLC RTX

Режим роботи	Опис
RUN	Дозволено: - завантаження програми з контролера до комп'ютера; - завантаження програми до контролера; - завантаження індивідуальних блоків програми у контролер; - використання STEP 7 для модифікації змінних та зміни режиму роботи контролера; - перезавантаження пам'яті з панелі управління або у STEP 7.
	Заборонено: - архівування та відновлення файлів STEP 7.
STOP	Дозволено: - Завантаження програми з контролера до комп'ютера або програмуючого пристрою; - завантаження програм або індивідуальних блоків до контролера; - використання STEP 7 для модифікації змінних; - перезавантаження пам'яті з панелі управління або у STEP 7; - архівування та відновлення файлів STEP 7.
	Заборонено: - використовувати STEP 7, для зміни режиму роботи у «RUN», якщо увімкнений режим «STOP» на панелі контролера WinLC RTX.

Налаштування параметрів контролера WinLC RTX.

Діалог властивостей WinLC. Налаштування у діалозі «WinLC Properties», які впливають на панель контролера або безпосередньо впливають на контролер.

Доступні такі методи, для відкриття діалогу «WinLC Properties»:

- опція 1:

1) відкриття Station Configuration Editor (редактор конфігурації станції);

2) переключення контролера в інший режим;

3) необхідно вибрати команду «Properties» в контекстному меню. Відкриється діалогове вікно, у якому вкладка «Submodul» активна.

- опція 2:

1) відкриття панелі контролера;

2) необхідно вибрати «CPU > Properties» в меню команд. Відкриється діалогове вікно, у якому вкладка «General» активна.

Можливі налаштування. Діалог «WinLC Properties» містить наступні вкладки, що містять наступні можливості для налаштування (табл. 9.2).

Таблиця 9.2 - Опис вкладок у діалозі властивостей

Вкладка	Налаштування
«General» tab (головна вкладка)	- запуск характеристик; - налаштування мови; - використання LED - екранів на PC; - запобігання зміни тактової частоти процесора.
«Data Starage» tab (вкладка зберігання даних)	відповідає за зберігання інформації
«PLC Memory» tab (вкладка пам'яті PLC)	шлях для зберігання програм STEP 7 і даних конфігурації
«Security» tab (вкладка безпеки)	дозвіл опцій захисту
«Submodule» tab (вкладка підмодуля)	налаштування підмодулів

Можна зробити наступні налаштування на вкладці «General» (рис. 9.2):

- запуск характеристик;

- налаштування мови;
- використання LED - екранів на PC;
- запобігання зміни тактової частоти процесора.

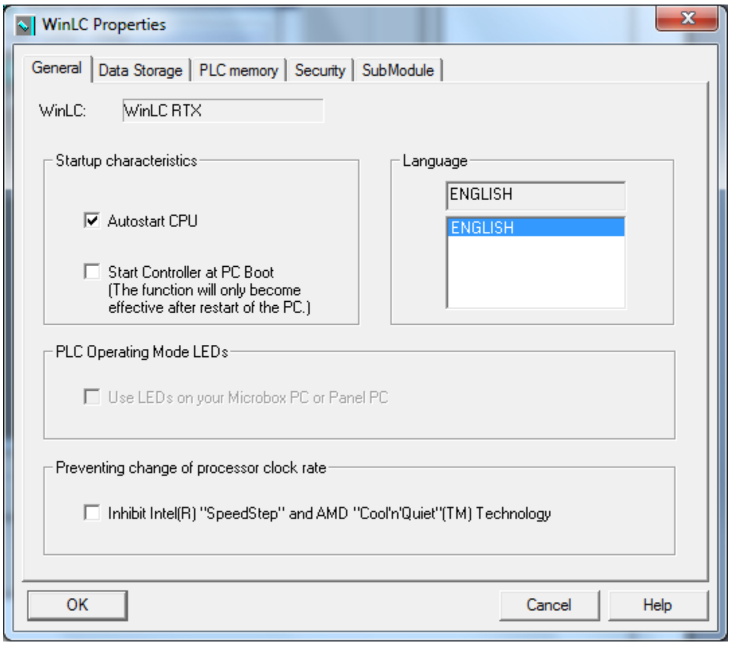


Рисунок 9.2 – Вкладка «General»

Основні налаштування WinLC RTX. За замовчуванням, необхідно запустити контролер вручну після завантаження PC. Однак, можна налаштувати контролер для автоматичного запуску під час процесу завантаження Windows, використовуючи логін користувача.

Вкладка «General». Варіант завантаження «Autostart CPU» вмикає контролер в тому ж режимі, в якому він був і перед виключенням. Після «синього екрану», контролер починає роботу у режимі в якому він працював до включення.

Якщо вибраний варіант «Start controller at PC boot», контролер починає працювати у режимі «STOP», при завантаженні PC.

Режим роботи індикаторів лицевої панелі WinLC «PLC Operating Mode LEDs» -

WinLC RTX буде індикувати світлодіодами для відображення режимів «RUN/STOP» або потенційну помилку, якщо буде вибрана дана опція і перезавантажений PC.

«Preventing change of processor clock rate» (запобігання зміни тактової частоти процесора) - функція енергозбереження.

За замовчуванням WinAC RTX відключає енергозберігаючу функцію «SpeedStep» (на комп'ютерах з процесорами Intel) і «Cool'n'Quiet» (на комп'ютерах з процесорами AMD). Загалом, WinAC RTX залишає вказані опції відключеними.

Ці енергозберігаючі функції економії енергії шляхом динамічної зміни процесу тактової частоти. Тактова частота зменшується при більш незначному завантаженні процесора «throttling» і знову зростає, якщо завантаження збільшується. Однак, динамічна зміна тактової частоти може призвести до тривалої затримки, що негативно впливає на продуктивність розширення RTX в режимі реального часу.

WinAC RTX налаштована таким чином, що якщо обраний «SpeedStep», то високопродуктивний процесор використовується на 100%, та завжди відображається в панелі настройки і диспетчері завдань.

Зміни опцій для WinLC RTX вступають в силу тільки після перезапуску WinLC RTX. Якщо WinLC RTX не вимкнений, необхідно встановити маркер, щоб перезавантажити контролер, після внесення змін. Якщо WinLC RTX вимкнений, зміни будуть активовані при наступному запуску контролера за допомогою команди «CPU > Start Controller».

Вкладка «Data Storage». Можна налаштувати наступні властивості на вкладці «Data Storage»:

- шлях для зберігання програм STEP 7 користувача і даних конфігурації;

- шлях для зберігання даних, для постійного зберігання.

Якщо NVRAM є доступною, то можна вказати, які дані повинні бути збережені на жорсткому диску або у NVRAM.

Зауваження:

При зміні шляху, WinLC більше не можуть отримати доступ до даних, які зберігаються у попередньому місці пам'яті. З цієї причини, необхідно архівувати програми STEP 7 користувача і дані їх конфігурації перед зміною параметрів зберігання даних. Можна відновити файл архіву після перезавантаження контролера з новими параметрами зберігання даних.

Налаштування параметрів пам'яті. Для налаштування зберігання даних, необхідно виконати наступне:

1) виконати архівування програми STEP 7 та її конфігурації;

2) вибати у меню «CPU > Properties»;

3) відкриття вкладки «Data Storage» (рис. 9.3);

4) прийняти за замовчуванням шлях у «Program and Configuration» або скористатися кнопкою , та вказати папку, в якій необхідно зберегти програму STEP 7 та її конфігурацію;

5) у полі «Retentive Data», необхідно прийняти шлях за замовчуванням або використати кнопку , для вибору шляху збереження даних;

6) необхідно вибрати між «NVRAM Storage» або «Hard Disk Storage» для зберігання даних, якщо комп'ютер має енергонезалежну пам'ять. В даному вікні відображаються об'єм доступної NVRAM-пам'яті. В іншому випадку, можуть бути обрані тільки «Hard Disk Storage»;

7) підтвердження за допомогою «OK». Після підтвердження, кнопка змінюється на «Confirm and restart controler».

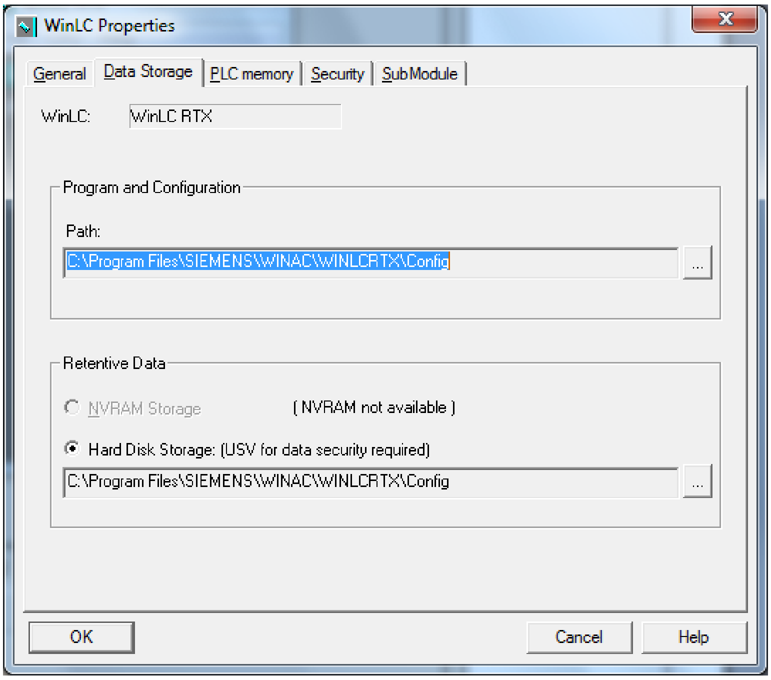


Рисунок 9.3 – Вкладка «Data Storage»

Необхідно переконатися у наявності незалежної пам'яті на к комп'ютері, перш ніж вибрати NVRAM для даних зберігання.

Зауваження:

Якщо активовано «Hard Disk Storage» для зберігання даних, то необхідно використовувати джерело безперебійного живлення UPS для збереження даних після збою живлення. Без UPS, дані в «Hard Disk Storage» втрачаються після відключення живлення.

Якщо активовано «NVRAM storage», використання UPS рекомендується у наступних випадках:

- інші програми, такі як бази даних та архіви повинні також бути збережені у разі відключенні живлення;

- управління Windows XP або Windows XP Embedded Standard з Enhanced Write Filter повинно бути вимкненим. Існує ризик того, що операційна система вийде з ладу.

Не треба виконувати стиснення файлів для системи через «Program and Configuration». Щоб визначити, чи дійсно файли стиснені, необхідно відкрити властивості диска, на якому дані файли зберігаються. Необхідно переконатися, що маркер «Compress drive to save disk space» не обраний (рис. 9.4).

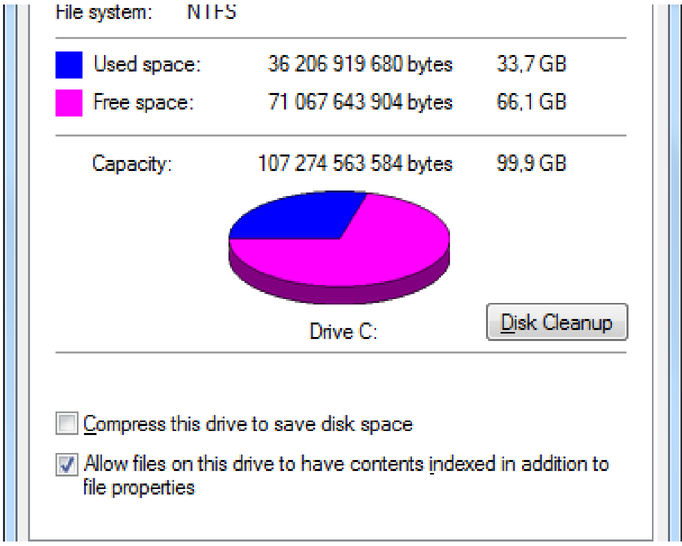


Рисунок 9.4 – Властивості диска на якому зберігаються файли

Вкладка «PLC Storage». На вкладці «PLC Storage», заданий розмір проекту STEP 7 відображається у полі «Memory Size» (рис. 9.5). Це рекомендований максимальний розмір проектів STEP 7, які можуть бути завантажені у WinLC.

WinLC поділяє загальний розмір проекту STEP 7 на наступні компоненти:

- розмір коду (наприклад, для організаційних блоків OB);
- розмір даних (наприклад, для блоків даних DB).

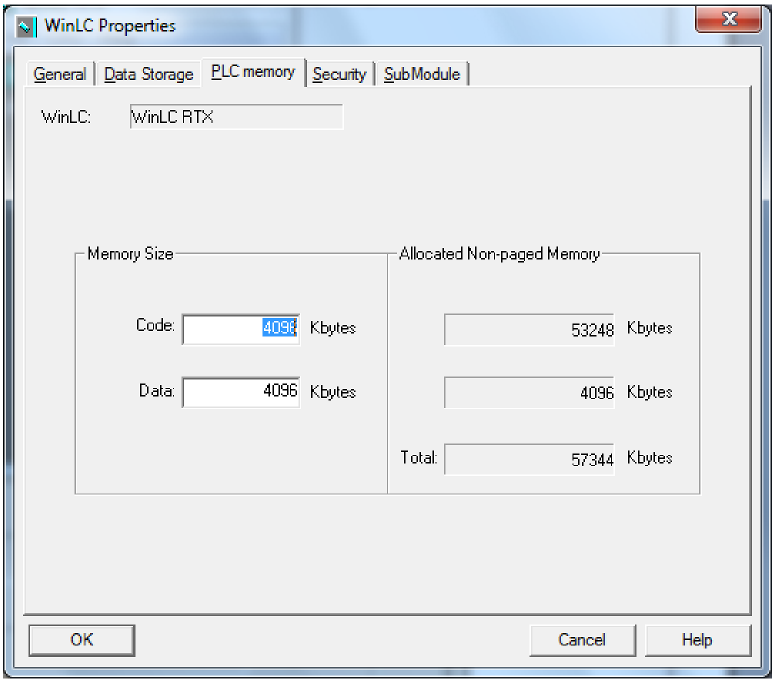


Рисунок 9.5 – Вкладка «PLC Memory»

Завантаження великих проектів STEP 7. Якщо необхідно завантажити проект STEP 7, в якому один з компонентів перевищує встановлений розмір, завантаження буде скасовано. Рекомендований розмір максимального об’єму можна збільшити вручну.

Вкладка «Security». Можете використовувати меню «CPU > Properties» у вкладці «Security» для зміни налаштувань захищеного доступу.

Для зміни параметрів захищеного доступу, необхідно виконати наступні дії:

1) активувати вкладку «Security» (відкривається діалогове вікно «Access Verification») (рис. 9.6);

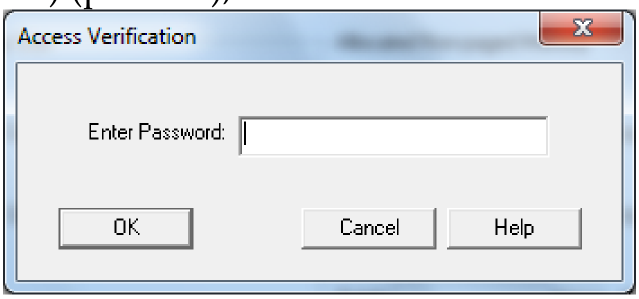


Рисунок 9.6 – Вікно діалогу «Access Verification»

2) необхідно ввести пароль. За замовчуванням пароля немає;
3) активувати кнопку «ОК» (рис. 9.7);
4) необхідно налаштувати захищений доступ до контролера;
5) активувати кнопку «ОК».

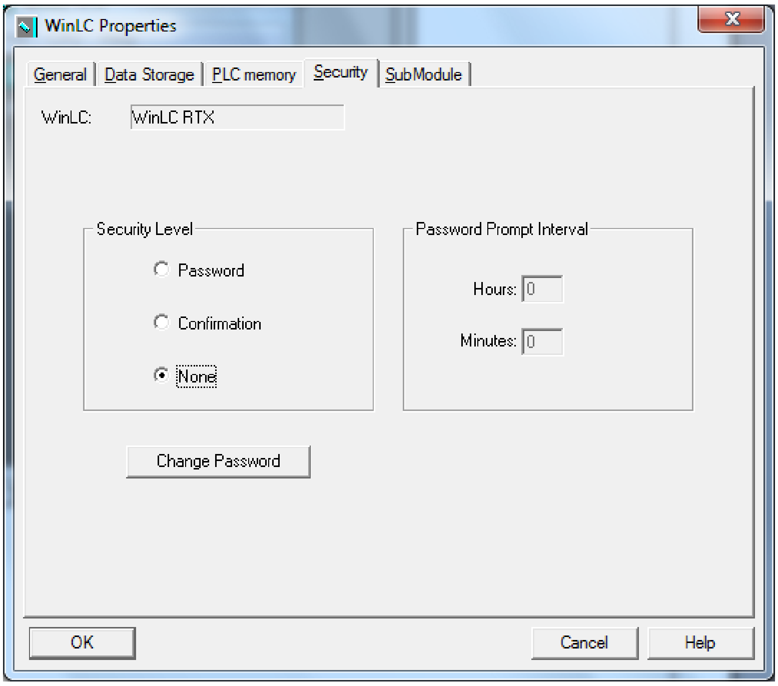


Рисунок 9.7 – Активна вкладка «Security»

Рівень захисту. На вкладці «Security», можна налаштувати рівень захисту паролем, який буде обмежувати доступ до контролера. Надаються наступні параметри безпечного доступу:

- пароль: при виборі «Password», деякі операції панелі контролера, такі як зміна режиму роботи, архівування та відновлення користувацької програми STEP 7, вимагають від користувача вводу пароля;

- підтвердження: при виборі «Confirmation», керуючий режим вимагає, від користувача підтвердження.

- none: При виборі «none», не потрібно підтвердження або пароля.

Інтервал запиту пароля. Можна встановити пароль, який буде дійсним на протязі заданого інтервалу часу, від 0 до максимум 23 годин, 59 хвилин. Після закінчення вказаного інтервалу необхідно знову вводити пароль. За замовчуванням «0» означає, що необхідно ввести пароль для кожної захищеної операції.

Виключення і запуск контролера не впливає на закінчення дії пароля протягом вказаного інтервалу, однак він обнуляється, якщо вимкнути панель контролера. При наступному запуску панелі контролера і доступу до захищених паролем операцій, буде здійснений запит на введення пароля.

Зміна пароля. Необхідно активувати кнопку «Change Password» для відображення діалогового вікна «Change Password». Діалог «Change Password» дозволяє змінити поточний пароль.

Процедура:

Необхідно застосувати наступну процедуру для зміни паролю:

1) необхідно ввести попередній пароль у поле «Old Password»;

2) у «New Password» необхідно ввести новий пароль (максимальна довжина 12 символів);

3) у вікні «Confirm New Password» необхідно підтвердити новий пароль;

4) активувати кнопку «OK».

Результат:

Налаштований пароль для доступу до операцій панелі контролера, такі як зміна режиму роботи, архівування та відновлення користувацьких програм STEP 7. Для того, щоб згодом мати доступ до параметрів безпеки, необхідно ввести пароль у полі діалогового вікна «Access Verification».

Управління контролером без пароля:

Якщо пароль створений, але рівень безпеки встановлений на «None» (відключений пароль), то як і раніше необхідно ввести налаштований пароль, перш ніж можна буде отримати доступ до діалогового вікна «Security».

Вкладка «Submodule». На вкладці «Submodule» відображається поточна конфігурація чотирьох слотів інтерфейсу для підмодулів WinLC RTX і список доступних мережевих карт, які можуть бути налаштовані як підмодулі. У цій вкладці можна додавати, редагувати або видаляти з'єднання мережевих карт, як підмодулів WinLC RTX. Можна перевірити підмодуль призначення Intel PRO/1000GT і відображення діагностичних даних (рис. 9.8).

Зауваження:

WinLC RTX підтримує одну карту CP 5611/21 або одну інтегровану CP 5611/21 PROFIBUS-інтерфейсу, як під модуль. Максимум один інтерфейс Industrial Ethernet (CP 1616, CP 1604, або IE General, наприклад Intel PRO/1000GT), як підмодуль, і не більше чотирьох підмодулів в загальній кількості. З чотирьох підмодулів, будь-який може бути комунікаційним інтерфейсом CP 5613. Можна налаштувати будь-яку комбінацію інтерфейсів зв'язку в чотирьох слотах з урахуванням вказаних обмежень.

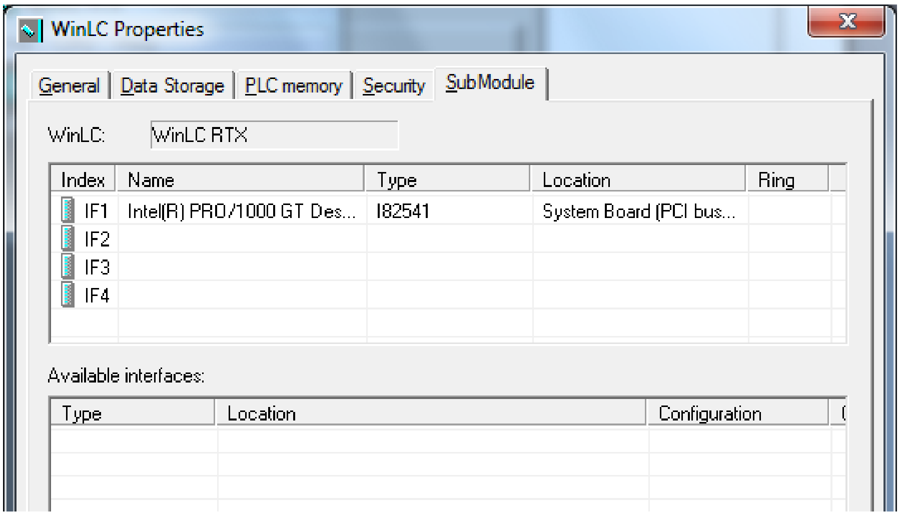


Рисунок 9.8 – Налаштування на вкладці «Submodule»

Доступні функції, при вимкненому WinLC RTX:

Доступні такі функції у вкладці «Submodule» при закритому WinLC RTX:

- Add: кнопка «Add» використовується для того, щоб вибрати мережеву карту зі списку доступних карт і вставити її у вільний слот слот. Мережева карта налаштована як підмодуль WinLC RTX вставляється у перший вільний слот інтерфейсу;

- Edit: можна вибрати слот і активувати кнопку «Edit» для зміни призначення слоту і налаштування DP інтерфейсу або назви. Можна використовувати клавіші стрілок на клавіатурі вгору/вниз для переміщення підмодуля до інших слотів;

- Delete: можна вибрати будь-який слот і активувати кнопку «Delete», щоб видалити налаштуваня мережевої карти як підмодуля WinLC RTX. Мережева карта, яка вилучена зі слоту, буде відображатися, як доступна на нижній панелі. Після вказаних дій це уже не підмодуль WinLC RTX. Клавіша «Del» на клавіатурі має таку саму функцію як і «Delete» в даному діалозі.

Доступні функції, при працюючому WinLC RTX:

Наступні функції є доступними у «WinLC Properties» для CP 5613. Якщо WinLC RTX є включеним - наступні кнопки не доступні для CP 5611/21 і для адаптерів Industrial Ethernet (Intel PRO/1000GT):

- «Ring On»: можна вибрати будь-який слот і активувати кнопку «Ring On», для перевірки налаштування CP 5613. WinLC RTX потребує, щоб слот PCI на комп'ютері та налаштуваний слот не конфліктували. Функція «Ring On» дозволяє перевірити, чи є налаштований CP 5613 для слота, який відповідає фактичному слоту комп'ютера. У ході цього випробування з функцією «Ring On», світлодіоди на налаштованому CP 5613 мигають. Комп'ютер також видає звуковий сигнал, якщо CP функціонує. Необхідно активувати кнопку «Ring Off» у діалоговому вікні «WinLC Properties» до кінця тестування. «Ring ON» не доступний для CP 5611/21;

- «Diagnostics»: можна вибрати слот, який займає CP та активувати кнопку «Diagnostics» для відображення інформації про зв'язок CP. Діалогове вікно «Submodule Network Diagnostics» відображає поточну версію вибраного CP і параметри шини.

Можна також відображати всі вузли в мережі зв'язку та статус кожного вузла. Необхідно активувати кнопку «Update» для відкриття цього діалогу. Даний діалог не є доступний постійно, оскільки запити кожного вузла створюють додаткове навантаження на мережу (рис. 9.9).

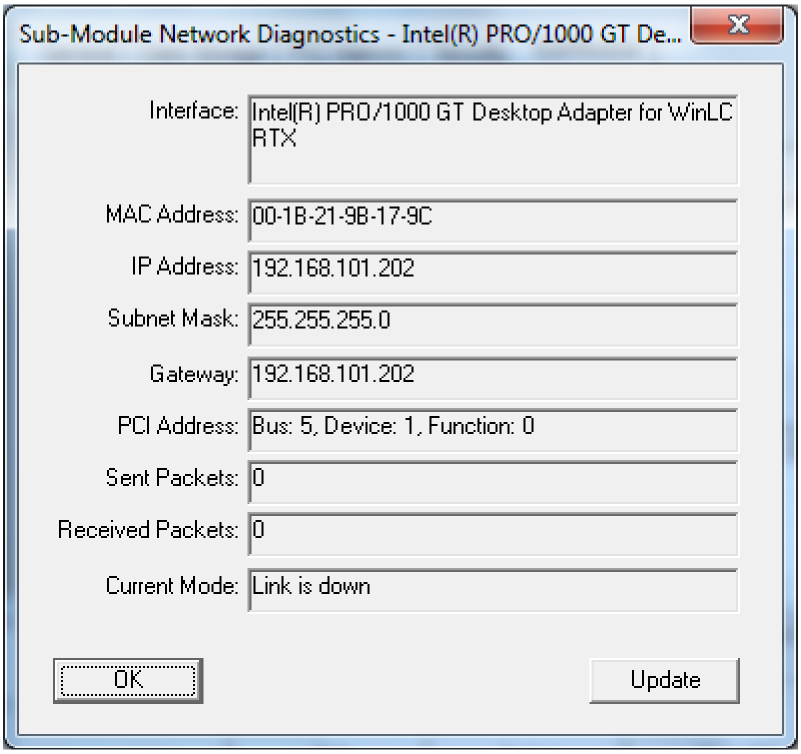


Рисунок 9.9 - Підмодуль для діагностики мережі

Перезавантаження пам'яті контролера WinLC RTX. Перезавантаження пам'яті відбувається шляхом скидання контролером свого первинного (за замовчуванням) стану. Перезавантаження видаляє користувацькі програми STEP 7 і системні дані (налаштування), а також відключає будь-які онлайн-комунікації.

«MRES» (меню процесора)

Як правило, виконується команда «MRES» перед завантаженням нової програми в контролер. Необхідно виконати перезавантаження пам'яті, коли індикатор «STOP» на панелі контролера повільно індику, це може свідчити про наступні умови:

- у робочій пам'яті були виявлені помилки, наприклад, розмір призначеної для користувача програми перевищує робочу пам'ять;
- перевищена тривалість циклу контролерав свідчить про його несправний стан.

Для перезавантаження пам'яті необхідно скористатися одним із таких методів:

- необхідно активуватим кнопку «MRES» на панелі контролера;
- необхідно вибрати «CPU > MRES» у меню команд;
- необхідно активувати комбінацію клавіш на комп'ютері «ALT + C + M»;
- виконати перезавантаження у STEP 7.

Під час перезавантаження пам'яті індикатор «STOP» продовжує індикувати.

Команда «MRES» перемикає контролер при необхідності в режим «STOP», а потім виконує наступні задачі:

- видаляє всі користувацькі програми STEP 7 (OB, бази даних, FC, FB, і системні дані), як з робочої пам'яті так і з завантажуваної пам'яті;
- скидає області пам'яті (I, O, M, T і C) до нульових значень;
- перезавантажує налаштування системи за замовчуванням;
- видаляє всі активні завдання по комунікаціям (наприклад, TIS) і всі відкриті комунікації;

Команда «MRES» не впливає на підмодуль мережевих адрес або на дані, які знаходяться у діагностичному буфері.

Після перезавантаження пам'яті, діагностичний буфер змінюється до свого розміру за замовчуванням. Області пам'яті входів (I) і виходів (Q) також змінюються до своїх розмірів за замовчуванням. Після перезавантаження пам'яті, необхідно налаштувати вищезгадані значення до необхідних.

Використання індикаторів статусу. Індикатори стану панелі контролера відображають поточний режим роботи і є корисними для усунення помилок. Ці індикатори знаходяться на LED-дисплеї та відображаються у PLC S7. Статус контролера змінюється при активації на індикатор.

Опис статусів індикаторів. В табл. 9.3 наведені різні стани індикаторів на панелі контролера.

Таблиця 9.3 – Стани індикаторів на панелі контролера

Індикатор	Опис
ON	Електричне живлення (для WinLC ввімкнений завжди)
BATF	Батарея несправна (для WinLC вимкнений завжди)
INTF	Цей індикатор вмикається (безперервно) для індикації стану помилки всередині контролера, наприклад: помилки програмування, помилки вбудованого програмного забезпечення, арифметичних помилок і помилки таймера
EXTF	Цей індикатор активується, щоб показати помилки, які є поза межами контролера, наприклад такі, як апаратні збої, помилки параметрів завдання, втрата зв'язку або інші помилки зв'язку, завантаження програми STEP 7 з налаштованою мережевою картою в WinLC RTX без будь-яких мережевих карт або несправних мережевих карт, а також помилок вводу/виводу. Цей індикатор активується разом з індикатором «BUSF», що означає несправність мережевих компонентів. Якщо користувачі програми STEP 7 обробляють помилку, виконуючи OB 122, індикатор «EXTF» вимкнеться через 3 секунди, якщо не виконається умова помилки
BUSF1 BUSF2 BUSF3 BUSF4	Дані індикатори активуються (позмінно) при виявлення помилки під час з'єднання з розподіленим вводом/виводом. Число «BUSF» LED відповідає числу підмодулів в яких виникла помилка.
FRCE	Цей індикатор активується (безперервно) для індикації активного управління виходами. Не використовується з WinLC
MAINT	Цей індикатор активується, якщо контролер PROFINET IO або пристрій вводу/виводу потребує обслуговування. За додатковою інформацією необхідно звернутися до онлайн довідки STEP 7.
RUN STOP	Активується, для індикації режиму роботи («RUN» або «STOP»). Індикатор «RUN» позмінно активується, а індикатор «STOP» ввімкнений під час переходу від режиму «STOP» в режим «RUN». Коли індикатор «STOP» вимикається, виходи контролера активуються. Примітка: індикатори «RUN» і «STOP» показують поточний режим роботи контролера. Режими «RUN» і «STOP» відображують вибраний режим (за аналогією з перемикачем режимів на S7-панелі процесора), який може відрізнятися від поточного режиму роботи.

Світлові індикатори. Активовані індикаторіи «RUN» і «STOP» можуть показувати додаткову інформацію про контролер або користувачів програми STEP 7(табл. 9.4).

Таблиця 9.4 – Опис роботи індикаторів

Індикатор RUN	Індикатор STOP	Опис
Перемикання з частотою 2 Гц	Перемикання з частотою 2 Гц	Контролер несправний. Всі індикатори активувалися.
Перемикання з частотою 0.5 Гц	ON	Користувацька програма STEP 7 зупинена на контрольній точці.
Перемикання з частотою 2 Гц	ON	Відбувається «холодне» або «гаряче» перезавантаження. Індикатор «RUN» продовжує бути активованим до завершення перезавантаження. Час, необхідний для перезавантаження залежить від необхідного часу для запуску організаційного блоку OB.
OFF	Перемикання з частотою 0.5 Гц	Контролер пам'яті потребує перезавантаження «MRES».
OFF	Перемикання з частотою 2 Гц	Триває перезавантаження пам'яті «MRES».

Коригувальні дії, при активації всіх індикаторів. Якщо всі індикатори активовані одночасно, то контролер перебуває у несправному стані і сталася помилка умови, яка не може бути виправлена шляхом перезавантаження пам'яті за допомогою команди меню «MRES». Для виходу з цього стану, необхідно виконати наступні кроки:

1) необхідно вибрати команду меню контролера «CPU > Shut Down», для вимкнення контролера;

2) необхідно перезапустити контролер (індикатор «STOP» активований, а індикатор «RUN» вимкнений);

3) необхідно використати команду «MRES» для перезавантаження пам'яті;

4) завантажити користувацьку програму STEP 7 та налаштування системи, або відновити архів користувацьких програм STEP 7.

Використання панелі налаштування WinLC. Панель налаштування призначена для настройки параметрів та перевірки роботи WinLC. Задаючи відповідні значення, можна налаштувати продуктивність контролера.

Для відкриття панелі налаштування WinLC RTX, необхідно виконати наступні дії:

1) необхідно вибрати команде меню «CPU > Tuning panel»;

2) програма WinLC RTX відкриває панель налаштування, як показано на рис. 9.10.

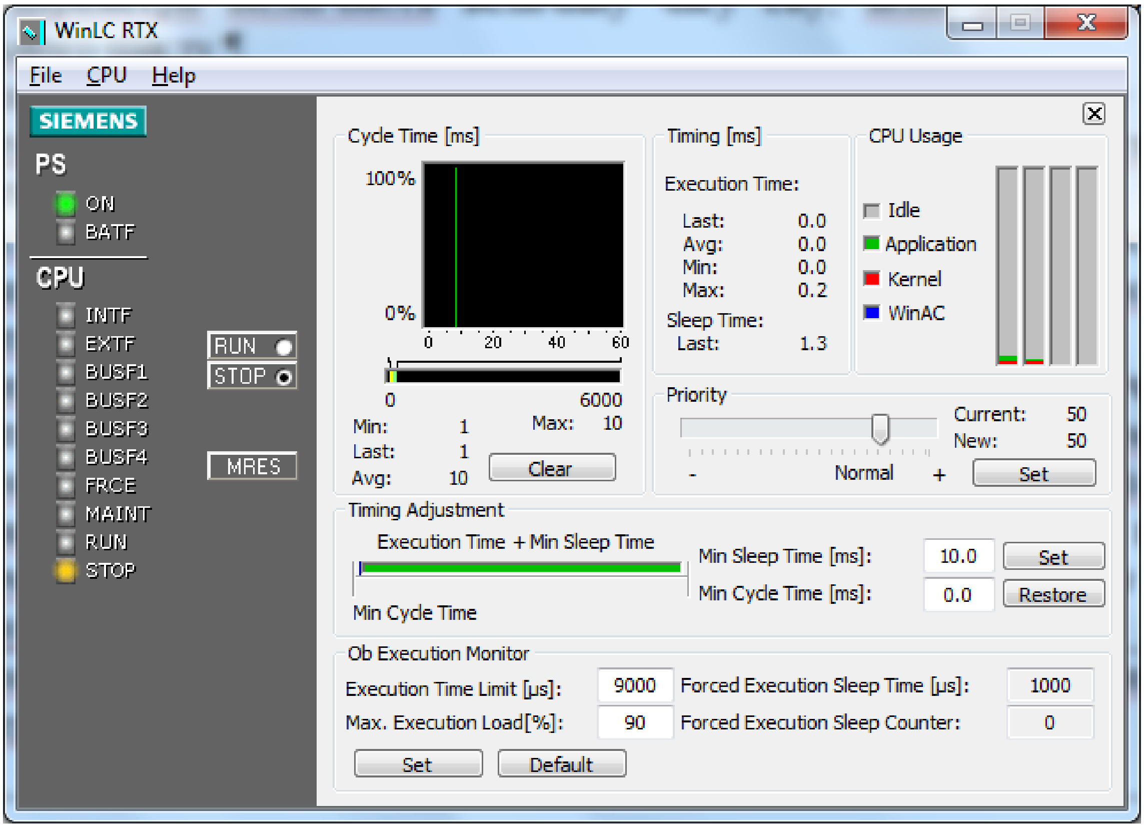


Рисунок 9.10 – Панель налаштування WinLC RTX

Функціональні області в налаштуваннях панелі WinLC RTX

Панель налаштування WinLC RTX містить наступні функціональні області (табл. 9.5).

Таблиця 9.5 – Опис функціональних областей панелі налаштування WinLC RTX

Область	Опис
Час циклу	- ця область відображає гістограму часу виконання «циклу сканування» у діапазоні 60 мс; - ця гістограма відображає мінімальний і максимальний час «сканування циклу», а також відсоток перевірених циклів, які потрапляють в різні діапазони часу; - кнопка «Clear» очищає попередні дані і створює нову гістограму; - на дисплеї «час циклу» скидається при: - режим роботи переходу від «STOP» до «RUN»; - закриття та відкриття налаштування панелі налаштування WinLC RTX.
Синхронізація	У данному полі відображається наступна інформація про «цикл сканування»: - «Execution Time» відображає час виконання останнього «циклу сканування», «середній час циклу», «мінімальний час циклу», і «максимальний час циклу» виконання програми; - «Sleep Time» відображає кількість «часу сну» для останнього «циклу сканування».
Використання CPU	Відображає наступні значення: - відсоток завантаження процесора комп'ютера, який знаходиться в режимі очікування; - відсоток завантаження, який використовується в додатках, ядрі, або WinAC.
Пріоритет	- встановлює рівень пріоритету для виконання WinLC RTX в порівнянні з іншими RTX-додатками, які запущені на комп'ютері; налаштування більш високого пріоритету означає: - операційна система реагує перш за все на WinLC RTX перед виконанням інших, менш пріоритетних завдань. Примітка: WinLC RTX має більш високий пріоритет, ніж всі інші програми Windows. Зміни до пріоритетів впливають на роботу WinLC RTX тільки тоді, коли інші RTX-додатки запущені.
Синхронізація та керування	Відбувається керування циклом, шляхом вводу наступних значень: - значення для мінімального «час сну»; - значення «мінімального часу циклу». Ці параметри визначають величину «часу сну», який додається в кінці основного «циклу». Кнопка «Set» застосовується для значень, що вводяться. Кнопка «Restore» обнуляє введені значеня до значень, які в даний час використовуються контролером: - налаштування панелі знову відправляє введені значення для контролера;

	- вплив цих значень на керування виконанням програми може бути контрольований. Примітка: для того, щоб задати «мінімальний час циклу», необхідно налаштувати час «циклу сканування» і мінімальні параметри «часу циклу» у «Circle/Clock Memory» на вкладці діалогового вікна «Properties» у STEP 7. Встановити «мінімальний час циклу» до значення, яке менше значення для «циклу сканування» контролю часу (за замовчуванням «час циклу» складає 6 сек.).
Виконання ОВ монітор	Гарантує, що контролер не перевищує максимальне навантаження на CP на інтервалі «циклу сканування».

Робота з налаштуванням панелі WinLC RTX. Значення, відмінні від мінімального часу циклу є унікальними для WinLC RTX і не зберігаються в налаштуваннях системи. Використання панелі налаштування для введення значень для мінімального часу циклу не змінює налаштування контролера.

Перемикання контролера WinLC RTX з режиму «STOP» у «RUN» обнулить мінімальні параметри часу циклу до значення, що задані у STEP 7. Щоб внести зміни в налаштування панелі, необхідно адаптувати відповідно конфігурацію STEP 7.

Час виконання і час простою. Якщо регулятор не забезпечує достатній час для простою для запуску інших програм, комп'ютер може перестати відповідати на вхідний оператор або контролера та програми, які можуть працювати некоректно. Крім того при виконанні користувацької програми STEP 7 може призвести до некоректної поведінки, що час виконання може змінюватися і подія може не виконатись.

Завжди надається схема аварійної зупинки. Крім того, завжди необхідно настроювати час простою і управляти продуктивністю контролера так, що користувацька програма STEP 7 виконується послідовно.

Використання діагностичного буферу. Діагностичний буфер дозволяє переглядати діагностичну інформацію системи без використання програмного забезпечення Simatic STEP 7.

Для відображення діагностичного буфера, необхідно вибрати у меню команди «CPU > Diagnostic Buffer», далі WinLC RTX активізує діагностичний буфер.

Структура діагностичного буфера. Діагностичний буфер відображає різну діагностичну інформацію:

- список подій відображається у верхній панелі діагностичного буферу;
- уточнення, що стосуються подій, відображаються в нижній панелі (рис. 9.11).

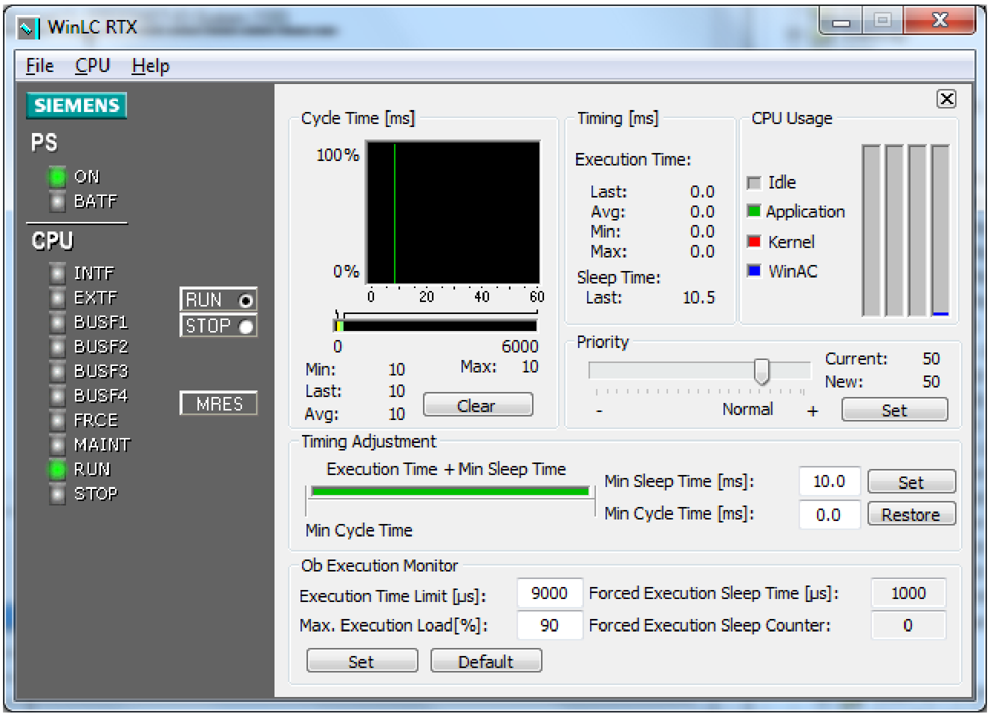


Рисунок 9.11 – Відображення інформації у діагностичному буфері

Функції діагностичного буфера. Діагностичний буфер реалізований у вигляді кільцевого буфера, в якому записані окремі події. Події відображаються в порядку спадання за часом. Якщо кільцевий буфер заповнений, то нові події записуються замість попередніх.

У діагностичному буфері відображається наступна інформація:

1) список подій (у цьому списку відображаються всі події в діагностичному буфері, зокрема відображається наступна інформація по кожній події):

- кількість записів (колонка може бути відсортована);
- дата та час події;
- короткий опис події (колонка може бути відсортована);

2) ID події (між верхньою і нижньою панеллю) - відображає ID - номер обраної події;

3) відомості про подію (нижня панель) - відображає відомості в текстовому або шістнадцятковому форматі.

Вибір формату діагностичної інформації. Можна відображати діагностичну інформацію в нижній частині вікна в текстовому або шістнадцятковому форматі. В шістнадцятковому форматі відображаються шістнадцяткові значення подій розміром 20 байтів.

Текстовий формат .Відображається наступна інформація про обрану подію:

- короткий опис;
- додаткова інформація, залежна від події, така як адреса інструкції, що викликана діагностичною подією та режимом переходу, який був викликаний подією;
- стани подій (вхідний або вихідний).

Якщо один з параметрів тексту не може бути ідентифікованим, діагностичний буфер показує рядок «###». Якщо тексту немає для нових модулів або нових подій, номер події і один з параметрів відображаються у вигляді шістнадцяткових значень.

Шістнадцятковий формат. Для обраної події відображаються шістнадцяткові значення.

Вибір типу часу. Якщо активізувати маркер «Time including CPU/local time difference», діагностичний буфер коректує значення часу дня.

Якщо не активовувати «Time including CPU/local time difference», діагностичний буфер показує час доби, який виставлений у модулі. Необхідно використовувати цей параметр, якщо час дня, який виставлений у модулі збігається з місцевим часом. Якщо не змінювати налаштування, діагностичний буфер одразу здійснить оновлення часу.

Оновлення діагностичного буфера. Для відображення найактуальнішої інформації, необхідно вибрати кнопку «Update».

Збереження діагностичного буфера. Щоб зберегти текстовий файл, що містить список подій і детальну інформацію по кожну подію, необхідно активувати кнопку «Save». Текстовий файл містить інформацію, як в текстовому, так і в шістнадцятковому форматі.

Відображення довідки. Для отримання довідки діагностичного буфера, необхідно активувати кнопку «Help». Для перегляду довідки по певній події необхідно:

- вибрати подію у верхній панелі;
- активувати кнопку «Help on Event».

Архівування та відновлення файлів з підтримкою STEP 7. Команда архівування дозволяє зберегти налаштування та користувацькі програми STEP 7 у архівному файлі (*.WLD). Архів файлів дозволяє легко відновити налаштування та користувацькі програми STEP 7 для контролера WinLC RTX.

Файли архіву функціонують, як знімний модуль пам'яті (карти EEPROM) S7-CPU, проте відрізняються тим, що контролер WinLC RTX автоматично невідновлює архів після перезавантаження пам'яті «MRES». Необхідно вручну відновлювати файл архіву. При архівуванні контролер WinLC RTX має бути в режимі «STOP».

Створення архіву. Для створення файлу архіву необхідно виконати наступні дії:

1) обрати меню команди «File > Archive» та зберегти при відкритті діалогового вікна;
2) перейти до місця зберігання архівного файлу;
3) вказати назву архівного файлу.

Після виконання вищезгаданих дій контролер WinLC RTX створює архівний файл з розширенням (*.WLD).

В архівному файлі зберігаються поточні користувацькі програми STEP 7, поточне налаштування системи і поточні значення бази даних.

Архівний файл не зберігає налаштування станції PC. Можна також використовувати Simatic Manager у STEP 7 для створення архіву, вибравши команду меню «File > Memory Card File > New».

Відновлення архіву. Для відновлення даних, які зберігаються у архіві, необхідно виконати наступні кроки:

1) вибрати команду меню «File > Restore»;

2) вибрати архівний файл з розширенням (*. WLD), який необхідно відновити;

3) активувати кнопку «OK».

Користувальницькі програми STEP 7 та налаштування для контролера будуть відновлені і завантажені в WinLC RTX.

Вибір методу перезавантаження. Метод перезавантаженняи визначає, які запускаючі організаційні блоки (OB) контролера WinLC RTX будуть виконуватись кожен раз, коли відбувається перехід від режиму «STOP» в режим «RUN». Запуск організаційного блока (OB) дозволяє ініціалізувати користувацькі програми STEP 7 та змінні. WinLC RTX підтримує два методи перезавантаження:

- «гарячий рестарт» (контролер WinLC RTX виконує (OB100) перед початком основного циклу (OB1). «гарячий рестарт» обнулює периферійні входи (PI), та змінює периферійні виходи (PQ) до попередньо заданого безпечного стану (за замовчанням в нульовий стан). «гарячий рестарт» також зберігає поточні значення області пам'яті для бітів пам'яті (M), таймерів (T), лічильників (C), і блоків даних (DB);

- «холодний рестарт» (контролер WinLC RTX виконує (OB102) перед початком основного циклу (OB 1). Як і «гарячий рестарт», «холодний рестарт» обнулює периферійні входи (PI) і змінює периферійні виходи (PQ) до попередньо заданого безпечного стану (за замовчанням в нульовий стан). Однак, «холодний рестарт» не зберігає поточні значення області пам'яті для бітів пам'яті (M), таймерів (T), лічильників (C), і блоків даних (DB), але встановлює вказані області до їхнього значення за замовчуванням.

Для налаштування за замовчуванням методу перезавантаження (рестарту) контролера WinLC RTX використовується програмний пакет STEP 7. За замовчуванням метод перезавантаження зберігається в налаштуваннях (системні дані) контролера, які завантажуються з вказаної користувацької програми STEP 7. WinLC RTX використовує цей метод, коли перезавантаження WinLC RTX є налаштованим для автозапуску і повертається в режим «RUN» перезапуску живлення.

Всякий раз, коли вибирається на панелі WinLC RTX селектор режиму «RUN» для переходу від режиму «STOP» в режим «RUN», WinLC RTX виконує «гарячий рестарт», виконуючи (OB100).

Для вибору конкретного методу перезавантаження (рестарту) WinLC RTX, необхідно вибрати один з наступних варіантів зміни режиму контролера з режиму «STOP» в режим «RUN»:

Варіант 1:

1) необхідно вибрати у меню команду «CPU > RUN», для зміни режиму контролеру від «STOP» до «RUN»;

2) при необхідності слід виконати підтвердження вибору або ввід пароля;

3) необхідно вибрати «гарячий рестарт», або «холодний рестарт» в діалоговому вікні «Restart Method».

Варіант 2:

1) необхідно активувати правою кнопкою маніпулятора «миш» для активізації режиму «RUN»;

2) при необхідності слід виконати підтвердження вибору або ввід пароля;

3) необхідно вибрати «гарячий рестарт», або «холодний рестарт» в діалоговому вікні «Restart Method».

9.2 Створення, конфігурування і параметрування проекту в WinAC RTX засобами Simatic STEP 7 Professional

Нижче наведено приклад інтеграції частотного перетворювача Sinamics G120 в проет на базі програмного контролера WinAC RTX і промислової шини PROFINET-IO.

Інтеграція частотного перетворювача в проект з WinAC RTX

Після встановлення відповыдної IP-адреси мережевоїкарти, за допомогою модуля (підпрограми) (HW-Config) виконуються всі інші установки.

Створення проекту в STEP 7. Необхідно створити новий проект в STEP 7 і присвоїти йому ім'я (рис. 9.12).

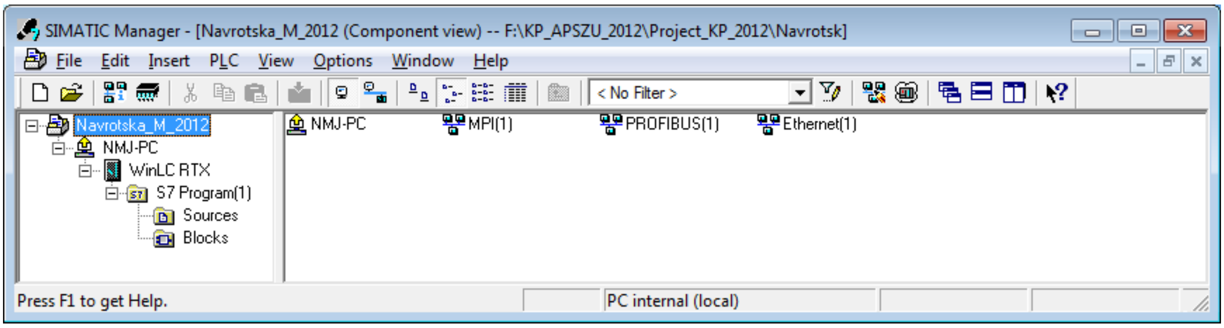


Рисунок 9.12 – Створення нового проекту в STEP 7

Проектування Simatic PC Station і створення мережі PROFINET-IO. Необхідно встановити в проекті об'єкт «Simatic PC Station» (рис. 9.13).

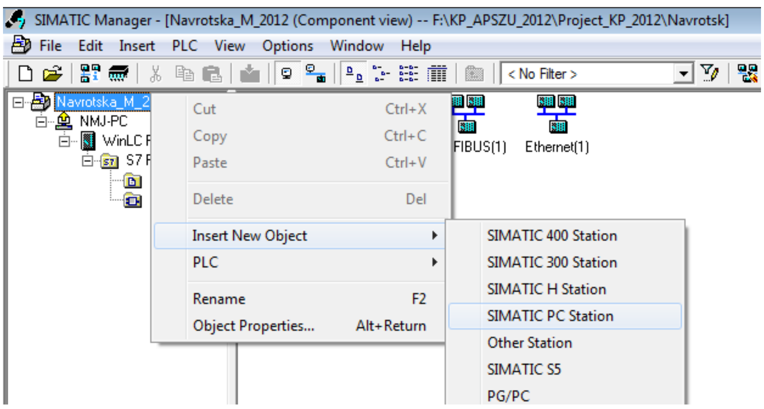


Рисунок 9.13 – Встановлення в проекті об'єкту Simatic «PC Station»

Відкрити апаратну конфігурацію (HW-Config) в STEP 7 (рис. 9.14).

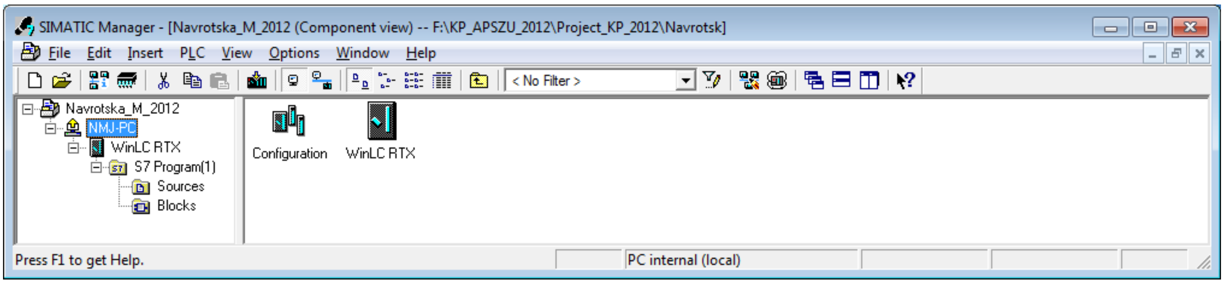


Рисунок 9.14 – Апаратна конфігурація (HW-Config) в STEP 7

Перетягнути з аппаратного каталога «Simatic 300» об'єкт «Rack» зі слотами для модулів S7-300 в проект. Встановити в перший слот модуль живлення, а в другий слот встановити Software PLC WinLC RTX.

При встановленні WinLC RTX автоматично відкривається вікно для визначення мережі. Створити мережу PROFINET-IO (рис. 9.15).

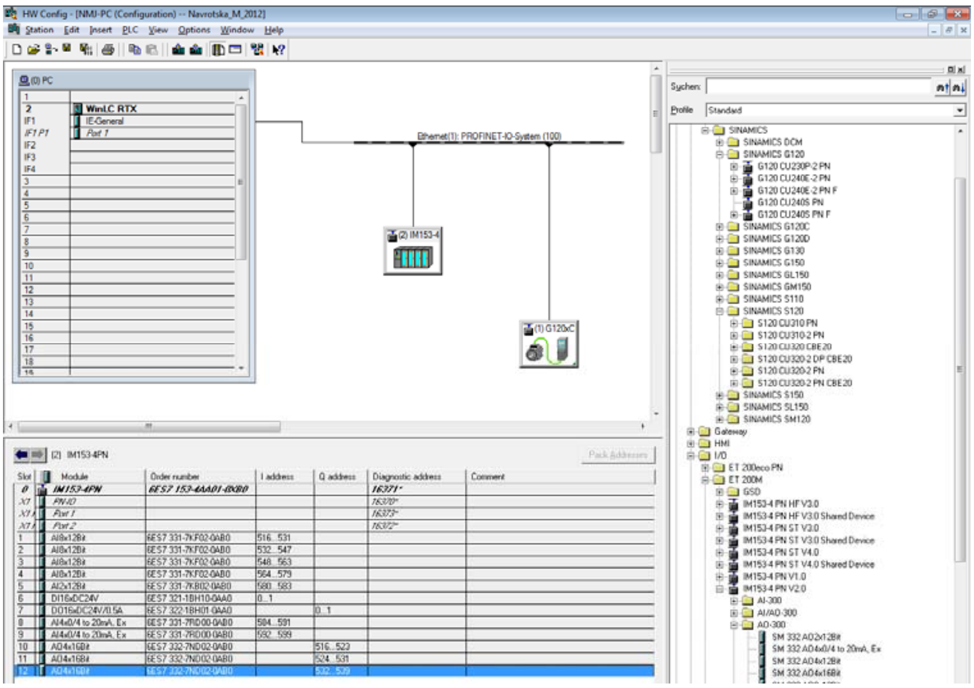


Рисунок 9.15 – Результат встановлення об'єктів Simatic «PC Station» з мережею PROFINET-IO і частотного перетворювача Sinamics G120

Інтеграція та параметрування перетворювача частоти та в проект STEP 7

У проекті STEP 7 існує два способи інтеграції перетворювача частоти:

1) через GSD файл (GSD це стандартизований файл опису компонентів). GSD підтримуються всіма контролерами, які є PROFIBUS-Master або в мережі Ethernet);

2) через менеджер об'єктів STEP 7 (більш зручний спосіб, доступний тільки для контролерів S7 і встановленого опційного программного пакету Drive_ES_Basic).

Нижче наведено процес інтеграції перетворювача частоти Sinamics G120 через менеджер об'єктів STEP 7 (рис. 9.16).

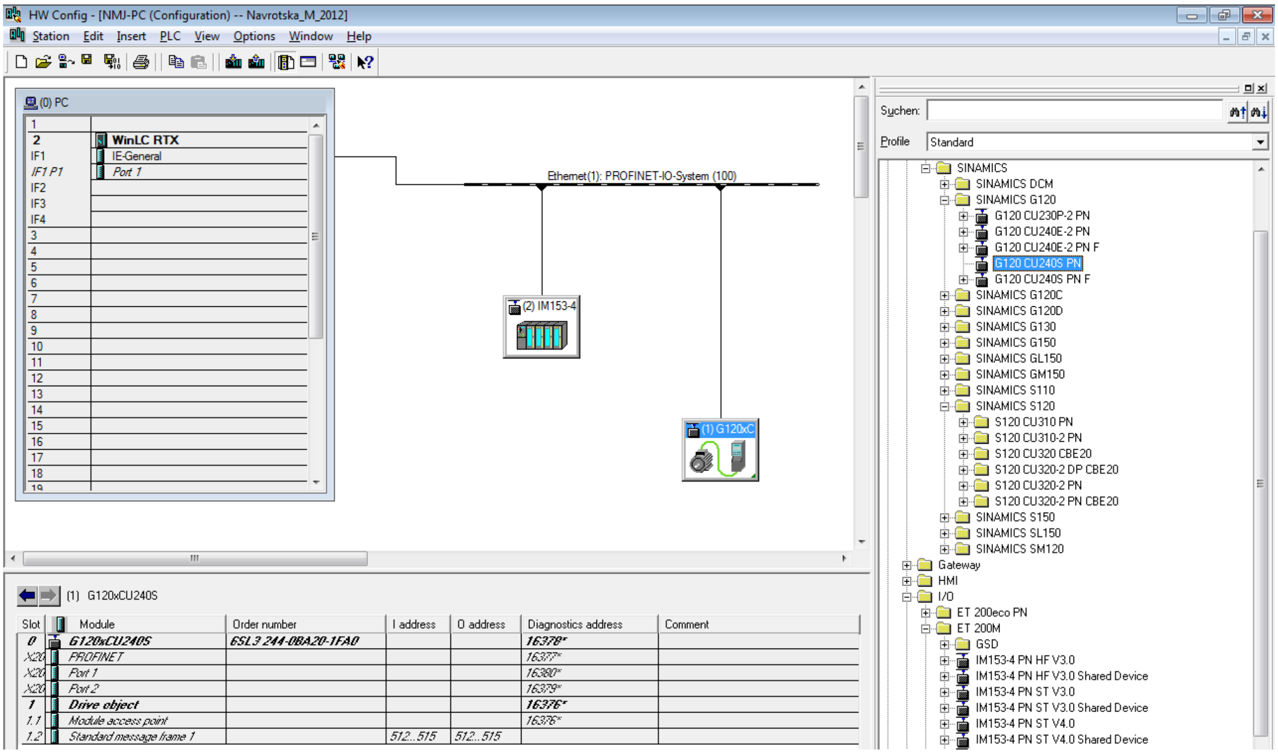


Рисунок 9.16 – Встановлення об'єктів проекту через менеджер об'єктів STEP 7

На рис. 9.17 наведено вікно параметрів частотного перетворювача Sinamics G120 з комунікаційним інтерфейсом Industrial Ethernet.

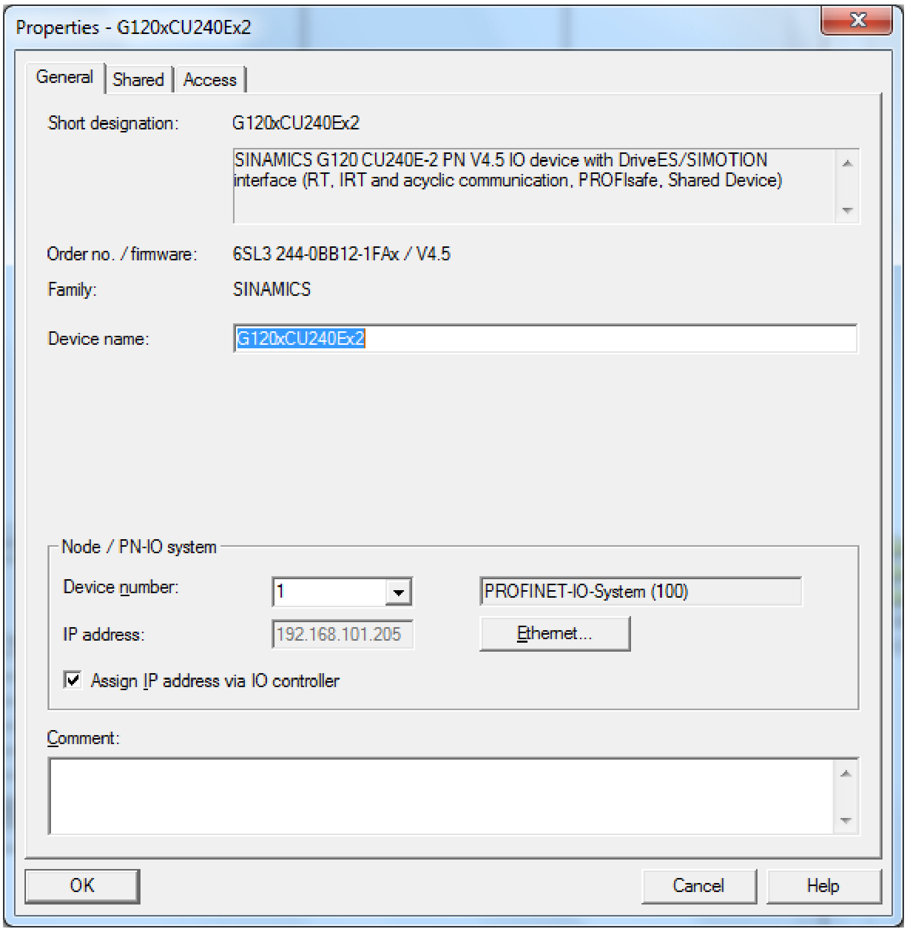


Рисунок 9.17 - Вікно параметрів частотного перетворювача Sinamics G120 в модулі (HW-Config)

Параметрування частотного перетворювача. Встановити перетворювач частоти Sinamics G120 шляхом перетягування даного об'єкту в мережу Industrial Ethernet і визначити його параметри (рис. 9.18).

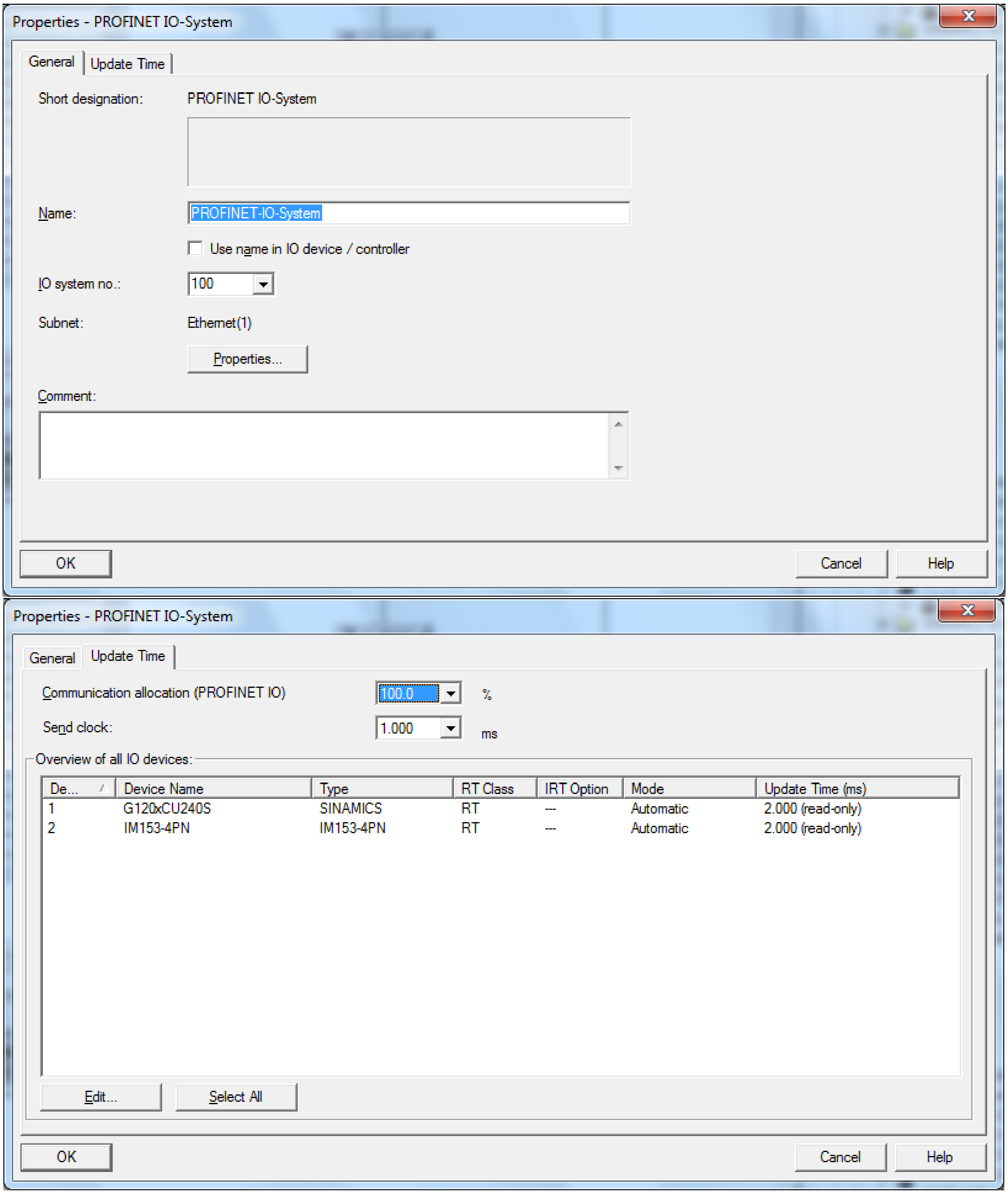


Рисунок 9.18 – Параметрування і встановлення перетворювача частоти Sinamics G120 в мережу PROFINET-IO

Об'єкт перетворювача частоти Sinamics G120 в каталозі продуктів (HW-Config) містить кілька типів телеграм. Тип телеграми визначає, якими циклічними даними контролер і перетворювач будуть обмінюватись. Наприклад, у разі стандартної «телеграми №1» перетворювач отримує дані від контролера WinLC RTX у вигляді «слова управління» і «заданої швидкості» і відправляє в контролер WinLC RTX «слово стану» і «фактичне значення швидкості».

Вставити перетягуванням необхідний тип «телеграми» з апаратного каталогу в слот 1 перетворювача частоти (рис. 9.19).

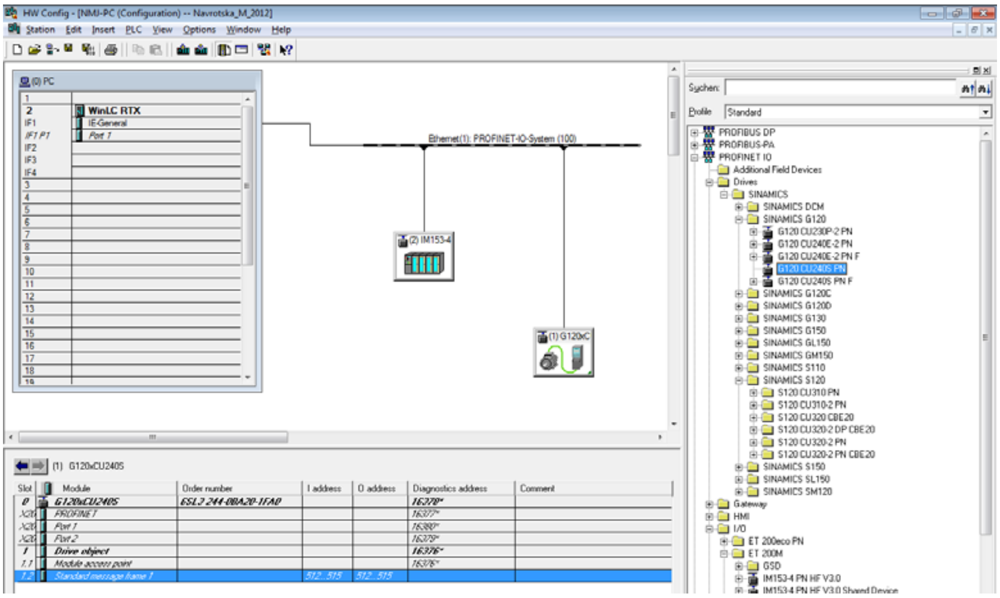


Рисунок 9.19 - Визначення типу «телеграми» перетворювача частоти Sinamics G120

STEP 7 автоматично присвоює діапазон адрес, в якому знаходяться дані процесу перетворювача частоти. Стандартна «телеграма №1» займає чотири байти вхідних і вихідних даних відповідно.

Установка типу телеграми в (HW-Config) діє тільки з боку контролера. У перетворювачі ідентичний тип телеграми повинен бути встановлений за допомогою програмного пакету «STARTER» або Basic Operator Panel (BOP) через параметр (P0922).

Далі, зберегти і скомпілювати проект у STEP 7. Встановити «on-line» з'єднання між PC і WinLC RTX і завантажити дані проекту в WinLC RTX.

Тепер перетворювач частоти пов'язаний з WinLC RTX. Комунікаційний інтерфейс між CPU і перетворювачем частоти задається через профіль PROFIdrive V4.1.

Реалізація системи частотного керування електроприводом на базі Software PLC WinAC RTX з WEB-інтерфейсом

Призначення і технічні характеристики системи

Система призначена для віддаленого керування електроприводом територіально розподілених технологічних об'єктів через промислову мережу Industrial Ethernet з WEB-інтерфейсом.

Технічні характеристики:

- апаратно-програмні засоби концерну «Siemens» Simatic S7 і Sinamics G120;
- діапазон вихідних частот перетворювачів частоти Sinamics G120 (0-600 Гц);
- діапазон вихідних потужностей електропривода (0.37-250 кВт);
- комунікації (PROFIBUS-DP, Industrial Ethernet, GSM, WEB-інтерфейс);
- стандарт профілю управління (PROFIdrive V4.1);
- способи управління («servo» і «vector»);
- режими управління (з заданими швидкістю або крутним моментом).

Склад системи:

- асинхронні трифазні електроприводи визначеної потужності;
- Software PLC (WinAC RTX 2010) «Siemens»;
- частотні перетворювачі Sinamics G120 «Siemens»;
- комунікація (Industrial Ethernet з WEB-інтерфейсом);
- SCADA-система (WinCC V7.x) «Siemens».

Засоби проектування системи керування

Проект системи частотного керування електроприводом на базі Software PLC WinAC RTX і промислової мережі Industrial Ethernet з WEB-інтерфейсом створюється шляхом:

- конфігурування об'єкту «PC-station» в модулі (HW-Config) базового програмного пакету STEP 7 (рис. 9.20);
- конфігурування комунікаційного середовища в модулі (підпрограмі) (NetPro) базового програмного пакету STEP 7 (рис. 9.21);
- програмування алгоритму керування на мові FBD (Functional Block Diagram – мова функціональних блоків) (рис. 9.22);
- проведення діагностики PLC через вбудований WEB-server (рис. 9.23);
- створення графічного HMI (Human Machine Interface – людино-машинного інтерфейсу) в SCADA-системі WinCC (рис. 9.24);
- організації WEB-доступу до об'єкту керування через мережу Internet на базі протоколу TCP/IP (рис. 9.25).

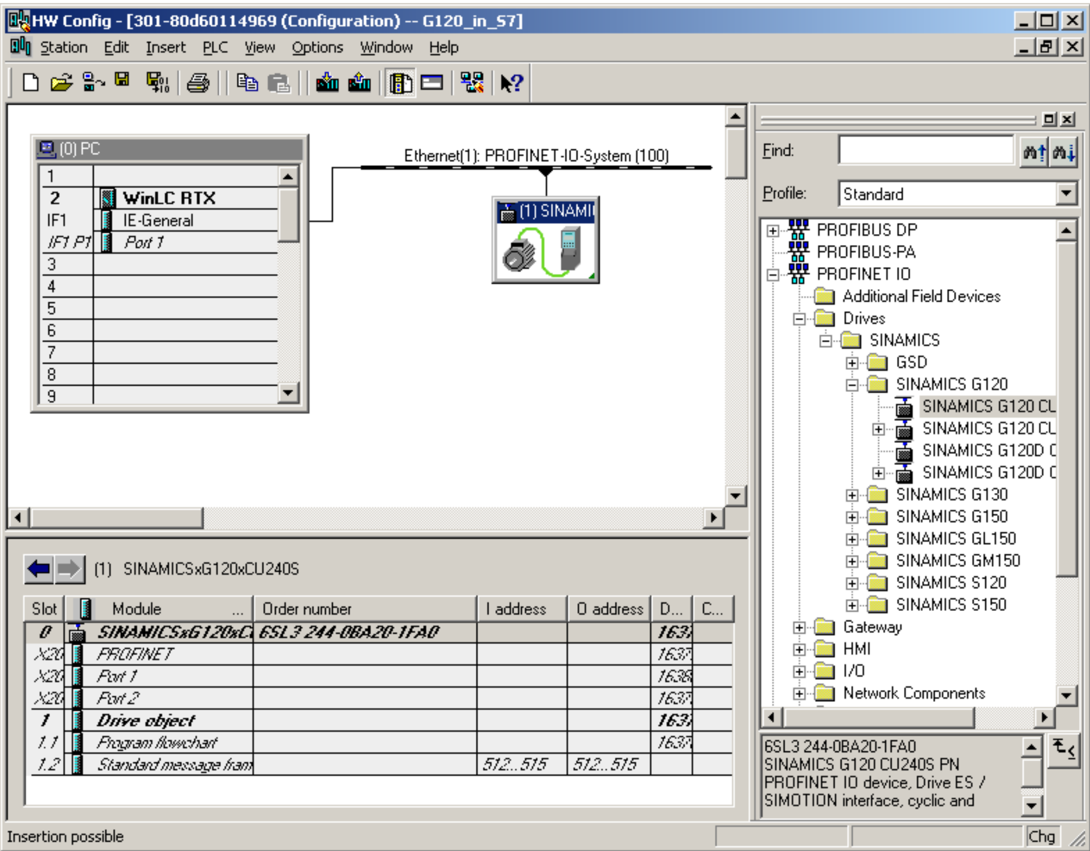


Рисунок 9.20 – Конфігурування PC-station в модулі HW-Config базового програмного пакету STEP 7

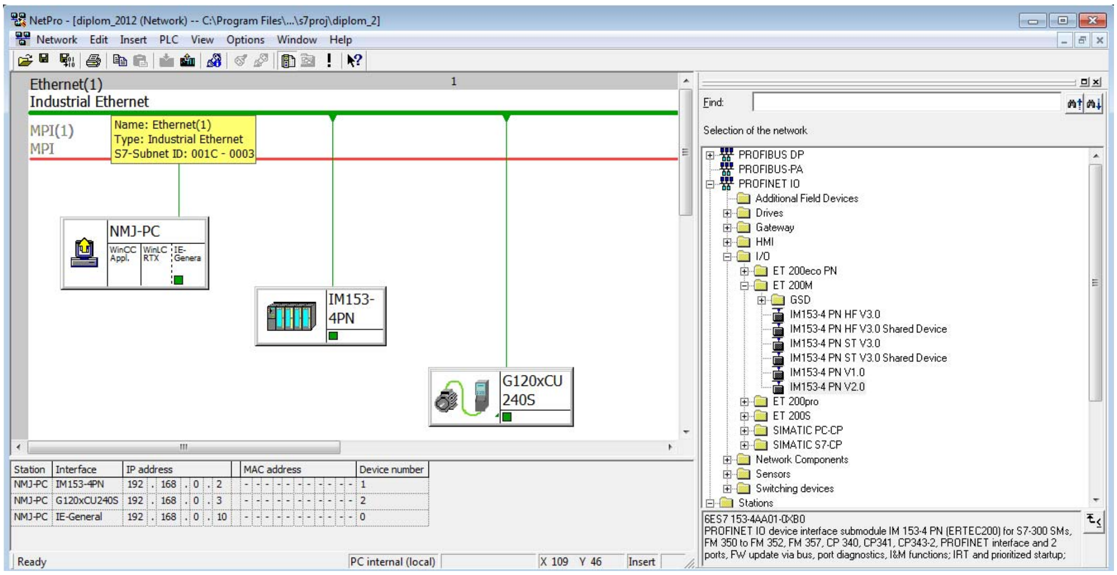


Рисунок 9.21 – Конфігурування комунікаційного середовища в модулі (NetPro) базового програмного пакету STEP 7

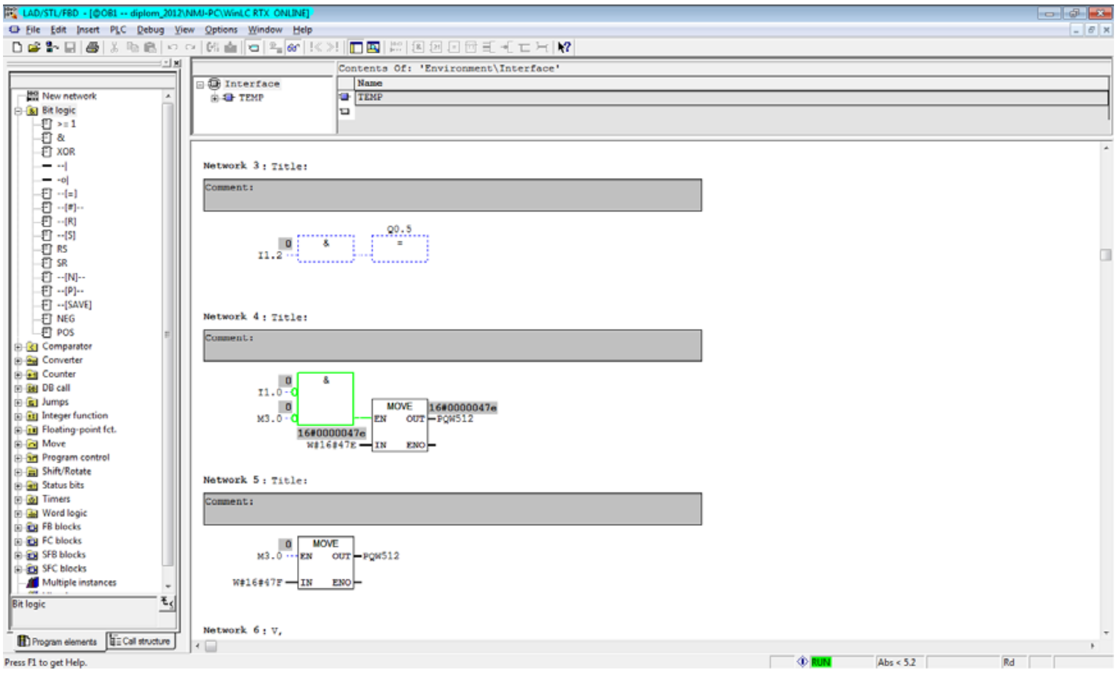


Рисунок 9.22 – Програмування алгоритму керування на мові FBD (Functional Block Diagram) – режим відладки «on-line»

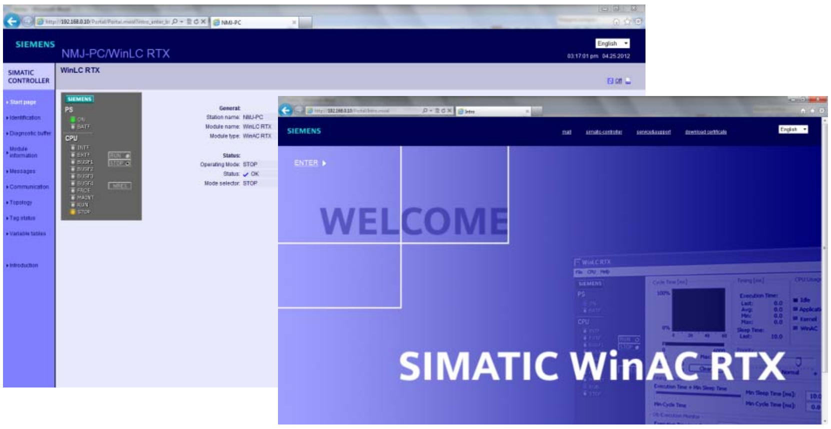


Рисунок 9.23 - Проведення діагностики Software PLC WinLC RTX через вбудований WEB-server

На рис. 9.24 показано графічний людино-машинний інтерфейс HMI системи частотного керування електроприводом в режимі керування швидкістю (графічні індикатори швидкості і поля вводу-виводу параметрів швидкості, а також вікна трендів зміни швидкості електропривода в часі, які можуть зберігатись в архів через визначені проміжки часу).

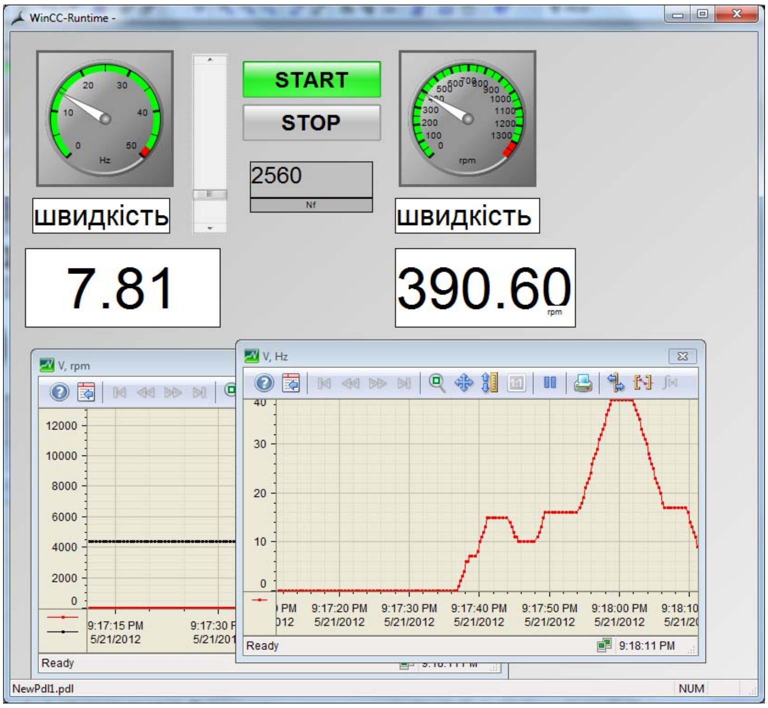


Рисунок 9.24 - Графічний інтерфейс керування перетворювачем частоти Sinamics G120 в SCADA-системі WinCC

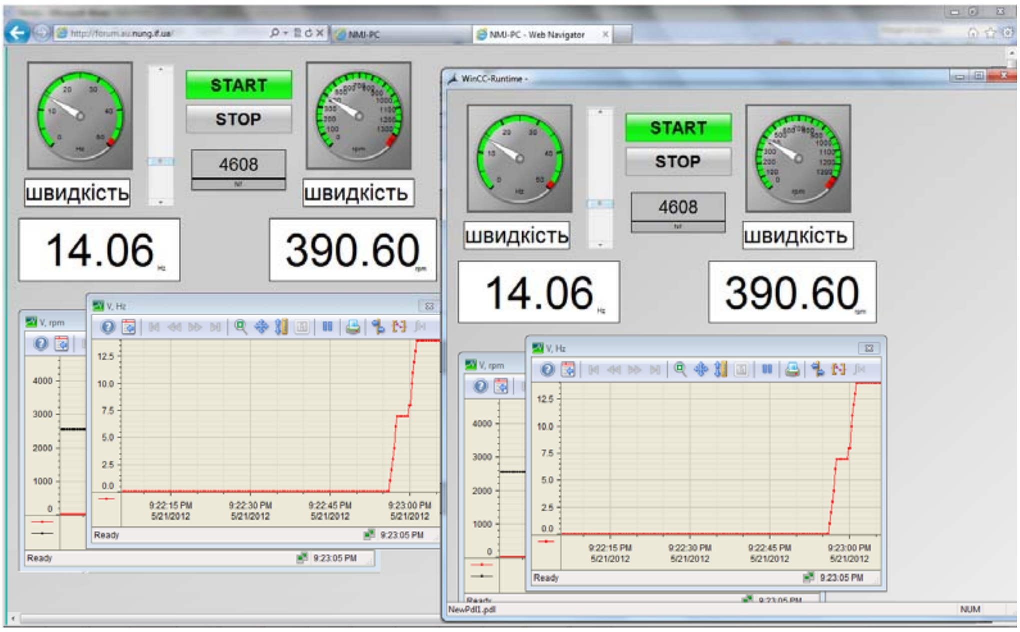


Рисунок 9.25 – Процес частотного керування електроприводом (вікно керування на локальному PC в режимі «run-time» і вікно керування в режимі «run-time»з віддаленого терміналу через WEB-interface

Контрольні запитання:

1) Software PLC «WinAC RTX», його організація і основні технічні характеристики;

2) Комунікації, що підтримуються Software PLC «WinAC RTX»;

3) Організація і параметрування апаратно-програмних засобів Personal Computer (PC) для функціонування Software PLC «WinAC RTX»;

4) Мови програмування Software PLC «WinAC RTX» стандарту IEC 61131;

5) Основні параметри і процедура запуску панелі параметрів «WinLC Properties» і панелі діагностики «WinLC Tuning Panel» для Software PLC «WinAC RTX»;

6) апаратні інтерфейси Software PLC «WinAC RTX»;

7) організація і параметрування ресурсів пам'яті Software PLC «WinAC RTX»;

8) Підключення сигнальних модулів «Siemens» до Software PLC «WinAC RTX»;

9) Підключення сигнальних модулів «Phoenix Contact» до Software PLC «WinAC RTX»;

10) Способи адресації змінних для для Software PLC «WinAC RTX»;

11) Призначення, імпорт та інсталяція GSD-файлів;

12) Процедура тестування управляючої програми в режимі «on-line».

РОЗДІЛ 10
ОГЛЯД ТЕХНОЛОГІЇ ОРС

10.1.1 ОРС – що це?

ОРС (OLE for Process Control, Зв'язування та Вбудовування Об'єктів для Процесу Керування) — сімейство програмних технологій, які надають єдиний інтерфейс для керування об'єктами автоматизації і технологічними процесами.

ОРС — це набір відкритих протоколів, які регламентують взаємодію між собою різноманітних об'єктів автоматизації, наприклад SCADA-системи.

Стандарт ОРС розроблявся з ***метою*** *скоротити витрати на розробку й супровождення додатків промислової автоматизації. Суть ОРС проста — надати розробникам промислових програм універсальний фіксований інтерфейс обміну даними с будь-якими пристроями.*

Технологія ОРС реалізується за принципом «клієнт – сервер»:

- ✓ **ОРС-сервер** – програма, яка отримує дані во внутрішньому форматі пристрою або системи та перетворює ці дані в формат ОРС. ОРС-сервер є джерелом даних для ОРС-клієнтів. По своїй суті ОРС-сервер – це деякий універсальний драйвер фізичного обладнання, який забезпечує взаємодію з будь-яким ОРС-клієнтом.
- ✓ **ОРС-клієнт** – програма, яка приймає від ОРС-серверів дані в форматі ОРС.

Стандарт ОРС розроблений міжнародною організацією ОРС Foundation, членами якої є більш 400 підприємств, діючих в області засобів автоматизації та вимірювальної техніки. Засновниками організації є фірми Fisher-Rosemount, Rockwell Software, Opto 22, Intellution и Intuitive Technology.

Перша версія ОРС стандарту була випущена в 1998 г.

Головною метою ОРС є створення можливості взаємодії засобів автоматизації, які розроблені різними виробниками, функціонують у різних середовищах та під керуванням різних протоколів.

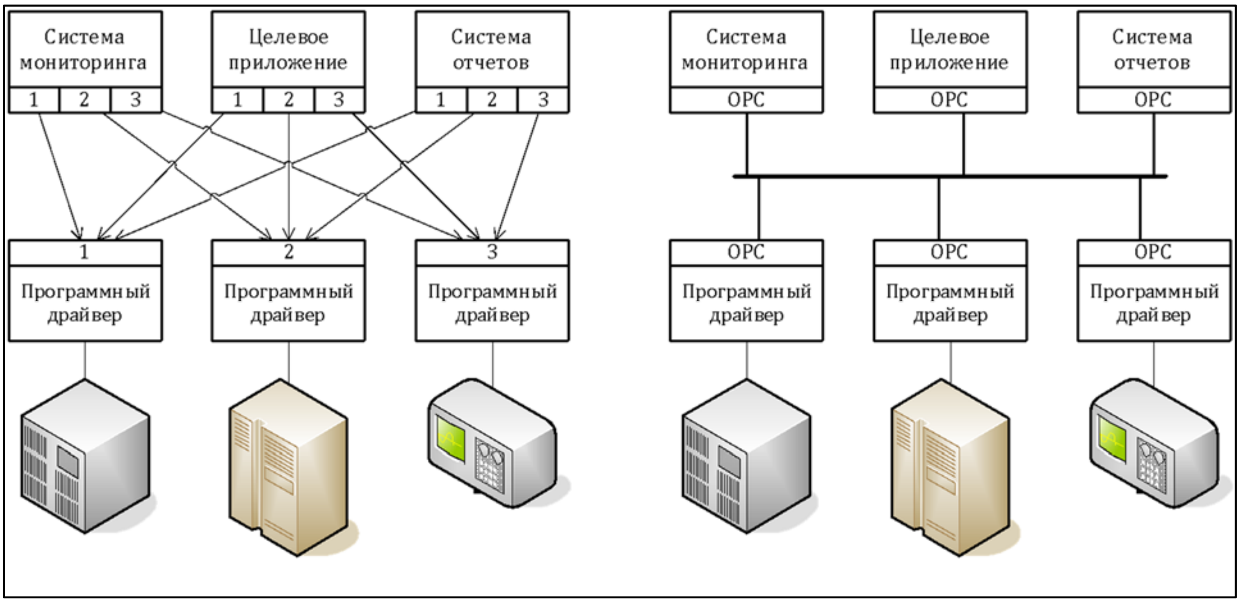


Рисунок 10.1 - Приклад системи автоматизації: а) на базі множини драйверів; б) на базі OPC.

OPC Foundation – міжнародна організація, яка об'єднала в собі множину компаній, які займаються розробкою програмного і апаратного забезпечення систем автоматизації.

10.1.2 Склад OPC

⇒ **OPC DA (OPC Data Access)** - специфікація для обміну даними між клієнтом (*наприклад*, *SCADA*) й апаратурою (*контролерами*, *модулями вводу-виводу та ін.*) у реальному часі;

⇒ **OPC HDA (Historical Data Access)** - специфікація для доступу до передісторії процесу (*до збережених в архіві даними*). Сервер забезпечує уніфікований спосіб доступу за допомогою DCOM технології. Забезпечує читання, запис і змінення даних;

⇒ **Batch** - специфікація для особливих фізико-хімічних технологічних процесів обробки матеріалів, які не є безперервними. У таких процесах виконується завантаження декількох видів сировини в певних пропорціях згідно з рецептом, встановлюються режими обробки, а після виконання циклу обробки і вивантаження готового матеріалу завантажується нова партія сировини. OPC сервер виконує обмін між клієнтом і сервером рецептами, характеристиками технологічного обладнання, умовами й результатами обробки;

⇒ **OPC Alarms & Events (A&E)** - специфікація для повідомлення клієнта про події та сигнали тривоги, що надсилаються клієнту в міру їхнього виникнення. Цей сервер пересилає аварійні сигнали, дії оператора, інформаційні повідомлення, результати контролю стану системи;

⇒ **OPC Data eXchange** - специфікація для обміну даними між двома OPC DA серверами через мережу Ethernet;

⇒ **OPC Security** - специфікація, яка визначає методи доступу клієнтів до сервера, які забезпечують захист важливої інформації від несанкціонованої модифікації;

⇒ **OPC XML-DA** - набір гнучких правил і форматів, що узгоджуються між собою для подання первинних даних із допомогою мови XML, веб-технологій і повідомлень SOAP;

⇒ **OPC Complex Data** - додаткові специфікації до OPC DA і XML-DA, які дають змогу серверам працювати зі складними типами даних, як-от бінарні структури і XML-документи;

⇒ **OPC Commands** - набір програмних інтерфейсів, який дає змогу OPC клієнтам і серверам ідентифікувати, посилати й контролювати команди, що виконуються в технічному пристрої (у контролері, модулі вводу-виводу);

⇒ **OPC Unified Architecture** - принципово новий набір специфікацій, який уже не базується на DCOM технології.

10.2 OPC DA - Data Access

10.2.1 Огляд

Виходячи з області застосування OPC-серверів в автоматичних системах керування, підприємства розрізняють кілька рівнів керування:

⇒ нижній рівень – польові шини (fieldbus) и окремі контролери;

⇒ середній рівень – цехові мережі;

⇒ рівень АСК ТП – рівень роботи систем типу SCADA;

⇒ рівень АСКП – рівень застосувань для керування ресурсами підприємства.

Кожний з цих рівнів може обслуговуватись OPC-сервером, надаючи дані OPC-клієнту на більш високому рівні або навіть «сусіду»

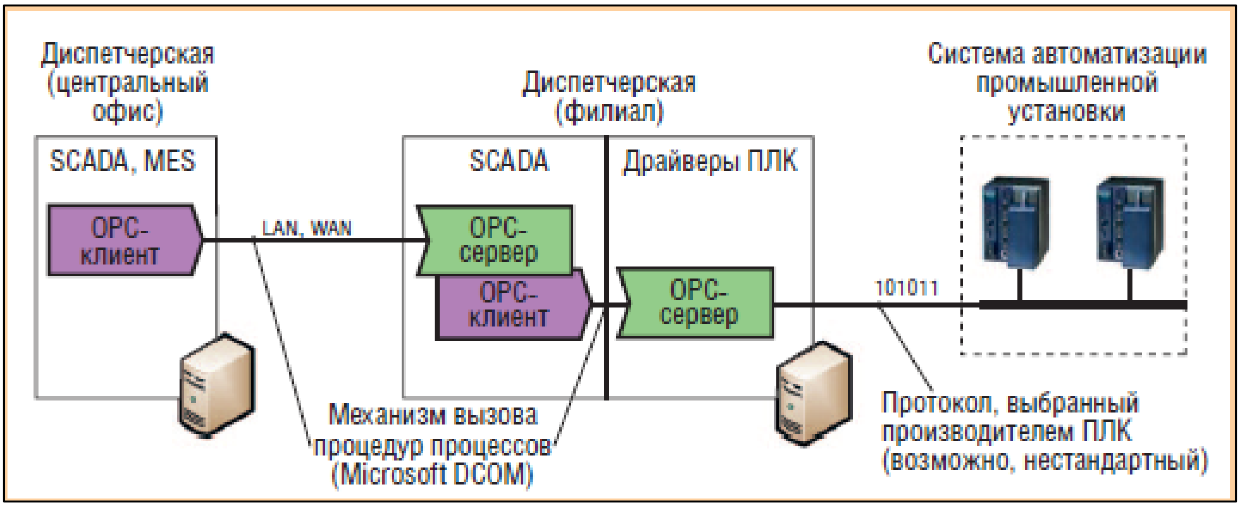


Рисунок 10.2 – Схема OPC

Сервер OPC DA - найбільш використовуваний в промислової автоматизації. Він забезпечує обмін даними (запис и читання) між клієнтською програмою і фізичними пристроями.

Дані складаються з трьох полів:

⇒ значення,

⇒ якість,

⇒ часова мітка.

Поле якості дозволяє передати від пристрою клієнтській програмі інформацію про вибіг вимірювальної величини за межі динамічного діапазону, про відсутність даних, помилку зв'язку тощо.

OPC DA – режими читання даних:

⇒ **синхронний режим**: клієнт посилає запит серверу і очікує відповідь від нього;

⇒ **асинхронний режим**: клієнт посилає запит і одразу переходить до виконання інших задач. Сервер, після виконання функції запиту, посилає клієнту повідомлення і той забирає наявні дані;

⇒ **режим підписки**: клієнт повідомляє серверу перелік тегів, значення яких сервер повинен відправляти клієнту тільки в разі їх зміни. Для того, щоб шум даних не був прийнятий за їх зміну, вводиться поняття "мертвої зони", яка перевищує максимально можливий розмах завади;

⇒ **режим обновлення даних**: клієнт визиває одночасне читання усіх активних тегів. Активними називаються усі теги, крім помічених як "пасивні". Такий поділ тегів зменшує завантаження процесору поновленням даних, які приймаються з фізичного пристрою.

У кожному з цих режимів дані можуть зчитатися або з кешу OPC сервера, або безпосередньо з фізичного пристрою. Читання з кеша виконується набагато швидше, але дані до моменту читання можуть втратити свою актуальність. Тому сервер має періодично оновлювати дані з максимально можливою частотою. Для зменшення завантаження процесора використовують параметр частоти оновлення, яка може встановлюватись для кожної групи тегів індивідуально. Крім того, деякі теги можна зробити пасивними, тоді їхні значення не будуть оновлюватися даними з фізичного пристрою.

Запис даних у фізичний пристрій може виконуватись тільки двома методами:

⇒ синхронним
⇒ асинхронним

та виконується відразу в пристрій, без проміжної буферизації.

У **синхронному** режимі функція запису виконується доти, поки з фізичного пристрою не надійде підтвердження, що запис виконано. Цей процес може займати багато часу, упродовж якого клієнт знаходиться в стані очікування завершення функції й не може продовжувати виконання своєї роботи.

У разі **асинхронного** запису, клієнт надсилає дані серверу і відразу продовжує свою роботу. Після закінчення запису сервер надсилає клієнту відповідне повідомлення.

OPC DA – режим користувача:

OPC DA сервер може мати (не обов'язково) користувальницький інтерфейс, який дозволяє виконувати будь-які допоміжні функції для полегшення роботи з обладнанням.

Наприклад:

– *пошук підключеного до промислової мережі обладнання;*
– *установку параметрів обладнання (імені, адреси, швидкості обміну даними, періоду сторожового таймера, наявність контрольної суми та ін.);*
– *організовувати ієрархічне представлення імен тегів;*

– *спостерігати значення тегів;*
– *управляти правами доступу до OPC серверу.*

У відповідності до стандарту, OPC сервер під час інсталяції автоматично реєструється в реєстрі Windows. Запуск сервера здійснюється аналогічно до будь-якої іншої програми або автоматично з клієнтської програми.

OPC DA – Архітектура систем:

В якості OPC клієнта може виступати програма на мові C++ (SCADA-пакет) або програма на мові Visual Basic, VBA, Delphi або будь-яка інша, яка підтримує впровадження COM-об'єктів. Програма на мові C++ взаємодіє з OPC сервером через інтерфейс OPC Custom, а програма на Visual Basic, VBA, Delphi - через інтерфейс автоматизації OPC Automation. OPC сервер и OPC клієнти можуть працювати тільки на комп'ютерах і контролерах с ОС, які підтримують технологію DCOM (наприклад, Windows XP и Windows CE).

Приклади архітектури систем, які включають OPC сервери и OPC клієнти представлені на рисунках нижче (рисунки 10.3, 10.4).

OPC сервер підключається до фізичних пристроїв будь-яким чином; дані засоби стандартом не передбачені. Клієнтська програма і OPC сервер можуть бути встановлені на одному и тому ж комп'ютері на різних комп'ютерах мережі Ethernet

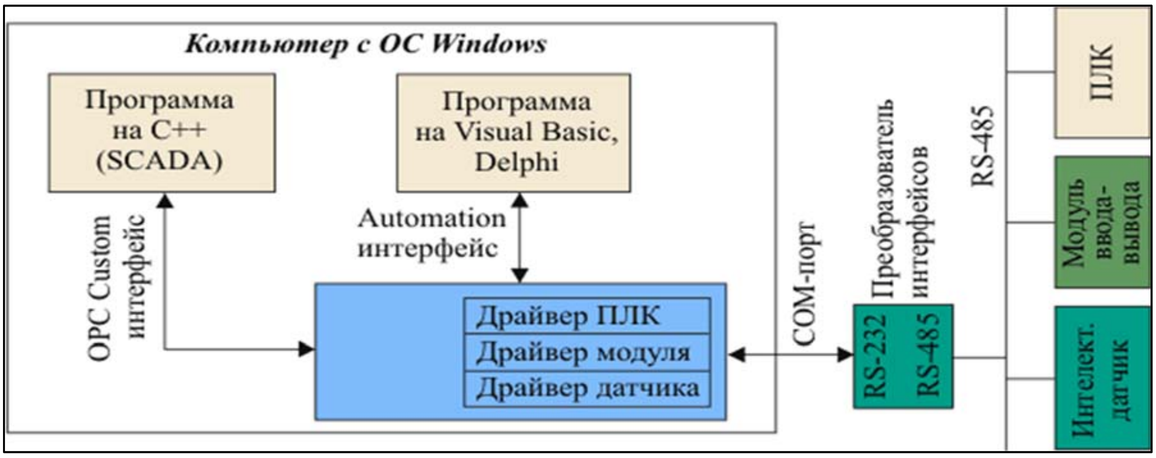


Рисунок 10.3 - Приклад взаємодії прикладних програм і фізичних пристроїв через OPC сервер на одному комп'ютері

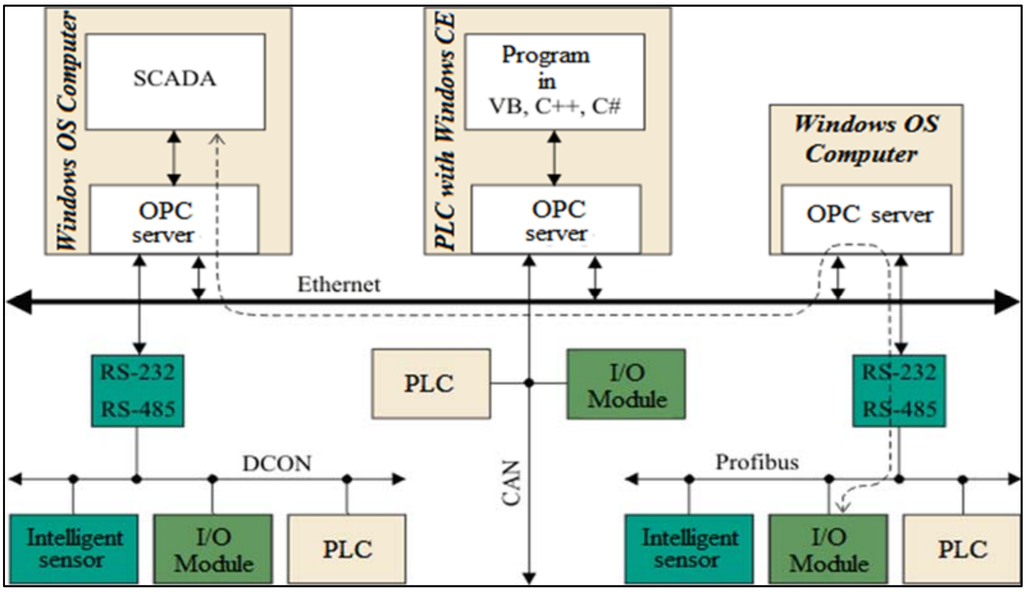


Рисунок 10.4 - Пример применения OPC технологии для сетевого доступа к данным в системах автоматизации

За наявності декількох комп'ютерів кожен із них може містити OPC сервер і під'єднані до нього фізичні пристрої. У такій системі будь-який OPC клієнт із будь-якого комп'ютера може звертатися до будь-якого OPC сервера, зокрема до розташованого на іншому комп'ютері мережі. Це досягається завдяки технології DCOM, що використовує віддалений виклик процедур (RPC — Remote Procedure Call).

Система SCADA може звернутися за даними до модулю вводу-виводу за шляхом, який вказаний на рисунку 4 штриховою лінією. Важливо, що комп'ютери і контролери в такій архітектурі можуть працювати з різними промисловими мережами. Обмін даними з ПЛК, що працюють з ОС Windows, виконуються аналогічно, як з комп'ютерами.

У разі використанні обладнання різних виробників на комп'ютері (контролері) може бути встановлено кілька OPC серверів різних виробників, однак OPC сервер монопольно займає COM-порт комп'ютера (оскільки безперервно виконує оновлення даних), тому кількість портів має дорівнювати кількості OPC серверів. Для нарощування кількості COM портів можна використовувати перетворювачі інтерфейсу USB на RS-232. До різних портів комп'ютера можуть бути під'єднані різні промислові мережі. У такому разі OPC сервери використовуються як міжмережеві шлюзи.

10.3 OPC HDA - Historical Data Access

10.3.1 Огляд

Метою OPC HDA сервера (сервера передісторії процесу) є надання клієнтській програмі єдиного інтерфейсу для обміну даними з будь-якими сховищами даних, у ролі яких може бути нестандартний файл із даними, стандартна СУБД, OPC DA сервер або інший OPC HDA сервер.

Стандарт поширюється тільки на інтерфейси для взаємодії HDA сервера з клієнтськими програмами й не встановлює способів отримання або зберігання даних.

Специфікація OPC HDA встановлює стандарт на інтерфейси COM об'єкта й методи його використання. Структура сервера й методи взаємодії з клієнтами повністю аналогічні загальній ідеології OPC.

Наприклад, OPC клієнт може приєднуватися до кількох OPC HDA серверів різних виробників і бути встановленим на різних комп'ютерах у мережі Ethernet.

Типи серверів HDA:

– простий сервер даних передісторії для побудови графіків (трендів);

– сервер для зберігання даних в упакованому вигляді з можливістю їх обробки і аналізу. (*До функцій обробки і аналізу даних відносяться пошук середнього, мінімального і максимального значень та ін.)*

<u>Робота з даними полягає в читанні, запису або зміні даних</u>

Читання архівних даних проводиться з використанням 3-х різних механізмів. Перший механізм передбачає зчитування архівних даних у певному часовому діапазоні для однієї або декількох змінних із певними іменами. Кількість зчитаних значень обмежується клієнтом. Другий механізм передбачає зчитування архівних даних за часом їхнього відновлення. Третій механізм дає змогу отримувати статистичну інформацію за збереженими даними.

Недоліки OPC технології:

– доступність тільки на операційних системах сімейства Microsoft Windows;

– зв'язок з технологією DCOM, вихідні коди якої є закритими.

Це не дозволяє вирішувати питання надійності ПЗ, а також виявляти і усувати програмні відмови, які виникають;

- можливі проблеми конфігурування, які пов'язані з DCOM;
- неточні повідомлення DCOM про переривання зв'язку;
- непристосованість DCOM для обміну даними через Інтернет;
- непристосованість DCOM для забезпечення інформаційної безпеки.

COM/DCOM:

Технологія COM (Component Object Model) є стандартом, призначеним для створення ПЗ на основі взаємодіючих компонентів, кожен із яких може використовуватися в багатьох програмах одночасно.

Основним поняттям, яким оперує стандарт COM, є COM-компонент.

Дані програми фактично не є автономними програмами, а являють набір взаємодіючих COM-компонентів.

Кожний компонент має унікальний ідентифікатор (GUID) и може одночасно використовуватися багатьма програми. Компонент взаємодіє з іншими програмами через COM-інтерфейси — набори абстрактних функцій і властивостей.

Кожний COM-компонент мусить, принаймні, підтримувати стандартний інтерфейс «IUnknown», який надає базові засоби для роботи з компонентом. Інтерфейс «IUnknown» включає в себе три методи: QueryInterface, AddRef, Release.

Технологія DCOM (Distributed COM — розподілена COM) основана на технології DCE/RPC. DCOM дозволяє COM-компонентам взаємодіяти один з одним по мережі.

Для виклику віддаленого об'єкта параметри мають бути вилучені зі стека (або з регістрів процесора), поміщені в буфер і передані через мережу.

Маршалинг **-** Процес вилучення параметрів і поміщення їх у буфер**.** Цей процес нетривіальний, оскільки параметри можуть містити вказівники на масиви і структури, які, так само, можуть містити вказівники на інші структури.

Демаршалинг - проводиться на сервері як зворотний процес відтворення стека, після чого викликається необхідний об'єкт. Після завершення виклику проводиться маршалінг значення, що повертається і вихідних параметрів та відправка їх клієнту.

Для виконання маршалінга й демаршалінга необхідно мати точний опис методу, включно з усіма типами даних і розміри масивів. Для опису використовується мова опису інтерфейсів (IDL), що входить у стандарт DCE RPC. Отримані файли опису компілюються спеціальним компілятором IDL у вихідний код на мові C, що виконує маршалінг і демаршалінг для зазначених інтерфейсів. Код, що запускається на стороні клієнта, називається «проксі», на стороні об'єкта — «стаб», і завантажується бібліотекою COM за потребою.

Протокол DCOM, відомий як об'єктний RPC (ORPC), є розширенням протоколу DCE RPC. ORPC використовує стандартні пакети RPC з додатковою, необхідної для DCOM інформацією. Заголовок виклику містить ідентифікатор вказівника інтерфейсу (IPID), який використовується для ідентифікації необхідного інтерфейсу необхідного об'єкта на сервері, а параметри починаються з додаткового неявного аргументу.

Клієнт має періодично підтверджувати свою активність шляхом «пінгування» сервера. Якщо період пінгування закінчився без отримання «пінгу», вважається, що клієнт завершив роботу аварійно й усі його посилання на інтерфейси об'єкта знищуються.

10.4 OPC UA - Unified Architecture

10.4.1 Огляд

У 2006 році OPC Foundation запропонувала стандарт специфікації для обміну даними в системах промислової автоматизації "**OPC Unified Architecture**" - **OPC з уніфікованої архітектурою**.

Стандарт OPC UA встановлює методи обміну повідомленнями між OPC сервером і клієнтом, які не залежать від апаратно-програмної платформи, від типу взаємодіючих систем і мереж.

OPC UA забезпечує:

– надійну и безпечну комунікацію;
– протидію вірусним атакам;
– гарантує ідентичність інформації клієнта і сервера.

Використовується поняття **об'єкту** тобто фізичний або абстрактний елемент системи.

Об'єкт (ООП) - визначається як екземпляр класу, а клас розглядається як тип даних. Об'єкти включають в себе змінні, події і методи.

Прикладами об'єктів можуть бути фізичні пристрої, системи і підсистеми, які їх включають. Датчик температури, наприклад, може бути представлений як об'єкт, який включає в себе значення температури, набір параметрів сигналізації та границі їх спрацьовування.

10.4.2 Формат даних

Основні формати даних:

– бінарні структури;

– XML документи.

Формат даних може визначатись постачальником OPC сервера або стандартом. Для роботи з довільними форматами клієнт може запросити в сервера інформацію про опис цього формату. У багатьох випадках використовується автоматичне визначення формату даних під час їхнього передавання.

OPC UA володіє високою ечутливістю даних до завад, збоїв, обчислювальних помилок та повідомлень про події.

Це забезпечується механізмом швидкого виявлення помилок комунікації і відновлення даних.

Сервери можуть мати доступ як до поточних, так і архівованих даних, до подій і аварійних сигналів.

OPC UA можна впроваджувати у різні комунікаційні протоколи, а дані можуть бути закодовані способами, оптимальними за співвідношенням ефективності й можливості перенесення на інші платформи.

10.4.3 Service Oriented Architecture

Основною відмінністю OPC UA від OPC є відмова від технології COM і DCOM фірми Microsoft і перехід до архітектури SOA (*Service Oriented Architecture — «Архітектура, орієнтована на сервіси»*).

Мета: обмін інформацією та забезпечення сумісності с безліччю різних апаратно-програмних платформ.

Сервис в OPC UA - деяка функціональність, укладена в програмному компоненті, який може транспортуватися від сервера до клієнта або назад і викликатися віддалено.

Виклик сервісу аналогічний виклику методу в мовах об'єктно-орієнтованого програмування. Інтерфейс між клієнтом OPC UA і сервером визначається як набір сервісів.

Завдяки побудові сервера OPC UA на основі сервісів з'явилась можливість змінювати розмір (масштабувати) сервери для його використання на платформах з різними обчислювальними ресурсами: для вбудованих застосувань може бути використаний скорочений набір сервісів, для корпоративних мережних серверів - повний набір.

Основним принципом SOA є незалежність від програмної технології, від обчислювальної платформи, від мов програмування, від конкретних застосунків, а також організація сервісів як слабосвязанних компонентів для побудови систем.

Сервіси включають у себе засоби для забезпечення інформаційної безпеки.

10.4.4 Сервіси

Сервіси OPC UA поділяються на логічні групи:

– сервіси безпечних каналів;
– сервіси сесій взаємодії застосунків з ініціативи користувача;
– сервіси для керування вузлами. Дають змогу клієнтам додавати, модифікувати або видаляти вузли в адресному просторі;
– сервіси видимості вузлів, що дають змогу задавати індивідуальні набори видимих вузлів для різних клієнтів;
– сервіси атрибутів дають змогу модифікувати атрибути вузлів;
– сервіси методів, які викликають функції, виконувані елементами системи;
– сервіси для моніторингу вузлів у режимі підписки. Ці сервіси періодично контролюють змінні, атрибути й події, а також генерують повідомлення у разі настання заданих умов;
– сервіси для здійснення підписки й публікації повідомлень.

Відмова від DCOM стала можливою завдяки появі нових транспортних механізмів, заснованих на SOAP, XML, HTTP и сервісах. Завдяки їм OPC UA дозволяє здійснити безпечну і надійну доставку інформації та об'єднати в одному сервері функціональність OPC DA, OPC HDA и OPC A&E серверів.

Стандарт OPC UA не призначений для заміни існуючих OPC специфікацій, проте доповнює і розширює їх можливості.

Для забезпечення інформаційної безпеки використовуються стандартні Web-сервіси безпеки такі як WS-Security, WS-Trust або WS-SecureConversation.

Засоби безпеки: від простої аутентифікації за допомогою паролю і обміну цифровими підписами до повного шифрування повідомлень.

OPC повідомлення передаються за допомогою повідомлень SOAP у вигляді XML тексту.

Оскільки кодування і декодування текстового формату займає достатньо багато часу, стандарт передбачає альтернативний спосіб представлення інформації у вигляді бінарного файлу.

10.4.5 Переваги

1. Реалізація на мові програмування ANSI C для забезпечення можливості перенесення на інші платформи, включно з вбудованими системами. Доступні також версії на.NET і Java (від OPC Foundation);
2. Орієнтація на сервіси замість орієнтації на об'єкти, що дає змогу використовувати OPC UA на будь-яких комп'ютерах, вбудованих системах, у комунікаторах і т.і. які використовують веб-сервіси;
3. Дає змогу здійснити масштабування OPC UA, тобто змінення обсягу програми залежно від обчислювальних ресурсів процесора й необхідної функціональності. Можливе також виконання компіляції як однопотокового або багатопотокового застосунку;
4. Підтримка надійного й сучасного транспортного механізму SOAP на базі XML із застосуванням HTTP протоколу;
5. Забезпечення доброї інформаційної безпеки;
6. Таймаут конфігурується для кожного сервісу;
7. Використання відкритих стандартів World Wide Web Consortium (W3C) замість закритого стандарту COM/DCOM.

10.4.6 Концепция системы на базе OPC UA

OPC UA клієнт і сервер можуть бути скомбіновані в одному застосуванні для взаємодії с іншими OPC UA клієнтами и серверами (рисунок 5).

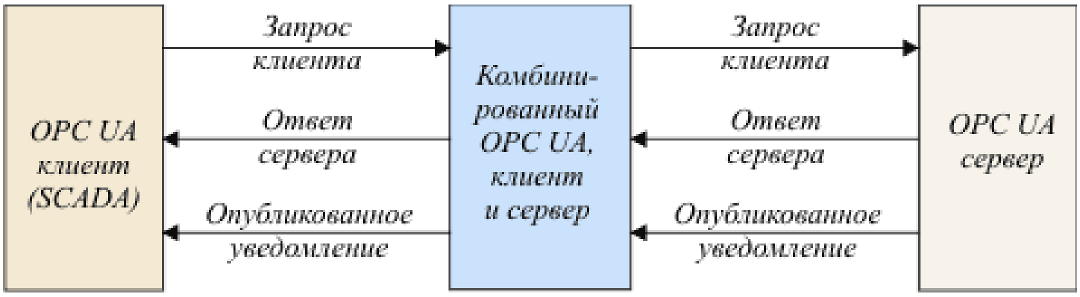


Рисунок 10.5 - Концепція системи на базі OPC UA

Система на базі OPC UA може включати багато клієнтів и серверів. Кожний клієнт може працювати паралельно с декількома серверами и кожний сервер може обслуговувати декілька клієнтів.

Користувальницьке застосування (SCADA) може створювати комбіновані групи клієнтів и серверів для ретрансляції повідомлень, якими воно обмінюється з іншими клієнтами и серверами

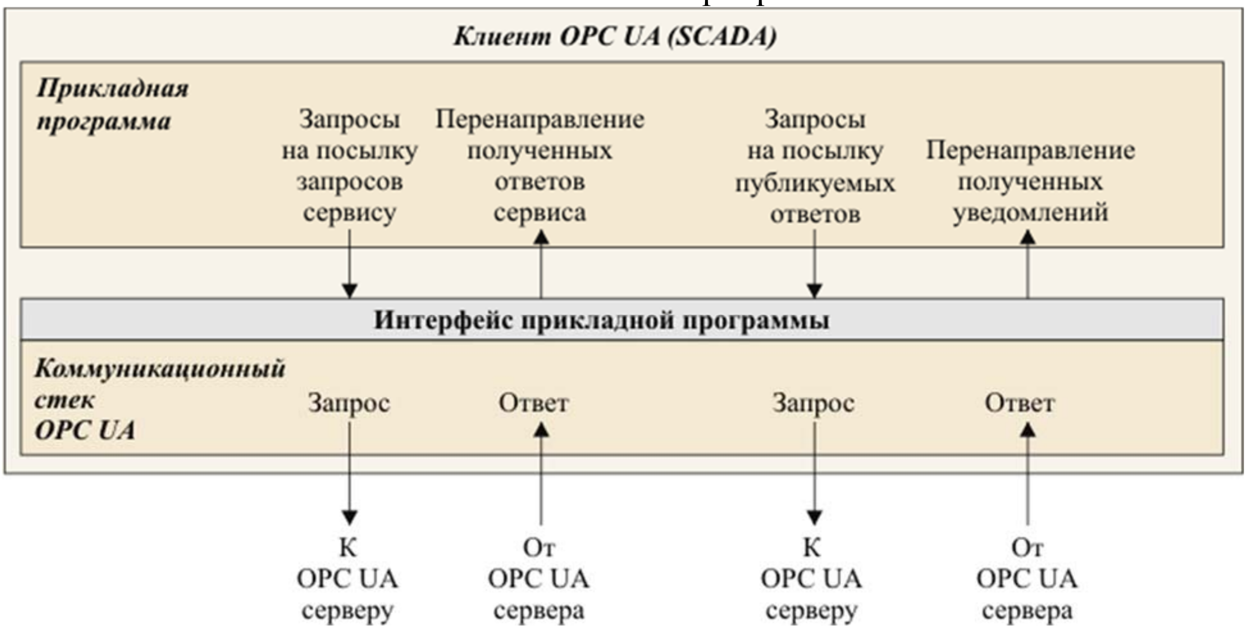


Рисунок 10.6 – Структура клієнта

Клієнтом при взаємодії з OPC сервером є прикладна програма, наприклад SCADA.

Клієнтська програма (рисунок 10.6) виконує запити сервісів OPC сервера через внутрішній інтерфейс, який є ізолюючим прошарком між програмою і комунікаційним стеком. Комунікаційний стек конвертує запити клієнтської прикладної програми в повідомлення для виклику необхідного сервісу, які посилає серверу. Після отримання відповіді на запити комунікаційний стек передає їх в клієнтську програму.

Серверне застосування (рисунок 10.7) представляє програмну реалізацію функцій, які має виконувати сервер.

Взаємодія OPC UA сервера з клієнтом здійснюється через інтерфейс прикладної програми шляхом відправки запитів і отримання відповідей.

Модулі вводу-виводу, ПЛК, інтелектуальні пристрої і програми, які здатні поставляти дані через OPC сервер, позначені як "реальні об'єкти".

Адресний простір OPC серверу є множиною вузлів, доступних клієнтській програмі за допомогою сервісів OPC UA. "Вузли" в адресному просторі використовуються задля представлення реальних об'єктів, їх визначення та перехресних посилань. В адресному просторі виділяється підпростір вузлів, які сервер робить "видимими" для клієнта. Видимі вузли організуються у вигляді ієрархічної структури, для зручності навігації їх клієнтської програмою.

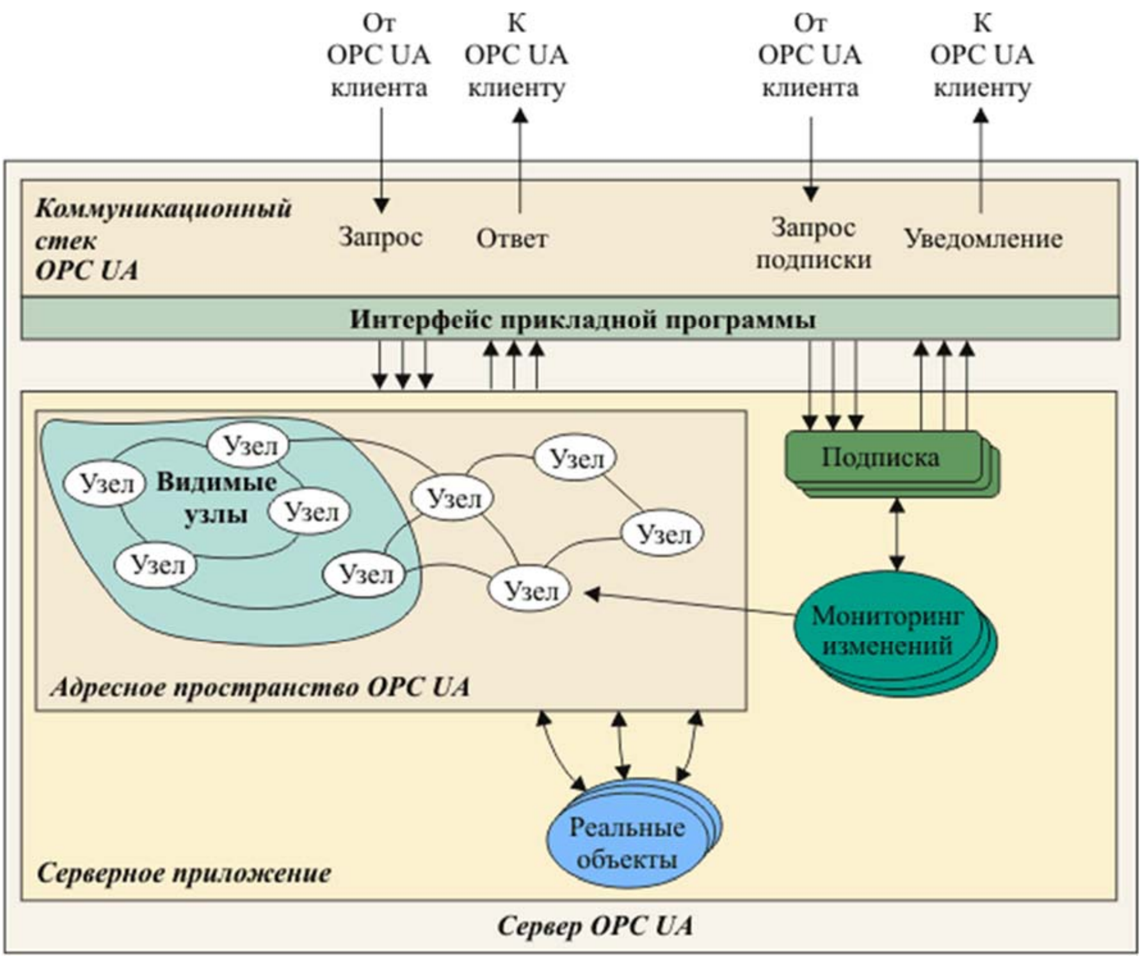


Рисунок 10.7 – Структура OPC UA серверу

Обмін даними між клієнтом и сервером може виконуватись як шляхом отримання миттєвих відповідей на запити, так й за схемою "видавець-передплатник". В другому випадку клієнтська програма здійснює "підписку" на отримання певних даних, які сервер має буде пред'явити по мірі їх появи. Для реалізації режиму підписки сервер здійснює безперервний контроль (моніторинг) вузлів і відповідних їм реальних об'єктів с ціллю виявлення змін. При виявленні змін в даних, подіях або аварійних сигналах (Alarm) сервер генерує повідомлення, яке передається клієнту по каналу підписки.

OPC UA допускає обмін між двома серверами. Для цього один з серверів виконує роль клієнта, другий - роль сервера. Таким чином можна об'єднати кілька серверів ланцюгом, при цьому кожний з них буде виступати з однієї сторони ланцюга у якості клієнта, з іншої сторони - в якості сервера (рисунок 5). Для захисту вже зроблених інвестицій в OPC на базі DCOM організація OPC Foundation розробила стратегію переходу на нову технологію з застосуванням " UA-оболонки", яка допускає обмін даними між старими і новими продуктами. Така оболонка дозволяє, наприклад, DCOM OPC серверу працювати с OPC UA клієнтом, і навпаки.

10.4.7 OPC UA сервер в середі MS Excel

MS Excel в якості середи для розробки нескладних систем автоматизації має такі переваги та недоліки.

Переваги MS Excel:

– містить зручний, призначений для користувача інтерфейс для обробки даних, для побудови графіків, вставки малюнків, виконання анімації і т. п.;
– містить також вбудовані елементи керування (списки, що розкриваються, "радіокнопки", "чек-бокси" і т. п.);
– дозволяє зробити спеціалізований призначений для користувача інтерфейс, який не поступається по дизайну професійним SCADA-пакетів. Особливо зручним є просте збереження виконаної роботи в файл.

Недоліки MS Excel**:**

– низька швидкодія при запису даних в осередку таблиці;
– відсутність в VBA можливості виконання завдань в декількох паралельних потоках.

VBA включає в себе Visual Basic і додаткові функції, що забезпечують роботу з додатком, наприклад, з осередками MS Excel або з параграфами MS Word.

10.4.8 OLE Automation

Можливість роботи MS Excel (і інших додатків Microsoft Office) з OPC сервером забезпечується завдяки тому, що Visual Basic, що входить до складу MS Excel, підтримує технологію Automation (OLE Automation) фірми Microsoft.

OLE Automation дозволяє одному застосуванню (клієнту) використовувати об'єкти іншої програми (сервера). Це означає, що додаток-клієнт може використовувати об'єкти OPC сервера, який, в свою чергу, може отримувати дані з фізичного пристрою або записувати їх в нього.

Automation дозволяє використовувати в середовищі MS Excel також об'єкти інших додатків, наприклад, ActiveX об'єкти, об'єкти MS Word, Outlook Express та ін. Інтерфейс Automation дозволяє використовувати OPC сервер з будь-якими іншими додатками, що мають вбудований Visual Basic for Application.

Для підключення OPC сервера до MS Excel спочатку необхідно інсталювати на комп'ютері OPC сервер. Стандартний OPC сервер повинен реєструватися в реєстрі Windows автоматично. Потім потрібно відкрити пункт меню Tools / References в редакторі Visual Basic, знайти в списку "Available References" і відзначити галочкою посилання на ім'я OPC сервера. Після цього в списку об'єктів "Object Browser" з'являється список об'єктів OPC сервера і їх можна використовувати за загальними правилами програмування на Visual Basic. Більш детальну інформацію про роботу з об'єктами OLE Automation можна ознайомитися в керівництві по програмуванню.

Приклад ієрархії об'єктів інтерфейсу OLE Automation представлений на рисунку 8. Значення даних, лічених з пристрою введення, параметр якості даних і тимчасова мітка зберігаються в об'єкті OPCItem.

Кілька OPCItem можуть становити колекцію.

Колекція - це термін VBA, вона являє собою групу пов'язаних об'єктів, кожен з яких називається елементом колекції та може бути викликаний по його індексу.

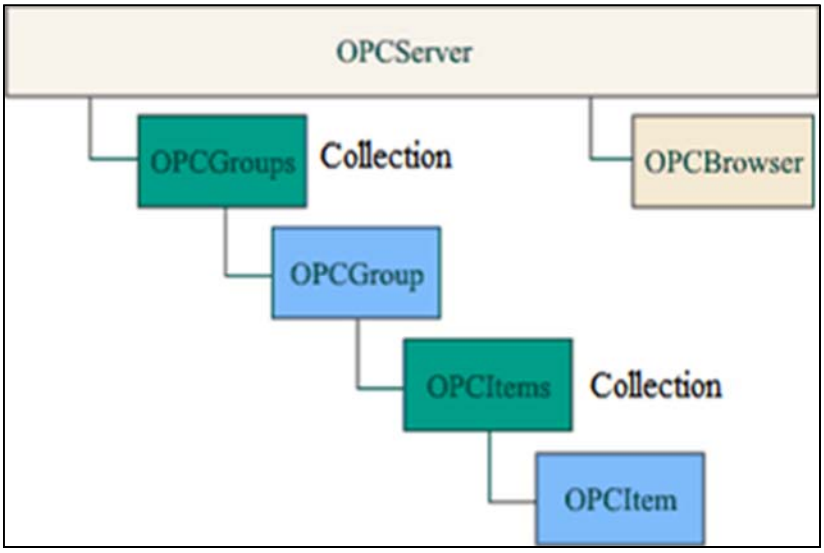


Рисунок 10.8 – Иерархия объектов интерфейса Automation

Кожна колекція має властивість Count, яка дозволяє підрахувати кількість елементів.

Об'єкт OPCServer є екземпляром OPC сервера. Цей об'єкт повинен бути створений до того, як будуть встановлені посилання на інші об'єкти. Він містить колекцію OPCGroups і створює об'єкт OPCBrowser, який використовується в OPC клієнті для побудови навігатора тегів.

10.5 Інтерфейс користувача - SCADA системи

Більшість систем автоматизації функціонує за участю людини (оператора, диспетчера). Інтерфейс між людиною і системою називають людино-машинним інтерфейсом (ЛМІ), в зарубіжній літературі - HMI (Human-Machinery Interface) або MMI (Man-Machinery Interface). А також коли ЛМІ призначений для взаємодії людини з автоматизованим технологічним процесом його називають SCADA-системою (Supervisory Control And Data Acquisition - "диспетчерське управління і збір даних").

Сучасні SCADA-пакети включають в себе широкий набір функціональних можливостей, що далеко виходить за рамки збору даних і диспетчерського управління.

Функції SCADA - групи:

– налаштування SCADA на конкретну задачу (розробка програмної частини системи автоматизації);

– диспетчерське управління;
– автоматичне керування;
– зберігання історії процесів;
– виконання функцій безпеки;
– виконання загальносистемних функцій.

Основна відмітна ознака - наявність інтерфейсу з користувачем.

При відсутності такого інтерфейсу, перераховані вище функції збігаються з функціями засобів програмування контролерів, а управління є автоматичним, на противагу диспетчерського.

Властивості SCADA:

– комфорт робочого місця,
– зрозумілість інтерфейсу,
– наявність підказок і блокування явних помилок оператора.

Подальший розвиток здійснюється в напрямку поліпшення ергономіки та створення експертних підсистем.

10.5.1 – Функції

Поняття **Alarm** і **подія**.

Подія - це вимірювання деяких станів у системі.

Наприклад, включення перевалки зерна в елеваторі, завершення циклу періодичного процесу обробки деталі, закінчення завантаження бункера, реєстрація нового оператора і т. п.

Події не вимагають термінового втручання оператора, а просто інформують його про стан системи.

Alarm ("сигнал тривоги") – попередження про важливу подію, у відповідь на яке потрібно терміново зробити деякі дії.

Наприклад, досягнення критичної температури зберігання зерна в елеваторі, після якого починається його загоряння, досягнення критичного значення тиску в автоклаві, після якого можливий розрив оболонки, спрацьовування датчика відкриття двері, перевищення допустимого рівня загазованості в котельні тощо.

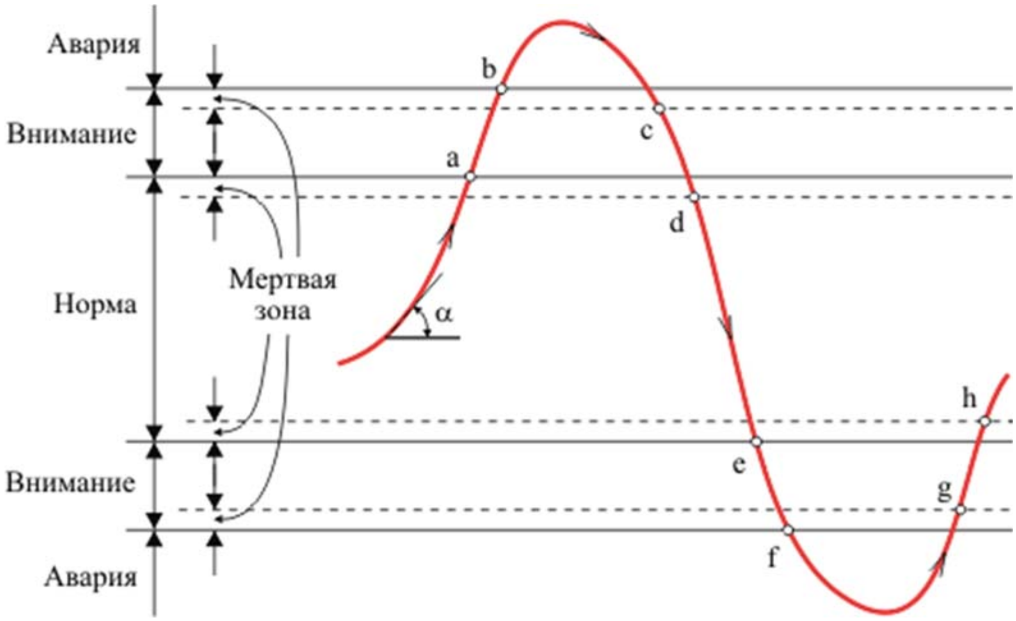


Рисунок 10.9 – Приклад виключної ситуації

Alarm:

– підтвердженні;

– не підтвердженні.

Підтверджений Alarm – це Alarm, у відповідь на який оператор ввів команду підтвердження. До цього моменту Alarm вважається непідтвердженим.

Сигнали Alarm діляться:

– дискретні

– аналогові

Дискретні сигналізують про зміну дискретної змінної;

Аналогові з'являються, коли неперервна змінна входить до завчасно Приклад (рисунок 9): Розподіл всього інтервалу зміни змінної y(t) на інтервали "Норма", "Увага" (передаварійний стан) і "Аварія"

Кожна критична межа має зону нечутливості (мертву зону), яка потрібна для того, щоб після зняття стану ALARM змінна не могла повернутися в нього внаслідок випадкових викидів в системі (шумів). Межі зон можуть змінюватися з плином часу.

Сигнал "Внимание" виникає при $y(a)<y(t)<y(b)$ під час наростання змінної, при $y(d)<y(t)<y(c)$ під час її зменшення;

Сигнал "Авария" виникає при $y(b)<y(t)$

Методика видачі сигналів Alarm повинна бути надійною. Зокрема, спливаючі вікна з повідомленнями сигналів Alarm повинні бути завжди поверх інших вікон, сигнали Alarm можуть дублюватися звуком і світлом. Оскільки сигналів Alarm в системі може бути багато, їм призначають різні пріоритети, різні гучності і тони звукового сигналу і т. п.

Розробка людино-машинного інтерфейсу

Однією з основних функцій SCADA є розробка людино-машинного інтерфейсу (тобто SCADA одночасно є і ЧМІ, і інструментом для його створення).

У процес розробки входять наступні операції:

- створення графічного інтерфейсу (мнемосхем, графіків, таблиць, спливаючих вікон, елементів для введення команд оператора і т д.);
- програмування і налагодження алгоритмів роботи системи автоматизації. Багато SCADA дозволяють виконувати налагодження системи як в режимі емуляції обладнання, так і з підключеним обладнанням;
- настройка системи комунікації (мереж, модемів, комунікаційні контролерів і т п.);
- створення баз даних і підключення до них SCADA.

SCADA як система диспетчерського керування

Задачі:

- взаємодія з оператором (видача візуальної і слухової інформації, передача в систему команд оператора);
- допомога оператору в прийнятті рішень (функції експертної системи);
- автоматична сигналізація про аварії і критичних ситуаціях;
- видача інформаційних повідомлень на пульт оператора;
- ведення журналу подій в системі;
- вилучення інформації з архіву та подання її оператору в зручному для сприйняття вигляді;
- підготовка звітів (наприклад, роздруківка таблиці температур, графіків зміни операторів, переліку дій оператора);
- облік напрацювання технологічного устаткування.

SCADA як частина системи автоматичного керування

Основна частина завдань автоматичного управління виконується за допомогою ПЛК, однак частина завдань може покладатися на SCADA.

Крім того, у багатьох невеликих системах управління ПЛК можуть взагалі бути відсутнім і тоді комп'ютер зі встановленою SCADA є єдиним засобом управління.

SCADA виконує такі завдання автоматичного управління:

– автоматичне регулювання;
– управління послідовністю операцій в системі автоматизації;
– адаптація до зміни умов протікання технологічного процесу;
– автоматичне блокування виконавчих пристроїв при виконанні заздалегідь заданих умов

Зберігання історії процесу

Знання передісторії керованого процесу дозволяє поліпшити майбутнє поведінку системи, проаналізувати причини виникнення небезпечних ситуацій або браку продукції, виявити помилки оператора.

Для створення історії система виконує наступні операції:

– збір даних і їх обробка (цифрова фільтрація, інтерполяція, стиск, нормалізація, масштабування і т. Д.);
– архівування даних (дій оператора, зібраних і оброблених даних, подій, сигналів Alarm, графіків, екранних форм, файлів конфігурації, звітів тощо);
– управління базами даних (реального часу і архівних).

загальносистемні функції

SCADA зазвичай є єдиною програмою для управління системою автоматизації, на неї покладаються деякі загальносистемні функції:

– здійснення взаємодій між декількома SCADA, між SCADA та іншими програмами (MS Office, базою даних, MATLAB);
– діагностика апаратури, каналів зв'язку і ПЗ.

Безпека SCADA

Застосування SCADA в системах віддаленого доступу через інтернет різко підвищило вразливість SCADA до дій ворожих осіб.

Наприклад, може привести до відмови в роботі мереж електропостачання, життєзабезпечення, зв'язку, відмови морських маяків, дорожніх світлофорів, до зараження води неочищеними стоками тощо.

Для підвищення безпеки SCADA використовують такі методи:

– розмежування доступу до системи між різними категоріями користувачів (у змінного оператора, технолога, програміста і директора повинні бути різні права доступу до інформації та до модифікації налаштувань системи);
– захист інформації (шляхом шифрування інформації і забезпечення секретності протоколів зв'язку);
– забезпечення безпеки оператора завдяки його віддалення від небезпечного керованого процесу (дистанційне керування)
– спеціальні методи захисту від кібер-атак;
– застосування міжмережевих екранів

10.5.2 Властивості SCADA

Аналіз властивостей різних SCADA дозволяє вибирати систему, оптимальну для вирішення поставленого завдання.

Групи:

– інструментальні властивості;
– експлуатаційні властивості;
– властивості відкритості;
– економічна ефективність

Експлуатаційні властивості

Якість SCADA в процесі експлуатації оцінюється кінцевими користувачами і характеризується наступним набором властивостей:

– нечутливість до помилок користувача, захищеність від вандалів і ворожих елементів, стійкість до помилок у вихідних даних;
– надійність;
– інформаційна захищеність;
– наявність засобів збереження даних при нештатних ситуаціях, відключеннях харчування і збоїв;
– наявність автомата перезапуску системи при її зависанні або після переривання харчування;
– підтримка резервування SCADA (операторської станції, мережевих серверів, клієнтських робочих станцій, резервне копіювання даних);

– підтримка перемикання екранів з різною деталізацією зображень;
– підтримка декількох моніторів.

Інструментальні властивості (властивості SCADA, що впливають на ефективність роботи системних інтеграторів):

– швидкість розробки проекту;
– легкість освоєння;
– підтримувані засоби комунікації;
– наявність функцій для складної обробки даних;
– наявність мов МЕК 61131-3 і універсального алгоритмічного мови типу Visual Basic;
– ступінь відкритості для розробника (підтримка COM і ActiveX для підключення програмних модулів користувача, а також OPC, ODBC, OLE DB;
– якість технічної документації (повнота, ясність викладу, кількість помилок);
– наявність режиму емуляції обладнання для налагодження;
– наявність внутрішніх графічних редакторів, що дозволяють відмовитися від застосування зовнішніх редакторів типу CorelDraw або Photoshop; підтримка типових графічних форматів файлів;
– якість технічної підтримки (час реакції на запитання користувачів, наявність "гарячої лінії" технічної підтримки.

Ступінь відкритості

Ступінь відкритості дуже сильно впливає на економічну ефективність системи, однак цей вплив носить випадковий характер, оскільки залежить від ступеня використання властивостей відкритості в конкретному проекті.

Відкритість для програмування користувачем SCADA забезпечується можливістю підключення програмних модулів, написаних користувачем або іншими виробниками. Це зазвичай досягається тим, що SCADA розробляється як контейнер для COM-об'єктів і ActiveX елементів.

Сумісність з апаратурою і базами даних інших виробників досягається за допомогою стандарту OPC, застосуванням інтерфейсу ODBC або OLE DB. Відкритість системи програмування досягається підтримкою мов МЕК 61131-3.

Особливо цікаво з точки зору відкритості застосування веб-інтерфейсу, оскільки він забезпечує доступ до SCADA з будь-якого комп'ютера, з будь-якої точки світу, незалежно від апаратної платформи, типу каналу зв'язку, операційної системи і використовуваного веб-навігатора

Економічна ефективність

Економічна ефективність SCADA визначається як відношення економічного ефекту від її впровадження до загальної суми витрат на впровадження і підтримку системи в працездатному стані.

На економічну ефективність впливають всі властивості SCADA, проте можна виділити наступні:

- масштабованість (можливість застосування як для великих, так і для малих систем);
- модульність. Дозволяє зробити замовну комплектацію системи в залежності від поставленого завдання. Типовими модулями можуть бути, наприклад, модуль введення-виведення, модуль візуалізації, модуль Alarm, модуль трендів, модуль звітів, модуль комерційного обліку енергоресурсів тощо;
- вартість обслуговування;
- умови оновлення версій;
- надійність постачальника, наявність досвіду практичного застосування;
- вартість навчання;
- вартість технічної підтримки;
- методи ціноутворення.

Система MasterSCADA призначена для створення повномасштабних систем автоматизації в різних галузях промисловості.

Особливість - об'єктний підхід, використаний на рівні опису системи при її налаштування на конкретний об'єкт автоматизації.

Наприклад, цех, ділянка, технологічний блок і фізичний пристрій при створенні проекту за допомогою MasterSCADA розглядаються як окремі об'єкти. Для кожного об'єкта створюється свій опис на технологічної мові програмування. Опис включає в себе властивості об'єкта і документи об'єкта. Властивостями можуть бути період опитування, спосіб лінеаризації датчика, діапазон вхідних сигналів. Документами об'єкта є його зображення, мнемосхема, графік зміни змінних тощо. Будь-який документ в системі відноситься до деякого об'єкту. Такий підхід дозволяє легко розмножувати один раз створені об'єкти, що підвищує швидкість настройки SCADA на завдання користувача.

До ознак об'єктного підходу відноситься також можливість успадкування всіх налаштувань від "батьківських" об'єктів.

Trace Mode - SCADA-система, яка складається з інструментальної системи і набору виконавчих модулів.

До складу TraceMode 6 входять засоби управління бізнес-процесами виробничого підприємства.

Для збільшення швидкості розробки проекту користувача застосовується технологія авто побудови (дозволяє знизити кількість помилок, що допускаються користувачем при ручному створенні проекту)

Автоматично в SCADA можуть бути побудовані:

- джерела даних ПЛК і модулів вводу-виводу за відомою конфігурації;
- канали за джерелами даних;
- зв'язку каналів з редактора аргументів;
- зв'язку контролер-сервер і сервер-сервер;
- SQL-запити;
- зв'язку з OPC-сервером;
- зв'язок з ODBC.

У систему TraceMode 6 входить п'ять мов програмування - Techno SFC, Techno LD, Techno FBD, Techno ST, і Techno IL, розширення відповідних мов стандарту МЕК 61131-3.

10.6 Вправи

10.6.1 Вправа 1 - Конфігурація OPC сервера за допомогою додатків програмного комплексу AutomationWorX від PhoenixContact

Покрокова інструкція створення сервера:

1. Створити програму за допомогою PC WorX додатки з програмного комплексу AutomationWorX.
2. Конфігурувати входи / виходи. Відзначити властивості PDD і OPC (рис. 10.10).

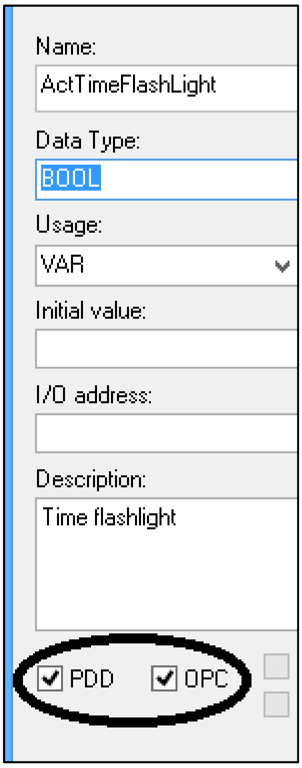


Рисунок 10.10 – Конфігурування входів/виходів

3. Запустить застосування OPC Configurator від AutomationWorX і створити з'єднання с ПЛК (рис. 10.11).

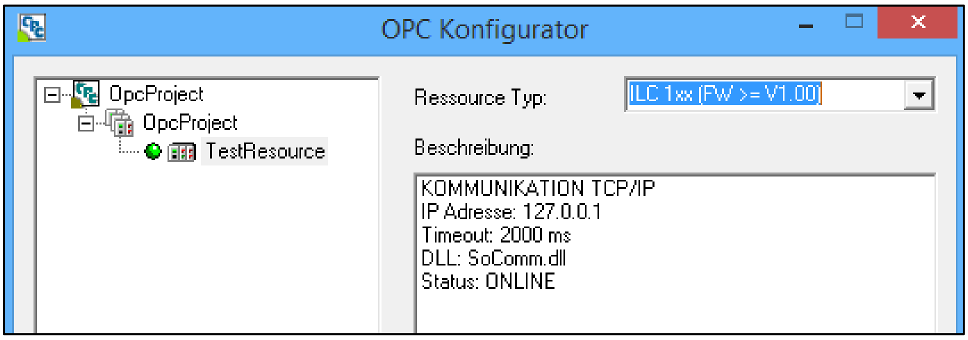


Рисунок 10.11 – Окно приложения OPC Configurator

4. Відкрити застосування OPC Test Client из AutomationWorX і вибрати "*Connect to Server*" (рис. 10.12).

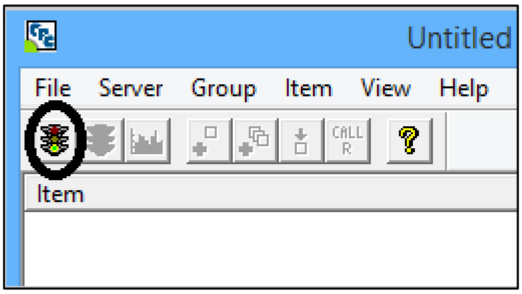


Рисунок 10.12 – Вікно застосування OPC Client: З'єднання с сервером

5. Вибрати "*Add all items*" (рис. 10.13).

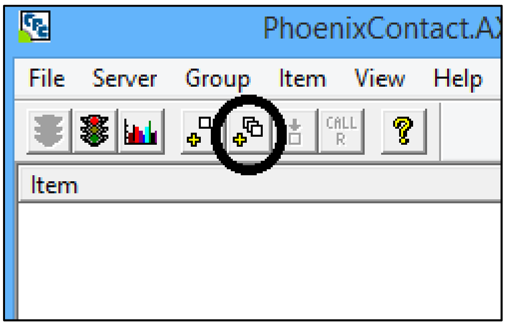


Рисунок 10.13 – Вікно застосування OPC Client: add all items (додати всі елементи)

6. Конфігурування завершено. Можна спостерігати роботу програми на клієнтському додатку. Для тестування, потрібно змінити значення змінних і спостерігати їх відображення на клієнті.

7. На наступному кроці слід розробити візуалізацію програми. Для візуалізації програми можна використовувати одну з двох програм з комплексу програм AutomationWorx: WebVisit або Visu +.

10.6.2 Вправа 2 - Конфігурування OPC сервера за допомогою застосування CoDeSys

1. Інсталювати та відкрити CODESYS Control Win V3.
2. В проекті потрібно створити символьну конфігурацію (рис. 10.14).

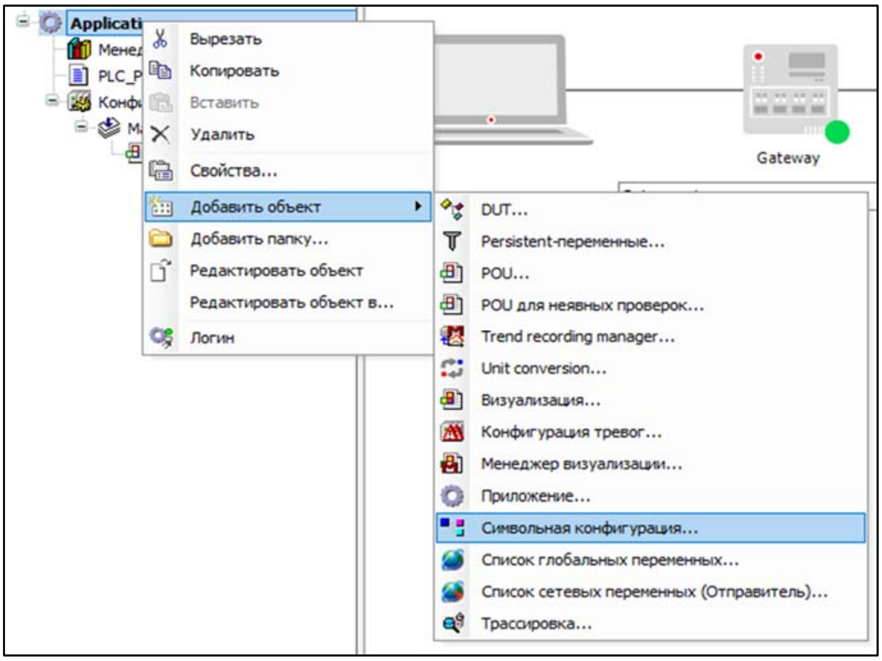


Рисунок 10.14 – Створення символьної конфігурації.

3. Виділити змінні, які будуть переміщені і відображуватись на сервері (рис. 10.15).

Рисунок 10.15 – Створення змінних

4. Запустити програму в CoDeSys.
5. Відкрити CoDeSys OPC Configuration (рис. 10.16).

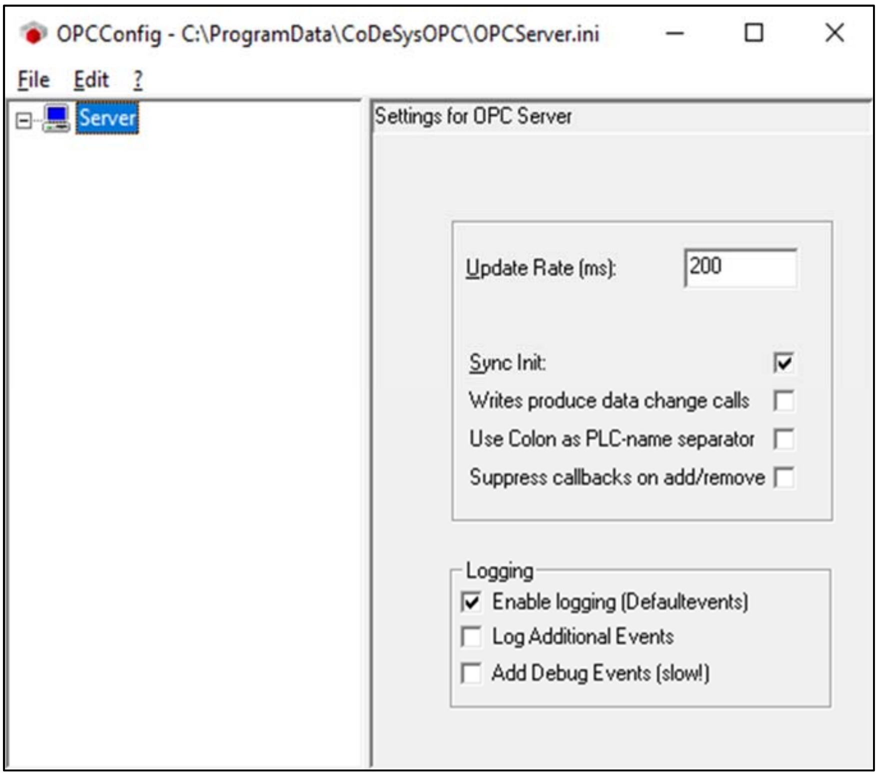


Рисунок 10.16 – Вікно застосування OPC Configuration

6. Підключитися до ПЛК (рис. 10.17).

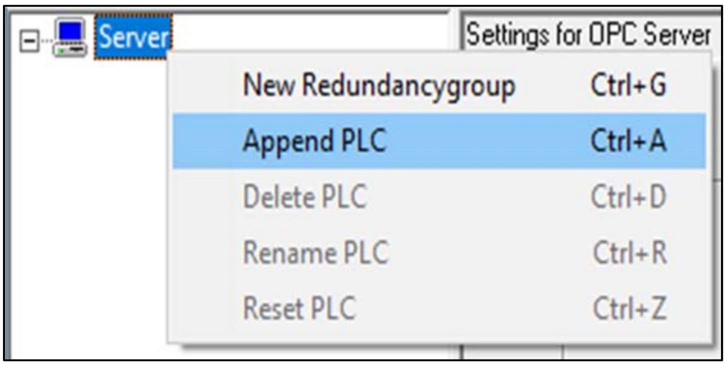


Рисунок 10.17 – З'єднання с ПЛК

7. Після підключення слід вибрати: Server → PLC1 → Connection →Edit… Вибрати потрібні параметри для підключення (рис. 10.18).

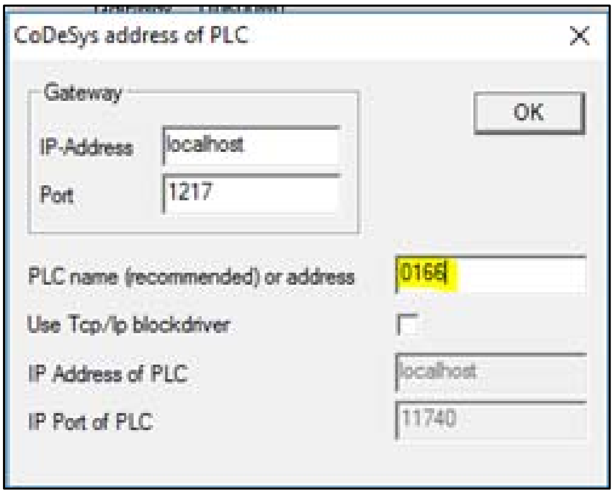


Рисунок 10.18 – Вибір мережних параметрів

8. Вибрати ім'я ПЛК у вікні Network scanning (рис. 10.19)

Рисунок 10.19 – Сканирование сети

8.1 Якщо немає підключення до ПЛК в реальному часі, то можна створити підключення до віртуального ПЛК і працювати з OPC сервером офлайн. Для цього потрібно зробити наступні кроки.

<u>По-перше</u>: створити стандартний проект в CoDeSys V3. (рис . 10.20)

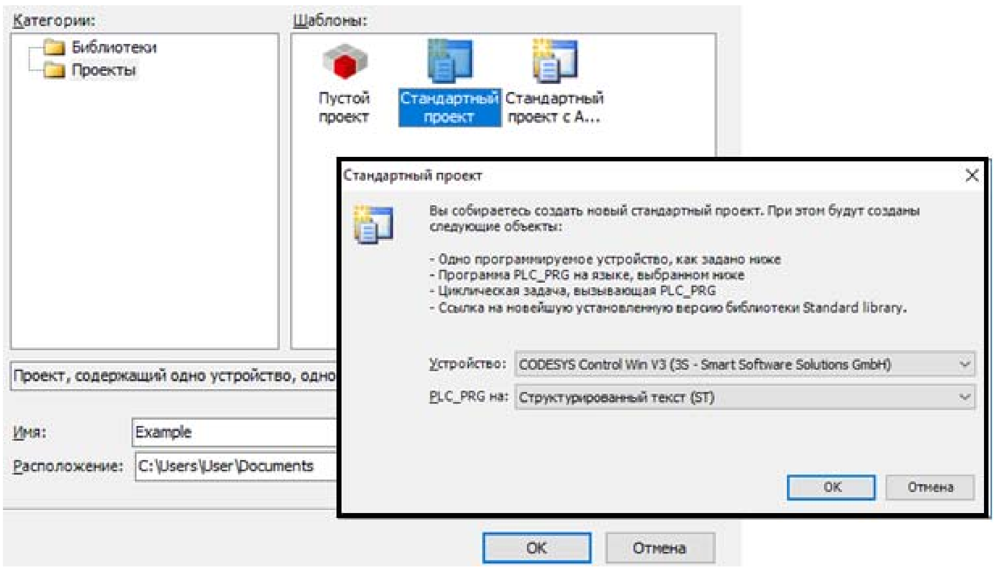


Рисунок 10.20 – Створення нового проекту

<u>По-друге</u>: створити просту програму (рис. 10.21).

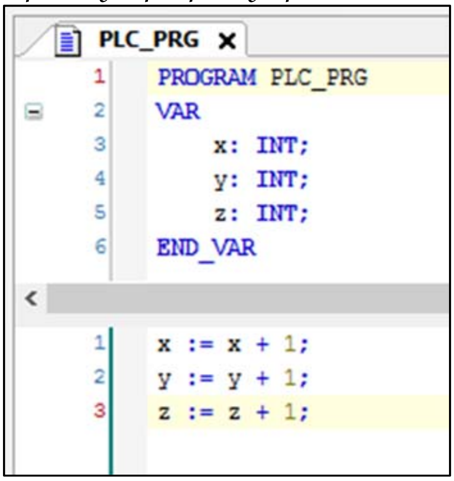


Рисунок 10.21 – Створення простої програми

<u>По-трете</u>: Вибрати Gateway. Це буде виглядати наступним чином:

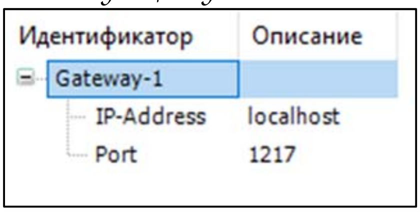


Рисунок 10.22 – Вибір gateway

Якщо необхідно, то слід створити нову конфігурацію з необхідними параметрами.

<u>По-четверте</u>: Вибрати на панелі задач (CODESYS Win Control SysTray) і знайти наступне застосування, запустити контекстне меню и запустити ПЛК.

<u>По-п'яте</u>: Повернутись в CoDeSys, вибрати Device selection і вибрати Scan network (рис. 10.23). Вибрати пристрій та натиснути OK.

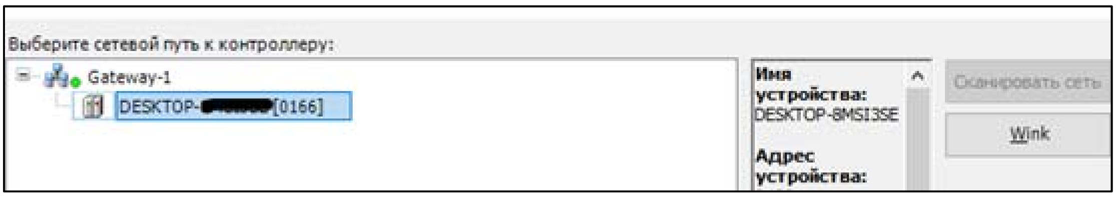


Рисунок 10.23 – Сканування сети

Потім показано як працює пристрійкоректно, можна з'єднати їх і можна також провести тестування програми (рис. 10.24)

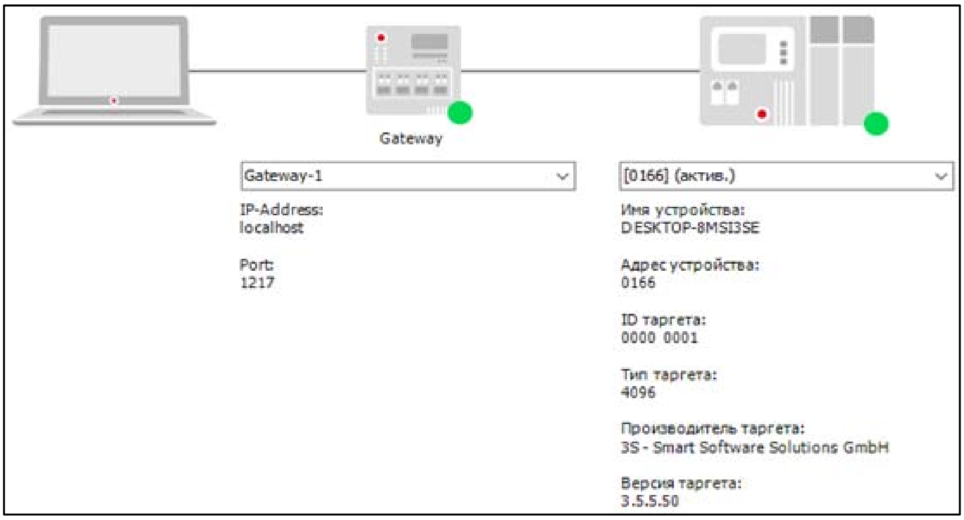


Рисунок 10.24 – Работа устройства

9. Тестування. Відкрити MatrikonOPC Explorer. Вибрати CoDeSys.OPC.DA. Вибрати "Connect to selected OPC Server" (рис. 10.25).

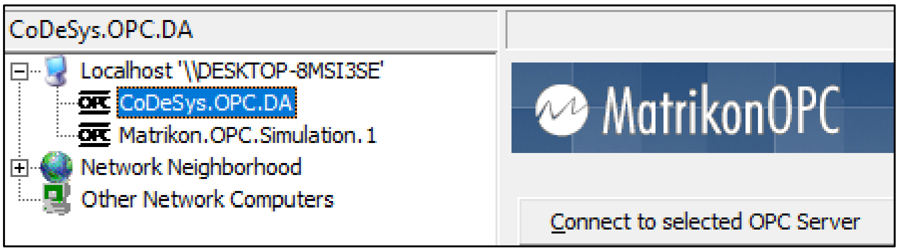


Рисунок 10.25 – Підключення до вибраного OPC Серверу

10. Конфігурування. Вибрати "Add items to selected OPC Server". Вибрати ім'я, оновити частоту і час. Натиснути кнопку OK (рис. 10.26).

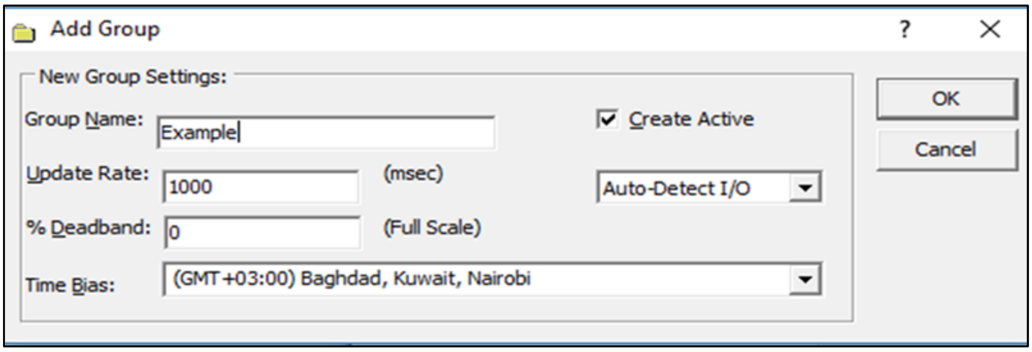


Рисунок 10.26 – Конфігурація нової групи

11. Відкрити нове вікно (рис. 10.27). Можна побачити створений ПЛК та його модуль PLC_PRG. Також можна побачити позначки які були створені раніше в CodeSys всередині модуля. Додати їх і натиснути кнопку OK.

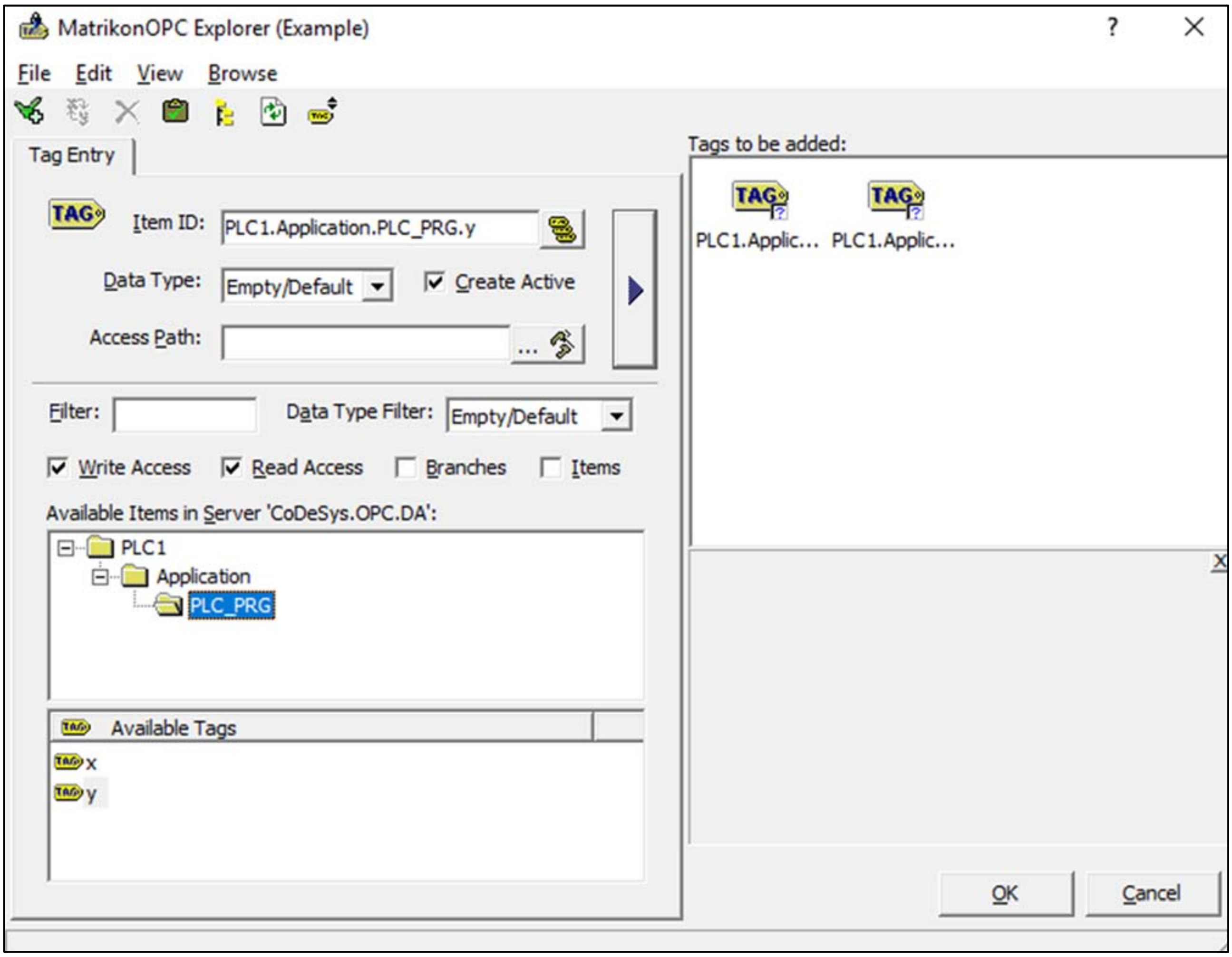


Рисунок 10.27 – Запуск програми

12. Після відкрити головне вікно Matricon Client і можна побачити створені раніше змінні x і y і їх значення (рис. 10.28).

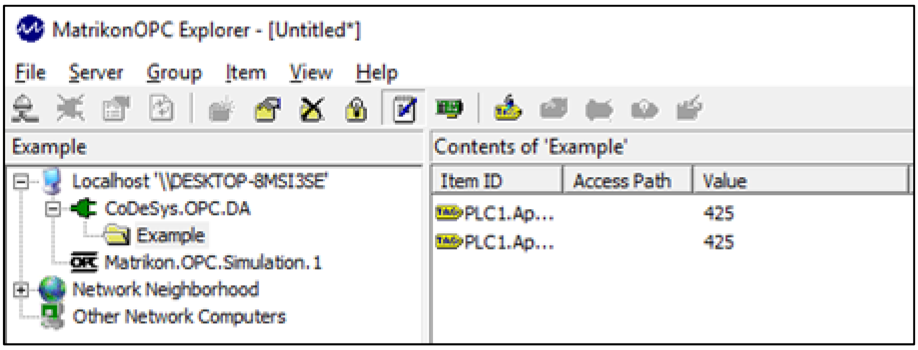


Рисунок 10.28 – Програма в роботі

Контрольні питання

1. Що являє собою стандарт OPC?
2. Які технології входять в OPC стандарт?
3. Коли застосовується стандарт OPC DA?
4. Назвіть область використання OPC HDA?
5. Назвіть особливості OPC UA?
6. Які особливості OPC DA?-
7. Яка архітектура системи OPC DA?
8. Які особливості OPC HDA?
9. Що таке процес маршалингу/демаршалингу?
10. Які особливості COM/DCOM технології?
11. Які особливості OPC UA?
12. Яка модель безпеки представлена в OPC UA?
13. Який формат даних використовується в OPC UA?

ЛАБОРАТОРНИЙ ПРАКТИКУМ

Робота з TATU Smart Lab

Вправа 1: Встановлення систем

1. Опис TSL BOX1 – відкрити TATU_TSL_user manual.
2. Встановити PCWorX версії v.6.20 або більше
3. Опис застосування PCWorX
   3.1 Створити новий проект та вибрати відповідний пристрій (рис. 1)

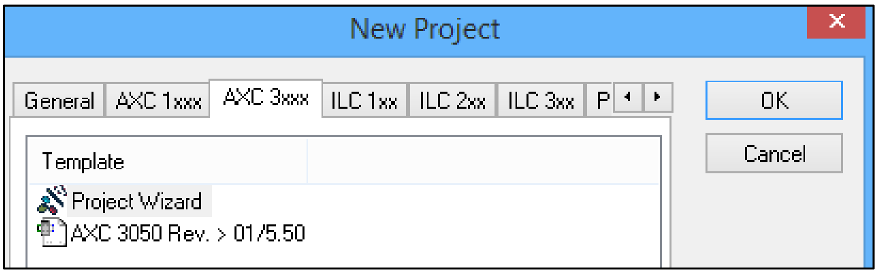


Рисунок 1 – створення нового проекту

   3.2 Вибрати робочу папку: Options – Directories – Set project directory
   3.3 Зберегти проект з вказаним ім'ям
4. Додати (import) необхідні файли опису "Import GSD File…". Додайте файл "GSDML-V2.25-Siemens-ET200S-20140317.xml" з папки TATU\ Attachments\ Software\ PC WORX\ GSD\ im151-3

Завдання до вправи 1. Інсталяція необхідних застосувань на ПК

Вправа 2: Налаштування пристроїв Profinet

<u>Перші кроки в роботі з TSL BOX1 Profinet</u>

1. З'єднайте ваш Ethernet кабель до комутатора PhC.
2. Встановіть IP адресу на вашому ПК.
3. Встановіть IP адресу *вашого* PLC: Відкрийте програму *IPAssign*. Натиснить *Next* та *Перезавантажте* ваш PLC. Перевірте Mac адресу і встановіть IP. Перезавантажте PLC знову (Рис. 2)

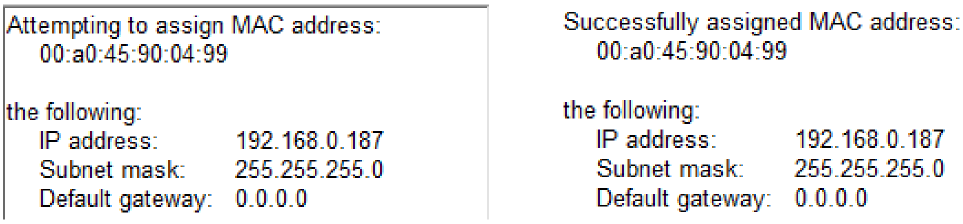


Рисунок 2 – Завдання IP адреси

4. Зайти PLC за вказаною адресую, використовуючи браузер і побачити web сторінку PLC.
5. Відкрити Ваш проект в PCWorX.
6. Натиснути *Bus Configuration*
7. Встановити діапазон IP адрес Вашого проекту.
8. Встановити IP адресу Вашого пристрою (в проекті), яку визначили раніше. Увага! Перевірте LAN інтерфейс.
9. Відкрити *Communication* та *перевірте* з'єднання з Вашим пристроєм.
10. Наявність Profinet з'єднань: *Extras* -> PROFINET Configuration. Встановити права мережного інтерфейсу.
11. Натиснути *Read Profinet* на Вашому Profinet пристрої. Натиснути *Refresh devices* (Рис.3)

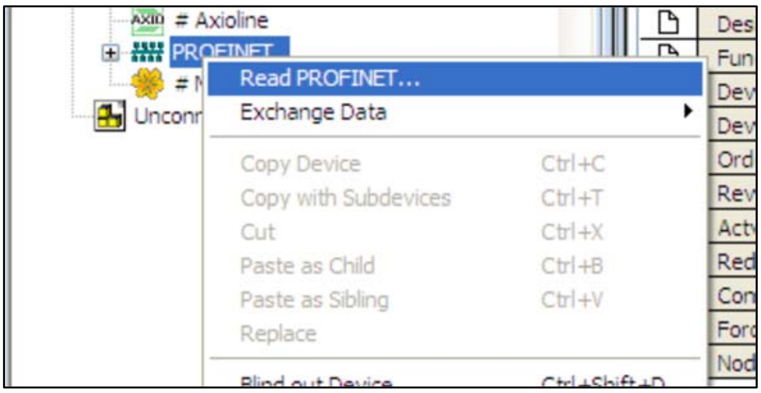


Рисунок 3 – Читання PROFINET

12. Скористатись *Insert button* для додавання пристроїв в проект (Також можна використовувати Device Catalog)

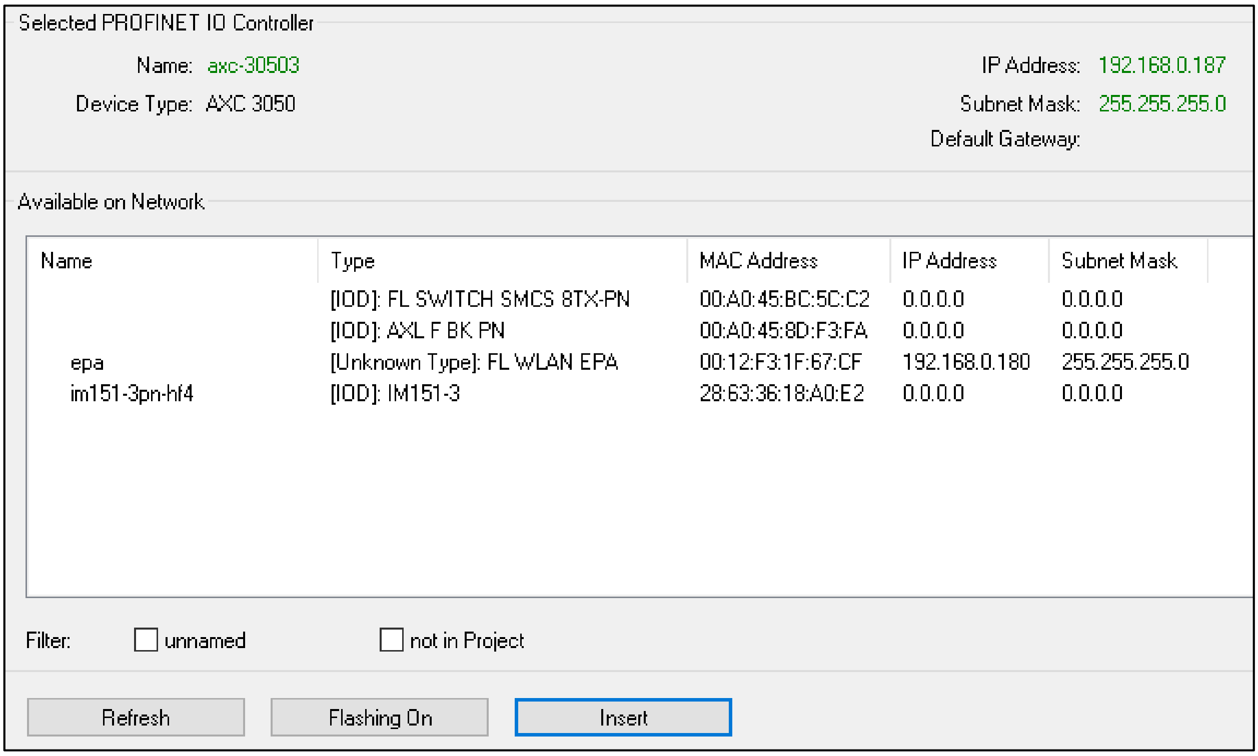


Рисунок 4 – Додавання пристроїв до проекту

13. Встановити діючу IP адресу для кожного PN пристрою: Відкрити *Bus Configuration* і натиснути пристрій який додали. Встановити IP адресу. Відкрити *PROFINET Stantionnames* і призначити *IP*.

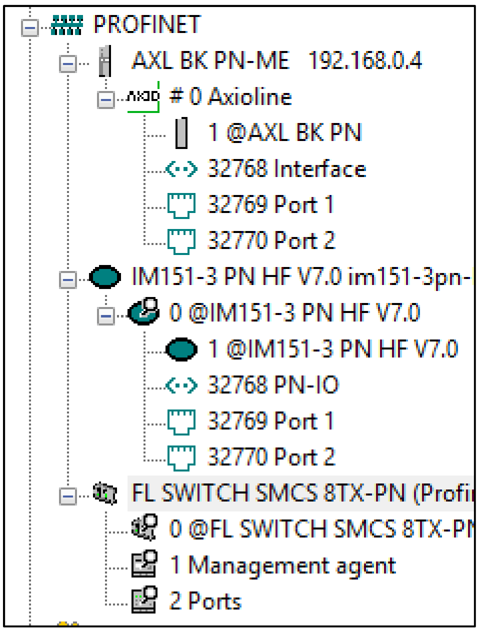


Рисунок 5 – Призначення IP кожному PN пристрою в проекті

14. Додати інший модуль в проект. Натиснути на Axioline в AXL BK PN та *Read Axioline*. Або можна скористуватись Module Catalog для додавання модулів до кожного пристрою. Також можна скористуватись Device Catalog. Якщо ви працюєте з Siemens, натиснути на IM151-3 PN HF та відкрити Siemens -> MODULE\PM -> PM-E DC24V S rev -> перетягнуту на "0 @IM151-3 PN" або правою кнопкою відкрити меню і вибрати Insert to the *Bus Structure*.

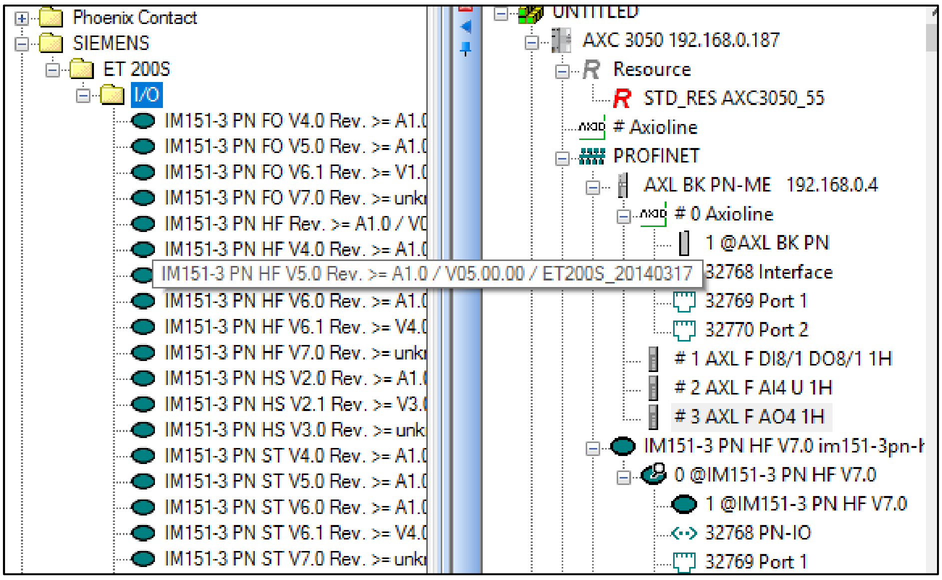


Рисунок 6 – Додавання модулів до Axioliine шини і Siemens BK

15. Встановити Ваші змінні в Data Process і дати їм відповідні імена. Написати Вашу програму і скомпілювати її. Завантажити Ваш проект на запустити PLC.

Завдання до вправи 2. Налаштування мережі PROFINET IO:

1. Налаштувати Profinet PLC та Profinet IO пристрої. Організувати зв'язок між обладнанням та відобразити роботу у вигляді простої програми та відображення індикації.
2. Організувати зв'язок між двома ПЛК та ПЛК з I/O додатками:
   2.1 виконати роботу та відобразити індикацію цифрових I/O;
   2.2 виконати роботу та відобразити індикацію аналогових I/O;

3. Організувати роботу між різними Profinet пристроями.
4. Організувати передачу даних між ПК та PLC; PLC та PLC з використанням різних мережевих технологій.
5. Розробити програму яка відображає технологічний процес за заданим варіантом та відповідним часовим проміжком.

Вправа 3: Діагностика пристроїв Profinet

Якщо Ви маєте зв'язок з PLC, можна показати діагностичні повідомлення.

Перший метод: Ви можете використати IP який був назначений PLC використовуючи браузер і побачити web сторінку PLC з відображенням діагностичних повідомлень, налаштувань та керуванням PLC.

Другий метод: Зовнішня програма Diag+: Засіб знайти деталі про PN пристрої та модулі. Для цього необхідно вибрати *View-> Diag+*.

Завдання до вправи 3: Виконати діагностику пристрою, якщо є потреба сконфігурувати новий фрейм.

Вправа 4: Аналіз трафіку між PN пристроями з використанням аналізатору мережних протоколів Wireshark

1.Запустіти Wireshark коли існує зв'язок між Вашими PN пристроями і ПК.
2. Перевірте трафік використовуючи інструмент діагностики Ethernet.
3. Прочитайте комунікаційні фрейми в різних фазах роботи мережі.

ЛАБОРАТОРНА РОБОТА № 1
ПОБУДОВА МЕРЕЖІ MODBUS/TCP З ВИКОРИСТАННЯМ ETHERNET

Мета – побудувати найпростішу натурну модель промислової мережі MODBUS/TCP.

Обладнання – для виконання лабораторної роботи необхідне таке обладнання та програмне забезпечення:

– функціональний блок TSL Box 1 – PLC & PROFINET;

– персональний комп'ютер з операційною системою сімейства Windows;

– програмний продукт PcWorx 6.x;

– набір кабелів Ethernet та USB.

Схему лабораторного стенду наведено на рис. 7

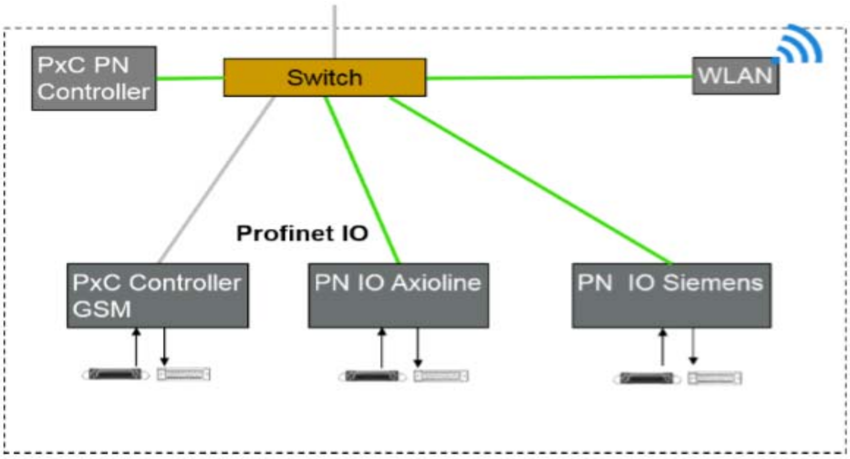


Рисунок 7– Схема лабораторного стенду PLC & PROFINET

Завдання на лабораторну роботу:

– налаштувати наведену схему, відповідно до індивідуального варіанту;

– розподілити та призначити IP-адреси;

– перевірити вірність проведеного налаштування системи за допомогою команди Ping;

– виконати перевірку роботи системи за допомогою http запитів до WEB-серверів, що інтегровані до системних модулів, контролерів та комутаторів.

Хід виконання роботи:

1. Підключити персональний комп'ютер до AXC 3050 через USB-порт.
2. У відповідності до завдання прописати IP-адрес для ПК.
3. У програмному середовищі PC WORX створюємо новий проект AXC 3050.
4. У вкладці «Bus structure» встановити діапазон адрес, а також IP-адреси для відповідних портів AXC 3050. Натиснути «Send» і перезапустити контролер.
5. У вкладці знайти пункт меню «PROFINET device interface», вибирати третій порт (порт AXC 3050 до якого підключена промислова мережа через комутатор FL SWITCH SMCS 8TX-PN), знову натиснути «Send» і відключити USB.
6. З'єднати комутатор FL SWITCH SMCS 8TX-PN та персональний комп'ютер за допомогою Ethernet-кабелю.
7. Задати IP-адресу з діапазону, що вказано на контролері. На верхній панелі перейти до вкладки «Extras» і перевірити тип з'єднання.
8. У вікні «BusStructure» знайти Profinet, натиснути правою кнопкою й вибрати з'єднання таким як Profinet.
9. Вибирати з переліку доступних пристроїв, такі, що представлені в схемі лабораторного стенду.
10. Прописати IP-адреси, DNS-імена доданим пристроям згідно з індивідуальним завданням.
11. Перейти до вкладки «PROFINET Stationames» і встановити маркер зберегти IP, після чого натиснути «Assing ip».
12. До BK PN підключити модулі, натиснути на «Axionline» правою кнопкою, вибрати пункт прочитати шину, після цього вибрати підключені пристрої з наданого списку.
13. Якщо все зроблено вірно, то в схемі лабораторного стенду є три налаштовані блоки. Перевірити зв'язок з ними можна за допомогою команди Ping, виконав її з командного рядка Cmd операційної системи Windows. Також правильність проведених налаштувань можна перевірити за допомогою http запитів до WEB-інтерфейсів налаштованих пристроїв, заздалегідь заборонивши використання Proxy-серверу в браузері для доступу до локальних адрес.
14. На заключному етапі аналогічним чином до зібраної моделі мережі додається контролер ILC 151 GSM / GPRS і підключається до нього модулі.

Перевірка проведених налаштувань проводиться відповідно до пункту 13.

Контрольні питання.

1. Як створити новий проект?
2. Яким чином перейти до режиму конфігурації контролера?
3. Де на контролері відображена інформація про його MAC-адресу?
4. Як перевірити зв'язок між персональним комп'ютером і програмованим контролером?
5. Як підключити додаткові модулі під час налаштування контролера?

ЛАБОРАТОРНА РОБОТА №2
МОНІТОРИНГ ТРАФІКУ МЕРЕЖІ

Мета – проаналізувати мережевий трафік промислової мережі. За допомогою програмних аналізаторів трафіку визначити пакети та інформацію в них відповідно до протоколу промислових мереж MODBUS/TCP.

Обладнання – для виконання лабораторної роботи необхідне таке обладнання та програмне забезпечення:

– функціональний блок TSL Box 1 – PLC & PROFINET;

– функціональний блок TSL Box 3 – Process Simulation;

– персональний комп'ютер з операційною системою сімейства Windows;

– програмне забезпечення – аналізатор трафіку WireShark, або його аналог з можливістю аналізу трафіку та перехоплення мережних пакетів;

– програмний продукт PcWorx 6.x;

– набір інтерфейсних кабелів TATUSmartLab;

– набір кабелів Ethernet та USB.

Схему лабораторного стенду наведено на рис. 8.

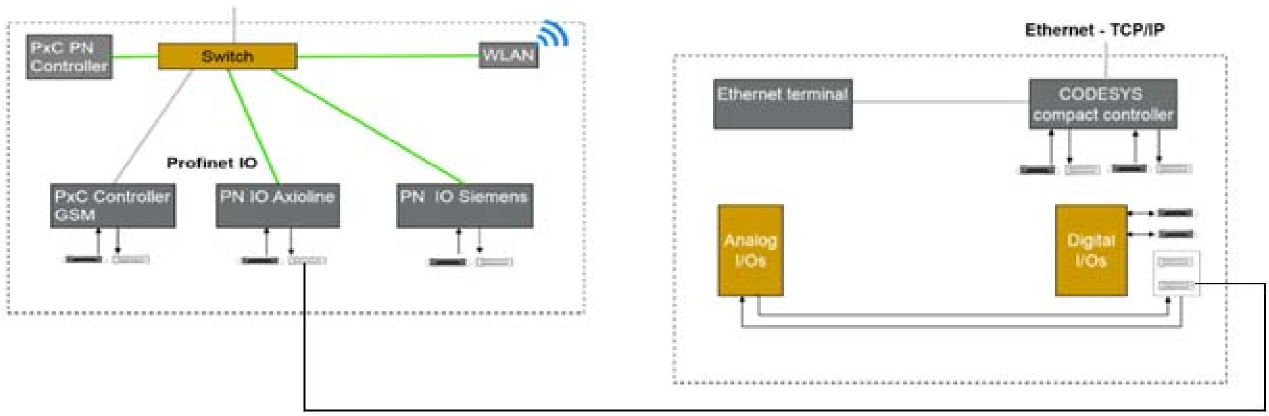


Рисунок 8 – Схема лабораторного стенду PLC & PROFINET та Process Simulation

Завдання на лабораторну роботу:

– за основу схеми дослідження вибирається налаштована модель з лабораторної роботи № 1, до якої додається стенд з виконавчим пристроями TSLBox 3 – Process Simulation;

– провести налаштування елементарної функції управління – реакцію індикаторної лампи на зміну стану кнопки на функціональному модулі TSLBox 3 – Process Simulation. Керування процесом виконується через контролер ILC 151 GSM/GPRS, що розташований на функціональному модулі TSL Box 1 – PLC & PROFINET;

– провести дослідження потоків трафіку в мережі;

– перехватити мережний трафік, відфільтрувати пакети MODBUS/TCP, визначити інформацію що передається в полі даних перехоплених пакетів.

Хід виконання лабораторної роботи:

1. Після того, як мережа настроєна (результат лабораторної роботи № 1), можна переходити до програмування блоків.

2. Фізично з'єднати за допомогою фірмового кабелю TATU Smart Lab функціональні модулі лабораторного стенду TSL Box 1 – PLC & PROFINET та TSLBox 3 – ProcessSimulation.

3. Зайти у вкладку програмування «IEC programming Worckspace».

4. У вікні «Project Tree Window» перейти до вкладки «Logical POUs», вибрати останній файл «Main».

5. Натиснути «Variable» у вікні, що відкрилося, задати ім'я (ONBOARD_INPUT_(BI0-15)). Як тип змінної використати «BOOL», VAR_GLOBAL.

6. Повторити крок «3», назву задати унікальну, в відповідності до індивідуального завдання. Прикладом «Out» для наочності. З'єднати вхід і вихід. Виконати компіляцію.

6. Якщо помилки відсутні, перейти до вкладки «Process Data Workspace». У вікнах, що відкрилися, вибирати програмувальний контролер, після цього вибирати модуль, конкретний біт (або слово цілком) і перемістити його до назви виводу (у цьому випадку в «Out»). Знову виконати компіляцію.

7. Наступний крок – запис програми. Для цього у верхніх вкладках знайти «Project Control Dialog», натиснути у вікні, що відкрилося, стоп.

8. Після цього натиснути «Download» і згодом перевести систему в режим «Cold».

9. Для наочності включити режим «Debug» і перевірити роботу системи в цілому.

10. Якщо всі налаштування проведені вірно, тоді необхідно перейти до

наступного блоку лабораторної роботи. Через вебраузер зайти на керований комутатор другого рівня моделі OSIFL SWITCH SMCS 8TX-PN.

11. Зайти до меню налаштування параметрів комутатора і перевести порт, до якого підключений персональний комп'ютер, до режиму «MIRROR». Як порт, який буде досліджуватись, вибрати порт до якого підключений контролер.

12. На персональному комп'ютері запустити програмний аналізатор мережних протоколів WireShark, вибрати як інтерфейс для прослуховування Etheret, що підключений до FL SWITCH SMCS 8TX-PN.

13. Натиснути кнопку «Capture» і перехопити мережевий потік, при цьому не забити періодично натискати на запрограмовану клавішу зміни стану на TSLBox 3 – ProcessSimulation.

14. Якщо все виконано вірно, тоді виконується налаштування екранних фільтр з прив'язкою до IP-адреси контролера та виконати аналіз перехоплених пакетів, а саме в полі даних відстежити зміни інформації та тип протоколу за яких проводиться обмін.

Контрольні питання.

1. Які основні типи даних, що використовуються в PC Worx?
2. Для зберігання якої інформації призначений тип даних INT?
3. Як створити нову змінну?
4. За що відповідає параметр змінної «Початкове значення»?
5. Як завантажити програму до контролера?
6. Як запустити завантажену програму?
7. Як змінити стан змінної під час роботи програми?

ПЕРЕЛІК ПОСИЛАНЬ

1. Борисов, А.М. Основы построения промышленных сетей автоматики / А.М. Борисов. – Челябинск: Издательский центр ЮУрГУ, 2012. – 108 с.
2. Васин, Н.Н. Основы сетевых технологий на базе коммутаторов и маршрутизаторов / Н.Н. Васин. – Интернет-университет информационных технологий – ИНТУИТ.ру, БИНОМ. Лаборатория знаний, 2011.– 270 с.
3. Денисенко В.В. Компьютерное управление технологическим процессом, экспериментом, оборудованием. – М. : Горячая линия – Телеком, 2008. –608 с.
4. Кангин, В.В. Аппаратные и программные средства систем управления. Промышленные сети и контроллеры: учебное пособие / В.В. Кангин, В.Н. Козлов. – М.: Изд. БИНОМ, 2010. – 418 с.
5. Новиков, Ю.В. Основы локальных сетей: курс лекций; учеб. пособие: для студентов, обучающихся по специальностям в обл. информ. технологий / Ю.В. Новиков, С.В. Кондратенко.– М.: Интернет – Ун-т Информ. Технологий, 2009. – 360 с.
6. Олифер, В.Г. Компьютерные сети. Принципы, технологии, протоколы. Учебное пособие для вузов./ В.Г. Олифер, Н.А. Олифер. – СПб., Питер, 2010. – 943 с.
7. Парр, Э. Программируемые контроллеры: руководство для инженера / Э. Парр; пер. 3-го англ. изд. – М.: БИНОМ. Лаборатория знаний, 2007. – 516 с.
8. Пупена, О.М. Промислові мережі та інтеграційні технології в автоматизованих системах: навчальний посібник / О.М. Пупена, І.В. Ельперін, Н.М. Луцька та ін. – К.: Вид-во «Ліра-К», 2011. – 552 с.
9. Modicon ModBus Protocol Reference Guide. PI-MBUS-300 Rev. J. - MODICON, Inc., Industrial Automation Systems. - June 1996. - 121 p.
10. ModBus messaging on TCP/IP implementation guide, vl.Oa [Електронний ресурс]. – Режим доступу: http://www.modbus-IDA.oig. - June 4, 2004 - 46 p
11. ModBus over serial line specification and implementation guide, vl.02 [Електронний ресурс]. - Режим доступе : http://www.modbus.org. - Dec. 20, 2006. – 44 p.

12. ModBus application protocol specification vl.la [Електронний ресурс]. – Режим доступу: http://www.modbus-lDA.org. - June 4. 2004. - 51 p.
13. Олссон Г., Пиани Дж. Цифровые системы автоматизации и управления . – СПб.: Невский Диалект, 2001 .– 557 с.
14. Изерман Р. Цифровые системы управления .– М.: Мир, 1984 .– 541 с.
15. Гук М. Конструктивы и шины компьютеров промышленного назначения (http://www.neva.ru/mgook/art/mg-inbus.htm)
16. Бердичевский М. Конструктивы Евромеханики во встраиваемых системах// Современные технологии автоматизации .– 2002 .– №4, С. 52 – 59.(http://www.cta.ru/pdf/2002-4/hardware2_2002_4.pdf)
17. PICMG Open Modular Computing Specifications (http://www.picmg.org)
18. Гук М. Интерфейсы ПК. Справочник .– СПб.: Питер, 1999 .– 416 с.
19. Локотков А. Интерфейсы последовательной передачи данных. Стандарты EIA RS-422A/RS-485 // Современные технологии автоматизации .–1997 .– №3, С. 110 – 119. (http://www.cta.ru/pdf/1997-3/note1_1997_3.pdf)
20. Енциклопедія АСУТП. - Інтернет-ресурс: http://bookasutp.ru
21. Інтернет-ресурс: http://profibus-dp.ru
22. Lifelong learning program project, ERASMUS Co-operation between Higher Education Institutions and Enterprises, 502106-LLP-1-2009-1-BE-ERASMUS- ECUE: CoNeT: Co-operative Network Training, October 2009 - September 2012.
23. Oleg Makarov, Yuliya Makarova TATU Smart Lab. User manual.
24. Dominique Daens. PROFINET IO. The Practical Aspects.
25. Christian Madritsch. Presentation of Profinet
26. Data Access Automation Interface Standard. Version 2.02, Feb. 4, 1999. - OPC Foundation, 1994. – 100p.
27. Iwanitz F., Lange J. OPC: fundamentals, implementation, and application. - Softing, 2-nd ed., Heidelberg: Huting, 2002. - 225 p.
28. Iwanitz F., Lange J. OPC fundamentals, Implementation, and Aplication. - 2nd rev. ed., Heidelberg: Huthig, 2002, 225 p.
29. Lange J.. Free choice of platform. The next OPC generation. - Softing AG, (www.softing.com), Nov. 2006. - 4 p.
30. OLE for Process Control. OPC Overview. Version 1.0. - Oct. 27, 1998. - 16 p.
31. OLE for Process Control. Data Access Standard. Version 1.0A. Sept. 11, 1997. – 229p.