

Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна

ПРОЕКТУВАННЯ ТА ТЕХНОЛОГІЇ КІБЕРФІЗИЧНИХ ВИРОБНИЧИХ СИСТЕМ

Методичні вказівки до проведення практичних занять
для здобувачів вищої освіти другого (магістерського) рівня
денної та заочної форм здобуття освіти за спеціальністю
G7 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка»

Електронний ресурс

Рецензенти:

О. О. Литвин – доктор фізико-математичних наук, професор, завідувач кафедри ХТЛПід Навчально-наукового інституту «Українська інженерно-педагогічна академія»;

А. М. Чернюк – кандидат технічних наук, професор, завідувач кафедри електроніки та електроенергетики Навчально-наукового інституту «Українська інженерно-педагогічна академія».

*Затверджено до розміщення в мережі Інтернет рішенням Науково-методичної ради
Харківського національного університету імені В. Н. Каразіна
(протокол № 1 від 23 жовтня 2025 року)*

- П 79 **Проектування** та технології кіберфізичних виробничих систем : методичні вказівки до проведення практичних занять для здобувачів вищої освіти другого (магістерського) рівня денної та заочної форм здобуття освіти за спеціальністю G7 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» [Електронний ресурс] / уклад. А. Ю. Мезеря. – Харків : ХНУ імені В. Н. Каразіна, 2026. – (PDF 88 с.)

Методичні вказівки призначені для виконання практичних робіт з дисципліни «Проектування та технології кіберфізичних виробничих систем» здобувачами спеціальності G7 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка». У матеріалах розглянуто побудову структурної та функціональної архітектури кіберфізичної виробничої системи, розроблення алгоритмів керування технологічними об'єктами, проектування промислових мереж та інформаційного обміну, а також формування цифрового двійника й оцінювання стану виробничого обладнання. Для кожної практичної роботи наведено теоретичні положення, розрахункові залежності, варіанти індивідуальних завдань, приклади розрахунків і вимоги до оформлення результатів. Методичні вказівки орієнтовані на формування системного підходу до проектування кіберфізичних виробничих систем з урахуванням взаємодії фізичних, керувальних, комунікаційних та інформаційних компонентів.

УДК 658.5.011.56 : 658.589 (075.8)

© Харківський національний університет
імені В. Н. Каразіна, 2026

© Мезеря А. Ю., уклад., 2026

Зміст

ВСТУП.....	5
ПРАКТИЧНА РОБОТА № 1. РОЗРОБЛЕННЯ СТРУКТУРНОЇ ТА ФУНКЦІОНАЛЬНОЇ АРХІТЕКТУРИ КІБЕРФІЗИЧНОЇ ВИРОБНИЧОЇ СИСТЕМИ.....	7
1. Мета практичного заняття	7
2. Прилади і матеріали	7
3. Теоретичні положення	8
4. Розрахункові показники архітектури	12
5. Методика виконання практичної роботи	14
6. Таблиця варіантів	17
7. Приклад виконання розрахунку	18
8. Приклад розроблення структурної архітектури	20
9. Функціональна архітектура прикладового варіанта	21
10. Вимоги до оформлення результатів роботи	22
11. Висновки до практичної роботи	23
ПРАКТИЧНА РОБОТА № 2. РОЗРОБЛЕННЯ АЛГОРИТМУ КЕРУВАННЯ ТЕХНОЛОГІЧНИМ ОБ'ЄКТОМ У СКЛАДІ КІБЕРФІЗИЧНОЇ СИСТЕМИ.....	23
2. Прилади і матеріали	24
3. Теоретичні положення	24
4. Методика виконання практичної роботи	34
5. Приклад розрахунку	36
6. Завдання до практичної роботи.....	41
7. Вимоги до оформлення звіту.....	42
ПРАКТИЧНА РОБОТА № 3. ПРОЄКТУВАННЯ ПРОМИСЛОВОЇ МЕРЕЖІ ТА ІНФОРМАЦІЙНОГО ОБМІНУ КІБЕРФІЗИЧНОЇ ВИРОБНИЧОЇ СИСТЕМИ	43
2. Прилади і матеріали	44
3. Теоретичні положення	44
4. Класифікація інформаційних потоків.....	49
5. Методика розрахунку.....	51

6. Методика виконання практичної роботи	54
7. Таблиця варіантів	56
8. Приклад розрахунку	57
9. Побудова мережі для розрахункового прикладу.....	61
10. Організація OPC UA-обміну	62
11. Розподіл трафіку за пріоритетом	62
12. Аналіз доцільності TSN	63
13. Завдання до практичної роботи.....	64
14. Вимоги до оформлення звіту	65
15. Висновки.....	65
ПРАКТИЧНА РОБОТА № 4. ПОБУДОВА ЦИФРОВОГО ДВІЙНИКА ТА ОЦІНЮВАННЯ СТАНУ КІБЕРФІЗИЧНОГО ВИРОБНИЧОГО ОБ'ЄКТА	66
2. Прилади і матеріали	67
3. Теоретичні положення	67
4. Методика виконання практичної роботи	76
5. Таблиця варіантів	77
5. Приклад розрахунку	78
6. Зв'язок цифрового двійника з попередніми практичними роботами	84
7. Завдання до практичної роботи.....	84
8. Вимоги до оформлення звіту.....	85
9. Висновки.....	86
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	87

ВСТУП

Кіберфізична виробнича система формується на межі двох середовищ – фізичного технологічного процесу та цифрової системи, яка отримує дані про цей процес, обробляє їх, формує керувальні рішення й передає відповідні дії назад до обладнання. На відміну від традиційної автоматизованої системи керування, де основна увага зосереджується на реалізації заданого алгоритму регулювання або логічного керування, для кіберфізичної системи принциповим є постійний інформаційний зв'язок між фізичними об'єктами, засобами автоматизації, обчислювальними ресурсами та програмними сервісами.

Таке розширення функцій змінює сам підхід до проєктування. Контролер, датчик або виконавчий механізм уже не розглядаються як ізольовані компоненти. Кожний елемент бере участь у формуванні загального інформаційного процесу: фізична величина вимірюється, перетворюється на цифрові дані, передається мережею, використовується локальним алгоритмом керування або програмним сервісом, зберігається, аналізується і, за потреби, стає основою для нового керувального впливу. Через це помилка у виборі структури системи може проявлятися не тільки як неправильне керування технологічним об'єктом, а й як надмірна затримка передавання інформації, втрата даних, несумісність програмних компонентів або неможливість масштабування системи.

У науковій літературі та нормативних документах описано значну кількість референтних архітектур для цифрового виробництва, Industrial Internet of Things, smart manufacturing та cyber-physical production systems. Вони відрізняються рівнями абстракції, способом функціональної декомпозиції, розподілом обчислювальних функцій та областю застосування. Тому референтну архітектуру не можна механічно переносити на конкретний виробничий об'єкт. Вона задає спосіб мислення, типові функціональні області та зв'язки між ними, тоді як склад обладнання, канали даних, цикли керування й програмні компоненти визначаються властивостями конкретного технологічного процесу.

У ISO/IEC TR 63306-1:2020 запропоновано системний підхід до взаємозв'язку стандартів, що стосуються smart manufacturing, причому розглядаються не лише окремі технологічні пристрої, а продукт, виробництво, ланцюги постачання та промислові сервіси. Серія ISO 23247, опублікована у 2021 році, формалізує принципи побудови цифрового двійника виробничої системи, визначає референтну архітектуру та вимоги до інформаційного обміну між її сутностями. Ці документи демонструють

характерну рису кіберфізичного підходу: фізична структура, інформаційна структура і функціональна структура повинні розглядатися спільно.

Практична частина дисципліни побудована так, щоб послідовно пройти основні етапи проєктування кіберфізичної виробничої системи. У першій роботі визначається її структурна та функціональна архітектура. У другій розробляється алгоритм керування технологічним об'єктом з урахуванням інформації, яка надходить із фізичного рівня. Третя робота присвячена організації промислової мережі та інформаційного обміну. У четвертій роботі фізичний об'єкт і система збирання даних пов'язуються з його цифровим представленням, після чого виконується оцінювання поточного стану об'єкта.

Практичні роботи не передбачають вибору обладнання лише за назвою або формального складання блок-схем. Для кожного рішення необхідно встановити його функцію в системі, визначити джерело та споживача інформації, напрямок передавання даних, характер керувального впливу й основні обмеження. Структура вважається обґрунтованою тоді, коли можна простежити повний цикл «фізичний процес – вимірювання – передавання даних – оброблення – прийняття рішення – керувальний вплив – зміна фізичного процесу».

Курс практичних робіт передбачає використання розрахункових оцінок. Їх завдання полягає не в тому, щоб замінити детальне інженерне проєктування, а в тому, щоб виявити очевидні структурні суперечності ще на етапі формування архітектури. Наприклад, пропускну здатність інформаційного каналу потрібно співвідносити з потоком даних; допустимий час реакції системи – із сумою затримок вимірювання, передавання, оброблення і виконання команди; вимоги до безперервності роботи – із надійністю компонентів, від яких залежить функціонування всього контуру.

Результатом виконання чотирьох практичних робіт має стати узгоджений проєкт кіберфізичної виробничої системи, у якому одна й та сама виробнича установка послідовно розглядається з архітектурної, алгоритмічної, мережевої та модельної точок зору. Такий підхід дозволяє не розривати окремі теми дисципліни на незалежні завдання, а показати зв'язок між фізичним обладнанням, автоматичним керуванням, промисловими комунікаціями та цифровими моделями.

ПРАКТИЧНА РОБОТА № 1. РОЗРОБЛЕННЯ СТРУКТУРНОЇ ТА ФУНКЦІОНАЛЬНОЇ АРХІТЕКТУРИ КІБЕРФІЗИЧНОЇ ВИРОБНИЧОЇ СИСТЕМИ

1. Мета практичного заняття

Мета роботи – набути навичок декомпозиції виробничого об’єкта та побудови структурної і функціональної архітектури кіберфізичної виробничої системи з визначенням фізичних компонентів, засобів вимірювання, виконавчих пристроїв, контролерів, мережеских вузлів, обчислювальних ресурсів, програмних компонентів та інформаційних зв’язків між ними.

Після виконання роботи здобувач повинен уміти:

- визначати межі кіберфізичної виробничої системи;
- відокремлювати фізичну, інформаційну та обчислювальну частини системи;
- встановлювати функції датчиків, виконавчих механізмів, контролерів та програмних компонентів;
- розрізняти структурне та функціональне представлення системи;
- визначати основні потоки даних і керувальних команд;
- оцінювати кількість каналів введення-виведення;
- визначати орієнтовну інтенсивність інформаційного обміну;
- оцінювати час проходження інформації в контурі керування;
- встановлювати компоненти, відмова яких призводить до втрати працездатності заданої функції.

2. Прилади і матеріали

Для виконання роботи використовуються:

1. персональний комп’ютер;
2. програмне середовище для побудови структурних і функціональних схем;
3. текстовий або табличний процесор для виконання розрахунків;
4. вихідні дані відповідно до номера варіанта;
5. технічні описи типових датчиків, виконавчих механізмів, промислових контролерів і мережевого обладнання;
6. матеріали лекційного курсу;
7. нормативні документи та наукові джерела [1–10].

Використання конкретного програмного пакета для побудови схем не є обов'язковою умовою. Схема повинна передавати структуру системи, а не особливості графічного редактора.

3. Теоретичні положення

3.1. Кіберфізична виробнича система як об'єкт проєктування

Кіберфізична виробнича система поєднує фізичні процеси з обчислювальними й комунікаційними засобами. Інформація про стан обладнання надходить від датчиків та інтелектуальних пристроїв, після чого використовується для локального керування, диспетчеризації, діагностування або інших програмних функцій.

Для виробничої системи принциповим є не сам факт наявності цифрового обладнання, а замкнений зв'язок між фізичним і цифровим середовищами. Наприклад, передавання температури технологічного середовища до архіву утворює канал спостереження. Якщо ж виміряне значення використовується алгоритмом для зміни положення клапана або продуктивності привода, виникає кіберфізичний контур керування.

У роботі доцільно розглядати систему як сукупність декількох взаємопов'язаних рівнів. Такий поділ є навчальною декомпозицією, а не окремою стандартизованою архітектурою.

Фізичний рівень

До нього належать технологічний об'єкт, механізми, матеріальні потоки, електричні та механічні приводи, технологічне середовище.

Цей рівень безпосередньо підпорядковується законам фізичного процесу. Його стан описується температурою, тиском, положенням, швидкістю, витратою, концентрацією, рівнем та іншими технологічними параметрами.

Рівень вимірювання і виконання

Містить датчики, вимірювальні перетворювачі, енкодери, кінцеві вимикачі, системи технічного зору, електроприводи, клапани, контактори та інші пристрої, через які цифрова система взаємодіє з фізичним процесом.

Саме цей рівень забезпечує двонапрямний перехід:

фізичний стан → дані;

керувальні дані → фізична дія.

Рівень локального керування

До нього можуть входити програмовані логічні контролери, промислові комп'ютери, контролери руху, локальні регулятори та вбудовані обчислювальні пристрої.

На цьому рівні виконуються операції, для яких необхідна визначена та достатньо мала затримка: логічне керування, регулювання, синхронізація механізмів, формування блокувань, оброблення сигналів аварійних станів.

Комунікаційний рівень

Його функція полягає у передаванні технологічних даних і команд між вузлами. Він включає фізичні канали зв'язку, мережеве обладнання, промислові мережі, шлюзи та протоколи обміну.

В ISO 23247-4:2021 інформаційний обмін розглядається як окрема складова архітектури цифрового виробничого середовища, причому вимоги встановлюються до декількох типів мереж, які з'єднують фізичні виробничі елементи, комунікаційні сутності, цифрове представлення та користувацькі застосунки [10].

Рівень периферійного оброблення

Edge-рівень розміщується між локальним обладнанням і системами вищого рівня. Тут можуть виконуватися агрегування даних, попередня фільтрація, перетворення протоколів, локальний аналіз, формування подій та підготовка даних для подальшого використання.

Введення такого рівня не є обов'язковим для будь-якої кіберфізичної системи. Якщо всі необхідні функції реалізуються контролером, а обсяг інформації малий, окремий edge-вузол лише ускладнить структуру.

Рівень виробничих інформаційних функцій

На ньому розташовуються засоби диспетчеризації, візуалізації, архівування даних, аналізу технологічних режимів, управління рецептами, виробничими завданнями тощо.

Саме тут окремі сигнали датчиків перетворюються на інформацію про стан технологічної ділянки в цілому.

Рівень корпоративних і зовнішніх сервісів

Цей рівень може забезпечувати взаємодію виробничої системи з інформаційними системами підприємства, базами даних, аналітичними платформами або іншими програмними сервісами.

Не кожний потік даних необхідно передавати на цей рівень. Сигнали, що використовуються для швидкого локального керування, повинні оброблятися якомога ближче до фізичного процесу. Подальша передача потрібна лише тоді, коли ці дані необхідні іншим функціям системи.

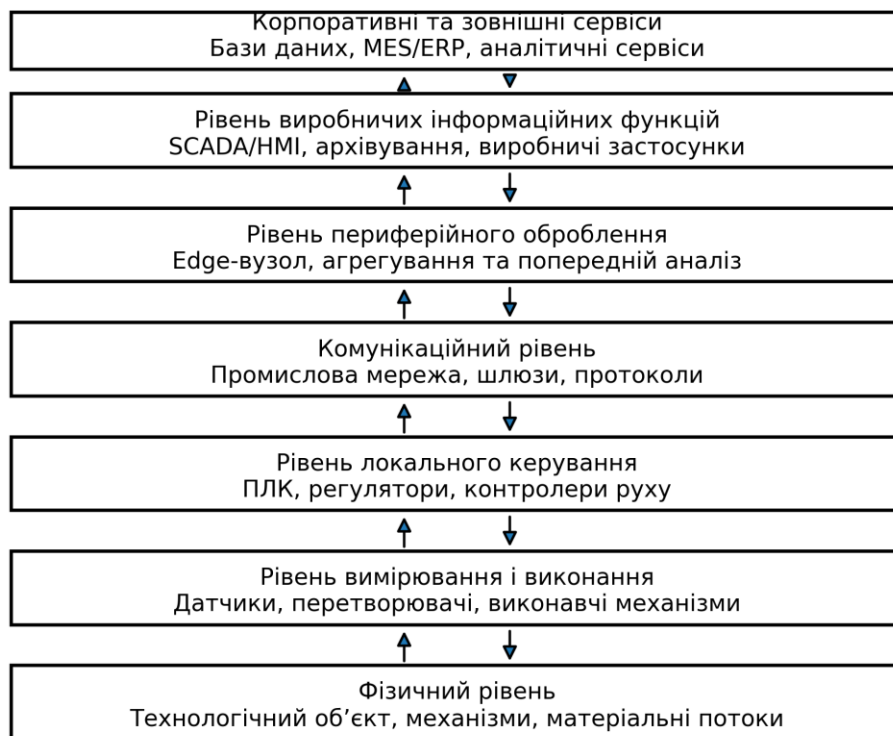


Рисунок 1.1. Узагальнена багаторівнева архітектура кіберфізичної виробничої системи

3.2. Структурна архітектура

Структурна архітектура показує, з яких фізичних та програмно-апаратних компонентів складається система і яким способом вони взаємопов'язані.

У структурній схемі доцільно показувати:

- технологічний об'єкт;
- датчики;
- виконавчі механізми;
- локальний контролер;
- пристрої зв'язку;
- edge-вузол, якщо він необхідний;
- операторську станцію;
- сервер або базу даних;
- виробничі інформаційні системи;
- зовнішні інформаційні системи;
- напрямки основних потоків даних.

Не слід змішувати фізичний компонент і його функцію. Наприклад, «контролер» – структурний елемент, тоді як «регулювання температури» – функція.

Сучасні дослідження reference architectures показують, що розмежування референтної архітектури і конкретної системної архітектури суттєве: перша задає загальні принципи та типові компоненти, друга описує реальну або проєктовану систему [6].

3.3. Функціональна архітектура

Функціональна архітектура відповідає на інше питання: які функції повинна виконувати система незалежно від того, на якому конкретно пристрої ці функції реалізовані?

Для типової кіберфізичної виробничої системи можна виділити такі функції:

- вимірювання технологічних координат;
- первинне оброблення сигналів;
- ідентифікація стану обладнання;
- локальне автоматичне керування;
- передавання технологічних даних;
- формування повідомлень про відхилення;
- візуалізація;
- накопичення історичних даних;
- обмін даними з виробничими інформаційними системами;
- підтримка цифрового представлення об'єкта.

ISO 23247-2:2021 також використовує функціональне представлення для опису функціональних сутностей цифрового двійника виробничого об'єкта [9]. Це підтверджує доцільність розділення фізичної структури системи та її функцій.

Для однієї функції можуть використовуватися декілька структурних елементів. І навпаки, один обчислювальний пристрій може виконувати декілька функцій.

Наприклад:

- структурний елемент: ПЛК;
- **функції:** опитування датчиків, логічне керування, ПІД-регулювання, діагностування, передавання технологічних даних.



Рисунок 1.2. Взаємозв'язок структурної та функціональної архітектури КФВС

4. Розрахункові показники архітектури

На етапі попереднього проєктування необхідно перевірити, чи відповідає запропонована структура обсягу сигналів та часовим вимогам.

4.1. Кількість каналів введення-виведення

Загальна кількість каналів визначається як

$$N_{IO} = N_{DI} + N_{DO} + N_{AI} + N_{AO},$$

де N_{IO} – загальна кількість каналів введення-виведення;
 N_{DI} – кількість дискретних входів; N_{DO} – кількість дискретних виходів;
 N_{AI} – кількість аналогових входів; N_{AO} – кількість аналогових виходів.

Отримане значення використовується для структурної оцінки необхідних засобів введення-виведення. На етапі реального проєктування до нього додається резерв каналів відповідно до вимог конкретного проєкту, але в цій практичній роботі резерв не задається нормативно.

4.2. Інтенсивність потоку даних

Для циклічного передавання пакета даних від одного джерела інформаційне навантаження без урахування службових полів протоколу можна оцінити як

$$R_i = \frac{8L_i}{T_i},$$

де R_i – корисна швидкість потоку даних від i -го джерела, біт/с; L_i – обсяг корисних даних одного повідомлення, байт; T_i – період передавання повідомлення, с.

Якщо в системі використовується декілька джерел, сумарний корисний потік становить

$$R_{\Sigma} = \sum_{i=1}^m R_i,$$

де R_{Σ} – сумарна швидкість корисного інформаційного потоку, біт/с; m – кількість незалежних потоків даних.

Цей розрахунок не включає заголовки протоколів, повторні передавання, службовий трафік і короткочасні піки навантаження. Тому його потрібно розглядати як нижню оцінку необхідної пропускної здатності, а не як завершений розрахунок мережі.

4.3. Завантаження інформаційного каналу

Відносне завантаження каналу за корисними даними визначається як

$$K_{\text{мер}} = \frac{R_{\Sigma}}{C},$$

де $K_{\text{мер}}$ – розрахункова частка використання пропускної здатності каналу; C – номінальна пропускна здатність каналу, біт/с.

Якщо результат виражається у відсотках, отримане значення множиться на 100%.

Цей показник сам по собі не доводить придатність конкретної промислової мережі. Дві мережі з однаковою номінальною пропускною здатністю можуть істотно відрізнятися за затримкою, джитером, детермінованістю та механізмами доставки повідомлень.

4.4. Час реакції кіберфізичного контуру

Орієнтовний час від зміни фізичного параметра до реалізації керувальної дії можна представити як

$$T_{\text{реак}} = T_{\text{вим}} + T_{\text{пер1}} + T_{\text{обр}} + T_{\text{пер2}} + T_{\text{вик}},$$

де $T_{\text{реак}}$ – сумарний час реакції системи; $T_{\text{вим}}$ – затримка вимірювання; $T_{\text{пер1}}$ – час передавання даних до обчислювального вузла; $T_{\text{обр}}$ – час оброблення і формування керувального рішення; $T_{\text{пер2}}$ – час передавання керувальної команди; $T_{\text{вик}}$ – час реакції виконавчого пристрою.

Ця залежність пояснює одну з причин розподілення обчислень у кіберфізичній системі. Якщо функція потребує швидкого реагування, передавання інформації через декілька віддалених рівнів може бути неприйнятним. Тоді алгоритм доцільно реалізувати на локальному контролері або edge-вузлі.

4.5. Частота оновлення інформації

Частота формування даних визначається співвідношенням

$$f_i = \frac{1}{T_i},$$

де f_i – частота оновлення даних i -го джерела, Гц.

Зменшення періоду опитування збільшує часову роздільну здатність даних, але одночасно збільшує інформаційний потік. Тому період передавання не слід вибирати мінімальним без зв'язку з динамікою фізичного процесу.

4.6. Оцінка готовності компонентів

Якщо для компонента відомі середній час безвідмовної роботи і середній час відновлення, коефіцієнт готовності можна визначити за залежністю

$$A_j = \frac{MTBF_j}{MTBF_j + MTTR_j}$$

де A_j – коефіцієнт готовності j -го компонента; $MTBF_j$ – середній час між відмовами; $MTTR_j$ – середній час відновлення після відмови.

Якщо функція виконується лише за одночасної працездатності всіх послідовно необхідних компонентів і резервування відсутнє, для навчальної оцінки готовності такого ланцюга визначають як

$$A_{\text{сист}} = \prod_{j=1}^n A_j,$$

де $A_{\text{сист}}$ – готовність розглянутого функціонального ланцюга; n – кількість необхідних компонентів.

При застосуванні резервування така проста модель уже непридатна, оскільки структура надійності змінюється.

5. Методика виконання практичної роботи

Етап 1. Визначення меж системи

За своїм варіантом необхідно встановити:

- технологічний об'єкт;
- основний технологічний процес;
- вхідні матеріальні або енергетичні потоки;
- вихідний результат;
- параметри, за якими оцінюється стан процесу;
- змінні, на які може впливати система керування.

Межі системи треба задавати конкретно. Наприклад, для конвеєрної сортувальної дільниці складську ERP-систему можна розглядати як зовнішню систему, а локальний ПЛК – як складову проєктованої КФВС.

Етап 2. Формування переліку фізичних компонентів

Скласти таблицю такого типу:

№	Компонент	Кількість	Фізична величина або дія	Тип інформації
1	Датчик	...	...	DI/AI/мережеві дані
2	Виконавчий механізм	...	...	DO/AO/мережева команда
3	Контролер	...	Оброблення	Цифрові дані
4	Мережевий вузол	...	Передавання	Цифрові дані

Для інтелектуального пристрою з мережевим інтерфейсом не потрібно штучно подавати всі його дані як дискретні входи ПЛК. У структурній схемі такий пристрій може бути окремим мережевим вузлом.

Етап 3. Визначення функцій

Для системи необхідно сформувати функціональні групи:

- **F1 – отримання інформації про фізичний процес;**
- **F2 – первинне оброблення даних;**
- **F3 – автоматичне керування;**
- **F4 – передавання даних;**
- **F5 – оперативне відображення стану;**
- **F6 – архівування;**
- **F7 – діагностування;**
- **F8 – інформаційна взаємодія із системами вищого рівня.**

Перелік можна змінювати відповідно до об'єкта. Функція, яка не потрібна для заданої установки, не повинна додаватися лише для збільшення кількості блоків.

Етап 4. Розроблення структурної схеми

На схемі необхідно розмістити мінімум такі групи:

Технологічний об'єкт → датчики та виконавчі механізми → локальне керування → промислова мережа → системи оброблення та візуалізації.

Додатково можуть бути введені:

- edge-вузол;
- сервер;
- база даних;
- цифровий двійник;

- MES;
- ERP;
- хмарний сервіс.

Додавати їх потрібно тільки за наявності функції, яку відповідний компонент реально виконує.

Етап 5. Розроблення функціональної схеми

На функціональній схемі показуються не назви пристроїв, а операції:
вимірювання → формування даних → передавання → оброблення → оцінювання стану → формування команди → виконання команди.

Окремим інформаційним шляхом необхідно показати:

оброблені технологічні дані → візуалізація → архівування / передавання системі вищого рівня.

Етап 6. Розрахунок кількості каналів

За вихідними даними визначити:

- DI;
- DO;
- AI;
- AO;
- загальну кількість каналів.

Інтелектуальні мережеві пристрої враховуються окремо.

Етап 7. Розрахунок інформаційних потоків

Для кожної групи мережевих даних визначити:

1. кількість джерел;
2. розмір повідомлення;
3. період передавання;
4. швидкість потоку даних;
5. сумарний потік;
6. відносне завантаження заданого каналу.

Етап 8. Оцінка часу реакції

Необхідно визначити суму затримок у найбільш критичному контурі:
датчик → мережа → контролер → мережа → виконавчий пристрій.

Якщо отриманий час перевищує допустиме значення, архітектуру необхідно змінити. Один із можливих способів – перенесення функції керування на нижчий рівень.

Етап 9. Формування остаточної архітектури

Після розрахунків здобувач повинен відповісти на три питання:

1. Чи всі фізичні змінні, потрібні для керування, вимірюються?
2. Чи всі керувальні дії мають фізичний виконавчий канал?
3. Чи забезпечує запропонована структура передавання та оброблення інформації в межах заданого часу?

Тільки після цього структурна схема вважається завершеною.

6. Таблиця варіантів

Для всіх варіантів мережеві повідомлення в розрахунку приймаються як корисне навантаження без службових полів протоколу. Детальніше протоколи розглядатимуться у практичній роботі №3.

Варіант	Виробничий об'єкт	DI	DO	AI	AO	Мережеві пристрої	Дані одного мережевого пристрою, байт	Період передавання, мс	Пропускна здатність каналу	Допустимий час реакції, мс
1	Конвеєрна дільниця транспортування деталей	12	8	2	2	2	24	100	100 Мбіт/с	150
2	Автоматизована пакувальна дільниця	16	10	4	2	3	32	100	100 Мбіт/с	120
3	Дільниця дозування сипких матеріалів	10	8	6	4	2	40	200	100 Мбіт/с	300
4	Автоматизована складальна комірка	18	12	4	4	4	32	50	100 Мбіт/с	100
5	Дільниця механічного оброблення деталей	14	10	6	2	3	48	100	100 Мбіт/с	150
6	Автоматизована насосна установка	8	6	8	4	2	32	200	10 Мбіт/с	400
7	Термічна технологічна установка	10	8	10	6	3	40	250	10 Мбіт/с	500
8	Роботизована виробнича комірка	20	14	4	4	5	64	50	100 Мбіт/с	80
9	Автоматизована лінія контролю якості	12	8	6	2	4	64	100	100 Мбіт/с	200
10	Дільниця сортування і накопичення продукції	16	12	4	2	3	40	100	100 Мбіт/с	120

7. Приклад виконання розрахунку

Об'єкт: автоматизована ділянка сортування металевих деталей.

Приймаємо такі вихідні дані:

- DI – 14;
- DO – 8;
- AI – 4;
- AO – 2;
- два інтелектуальні мережеві пристрої;
- дані дискретних датчиків – 4 байти на один датчик через 50 мс;
- дані аналогових датчиків – 8 байт через 100 мс;
- дані одного інтелектуального пристрою – 32 байти через 200 мс;
- пакет стану контролера – 48 байт через 100 мс;
- пропускна здатність мережі – 100 Мбіт/с;
- затримка вимірювання – 20 мс;
- затримка передавання даних до контролера – 5 мс;
- час оброблення – 10 мс;
- час передавання команди – 5 мс;
- реакція виконавчого механізму – 40 мс;
- допустимий час реакції – 100 мс.

7.1. Кількість каналів введення-виведення

$$N_{\text{ю}} = 14 + 8 + 4 + 2 = 28$$

Отже, для фізичних сигналів необхідно передбачити **28 каналів введення-виведення**.

Два інтелектуальні пристрої, які передають дані безпосередньо мережею, до цього числа не додаються.

7.2. Потік від дискретних датчиків

Період 50 мс становить 0,05 с.

Для одного датчика:

$$R_i = \frac{8 \cdot 4}{0,05} = 640 \text{ біт/с}$$

Для 14 датчиків:

$$R_{\text{DI}} = 14 \cdot 640 = 8960 \text{ біт/с}$$

7.3. Потік від аналогових датчиків

$$R_{\text{AI}} = 4 \cdot \frac{8 \cdot 8}{0,1} = 2560 \text{ біт/с}.$$

7.4. Потік від інтелектуальних пристроїв

$$R_{\text{INT}} = 2 \cdot \frac{8 \cdot 32}{0,2} = 2560 \text{ біт/с}.$$

7.5. Передавання стану контролера

$$R_{\text{ПЛК}} = \frac{8 \cdot 48}{0,1} = 3840 \text{ біт/с}.$$

7.6. Сумарний корисний інформаційний потік

$$R_{\Sigma} = 8960 + 2560 + 2560 + 3840 = 17920 \text{ біт/с}$$

Або

$$R_{\Sigma} = 17,92 \text{ кбіт/с}.$$

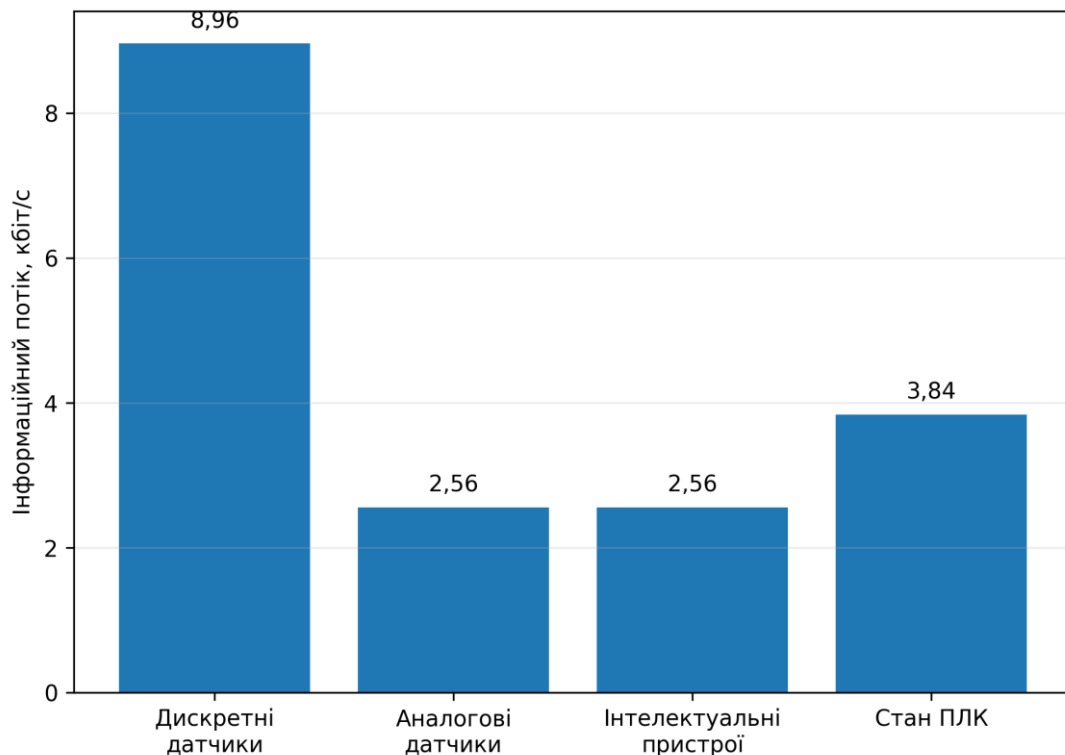


Рисунок 1.3. Розподіл корисного інформаційного потоку за джерелами даних

7.7. Розрахункове завантаження мережі

Пропускна здатність 100 Мбіт/с дорівнює 100 000 000 100\,000\,000 біт/с.

$$K_{\text{мер}} = \frac{17920}{100000000} = 0,0001792.$$

У відсотках:

$$K_{\text{мер}} = 0,01792\%.$$

Отримане значення стосується тільки корисних даних. Реальний мережевий трафік буде більшим через службову інформацію протоколів та інші повідомлення. Проте вже на цьому етапі видно, що обсяг

технологічних даних не створює помітного навантаження на канал із номінальною швидкістю 100 Мбіт/с.

Це, однак, не означає автоматичної придатності будь-якої мережі 100 Мбіт/с для контуру керування. Для швидких технологічних операцій потрібно додатково аналізувати часові властивості обміну, що буде виконуватися у практичній роботі №3.

7.8. Час реакції системи

$$T_{\text{реак}} = 20 + 5 + 10 + 5 + 40 = 80 \text{ мс}.$$

Для прикладового варіанта допустимий час становить 100 мс.

Отже,

$$80 < 100 \text{ мс}.$$

і запропонований інформаційний шлях за заданими вихідними параметрами задовольняє часову умову.

Запас за часом дорівнює

$$\Delta T = 100 - 80 = 20 \text{ мс}$$

де ΔT – запас між допустимим і розрахованим часом реакції.

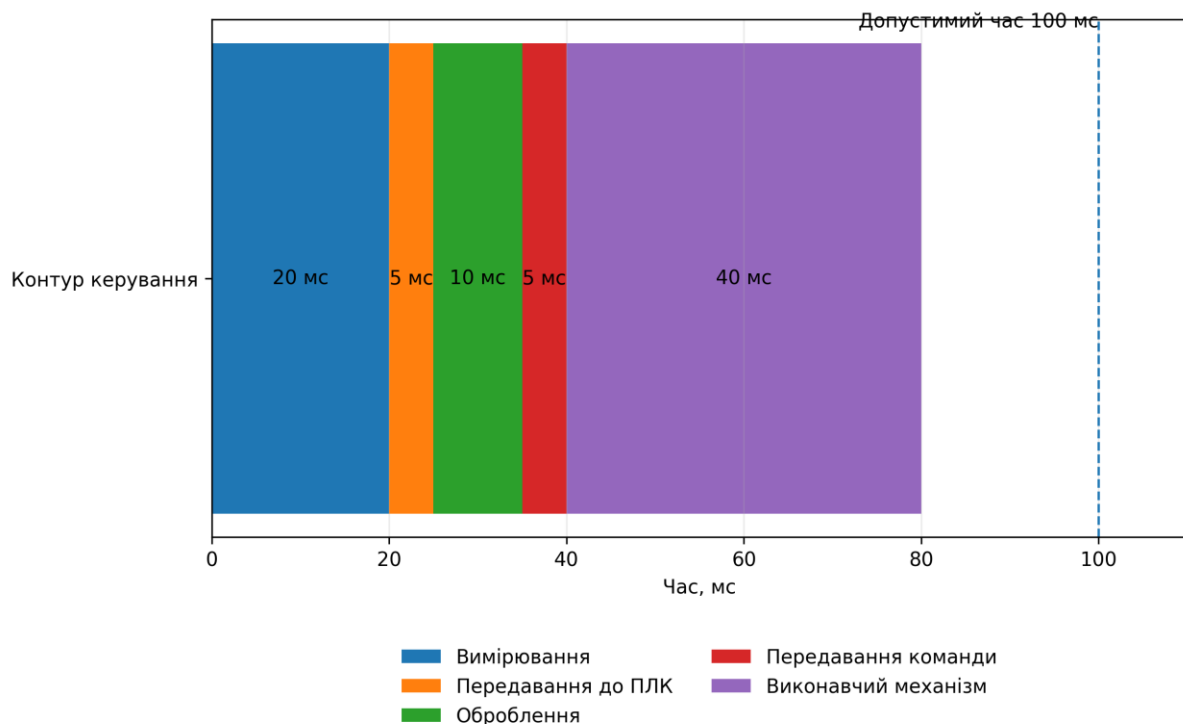


Рисунок 1.4 Складові часу реакції кіберфізичного контуру керування

8. Приклад розроблення структурної архітектури

Для ділянки сортування доцільна така структура:

Рівень 1 – фізичний процес

Конвеєр, механізм переміщення деталей, сортувальні заслінки, накопичувачі.

Рівень 2 – вимірювання та виконання

Датчики наявності деталей, датчики положення, датчики технологічних параметрів, виконавчі приводи сортувальних механізмів.

Рівень 3 – локальне керування

ПЛК, який виконує опитування сигналів, логіку сортування, блокування та формування команд приводам.

Рівень 4 – промислова комунікація

Мережеве з'єднання ПЛК, інтелектуальних пристроїв та операторської станції.

Рівень 5 – периферійна обробка

За потреби – промисловий комп'ютер або edge-вузол для агрегування та попереднього аналізу даних.

Рівень 6 – диспетчеризація

Операторський інтерфейс, відображення станів, аварій, лічильників продукції та параметрів процесу.

Рівень 7 – інформаційне зберігання та виробничі сервіси

Архів технологічних даних, база історичних параметрів, передавання агрегованої інформації до системи вищого рівня.

У такій архітектурі контур швидкого керування сортуванням не проходить через сервер або корпоративну інформаційну систему. Це принципове рішення: час виконання виробничої операції не повинен залежати від доступності програмного забезпечення, яке не потрібне безпосередньо для локального керування.

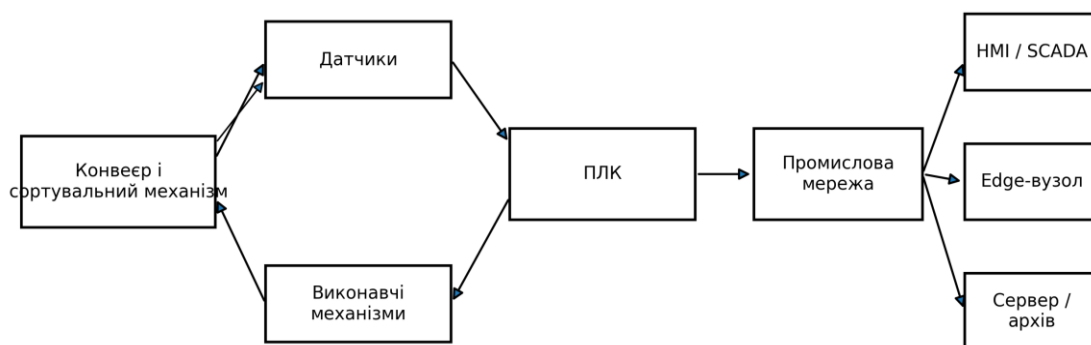


Рисунок 1.5. Структурна архітектура автоматизованої дільниці сортування металевих деталей

9. Функціональна архітектура прикладового варіанта

Функції можна розподілити таким чином:

Функція	Вхідна інформація	Результат	Місце виконання
Виявлення деталі	Сигнал датчика	Ознака наявності	Рівень вимірювання

Функція	Вхідна інформація	Результат	Місце виконання
Визначення параметрів деталі	Дані датчиків	Набір характеристик	ПЛК / інтелектуальний пристрій
Класифікація	Характеристики деталі	Номер категорії	ПЛК або edge-вузол
Формування команди	Категорія, положення	Команда сортування	ПЛК
Виконання сортування	Керувальний сигнал	Переміщення механізму	Виконавчий пристрій
Контроль результату	Сигнал датчика	Підтвердження операції	ПЛК
Візуалізація	Поточні дані	Стан установки	Операторська станція
Архівування	Технологічні дані	Історія роботи	Сервер
Виробнича статистика	Архівні дані	Агреговані показники	Інформаційний рівень

Структурний і функціональний описи в цьому випадку доповнюють один одного. Структурна схема визначає, де розташовані компоненти, а функціональна – що саме відбувається з інформацією.

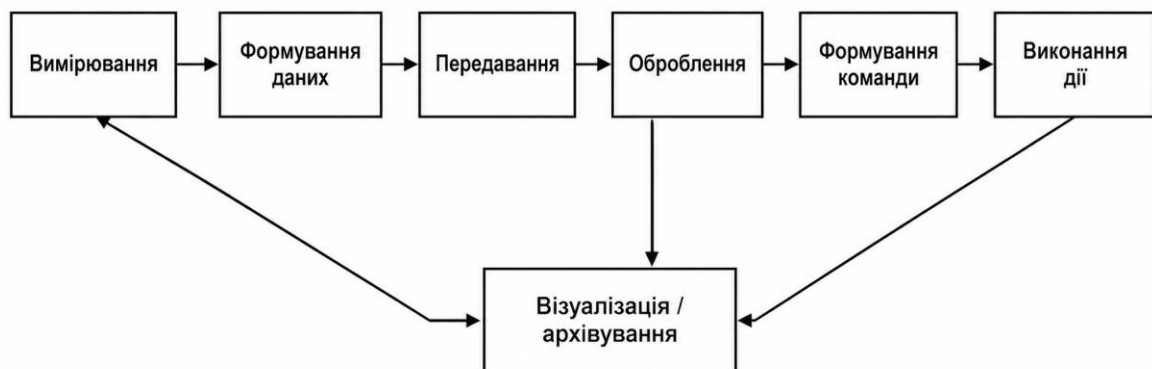


Рисунок 1.6. Функціональна схема інформаційно-керувального циклу дільниці сортування

10. Вимоги до оформлення результатів роботи

У звіті необхідно подати:

1. назву та мету практичної роботи;
2. вихідні дані відповідно до варіанта;
3. короткий опис технологічного об'єкта;
4. перелік вимірюваних параметрів;
5. перелік керувальних впливів;
6. таблицю фізичних компонентів системи;
7. розрахунок кількості каналів введення-виведення;
8. структурну схему КФВС;

9. функціональну схему КФВС;
- 10.таблицю основних інформаційних потоків;
- 11.розрахунок інформаційного навантаження;
- 12.оцінку часу реакції системи;
- 13.обґрунтування розподілу функцій між рівнями;
- 14.висновки.

Структурна схема повинна містити стрілки напрямку інформаційного та керувального обміну. Недостатньо з'єднати блоки лініями без пояснення характеру зв'язку.

11. Висновки до практичної роботи

У результаті виконання практичної роботи формується структурне та функціональне представлення кіберфізичної виробничої системи для заданого технологічного об'єкта. Структурна модель визначає склад фізичних, обчислювальних і комунікаційних компонентів, тоді як функціональна модель встановлює послідовність отримання, передавання, оброблення та використання технологічної інформації.

Розрахунок кількості каналів введення-виведення дозволяє узгодити фізичну частину установки із засобами автоматизації. Оцінка інформаційного потоку показує співвідношення між кількістю технологічних даних та пропускнуою здатністю каналу. Розрахунок часу реакції дає змогу перевірити, чи може запропонований шлях «датчик – оброблення – виконавчий пристрій» виконувати функцію в заданих часових межах.

Основним результатом роботи є не сама блок-схема, а обґрунтований розподіл функцій. Швидкі контури керування повинні замикатися на рівні, де забезпечується необхідний час реакції. Дані, потрібні для диспетчеризації, діагностування, накопичення та подальшого аналізу, можуть передаватися на вищі рівні. Такий розподіл усуває залежність критичних технологічних операцій від компонентів, які безпосередньо не беруть участі в керуванні.

ПРАКТИЧНА РОБОТА № 2. РОЗРОБЛЕННЯ АЛГОРИТМУ КЕРУВАННЯ ТЕХНОЛОГІЧНИМ ОБ'ЄКТОМ У СКЛАДІ КІБЕРФІЗИЧНОЇ СИСТЕМИ

1. Мета практичного заняття

Мета роботи – набути навичок розроблення алгоритму керування технологічним об'єктом як складовою кіберфізичної виробничої системи з урахуванням режимів роботи, стану технологічного обладнання,

вимірюваних координат, обмежень на керувальні впливи, аварійних умов та інформаційного обміну з іншими компонентами системи.

Під час виконання роботи необхідно навчитися:

- формалізувати послідовність роботи технологічного об'єкта;
- визначати стани та умови переходу між ними;
- формувати алгоритм циклічного опитування датчиків;
- перевіряти достовірність отриманих даних;
- розраховувати похибку регулювання;
- формувати керувальний вплив за ПІ-законом;
- враховувати обмеження виконавчого механізму;
- реалізовувати технологічні блокування;
- визначати реакцію алгоритму на аварійні ситуації;
- оцінювати час виконання циклу керування;
- визначати, які дані використовуються локально, а які передаються на вищий рівень КФВС;
- подавати алгоритм у вигляді блок-схеми або діаграми станів.

2. Прилади і матеріали

Для виконання практичної роботи використовуються:

1. персональний комп'ютер;
2. програмне середовище для побудови алгоритмічних схем;
3. текстовий або табличний процесор;
4. вихідні дані відповідно до номера варіанта;
5. матеріали практичної роботи № 1;
6. опис технологічного об'єкта;
7. перелік датчиків та виконавчих механізмів;
8. матеріали лекційного курсу;
9. нормативні документи .

Для побудови алгоритму допускається застосовувати блок-схеми, діаграми станів, SFC або текстове представлення логіки. IEC 61131-3:2025 визначає для програмованих контролерів текстову мову Structured Text, графічні мови Ladder Diagram і Function Block Diagram, а також Sequential Function Chart як засіб структуризації послідовності виконання програм і функціональних блоків.

3. Теоретичні положення

3.1. Алгоритм керування у складі кіберфізичної виробничої системи

Алгоритм локального контролера в КФВС не можна зводити лише до обчислення керувального сигналу. Перед розрахунком регулятора система повинна отримати дані від фізичного об'єкта, перевірити їхню придатність, визначити поточний режим, перевірити технологічні блокування і тільки після цього сформувати вплив на виконавчий механізм.

Після виконання керувальної операції формується зворотний інформаційний потік: значення технологічних координат, режим роботи, команди, аварії та діагностичні ознаки передаються до інших компонентів КФВС. Таке поєднання фізичної дії, локального обчислення та інформаційного обміну характерне для кіберфізичних виробничих систем

Для практичної роботи цикл керування доцільно представити послідовністю:

отримання даних → перевірка даних → визначення стану → перевірка блокувань → розрахунок керувального впливу → обмеження сигналу → передавання команди виконавчому механізму → контроль результату → передавання стану до інформаційного рівня.

Деякі функції в КФВС можуть виконуватися розподілено. Проте операції, для яких задано жорсткий час реакції, доцільно залишати на локальному рівні. Передавання їх до віддаленого сервісу збільшує кількість складових затримки та створює додаткову залежність від мережевого з'єднання. Дослідження CPPS також розглядають розподілене й автономне прийняття рішень як одну з характерних властивостей таких систем.

3.2. Представлення алгоритму через стани

Для технологічного обладнання з дискретними режимами зручно використовувати модель скінченного автомата. Стан визначає, який набір операцій дозволено виконувати в конкретний момент.

Узагальнена зміна стану

$$S_{k+1} = F(S_k, X_k),$$

де S_k – стан технологічного об'єкта на поточному кроці; S_{k+1} – стан на наступному кроці; F – функція переходів; X_k – сукупність логічних умов, команд та діагностичних ознак.

Для навчального завдання приймається така базова множина станів:

- **S0 – «Ініціалізація»;**
- **S1 – «Готовність»;**
- **S2 – «Автоматична робота»;**
- **S3 – «Зупинка»;**
- **S4 – «Аварія».**

За потреби до неї можуть додаватися стани «Ручний режим», «Налагодження», «Очікування матеріалу», «Технологічна пауза» тощо.

Перехід між станами повинен виконуватися за однозначною умовою. Наприклад:

S0 → S1: завершена самодіагностика, датчики доступні, аварій немає;

S1 → S2: надійшла команда «Пуск» і виконані технологічні блокування;

S2 → S3: надійшла команда штатної зупинки;

будь-який робочий стан → S4: виникла аварійна умова;

S4 → S1: причина аварії усунута, виконане підтверджене скидання аварії та дозволений повторний запуск.

Остання умова принципова: усунення аварійної ознаки не повинно автоматично означати відновлення технологічного руху, якщо це може спричинити неочікуваний пуск обладнання.

3.3. Опитування та перевірка вхідних даних

Першим етапом циклу автоматичного керування є зчитування інформації про фізичний об'єкт.

Для аналогової координати перевіряється належність виміряного значення допустимому діапазону:

$$y_{\min} \leq y_k \leq y_{\max},$$

де y_k – поточне виміряне значення регульованої координати; $y_{\min}$, $y_{\max}$ – допустимі нижня та верхня межі вимірювання.

Окремо може контролюватися швидкість зміни сигналу:

$$v_{y,k} = \frac{|y_k - y_{k-1}|}{T_s},$$

де $v_{y,k}$ – швидкість зміни вимірюваної координати; y_{k-1} – попереднє виміряне значення; T_s – період дискретизації алгоритму.

Умова достовірності за цим критерієм має вигляд

$$v_{y,k} \leq v_{y,\max},$$

де $v_{y,\max}$ – максимально допустима для заданого алгоритму швидкість зміни сигналу.

Ця перевірка не доводить фізичну достовірність вимірювання, але дозволяє виявити частину очевидно некоректних значень: різкі стрибки, значення поза заданим діапазоном або втрату оновлення даних.

Для мережевого датчика до логічної перевірки доцільно додати контроль ознаки якості або актуальності даних.

3.4. Розрахунок похибки керування

Для замкненого контуру керування похибка визначається різницею між заданим і вимірним значенням:

$$e_k = r_k - y_k,$$

де e_k – поточна похибка регулювання; r_k – задане значення технологічної координати.

Знак похибки повинен узгоджуватися з напрямом дії виконавчого механізму.

Якщо збільшення керувального сигналу призводить до збільшення регульованої величини, позитивна похибка повинна формувати збільшення керувального впливу.

3.5. Зона нечутливості

Для частини технологічних об'єктів непотрібно реагувати на дуже малі відхилення параметра. Тоді в алгоритм вводиться зона нечутливості:

$$|e_k| \leq \Delta,$$

де Δ – допустима ширина відхилення від заданого значення.

Якщо ця умова виконується, алгоритм може зберігати попередній керувальний сигнал.

Зона нечутливості застосовується лише тоді, коли невелике відхилення не погіршує технологічний процес. Вона не повинна вводитися механічно для кожного об'єкта.

3.6. Формування керувального впливу

3.6.1. ПІ-закон керування

Для практичної роботи використовується дискретний ПІ-регулятор. Він достатній для демонстрації зв'язку між вимірюванням, обчисленням і формуванням керувального впливу без введення додаткової диференціальної складової.

У приростній формі закон керування записується як

$$u_k = u_{k-1} + K_p(e_k - e_{k-1}) + K_i T_s e_k,$$

де u_k – поточний керувальний сигнал; u_{k-1} – його значення на попередньому кроці; K_p – коефіцієнт пропорційної складової; e_{k-1} – похибка на попередньому кроці; K_i – коефіцієнт інтегральної складової.

Пропорційна частина реагує на зміну похибки, а інтегральна складова накопичує її вплив у часі.

У практичній роботі коефіцієнти регулятора задані у таблиці варіантів як навчальні вихідні дані.

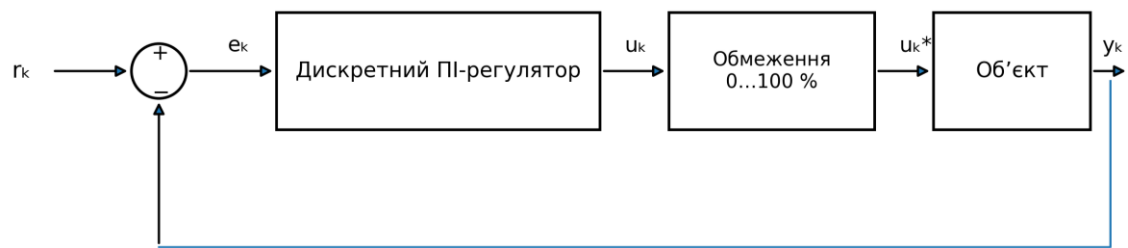


Рисунок 2.1. Структурна схема замкнутого контуру з дискретним ПІ-регулятором

3.6.2. Обмеження керувального сигналу

Результат обчислення регулятора не може перевищувати фізичні можливості виконавчого пристрою.

Обмежений сигнал визначається як

$$u_k^* = \min(u_{\max}, \max(u_{\min}, u_k)),$$

де u_k^* – сигнал після обмеження; $u_{\min}$, $u_{\max}$ – нижня та верхня межі керувального впливу.

Для нормованого керувального сигналу в цій роботі приймається

$$0\% \leq u_k^* \leq 100\%.$$

Наприклад, розраховане значення 112 % повинно бути обмежене до 100 %, а значення –8 % – до 0 %.

3.7. Логічні блокування

ПІ-регулятор сам по собі не визначає, чи дозволено фізично виконувати отриману команду.

Тому керувальний сигнал передається виконавчому пристрою тільки за умови виконання технологічного дозволу:

$$P_k = B_{1,k} \wedge B_{2,k} \wedge \dots \wedge B_{m,k},$$

де P_k – загальний дозвіл на виконання керувальної дії; $B_{i,k}$ – стан i -го блокування; m – кількість блокувань.

Наприклад, для електропривода такими умовами можуть бути:

- захисне огороження знаходиться у допустимому стані;
- аварійна команда відсутня;
- привод готовий;
- технологічний механізм не знаходиться у конфліктному положенні; достовірність необхідних датчиків підтверджена.

Якщо хоча б одна обов'язкова умова не виконана, алгоритм не повинен передавати звичайний автоматичний керувальний сигнал.

3.8. Оброблення аварійного стану

Аварійна логіка має пріоритет над штатним регулюванням.

Загальна ознака аварійного стану для навчальної моделі:

$$A_k = A_{1,k} \vee A_{2,k} \vee \dots \vee A_{q,k},$$

де A_k – загальна аварійна ознака; $A_{j,k}$ – окрема аварійна умова; q – кількість контрольованих аварійних умов.

За умови $A_k=1$ система повинна перейти до визначеного проєктом безпечного стану.

Саме значення безпечного керувального сигналу не можна задавати універсально. Для одного механізму безпечним буде вимкнення, для іншого – підтримання певного режиму або виконання спеціальної послідовності. Тому в реальному проєкті аварійна реакція визначається аналізом конкретного технологічного процесу.



Рисунок 2.2. Формування технологічного дозволу та аварійної ознаки

3.9. Час виконання алгоритму

Цикл керування повинен завершуватися раніше, ніж настане наступний момент його запланованого виконання.

Загальний час одного циклу можна оцінити як

$$T_{\text{ц}} = T_{\text{зч}} + T_{\text{пер}} + T_{\text{обч}} + T_{\text{вих}} + T_{\text{інф}}$$

де $T_{\text{ц}}$ – час циклу алгоритму; $T_{\text{зч}}$ – час зчитування даних; $T_{\text{пер}}$ – час їх перевірки; $T_{\text{обч}}$ – час виконання керувальних обчислень; $T_{\text{вих}}$ – час формування вихідної команди; $T_{\text{інф}}$ – час формування інформації для інших компонентів КФВС.

Для заданого максимального часу циклу повинна виконуватися умова

$$T_{\text{ц}} \leq T_{\text{доп}},$$

де $T_{\text{доп}}$ – максимально допустимий час виконання алгоритмічного циклу.

Запас часу становить

$$\Delta T_{\text{ц}} = T_{\text{доп}} - T_{\text{ц}},$$

де $\Delta T_{\text{ц}}$ – часовий резерв алгоритму.

Значний інформаційний обмін не слід механічно включати до швидкого локального циклу. Частину діагностичних або архівних даних

можна передавати з меншим пріоритетом або з більшим періодом оновлення. Такий розподіл узгоджується з підходами до децентралізованого та контекстно-залежного керування CPPS.

3.10. Інформаційна складова алгоритму

Локальний алгоритм повинен не тільки керувати фізичним процесом, а й формувати цифрове представлення його поточного стану.

Для передачі на інформаційний рівень доцільно формувати набір:

- поточний стан технологічного автомата;
- значення регульованої координати;
- задане значення;
- похибка;
- керувальний сигнал;
- стан блокувань;
- код аварії;
- режим керування;
- ознака достовірності даних;
- часову мітку.

Дослідження кіберфізичних виробничих систем розглядають використання виробничих даних для локального та розподіленого прийняття рішень, оптимізації та адаптації керування.

3.11. Загальна структура алгоритму

Алгоритм автоматичного режиму рекомендується формувати у такій послідовності.

Крок 1. Ініціалізація

Контролер:

перевіряє доступність необхідних сигналів;
встановлює початковий стан;
скидає тимчасові внутрішні змінні;
перевіряє наявність аварійних ознак.

Крок 2. Зчитування вхідних даних

Зчитуються:

аналогові параметри;
дискретні стани;
команди оператора;
дані інтелектуальних пристроїв;
ознаки готовності виконавчих механізмів.

Крок 3. Перевірка даних

Виконуються:

контроль діапазону;
контроль швидкості зміни;

перевірка якості мережевих даних;
перевірка актуальності сигналу.

Крок 4. Перевірка аварій

Якщо виявлено аварійну умову, штатний алгоритм припиняється і система переходить до стану S4.

Крок 5. Визначення режиму

Встановлюється один зі станів:

ініціалізація → готовність → автоматична робота → зупинка
або

будь-який стан → аварія.

Крок 6. Обчислення похибки

В автоматичному режимі визначається e_k .

Крок 7. Перевірка зони нечутливості

Якщо $|e_k| \leq \Delta$, попередній керувальний сигнал може бути збережений.

Якщо $|e_k| > \Delta$, виконується розрахунок ПІ-регулятора.

Крок 8. Обмеження сигналу

Розраховане значення приводиться до фізично допустимого діапазону.

Крок 9. Перевірка блокувань

Команда передається виконавчому пристрою тільки за умови $P_k=1$.

Крок 10. Контроль виконання

У наступних циклах перевіряється, чи призвела керувальна дія до очікуваної зміни фізичної координати.

Крок 11. Передавання даних

Формується інформаційний пакет для SCADA, edge-вузла, сервера або іншого компонента, визначеного у практичній роботі №1.

Крок 12. Перехід до наступного циклу

Після завершення поточного циклу контролер очікує наступного моменту запуску алгоритму.

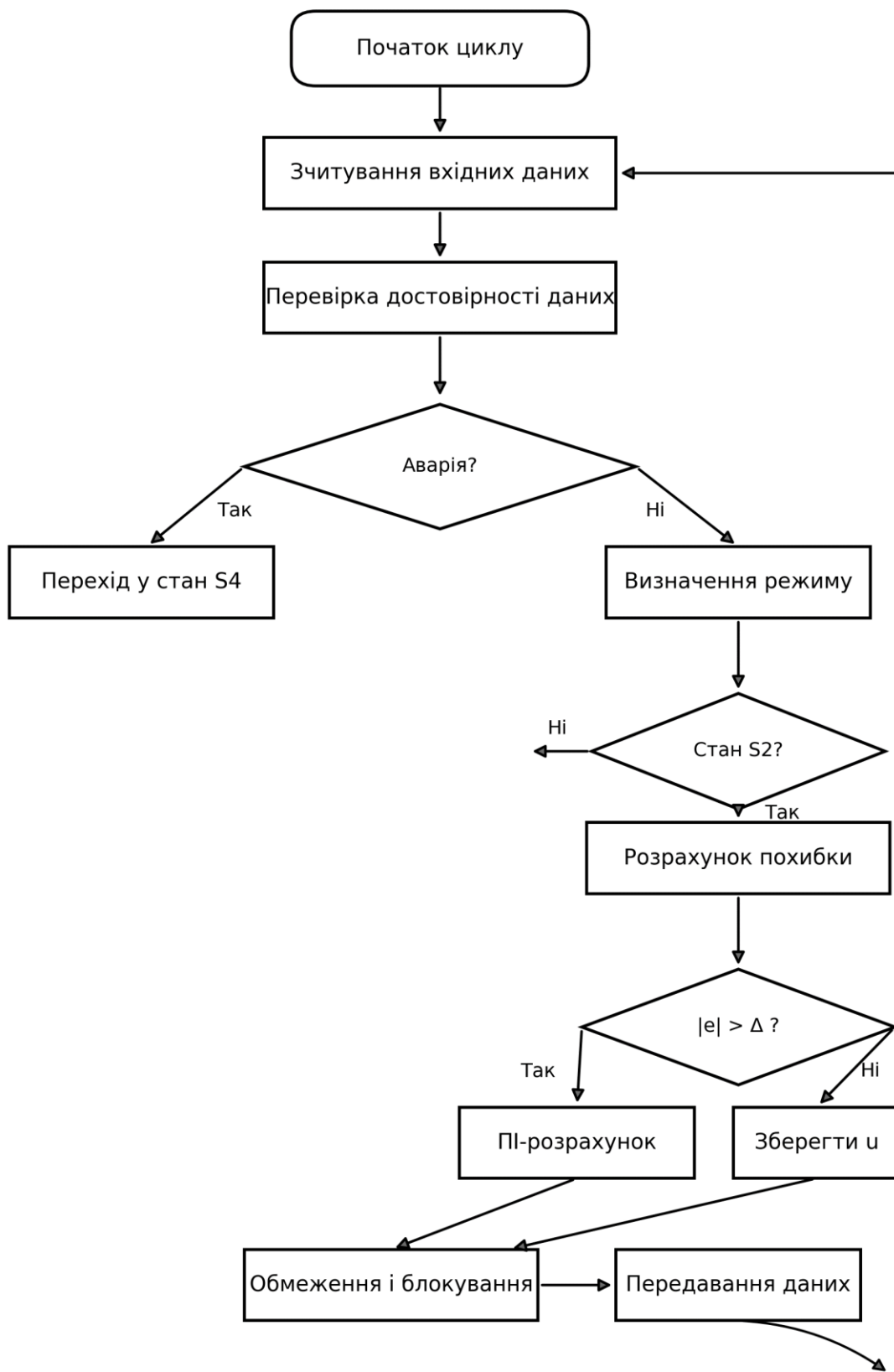


Рисунок 2.3. Загальний алгоритм керування технологічним об'єктом у складі КФВС

4. Методика виконання практичної роботи

4.1. Аналіз об'єкта

За варіантом визначити:

- фізичний об'єкт;
- регульовану координату;
- керувальну дію;
- основний датчик;
- виконавчий механізм;
- дискретні умови дозволу;
- аварійні ознаки.

4.2. Складання таблиці сигналів

Рекомендується використати форму:

№	Сигнал	Тип	Джерело	Призначення
1	Поточне значення координати	AI / мережевий	Датчик	Регулювання
2	Команда «Пуск»	DI / HMI	Оператор	Зміна стану
3	Готовність привода	DI / мережевий	Привод	Блокування
4	Керувальний сигнал	AO / мережевий	ПЛК	Виконавчий механізм
5	Аварія	DI / мережевий	Обладнання	Аварійний перехід

4.3. Розроблення діаграми станів

Для свого варіанта студент повинен побудувати щонайменше чотири стани:

- **S0** – ініціалізація;
- **S1** – готовність;
- **S2** – робота;
- **S4** – аварія.

За потреби додається S3 – штатна зупинка.

Для кожного переходу необхідно записати логічну умову.

Неприпустимою є діаграма, у якій стрілки між станами не мають умов переходу.

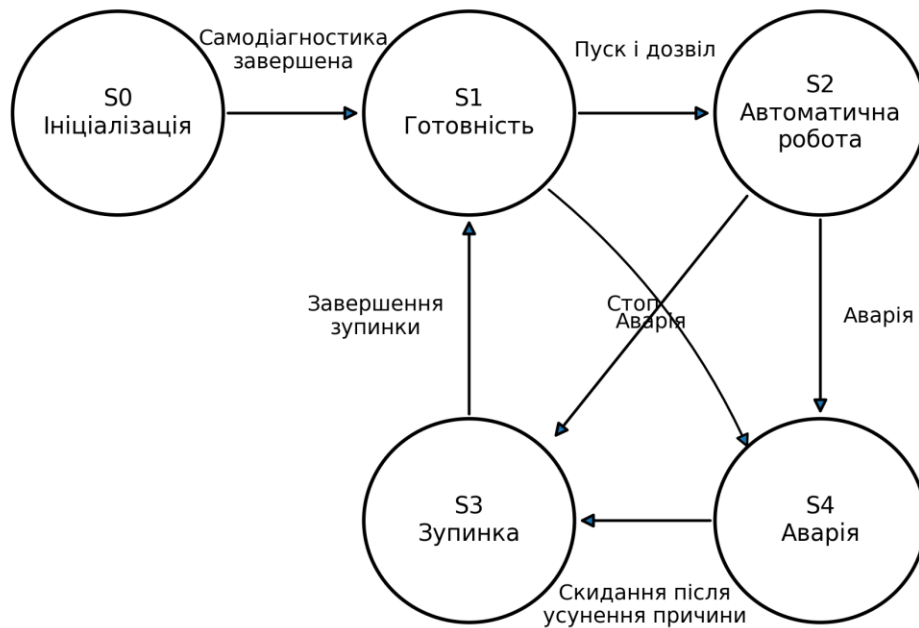


Рисунок 2.4. Діаграма станів технологічного об'єкта

4.4. Таблиця варіантів

Для порівнянності розрахунків технологічні координати та керувальні сигнали в таблиці задано у нормованих відсотках від робочого діапазону.

Варіант	Технологічний об'єкт	Регульована координата	$g_k, \%$	$u_k, \%$	$e_{k-1}, \%$	$u_{k-1}, \%$	K_p	K_{I1}, c^{-1}	T_{11}, c	$\Delta, \%$	T_{22}, ms
1	Конвеєрна транспортна ділянка	Швидкість конвеєра	70	63	9	54	1.4	0.08	0.5	1	100
2	Дозувальний бункер	Рівень матеріалу	65	58	8	48	1.6	0.10	1.0	2	150
3	Насосна установка	Витрата	75	68	9	60	1.2	0.06	0.5	1	120
4	Термічна камера	Температура	80	72	10	58	2.0	0.05	1.0	1	200
5	Пакувальний модуль	Швидкість подачі	60	54	8	47	1.5	0.08	0.5	1	100
6	Змішувальна установка	Частота обертання	68	61	9	52	1.3	0.07	0.5	1	120
7	Система подачі охолоджувача	Витрата	55	49	8	43	1.7	0.09	1.0	2	180
8	Роботизована виробнича комірка	Швидкість технологічного руху	72	65	10	55	1.2	0.05	0.2	1	80
9	Сушильна установка	Витрата повітря	75	66	11	57	1.8	0.08	1.0	2	200
10	Ділянка намотування матеріалу	Швидкість обертання	65	59	8	50	1.4	0.07	0.5	1	100

Для всіх варіантів приймається:

$$u_{\min}=0\%$$

$$u_{\max}=100\%$$

Студент повинен самостійно визначити щонайменше три технологічні блокування та дві аварійні умови, виходячи зі змісту свого технологічного об'єкта. Вони повинні бути фізично змістовними, а не довільними.

4.5. Послідовність розрахунку за варіантом

1. Визначити поточну похибку

Розрахувати e_{k_k} .

2. Порівняти похибку із зоною нечутливості

Перевірити умову

$$|e_k| \leq \Delta$$

3. Якщо похибка знаходиться поза зоною нечутливості – розрахувати новий керувальний сигнал

Використати дискретний ПІ-закон.

4. Виконати обмеження сигналу

Перевірити відповідність діапазону 0–100 %.

5. Перевірити технологічний дозвіл

Сформувати логічний вираз $P_k P_{_k}$.

6. Сформувати аварійний вираз

Визначити $A_k A_{_k}$.

7. Розробити діаграму станів

Визначити умови всіх переходів.

8. Розрахувати час циклу

Використати задані або отримані під час моделювання складові часу.

9. Порівняти час із допустимим

Визначити часовий резерв.

10. Розробити блок-схему повного алгоритму

На схемі повинні бути присутні:

початок → ініціалізація → опитування → перевірка даних →
перевірка аварій → визначення режиму → розрахунок → обмеження →
блокування → вихідна команда → передавання даних → повтор циклу.

5. Приклад розрахунку

5.1. Вихідні дані

Об'єкт: вузол регулювання витрати повітря сушильної камери.

Приймаємо:

- задана витрата – 70 % робочого діапазону;
- поточне значення – 62 %;
- попередня похибка – 10 %;
- попередній керувальний сигнал – 45 %;
- $K_p=1,8$;
- $K_i=0,12 \text{ с}^{-1}$;
- $T_s=0,5 \text{ с}$;
- зона нечутливості – 1 %;
- попереднє вимірне значення – 60 %;
- допустима швидкість зміни вимірювання – 10 %/с;
- максимально допустимий час циклу – 80 мс.

Для оцінки часу циклу приймаємо:

- зчитування – 8 мс;
- перевірка даних – 3 мс;
- обчислення – 5 мс;
- формування вихідної команди – 4 мс;
- формування інформаційних даних – 10 мс.

5.2. Перевірка швидкості зміни сигналу

$$v_{y,k} = \frac{|62 - 60|}{0,5} = 4 \text{ \%}/\text{с}$$

Перевіряємо:

$$4 < 10 \text{ \%}/\text{с}$$

За прийнятим критерієм стрибок вимірюваного сигналу не виходить за задану межу.

5.3. Розрахунок похибки

$$e_k = 70 - 62 = 8\%$$

5.4. Перевірка зони нечутливості

$$|8| > 1$$

Отже, похибка знаходиться за межами зони нечутливості, тому необхідно обчислити новий керувальний сигнал.

5.5. Розрахунок ПІ-регулятора

$$u_k = 45 + 1,8(8 - 10) + 0,12 \cdot 0,5 \cdot 8$$

$$u_k = 45 - 3,6 + 0,48 = 41,88\%$$

Зменшення сигналу від 45 до 41,88 % узгоджується з тим, що похибка зменшилася з 10 до 8 %.

5.6. Перевірка обмеження

$$0 < 41,88 < 100$$

Отже

$$u_k^* = 41,88\%$$

Обмеження сигналу не активується.

5.7. Формування технологічного дозволу

Для прикладу приймаємо три логічні умови:

- B1B_1 – вентилятор готовий до роботи;
- B2B_2 – захисне блокування справне;
- B3B_3 – датчик витрати має достовірний стан.

Тоді

$$P_k = B_{1,k} \wedge B_{2,k} \wedge B_{3,k}$$

Припустимо:

$$B_{1,k}=1$$

$$B_{2,k}=1$$

$$B_{3,k}=1$$

Тоді

$$P_k = 1 \wedge 1 \wedge 1 = 1$$

Керувальний сигнал 41,88 % дозволено передати частотному приводу вентилятора.

5.8. Перевірка аварійних умов

Для прикладу приймаються дві аварійні ознаки:

- A1A_1 – аварія частотного привода;
- A2A_2 – неприпустиме перевищення температури сушильної камери.

У штатному режимі:

$$A_{1,k} = 0$$

$$A_{2,k} = 0$$

Тоді

$$A_k = 0 \vee 0 = 0$$

Перехід до аварійного стану не виконується.

5.9. Розрахунок часу алгоритмічного циклу

$$T_{\text{ц}} = 8 + 3 + 5 + 4 + 10 = 30 \text{ мс}$$

Перевіряємо умову:

$$30 < 80 \text{ мс}$$

Отже, задане обмеження виконується.

Часовий резерв становить

$$\Delta T_{\text{ц}} = 80 - 30 = 50 \text{ мс}$$

Алгоритм має 50 мс резерву відносно заданого максимально допустимого часу циклу.

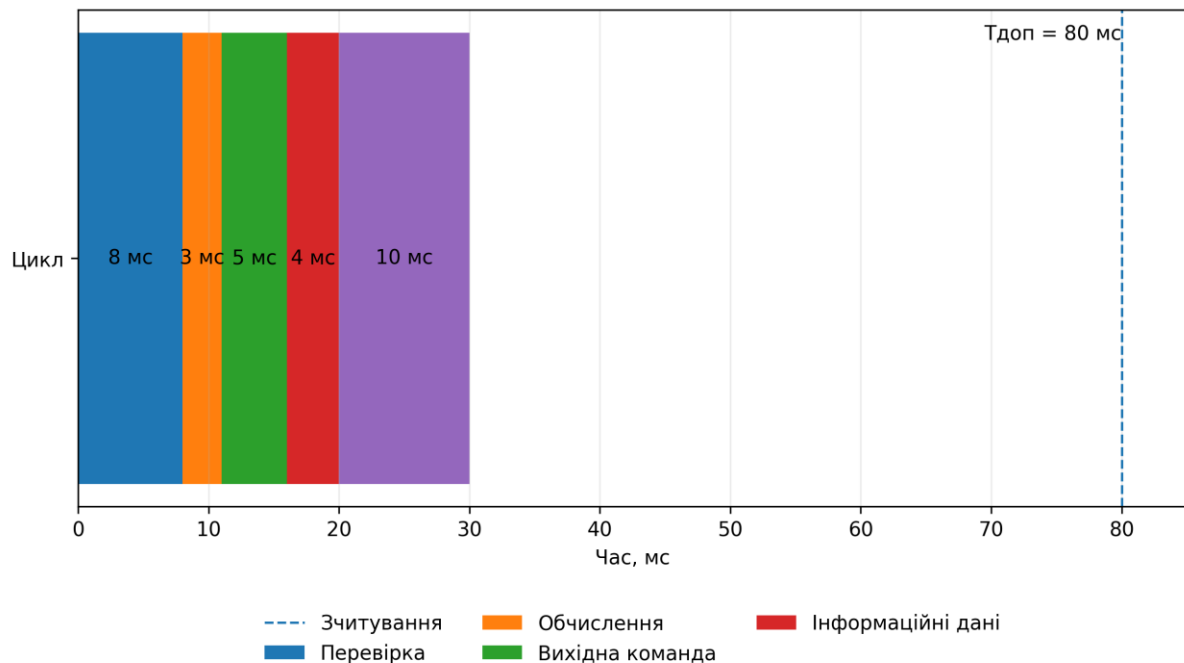


Рисунок 2.5. Часова структура алгоритмічного циклу для розрахункового прикладу

5.10. Діаграма станів для прикладу

Для вузла сушильної камери приймається така послідовність.

S0 – Ініціалізація

Виконуються:

- запуск контролера;
 - перевірка зв'язку;
 - перевірка датчика витрати;
 - перевірка привода;
- встановлення початкових змінних.

Якщо перевірки завершені без аварій:

S0 → S1.

S1 – Готовність

Виконавчий механізм знаходиться у стані очікування.

Умова переходу:

Пуск = 1 AND усі блокування виконані → S2.

S2 – Автоматична робота

Циклічно виконуються:

- зчитування витрати;
- перевірка вимірювання;
- розрахунок похибки;
- ПІ-регулювання;
- обмеження;
- передавання сигналу приводу;
- контроль результату;
- передавання даних SCADA.

За командою штатної зупинки:

S2 → S3.

При аварії:

S2 → S4.

S3 – Зупинка

Виконується передбачена технологією процедура зупинки.

Після завершення:

S3 → S1.

S4 – Аварія

Штатне ПІ-регулювання припиняється.

Виконується аварійна реакція, визначена для об'єкта.

Повернення дозволяється тільки після:

- усунення причини;
- підтвердження скидання;
- повторної перевірки блокувань.

Після цього:

S4 → S1.

5.11. Інформаційний обмін прикладового алгоритму

Після кожного циклу локальний контролер формує такі дані:

Параметр	Значення прикладу	Призначення
Стан автомата	S2	Відображення режиму
Задана координата	70 %	Контроль завдання
Поточна координата	62 %	Моніторинг процесу
Похибка	8 %	Аналіз регулювання
Керувальний сигнал	41,88 %	Контроль приводу
Дозвіл	1	Діагностика блокувань
Аварійний стан	0	Сигналізація
Достовірність датчика	1	Контроль якості даних

Не всі ці дані повинні передаватися з тим самим періодом, з яким виконується локальний регулятор. Наприклад, контур може обчислюватися через 0,5 с, а архівування частини параметрів виконуватися рідше. Так розділяються вимоги керування та вимоги інформаційного моніторингу.

5.12. Узагальнений алгоритм для прикладового варіанта

Послідовність роботи має такий вигляд:

1. Початок циклу.

2. Зчитування датчика витрати.

3. Перевірка діапазону та швидкості зміни сигналу.

4. Перевірка аварій.

Якщо аварія є $\rightarrow$ S4 «Аварія».

Якщо аварії немає $\rightarrow$ перехід далі.

5. Визначення поточного стану.

Якщо система не перебуває у S2 $\rightarrow$ регулятор не виконується.

Якщо система перебуває у S2 $\rightarrow$ розраховується похибка.

6. Перевірка зони нечутливості.

Якщо похибка мала $\rightarrow$ зберігається попереднє значення.

Якщо похибка перевищує зону нечутливості $\rightarrow$ виконується ПП-розрахунок.

7. Обмеження керувального сигналу.

8. Перевірка блокувань.

Якщо $P_k=0$ $P_k=0 \rightarrow$ команда не передається.

Якщо $P_k=1$ $P_k=1 \rightarrow$ команда передається приводу.

9. Формування даних про стан КФВС.

10. Передавання даних до SCADA / edge-рівня.

11. Завершення поточного циклу.

12. Повтор алгоритму через T_s .

6. Завдання до практичної роботи

За номером свого варіанта необхідно:

1. описати технологічний об'єкт;
2. визначити регульовану координату;
3. визначити керувальну координату;
4. сформулювати перелік вхідних та вихідних сигналів;
5. визначити стани технологічного автомата;
6. задати умови переходів;
7. визначити не менше трьох технологічних блокувань;
8. визначити не менше двох аварійних умов;
9. розрахувати поточну похибку;
10. перевірити зону нечутливості;

11. розрахувати керувальний сигнал;
12. перевірити обмеження керувального сигналу;
13. сформувати логічний вираз технологічного дозволу;
14. сформувати аварійну логіку;
15. оцінити час виконання циклу;
16. визначити дані, які передаються до інформаційного рівня КФВС;
17. побудувати діаграму станів;
18. побудувати повну блок-схему алгоритму;
19. зробити висновок щодо працездатності запропонованого алгоритму.

7. Вимоги до оформлення звіту

Звіт повинен містити:

1. тему роботи;
2. мету;
3. номер варіанта;
4. опис технологічного об'єкта;
5. структурний фрагмент об'єкта керування;
6. таблицю вхідних і вихідних сигналів;
7. перелік режимів роботи;
8. діаграму станів;
9. умови переходу між станами;
10. розрахунок похибки;
11. розрахунок ПІ-регулятора;
12. перевірку обмежень;
13. перелік технологічних блокувань;
14. аварійну логіку;
15. розрахунок часу циклу;
16. перелік даних для інформаційного обміну;
17. блок-схему повного алгоритму;
18. висновки.

У блок-схемі не слід використовувати один блок «керування об'єктом», що приховує всю логіку. Необхідно показати окремо отримання даних, перевірку, прийняття рішення, розрахунок та формування керувального сигналу.

8. Висновки до практичної роботи

Під час виконання практичної роботи алгоритм керування технологічним об'єктом розглядається як частина кіберфізичного контуру, а не як ізольований регулятор. Такий алгоритм одночасно працює з фізичними сигналами, дискретними станами обладнання, математичним законом керування, технологічними обмеженнями та інформаційними даними.

Поділ алгоритму на стани дозволяє відокремити штатне регулювання від операцій пуску, зупинки та аварійної реакції. ПІ-регулятор формує керувальний вплив тільки у стані, де автоматичне регулювання дозволене. Перед видачею команди результат обчислення обмежується можливостями виконавчого механізму та перевіряється системою блокувань.

Оцінювання часу циклу дає змогу перевірити, чи може алгоритм виконуватися із заданою періодичністю. Якщо ця умова не виконується, проблема не усувається збільшенням коефіцієнтів регулятора: необхідно змінювати структуру програмного циклу, перерозподіляти обчислення або переглядати інформаційний обмін.

Для КФВС додатковим результатом алгоритму є цифровий опис поточного стану об'єкта. Вимірювання, завдання, керувальний сигнал, стан автомата, блокування й аварійні ознаки можуть використовуватися SCADA, edge-вузлом, системою діагностування або цифровим двійником. Саме цей інформаційний зв'язок між фізичним керуванням і цифровими функціями відрізняє розглянуту задачу від розроблення звичайного локального алгоритму автоматизації.

ПРАКТИЧНА РОБОТА № 3. ПРОЄКТУВАННЯ ПРОМИСЛОВОЇ МЕРЕЖІ ТА ІНФОРМАЦІЙНОГО ОБМІНУ КІБЕРФІЗИЧНОЇ ВИРОБНИЧОЇ СИСТЕМИ

1. Мета практичного заняття

Мета роботи – набути навичок проєктування комунікаційної підсистеми кіберфізичної виробничої системи, визначення структури промислової мережі, формування інформаційних потоків між її вузлами, оцінювання мережевого навантаження та часових характеристик передавання технологічних даних.

У процесі виконання роботи необхідно навчитися:

- визначати склад вузлів промислової мережі;
- встановлювати джерела та споживачів технологічних даних;
- розділяти циклічний, подієвий, діагностичний та інформаційний трафік;
- обирати структуру мережі відповідно до архітектури КФВС;
- визначати доцільність використання OPC UA Client/Server та OPC UA PubSub;
- розраховувати фактичний інформаційний потік з урахуванням службових даних;
- оцінювати завантаження каналу;

- визначати інтенсивність передавання повідомлень;
- оцінювати час передавання кадру;
- розраховувати затримку проходження критичного повідомлення через мережу;
- порівнювати отриману затримку з допустимою;
- визначати функціональне призначення сегментів мережі;
- формувати схему інформаційного обміну між польовим, керувальним та інформаційним рівнями.

2. Прилади і матеріали

Для виконання практичної роботи використовуються:

1. персональний комп'ютер;
2. програмне середовище для побудови мережевих та структурних схем;
3. табличний процесор для виконання розрахунків;
4. вихідні дані відповідно до номера варіанта;
5. результати практичних робіт № 1 і № 2;
6. опис технологічного об'єкта;
7. перелік контролерів, датчиків, виконавчих пристроїв та інформаційних вузлів;
8. матеріали лекційного курсу;
9. нормативні документи IEC та IEEE;

3. Теоретичні положення

3.1. Промислова мережа як складова КФВС

Комунікаційна підсистема кіберфізичної виробничої системи забезпечує зв'язок між фізичним обладнанням, локальними контролерами, периферійними обчислювальними вузлами та програмними системами вищого рівня. Її функція не зводиться до передавання максимально можливого обсягу даних. Для технологічного керування значення мають також затримка доставки повідомлення, її варіація, послідовність надходження даних, доступність каналу та можливість розділення потоків з різними вимогами.

Для КФВС характерна одночасна передача кількох типів інформації. Контролер може кожні десятки мілісекунд отримувати дані від польових пристроїв, значно рідше передавати діагностичну інформацію, а події аварійного характеру – передавати без очікування чергового інформаційного циклу.

Тому під час проєктування необхідно встановити не тільки загальний обсяг трафіку, а й характеристики кожного потоку:

- джерело;
- приймач;
- розмір повідомлення;
- період передавання;
- критичність;
- допустиму затримку;
- спосіб доставки.

Вимоги промислових мереж істотно залежать від конкретного застосування, тому одна й та сама мережна конфігурація не може вважатися оптимальною для всіх виробничих систем [24].

3.2. Рівні інформаційного обміну

Для практичної роботи доцільно виділити три основні групи мережевих взаємодій.

Полевий обмін

Виконується між:

- датчиками;
- інтелектуальними вимірювальними пристроями;
- електроприводами;
- віддаленими модулями введення-виведення;
- локальними контролерами.

Для цього рівня характерні невеликі повідомлення та підвищені вимоги до часу їх доставки.

Обмін між засобами керування

До нього належать зв'язки:

- ПЛК – ПЛК;
- ПЛК – контролер руху;
- ПЛК – промисловий комп'ютер;
- контролер – edge-вузол.

На цьому рівні може передаватися як циклічна, так і подієва інформація.

Вертикальний інформаційний обмін

Виконується між локальним рівнем автоматизації та:

- SCADA;
- сервером історичних даних;
- edge-платформою;
- цифровим двійником;
- MES;
- іншими інформаційними сервісами.

Тут більшу роль відіграє семантична структура даних, можливість стандартизованого доступу та незалежність програмного забезпечення від конкретного виробника контролера.

OPC UA використовується саме як стандартизована архітектура обміну інформацією між промисловими програмними й апаратними компонентами. Чинний IEC 62541-1:2025 визначає загальні концепції OPC UA, а IEC 62541-4:2025 – набір сервісів взаємодії клієнтів і серверів.

3.3. Топологія промислової мережі

Топологія визначає фізичний або логічний спосіб з'єднання мережевих вузлів.

Для виробничих систем найбільш характерними є:

зіркова структура – пристрої з'єднуються з центральним або локальним комутатором;

ієрархічна зірка – декілька локальних комутаторів об'єднуються через комутатор вищого рівня;

лінійна структура – вузли або пристрої послідовно включені в інформаційний шлях;

кільцева структура – формується альтернативний шлях передавання даних, який може використовуватися механізмами резервування.

Вибір топології не повинен визначатися лише кількістю кабелів. Необхідно враховувати:

- фізичне розташування обладнання;
- кількість вузлів;
- вимоги до резервування;
- допустиму кількість проміжних комутаторів;
- розподіл критичного трафіку;
- можливість подальшого розширення.

Для навчальної КФВС у більшості варіантів використовується ієрархічна зіркова структура. Вона дозволяє окремо сформувати польовий сегмент і сегмент керування, а потім зв'язати їх з інформаційним рівнем.

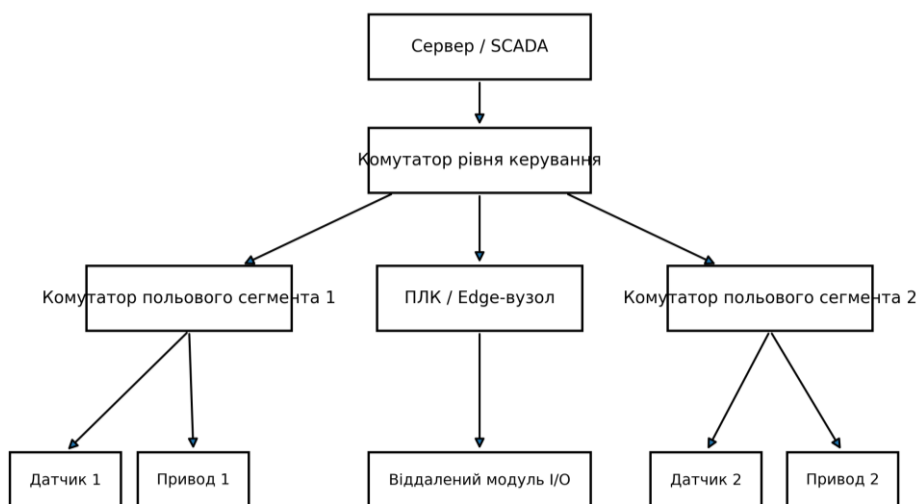


Рисунок 3.1. Ієрархічна топологія промислової мережі кіберфізичної виробничої системи

3.4. Комутація промислового Ethernet

Комутатор приймає Ethernet-кадр на одному порту та передає його через порт, який відповідає маршруту до потрібного вузла.

Кожний проміжний комутатор додає певну затримку. Крім неї, виникають:

- час передавання кадру фізичним каналом;
- затримка очікування в черзі;
- затримка поширення сигналу;
- час оброблення даних на кінцевих вузлах.

Тому номінальної швидкості Ethernet недостатньо для оцінювання часових властивостей мережі.

Мережа 100 Мбіт/с може мати невелике середнє завантаження і водночас не задовольняти вимогу певного контуру за максимальною затримкою, якщо кадр потрапляє до некерованої черги.

3.5. OPC UA Client/Server

Модель Client/Server передбачає адресну взаємодію клієнта із сервером. OPC UA Client формує запит до OPC UA Server, після чого сервер виконує відповідний сервіс і повертає результат.

IEC 62541-4:2025 визначає OPC UA Services як абстрактні віддалені виклики, через які реалізується взаємодія клієнтів та серверів.

Такий механізм доцільний, наприклад, коли:

- SCADA зчитує параметри ПЛК;
- операторська станція змінює технологічне завдання;
- інформаційний застосунок запитує стан конкретного вузла;

- клієнт звертається до структурованої інформаційної моделі обладнання.

Client/Server зручний для адресного доступу до даних, але не кожний виробничий обмін природно описується схемою «запит – відповідь».

Read / Write / Subscription



Рисунок 3.2. Схема інформаційної взаємодії OPC UA Client/Server

3.6. OPC UA PubSub

У моделі Publish/Subscribe джерело публікує набір даних, а підписники отримують його без формування окремого запиту кожним споживачем.

IEC 62541-14:2026 визначає PubSub як окрему модель OPC UA, яка доповнює Client/Server. Стандарт охоплює конфігурацію PubSub, структуру повідомлень та їх відображення на транспортні механізми. Редакція 2026 року замінила IEC 62541-14:2020.

PubSub доцільний, коли одні й ті самі виробничі дані потрібні кільком споживачам.

Наприклад, контролер може публікувати:

поточний стан обладнання → SCADA + edge-вузол + цифровий двійник.

У цьому випадку інформаційне джерело не повинно підтримувати окремий логічний процес опитування для кожного споживача.

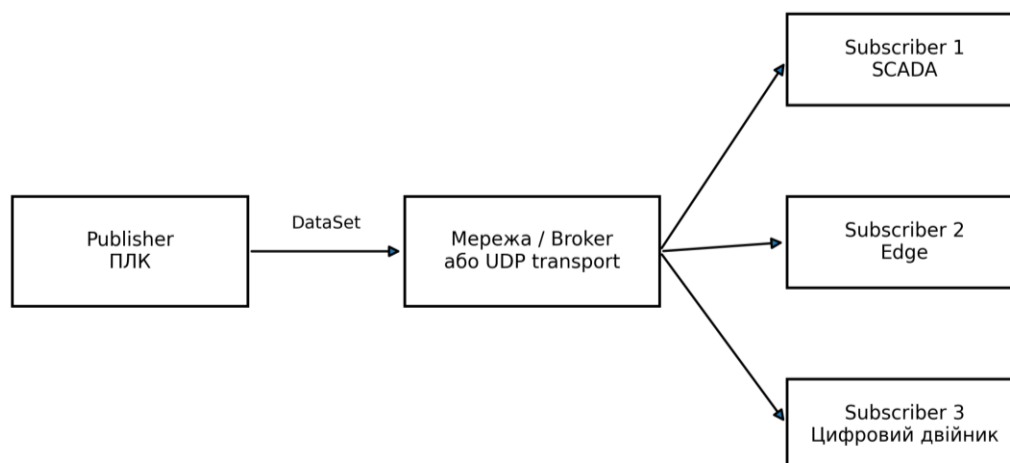


Рисунок 3.3. Схема інформаційної взаємодії OPC UA PubSub

3.7. OPC UA і TSN

OPC UA та TSN вирішують різні задачі.

OPC UA визначає структуру інформаційної взаємодії, моделі даних та сервіси прикладного рівня. TSN є сукупністю механізмів сімейства IEEE 802.1 для Ethernet-мереж, спрямованих, зокрема, на керування передаванням потоків із часовими вимогами.

Тому некоректно розглядати TSN як заміну OPC UA.

Логічно їх співвідношення можна подати так:

- OPC UA – що і в якій інформаційній формі передається;
- TSN – за якими правилами Ethernet-мережа забезпечує передавання критичних потоків.

Експериментальна реалізація показала можливість побудови виробничої комунікаційної архітектури, в якій TSN використовується як мережевий базис, а OPC UA – для горизонтального й вертикального інформаційного обміну.

3.8. Часочутливі мережі TSN

TSN не є одним окремим протоколом. Це група стандартизованих механізмів Ethernet, пов'язаних із:

- синхронізацією часу;
- формуванням та плануванням трафіку;
- обмеженням затримок;
- підвищенням надійності доставки;
- резервуванням мережевих ресурсів.

IEEE 802.1Q-2022 є чинним базовим стандартом для мостів і мостових мереж; до нього також належать механізми керування різними класами Ethernet-трафіку.

У навчальній роботі детальна конфігурація TSN-розкладу не виконується. Здобувач повинен визначити, які потоки потребують пріоритетної або детермінованої передачі, а які можуть працювати в режимі звичайного інформаційного трафіку.

4. Класифікація інформаційних потоків

Для виконання роботи інформацію необхідно поділити на чотири класи.

4.1. Циклічний технологічний трафік

Передається з постійним періодом.

Наприклад:

- поточна швидкість привода;

- положення механізму;
- тиск;
- температура;
- керувальне завдання.

Час доставки такого повідомлення повинен бути узгоджений із періодом роботи алгоритму керування.

4.2. Подієвий трафік

Формується після виникнення певної події:

- аварія;
- зміна режиму;
- досягнення граничного положення;
- зміна стану обладнання.

Для подієвого повідомлення середня частота передавання може бути малою, але конкретне повідомлення може мати жорстку вимогу до затримки.

4.3. Діагностичний трафік

Містить:

- стан пристроїв;
- коди помилок;
- температуру електронних вузлів;
- лічильники;
- інформацію про якість зв'язку.

Такий трафік зазвичай не повинен конкурувати за мережеві ресурси з критичними командами керування.

4.4. Інформаційний трафік вищого рівня

До нього можуть входити:

- архівні дані;
- виробнича статистика;
- технологічні звіти;
- параметри цифрового двійника;
- дані MES;
- агреговані параметри обладнання.

Сам факт наявності вільної пропускну здатності не означає, що ці дані необхідно передавати з максимальною частотою.

5. Методика розрахунку

Позначення L_i , T_i , C та $K_{\text{мер}}$, введені у практичній роботі №1, зберігаються.

У практичній роботі №3 до розрахунку додається службова складова повідомлення та мережевий шлях через комутатори.

5.1. Розрахунок фактичного мережевого потоку

Для групи однотипних вузлів мережевий потік визначається як

$$R_{\text{мер},i} = \frac{8n_i(L_i + H_i)}{T_i},$$

де $R_{\text{мер},i}$ – мережевий потік для i -ї групи пристроїв, біт/с; n_i – кількість пристроїв у групі; H_i – прийнятий у розрахунковій моделі обсяг службових даних одного повідомлення, байт.

Сумарний потік становить

$$R_{\text{мер},\Sigma} = \sum_{i=1}^m R_{\text{мер},i},$$

де $R_{\text{мер},\Sigma}$ – сумарний мережевий інформаційний потік.

У цій практичній роботі H_i є параметром, у якому сумарно враховуються прийняті для варіанта службові витрати. Він не подається як універсальний розмір заголовка Ethernet або OPC UA, оскільки реальний обсяг службової інформації залежить від конкретного стека протоколів і способу передавання.

5.2. Завантаження мережі

$$K_{\text{мер}} = \frac{R_{\text{мер},\Sigma}}{C}$$

У відсотках

$$K_{\text{мер},\%} = 100K_{\text{мер}},$$

де $K_{\text{мер},\%}$ – частка використаної номінальної пропускної здатності, %.

Це середнє розрахункове навантаження. Воно не характеризує максимальну затримку повідомлення в черзі.

5.3. Інтенсивність повідомлень

Загальну кількість повідомлень, що формуються за одиницю часу, можна оцінити як

$$\nu_{\Sigma} = \sum_{i=1}^m \frac{n_i}{T_i},$$

де ν_{Σ} – інтенсивність формування повідомлень, повідомлень/с.

Цей показник доповнює розрахунок бітової швидкості. Два варіанти з однаковим потоком даних можуть мати різну кількість малих повідомлень і, відповідно, різний характер навантаження на комунікаційні вузли.

5.4. Час передавання одного повідомлення

Для одного мережевого повідомлення час його фізичного передавання через канал у прийнятій спрощеній моделі становить

$$t_{\text{сер},i} = \frac{8(L_i + H_i)}{C},$$

де $t_{\text{сер},i}$ – час серіалізації повідомлення в одному сегменті мережі.

Наприклад, збільшення швидкості каналу зменшує цю складову, але не усуває затримку комутатора, черги чи оброблення.

5.5. Оцінка наскрізної затримки

Для навчального розрахунку критичного потоку використовується спрощена модель:

$$T_{\text{мер}} = T_{\text{форм}} + h t_{\text{сер}} + n_{\text{ком}} t_{\text{ком}} + T_{\text{чер}} + T_{\text{пош}} + T_{\text{пр}},$$

де $T_{\text{мер}}$ – розрахункова наскрізна затримка повідомлення; $T_{\text{форм}}$ – час формування повідомлення передавальним вузлом; h – кількість фізичних сегментів зв'язку на маршруті; $n_{\text{ком}}$ – кількість проміжних комутаторів; $t_{\text{ком}}$ – прийнята затримка одного комутатора без урахування часу серіалізації; $T_{\text{чер}}$ – прийнятий максимальний час очікування в черзі; $T_{\text{пош}}$ – сумарний час поширення сигналу по фізичному середовищу; $T_{\text{пр}}$ – час оброблення повідомлення приймальним вузлом.

Для простої послідовності

відправник → комутатор 1 → ... → комутатор nn → приймач

кількість сегментів становить

$$h = n_{\text{ком}} + 1.$$

5.6. Перевірка часової умови

Промислова мережа задовольняє вимогу конкретного інформаційного потоку, якщо

$$T_{\text{мер}} \leq T_{\text{мер,доп}},$$

де $T_{\text{мер,доп}}$ – допустима затримка критичного повідомлення.

Запас часу становить

$$\Delta T_{\text{мер}} = T_{\text{мер,доп}} - T_{\text{мер}},$$

де $\Delta T_{\text{мер}}$ – резерв за допустимою затримкою.

Якщо результат від'ємний, збільшення пропускної здатності може бути лише одним із можливих рішень. Причиною порушення можуть бути також черги, надмірна кількість комутаторів або неправильний розподіл критичного й некритичного трафіку.

5.7. Джитер

Для періодичного технологічного трафіку додатково контролюється зміна часу доставки.

$$J = T_{\text{max}} - T_{\text{min}},$$

де J – джитер; T_{max} , T_{min} – найбільший і найменший виміряні або змодельовані часи доставки повідомлення.

Середня затримка сама по собі недостатня для оцінювання детермінованої мережі. Якщо однакове циклічне повідомлення одного разу

надходить через 0,2 мс, а іншого – через 4 мс, середнє значення приховує істотну нерівномірність доставки.

TSN-механізми розробляються саме для керованого передавання потоків з часовими вимогами та зменшення невизначеності, характерної для звичайного Ethernet при конкуренції трафіку.

6. Методика виконання практичної роботи

Етап 1. Визначення мережевих вузлів

За структурою, розробленою у практичній №1, необхідно виділити:

1. ПЛК;
2. мережеві датчики;
3. інтелектуальні приводи;
4. модулі введення-виведення;
5. промислові комутатори;
6. HMI;
7. SCADA;
8. edge-вузол;
9. сервер;
- 10.цифровий двійник.

Компонент потрібно включати до мережевої схеми лише тоді, коли він реально бере участь в інформаційному обміні.

6.1. Формування таблиці інформаційних потоків

Необхідно скласти таблицю:

№	Джерело	Приймач	Дані	Тип трафіку	Період	Критичність
1	Датчик	ПЛК	Технологічна координата	Циклічний	...	Висока
2	ПЛК	Привод	Завдання	Циклічний	...	Висока
3	Привод	ПЛК	Діагностика	Циклічний / подієвий	...	Середня
4	ПЛК	SCADA	Стан процесу	Циклічний	...	Середня
5	ПЛК	Edge	Дані для аналізу	Циклічний	...	Низька / середня
6	ПЛК	SCADA	Аварія	Подієвий	За подією	Висока

Етап 2. Класифікація трафіку

Для кожного інформаційного потоку встановлюється один із класів:

1. критичний циклічний;
2. подієвий;
3. діагностичний;
4. інформаційний.

Ця класифікація використовується при розробленні логічної структури мережі.

Етап 3. Вибір топології

Необхідно побудувати фізичну схему мережі.

Для типової роботи допускається структура:

польові пристрої → комутатор польового сегмента → ПЛК → комутатор рівня керування → SCADA / Edge / сервер.

Якщо застосовується резервування, альтернативний шлях потрібно показати окремо.

Етап 4. Розподіл протоколів та способів обміну

Для кожного зв'язку необхідно зазначити його функціональне призначення.

Для навчального проєкту можна прийняти таку логіку:

1. локальні критичні дані – промисловий Ethernet;
2. ПЛК ↔ SCADA – OPC UA Client/Server;
3. передавання одного набору даних декільком цифровим споживачам – OPC UA PubSub;
4. трафік, що має детерміновані часові вимоги, – окремий критичний клас із подальшим аналізом доцільності TSN.

Вказати просто «OPC UA» для всіх зв'язків недостатньо. Необхідно зазначити, яку інформаційну задачу він виконує.

Етап 5. Розрахунок мережевого трафіку

Для кожного класу необхідно визначити:

1. кількість вузлів;
2. корисні дані одного повідомлення;
3. прийняті службові дані;
4. період;
5. потік у біт/с.

Після цього визначити сумарний потік.

Етап 6. Розрахунок завантаження каналу

За заданою швидкістю Ethernet розрахувати $K_{\text{мер}}, \%$.

Низьке значення не є самостійною підставою для висновку про детермінованість мережі.

Етап 7. Розрахунок інтенсивності повідомлень

Необхідно визначити v_{Σ} .

Параметр використовується для оцінювання кількості інформаційних операцій за секунду.

Етап 8. Визначення критичного маршруту

З усіх потоків необхідно вибрати той, для якого задано найжорсткішу часову вимогу.

Встановити:

1. кількість сегментів;
2. кількість комутаторів;
3. розмір повідомлення;
4. час формування;
5. затримку комутаторів;
6. максимальне очікування в черзі;
7. час поширення;
8. час приймального оброблення.

Етап 9. Розрахунок наскрізної затримки

Виконати розрахунок $T_{\text{мер}}$.

Отримане значення порівняти з $T_{\text{мер,доп}}$.

Етап 10. Аналіз результату

Якщо затримка не відповідає вимозі, необхідно запропонувати структурну зміну, наприклад:

1. скорочення маршруту;
2. винесення критичного контуру на локальний рівень;
3. розділення технологічного та інформаційного трафіку;
4. встановлення керованого комутатора;
5. використання пріоритезації;
6. застосування TSN-механізмів для потоків, які справді потребують детермінованого обслуговування.

7. Таблиця варіантів

У кожному варіанті задаються дві групи трафіку:

- **група 1** – технологічний циклічний трафік;
- **група 2** – діагностичний / інформаційний трафік.

Вар.	Виробничий об'єкт	n_1	L_1 , байт	T_1 , мс	n_2	L_2 , байт	T_2 , мс	H , байт	C , Мб/с	$n_{\text{зод}}$	$t_{\text{зод}}$, мкс	$T_{\text{зод}}$, мкс	$T_{\text{зод, доп}}$, мс
1	Конвеєрна дільниця	6	24	20	3	48	200	28	100	2	12	60	1,0
2	Пакувальна дільниця	8	32	20	4	64	250	28	100	2	15	70	1,2
3	Дозувальна установка	6	40	50	3	64	500	32	100	2	15	80	2,0
4	Складальна комірка	10	32	10	5	64	200	28	100	3	12	50	0,8
5	Механообробна дільниця	8	48	20	4	96	500	32	100	3	15	70	1,5
6	Насосна установка	6	32	50	4	64	500	28	100	2	15	100	3,0
7	Термічна установка	8	40	100	4	96	1000	32	100	2	15	120	5,0
8	Роботизована комірка	12	48	5	6	96	100	32	1000	3	10	40	0,5
9	Лінія контролю якості	10	64	20	5	128	500	36	1000	3	12	60	1,0
10	Сортувальна дільниця	8	40	10	4	80	200	28	100	2	12	50	0,8

Для розрахунку наскрізної затримки у всіх варіантах додатково приймається:

$$T_{\text{форм}}=20 \text{ мкс}$$

$$T_{\text{пош}}=3 \text{ мкс}$$

$$T_{\text{пр}}=15 \text{ мкс}$$

Ці значення також є навчальними параметрами розрахункової моделі.

8. Приклад розрахунку

8.1. Вихідні дані

Об'єкт: автоматизована дільниця переміщення та позиціонування деталей.

Приймаємо три інформаційні групи.

Група 1 – циклічні технологічні пристрої

- кількість – 8;
- корисні дані – 32 байти;
- службові дані – 28 байт;
- період – 20 мс.

Група 2 – діагностичні пристрої

- кількість – 4;
- корисні дані – 64 байти;
- службові дані – 28 байт;
- період – 200 мс.

Група 3 – стан локального контролера

- кількість – 1;
- корисні дані – 96 байт;
- службові дані – 28 байт;
- період – 100 мс.

Швидкість мережі:

- $C=100$ Мбіт/с

Критичний маршрут містить два комутатори.

Приймаємо:

- $n_{\text{ком}}=2$

- $t_{\text{ком}}=12$ мкс

- $T_{\text{чер}}=50$ мкс

- $T_{\text{мер,доп}}=1$ мс

8.2. Розрахунок циклічного технологічного потоку

Період 20 мс дорівнює 0,02 с.

$$R_{\text{мер,1}} = \frac{8 \cdot 8(32 + 28)}{0,02} = 192000 \text{ біт/с}$$

Отже,

$$R_{\text{мер,1}} = 192 \text{ кбіт/с.}$$

8.3. Розрахунок діагностичного потоку

$$R_{\text{мер,2}} = \frac{8 \cdot 4(64 + 28)}{0,2} = 14720 \text{ біт/с}$$

Або

$$R_{\text{мер,2}} = 14,72 \text{ кбіт/с.}$$

8.4. Потік даних контролера

$$R_{\text{мер,3}} = \frac{8 \cdot 1(96 + 28)}{0,1} = 9920 \text{ біт/с} = 9,92 \text{ кбіт/с}$$

8.5. Сумарний інформаційний потік

$$R_{\text{мер,}\Sigma} = 192000 + 14720 + 9920 = 216640 \text{ біт/с} = 216,64 \text{ кбіт/с}$$

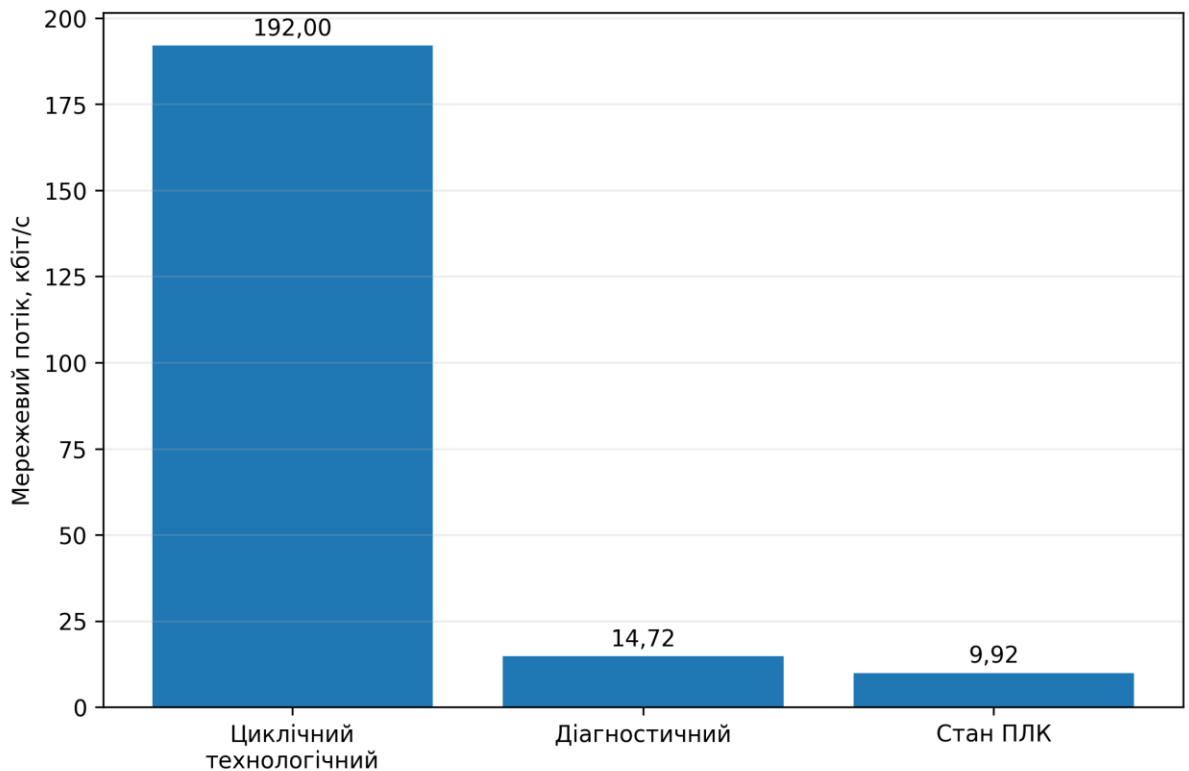


Рисунок 3.4. Розподіл мережевого інформаційного потоку за типами даних для розрахункового прикладу

8.6. Завантаження каналу

$$K_{\text{мер}} = \frac{216640}{100000000} = 0,0021664 = 0,21664\%$$

Середнє розрахункове навантаження від заданих потоків є невеликим відносно номінальної швидкості 100 Мбіт/с.

Цей результат не дає підстав автоматично вважати мережу придатною для критичного контуру. Необхідно окремо перевірити затримку.

8.7. Інтенсивність повідомлень

Для першої групи:

$$\nu_1 = \frac{8}{0,02} = 400 \text{ повідомлень/с}$$

Для другої

$$\nu_2 = \frac{4}{0,2} = 20 \text{ повідомлень/с}$$

Для третьої

$$\nu_3 = \frac{1}{0,1} = 10 \text{ повідомлень/с}$$

Тоді

$$\nu_{\Sigma} = 400 + 20 + 10 = 430 \text{ повідомлень/с}$$

Отже, задана конфігурація формує в середньому 430 повідомлень за секунду.

8.8. Час серіалізації критичного повідомлення

Для технологічного повідомлення першої групи загальний прийнятий розмір становить

$$32 + 28 = 60 \text{ байт}$$

Тоді

$$t_{\text{сер}} = \frac{8 \cdot 60}{100000000} = 4,8 \text{ мкс}$$

8.9. Кількість фізичних сегментів

За наявності двох комутаторів:

$$h = 2 + 1 = 3$$

Маршрут має вигляд:

польовий пристрій → комутатор 1 → комутатор 2 → ПЛК.

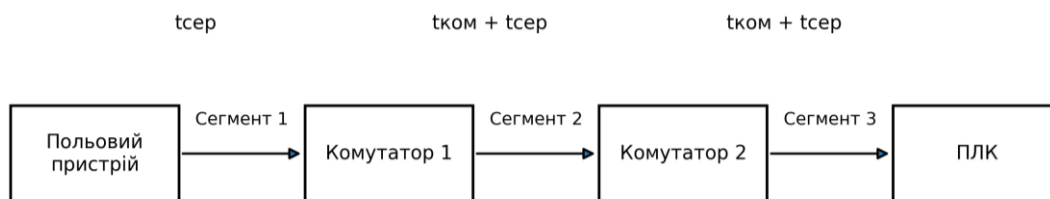


Рисунок 3.5. Критичний маршрут передавання технологічного повідомлення в промисловій мережі

8.10. Розрахунок наскрізної затримки

$$T_{\text{мер}} = 20 + 3 \cdot 4,8 + 2 \cdot 12 + 50 + 3 + 15 = 126,4 \text{ мкс} = 0,1264 \text{ мс}$$

8.11. Перевірка часової вимоги

$$0,1264 < 1 \text{ мс}$$

Отже, за прийнятими навчальними параметрами вимога до максимальної затримки виконується.

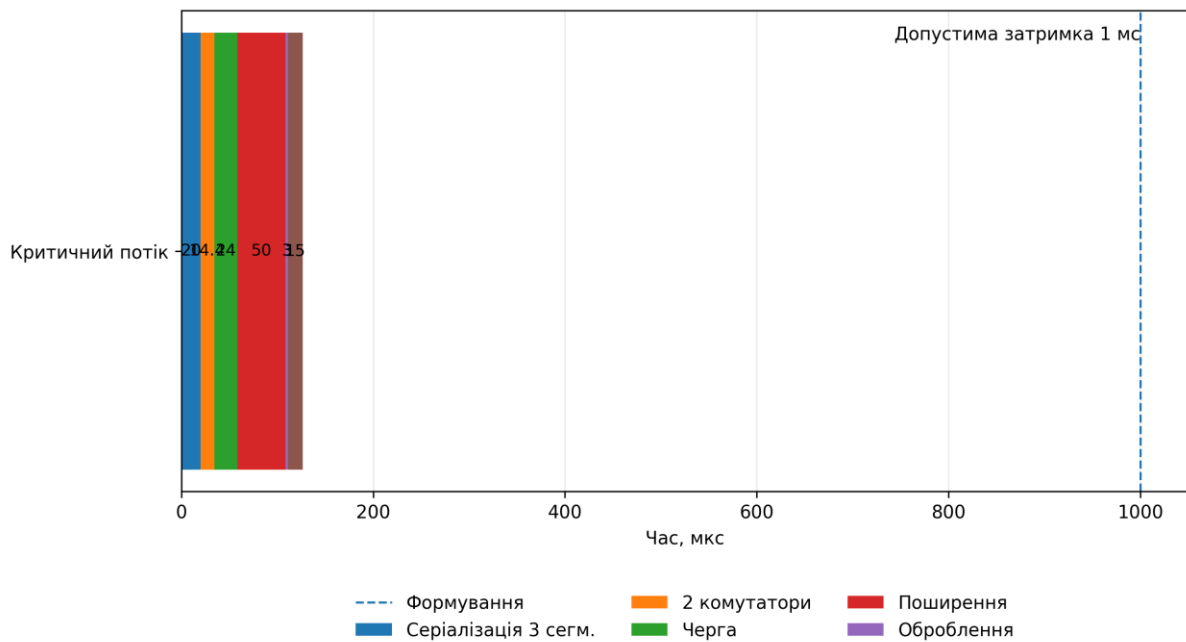


Рисунок 3.6. Складові наскрізної затримки критичного інформаційного потоку

8.12. Розрахунок резерву

$$\Delta T_{\text{мер}} = 1 - 0,1264 = 0,8736 \text{ мс}$$

Отриманий резерв становить 0,8736 мс.

9. Побудова мережі для розрахункового прикладу

Для прикладу пропонується така фізична структура.

Польовий сегмент

До першого керованого комутатора підключаються:

- мережеві датчики;
- приводи;
- віддалені модулі введення-виведення.

Сегмент керування

До другого комутатора підключаються:

- ПЛК;
- HMI;
- edge-вузол.

Інформаційний сегмент

Через відповідний мережевий інтерфейс здійснюється зв'язок із:

- SCADA;
- сервером історичних даних;
- цифровим двійником.

Критичний цикл

датчик → ПЛК → привод

не повинен залежати від доступності сервера історичних даних або цифрового двійника.

Це продовжує принцип, прийнятий у практичних роботах №1 та №2: фізично критичне керування залишається локальним, тоді як інформаційна частина може працювати через вищі рівні КФВС.

10. Організація OPC UA-обміну

Для прикладової системи можна сформувати два різні механізми взаємодії.

OPC UA Client/Server

ПЛК / OPC UA Server → SCADA / OPC UA Client

Передаються:

- поточні технологічні координати;
- стан установки;
- значення завдань;
- режими;
- діагностичні ознаки.

SCADA отримує структурований доступ до змінних і може виконувати передбачені конфігурацією операції.

OPC UA використовується у виробничих CPS як засіб вертикального та горизонтального інформаційного обміну; практичні дослідження також показують, що перехід до OPC UA впливає не тільки на протокол, а й на структуру програмних компонентів системи.

OPC UA PubSub

ПЛК / Publisher → SCADA + Edge + цифровий двійник / Subscribers

Публікуються:

1. поточний режим;
2. технологічні координати;
3. керувальний сигнал;
4. стан блокувань;
5. діагностичні дані.

PubSub у цьому випадку усуває необхідність трактувати кожного цифрового споживача як незалежний цикл адресного опитування одного джерела.

11. Розподіл трафіку за пріоритетом

Для прикладу пропонується така функціональна класифікація:

Потік	Приклад	Пріоритет
Керувальний	ПЛК → привод	Найвищий
Циклічний технологічний	Датчик → ПЛК	Високий
Аварійний подієвий	Привод → ПЛК / HMI	Високий
Оперативний	ПЛК → SCADA	Середній
Діагностичний	Пристрій → сервер	Нижчий
Архівний	Edge → історична база	Нижчий

Ця таблиця визначає функціональну вимогу, а не конкретне числове значення Ethernet PCP або іншого поля пріоритету. Конкретне відображення класів у мережеві пріоритети залежить від обраної технології та конфігурації обладнання.

12. Аналіз доцільності TSN

TSN не слід вводити до проєкту лише тому, що система є кіберфізичною.

Необхідність TSN має бути пов'язана з конкретними вимогами:

1. обмеженою максимальною затримкою;
2. малим джитером;
3. співіснуванням критичного і некритичного трафіку в одній Ethernet-інфраструктурі;
4. необхідністю передбачуваного обслуговування часових потоків;
5. вимогами до резервування або синхронізації.

Якщо локальний технологічний контур без проблем працює у відокремленому сегменті промислового Ethernet, а інформаційний трафік не впливає на його часові характеристики, додавання складної TSN-конфігурації потребує окремого обґрунтування.

Дослідження сучасного TSN також показують, що конфігурація планування трафіку є окремою складною задачею і не існує одного методу розкладу, оптимального для всіх мережевих сценаріїв.

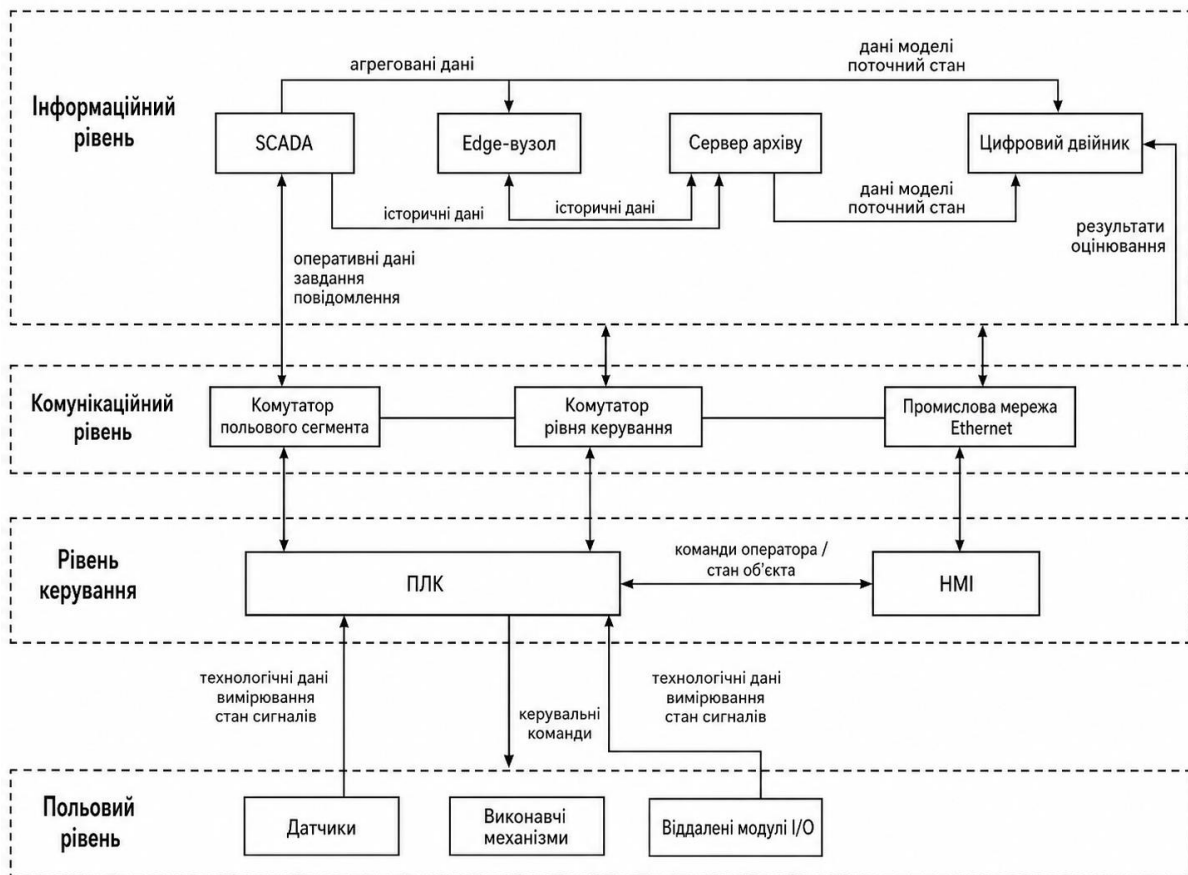


Рисунок 3.7 Структурна схема інформаційного обміну КФВС

13. Завдання до практичної роботи

За номером свого варіанта необхідно:

1. визначити вузли промислової мережі;
2. побудувати фізичну топологію;
3. сформувати таблицю інформаційних потоків;
4. поділити трафік на функціональні класи;
5. визначити критичний інформаційний потік;
6. визначити зв'язки, для яких доцільний OPC UA Client/Server;
7. визначити зв'язки, для яких доцільний OPC UA PubSub;
8. розрахувати потік першої групи;
9. розрахувати потік другої групи;
10. визначити сумарний мережевий потік;
11. розрахувати завантаження каналу;
12. визначити інтенсивність повідомлень;
13. визначити час серіалізації критичного повідомлення;
14. встановити критичний маршрут;
15. визначити кількість сегментів;
16. розрахувати наскрізну затримку;
17. порівняти затримку з допустимою;

18. визначити часовий резерв;
19. розподілити інформаційні потоки за пріоритетами;
20. обґрунтувати необхідність або відсутність необхідності застосування TSN;
21. побудувати структурну схему інформаційного обміну КФВС;
22. зробити висновки.

14. Вимоги до оформлення звіту

Звіт повинен містити:

1. тему та мету практичної роботи;
2. номер варіанта;
3. короткий опис технологічного об'єкта;
4. перелік мережевих вузлів;
5. фізичну топологію мережі;
6. таблицю інформаційних потоків;
7. класифікацію трафіку;
8. обґрунтування вибраних способів обміну;
9. схему OPC UA Client/Server;
10. схему OPC UA PubSub, якщо вона використовується;
11. розрахунок кожної групи мережевого трафіку;
12. розрахунок сумарного потоку;
13. розрахунок завантаження мережі;
14. розрахунок інтенсивності повідомлень;
15. критичний маршрут;
16. розрахунок часу серіалізації;
17. розрахунок наскрізної затримки;
18. перевірку часової умови;
19. розрахунок резерву;
20. таблицю пріоритетів інформаційних потоків;
21. висновок щодо доцільності TSN;
22. загальну схему інформаційного обміну КФВС;
23. висновки.

15. Висновки

У практичній роботі промислова мережа розглядається як частина загального кіберфізичного контуру, а не як пасивний канал між обладнанням. Вибір структури мережі пов'язується з функціями технологічного об'єкта, архітектурою локального керування та характером інформації, яка передається між фізичним і цифровим рівнями.

Розрахунок мережевого потоку показує, яку частину номінальної пропускної здатності займають задані технологічні та діагностичні повідомлення. Водночас невелике середнє завантаження каналу не гарантує

виконання часових вимог. Для критичного потоку окремо враховуються час серіалізації, кількість комутаторів, очікування в черзі, оброблення та інші складові наскрізної затримки.

OPC UA Client/Server і PubSub виконують різні функції. Client/Server доцільний для адресного доступу до структурованих даних, тоді як PubSub дає змогу організувати передавання одного інформаційного набору кільком споживачам.

TSN, своєю чергою, не замінює OPC UA і не повинен додаватися до будь-якої КФВС формально. Його застосування потребує конкретної часової причини: необхідності контрольованої затримки, обмеження джитера або сумісного передавання критичних та некритичних потоків через спільну Ethernet-інфраструктуру.

Результатом роботи має бути мережа, для якої можна простежити кожний суттєвий інформаційний потік від джерела до споживача, визначити його обсяг, період, маршрут та часові обмеження. Такий опис утворює основу для наступної практичної роботи, де отримані від фізичного об'єкта дані використовуватимуться для побудови й оновлення цифрового двійника.

ПРАКТИЧНА РОБОТА № 4. ПОБУДОВА ЦИФРОВОГО ДВІЙНИКА ТА ОЦІНЮВАННЯ СТАНУ КІБЕРФІЗИЧНОГО ВИРОБНИЧОГО ОБ'ЄКТА

1. Мета практичного заняття

Мета роботи – набути навичок побудови спрощеного цифрового двійника виробничого об'єкта, організації його синхронізації з фізичною системою та оцінювання технічного стану обладнання за результатами зіставлення вимірюваних і модельних параметрів.

У процесі виконання роботи необхідно навчитися:

- визначати фізичний об'єкт, для якого створюється цифровий двійник;
- встановлювати склад даних, необхідних для його функціонування;
- відокремлювати математичну модель від цифрового двійника як інформаційної системи;
- визначати параметри стану фізичного об'єкта;
- формувати цифрове представлення технологічного обладнання;
- організовувати оновлення цифрової моделі за вимірюваними даними;
- розраховувати прогнозоване значення технологічної координати;

- визначати розбіжність між фізичним об'єктом і його цифровою моделлю;
- оцінювати точність цифрового двійника;
- нормувати діагностичні параметри;
- формувати інтегральну оцінку стану обладнання;
- розрізняти штатний, попереджувальний і критичний стани;
- визначати інформацію, яка повинна передаватися до SCADA, системи діагностування або цифрового сервісу;
- встановлювати зв'язок цифрового двійника з архітектурою, алгоритмами керування та мережею, розробленими у практичних роботах №1–3.

2. Прилади і матеріали

Для виконання практичної роботи використовуються:

1. персональний комп'ютер;
2. табличний процесор або середовище математичних розрахунків;
3. програмне середовище для побудови структурних схем;
4. результати практичних робіт №1–3;
5. вихідні дані відповідно до номера варіанта;
6. структурна схема виробничого об'єкта;
7. дані датчиків технологічних та діагностичних параметрів;
8. параметри математичної моделі об'єкта;
9. матеріали лекційного курсу;
10. стандарти серії ISO 23247.

3. Теоретичні положення

3.1. Цифровий двійник виробничого об'єкта

Цифровий двійник виробничого об'єкта являє собою цифрове представлення фізичного виробничого елемента, стан якого пов'язаний із даними реального об'єкта. Для виробничих систем принципи такого представлення формалізовані серією ISO 23247. ISO 23247-1:2021 визначає загальні принципи цифрового двійника у виробництві, ISO 23247-2:2021 – референтну архітектуру, ISO 23247-3:2021 – інформаційні атрибути цифрового представлення виробничих елементів.

Цифровий двійник не слід ототожнювати з одним рівнянням, тривимірною моделлю обладнання або набором даних із датчиків. Для функціонального цифрового двійника необхідний зв'язок кількох складових:

фізичний об'єкт → вимірювання → передавання даних → цифрове представлення → математична або інформаційна модель → оцінювання стану → цифровий сервіс.

Залежно від завдання результатом роботи цифрового двійника може бути відображення поточного стану, моделювання поведінки, виявлення розбіжностей, діагностування, прогнозування або підтримка прийняття рішення. Дослідження виробничих цифрових двійників показують, що одна з принципових проблем їх реалізації полягає саме в узгодженні моделей, реальних даних, комунікаційної інфраструктури та обчислювальних сервісів.

3.2. Фізичний об'єкт і його цифрове представлення

У практичній роботі фізичний об'єкт характеризується трьома групами параметрів:

1. Керувальні параметри

Визначають вплив системи керування на технологічний процес.

Наприклад:

- завдання частоти обертання;
- положення клапана;
- частота перетворювача;
- потужність нагрівача;
- завдання швидкості привода.

2. Технологічні координати

Характеризують поточну поведінку об'єкта:

- частота обертання;
- швидкість;
- температура процесу;
- витрата;
- тиск;
- положення механізму.

3. Діагностичні параметри

Використовуються для оцінювання стану:

1. вібрація;
2. температура підшипникового вузла;
3. струм електродвигуна;
4. споживана потужність;
5. час виконання операції;
6. інші параметри, які мають фізичний зв'язок зі станом обладнання.

Цифрове представлення повинно містити тільки ті параметри, які потрібні для конкретного завдання. Намагання включити до двійника всі доступні дані збільшує обчислювальні витрати та складність системи, але саме по собі не підвищує точність оцінювання стану. У систематичному

огляді цифрових двійників для прогнозного обслуговування складність моделей і різноманітність даних також визначаються як суттєві практичні проблеми.

3.3. Математична модель цифрового двійника

Математична модель описує одну або декілька властивостей фізичного об'єкта.

У загальному вигляді модель динамічного об'єкта можна представити рівнянням стану

$$\hat{\mathbf{x}}_{k+1} = f(\hat{\mathbf{x}}_k, \mathbf{u}_k),$$

де $\hat{\mathbf{x}}_k$ – оцінений вектор стану цифрової моделі; f – функція переходу між станами; $\mathbf{u}_k$ – вектор керувальних впливів.

Розрахунковий вихід моделі визначається як

$$\hat{\mathbf{y}}_k = h(\hat{\mathbf{x}}_k),$$

де $\hat{\mathbf{y}}_k$ – вектор модельних вихідних координат; h – функція, яка пов'язує внутрішній стан моделі з її виходами.

У реальній виробничій задачі функції f та h можуть бути сформовані на основі фізичних законів, експериментальної ідентифікації, машинного навчання або поєднання цих підходів.

Дослідження гібридних моделей показують, що фізичні та data-driven підходи можуть доповнювати один одного: фізична модель зберігає інтерпретованість, тоді як модель на основі даних може компенсувати процеси, які складно описати аналітично.

3.4. Фізичні, data-driven та гібридні моделі

Фізична модель

Будується на рівняннях, що описують механічні, електричні, теплові, гідравлічні або інші процеси.

Її перевага полягає у фізичній інтерпретованості параметрів. Недолік виникає, якщо реальний об'єкт має невідомі нелінійності, змінні характеристики або процеси, які неможливо достатньо точно представити спрощеною моделлю.

Модель на основі даних

Встановлює залежність за експериментальними вимірюваннями.

Для неї не завжди потрібний повний аналітичний опис фізики об'єкта, але результат істотно залежить від повноти та якості навчальних даних.

Гібридна модель

Поєднує фізичну структуру та параметри, отримані з даних.

У цій практичній роботі використовується спрощена фізично інтерпретована модель першого порядку, оскільки метою є вивчення

принципу побудови цифрового двійника, а не ідентифікація конкретного промислового агрегату.

3.5. Модель об'єкта першого порядку

Для розрахунку динаміку технологічної координати приймаємо у вигляді

$$\tau \frac{d\hat{y}}{dt} + \hat{y} = Ku,$$

де $\hat{y}$ – модельне значення технологічної координати; τ – стала часу моделі; K – коефіцієнт передачі моделі.

Після дискретизації методом Ейлера отримуємо

$$\hat{y}_{k+1} = \hat{y}_k + \frac{T_s}{\tau} (Ku_k - \hat{y}_k).$$

Рівняння показує, як цифрова модель прогнозує наступне значення координати за своїм попереднім станом та поточним керувальним впливом.

Для реального об'єкта перед використанням такої моделі потрібно експериментально підтвердити, що модель першого порядку достатньо точно описує його динаміку.

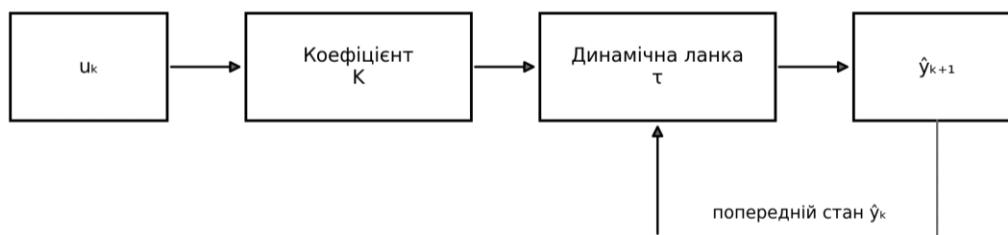


Рисунок 4.1. Структурне представлення дискретної моделі об'єкта першого порядку

3.6. Синхронізація цифрового двійника

Математична модель може працювати без фізичного об'єкта. У такому випадку вона залишається моделлю або симуляцією.

У цифровому двійнику її стан коригується або перевіряється за реальними даними.

Послідовність синхронізації має вигляд:

фізичний об'єкт → датчик → промислова мережа → актуалізація цифрового стану → модельний розрахунок → порівняння з фізичним об'єктом.

Затримки мережі, розраховані у практичній роботі №3, безпосередньо впливають на актуальність даних цифрового двійника.

Вік отриманого вимірювання визначається як

$$T_{\text{вік},k} = t_{\text{пот},k} - t_{\text{м},k},$$

де $T_{\text{вік},k}$ – вік даних на момент їх використання; $t_{\text{пот},k}$ – поточний час; $t_{\text{м},k}$ – часова мітка вимірювання.

Для придатних даних повинна виконуватися задана проектом умова

$$T_{\text{вік},k} \leq T_{\text{вік,доп}},$$

де $T_{\text{вік,доп}}$ – максимально допустимий вік даних для заданої функції.

Одне універсальне значення $T_{\text{вік,доп}}$ встановити неможливо. Воно залежить від динаміки конкретного фізичного процесу та функції, для якої використовуються дані.



Рисунок 4.2. Цикл синхронізації фізичного об'єкта та його цифрового двійника

3.7. Оцінювання відповідності цифрової моделі фізичному об'єкту

3.7.1. Залишкова похибка моделі

Розбіжність між реальним і модельним значеннями визначається як

$$\varepsilon_k = y_k - \hat{y}_k,$$

де ε_k – залишкова похибка цифрової моделі.

Якщо ε_k близька до нуля, модель та фізичний об'єкт у цій точці узгоджуються.

Стійке збільшення залишкової похибки може мати різні причини:

1. зміна параметрів фізичного об'єкта;

2. неточна модель;
3. похибка датчика;
4. зміна режиму;
5. некоректна синхронізація;
6. пошкодження обладнання.

Тому одна велика похибка не повинна автоматично трактуватися як несправність.

3.7.2. Абсолютна похибка

$$E_{a,k} = |\varepsilon_k|,$$

де $E_{a,k}$ – абсолютна похибка прогнозування.

3.7.3. Середньоквадратична похибка

Для послідовності з N спостережень відповідність моделі оцінюється за RMSE:

$$E_{\text{RMSE}} = \sqrt{\frac{1}{N} \sum_{k=1}^N \varepsilon_k^2},$$

де E_{RMSE} – середньоквадратична похибка моделі; N – кількість порівнюваних спостережень.

Цей показник реагує сильніше на великі розбіжності, оскільки похибка підноситься до квадрата.

Для практичної роботи допустиме значення $E_{\text{RMSE,доп}}$ задається у вихідних даних.

Критерій відповідності моделі має вигляд

$$E_{\text{RMSE}} \leq E_{\text{RMSE,доп}}.$$

3.8. Оцінювання технічного стану

Цифровий двійник може використовувати не лише регульовану координату, а й параметри, які змінюються при погіршенні стану обладнання.

У цифрових двійниках для condition monitoring і predictive maintenance дані фізичного об'єкта поєднуються з моделями для виявлення, локалізації або прогнозування змін стану.

У роботі використовується спрощена багатопараметрична оцінка.

3.8.1. Нормована зміна діагностичного параметра

Для кожного діагностичного параметра визначається безрозмірне відхилення

$$z_i = \frac{|p_i - p_{i,0}|}{\Delta p_i},$$

де z_i – нормоване відхилення i -го параметра; p_i – його поточне значення; $p_{i,0}$ – опорне значення для прийнятого справного стану; Δp_i – масштаб відхилення, заданий для оцінювання.

Параметр Δp_i не є нормативною аварійною межею. Це масштаб нормування, який задається у варіанті.

При $z_i = 0$ параметр збігається з опорним значенням.

При $z_i = 1$ відхилення дорівнює прийнятому масштабу Δp_i .

3.8.2. Інтегральний діагностичний показник

Для одночасного врахування декількох параметрів використовується зважена евклідова норма:

$$D = \sqrt{\sum_{i=1}^{m_d} w_i z_i^2},$$

де D – інтегральний діагностичний показник; m_d – кількість діагностичних параметрів; w_i – ваговий коефіцієнт параметра.

Вагові коефіцієнти нормуються:

$$\sum_{i=1}^{m_d} w_i = 1.$$

У практичній роботі використовуються два параметри:

p_1 – середньоквадратичне значення віброшвидкості;

p_2 – температура контрольованого вузла.

Для розрахунку приймається:

$$w_1 = 0,6$$

$$w_2 = 0,4$$

Вибір ваг у реальному діагностичному комплексі потребує обґрунтування за даними надійності, діагностичної чутливості або експертного аналізу. Значення 0,6 та 0,4 у цій роботі є умовою завдання.

3.8.3. Класифікація стану

Для практичної роботи задаються дві межі:

$$D_1 = 1,0$$

$$D_2 = 1,5$$

Вони є навчальними порогами, а не нормативними границями вібрації чи температури.

Стан класифікується так:

штатний стан

$$D < D_1$$

стан, що потребує уваги

$$D_1 \leq D < D_2.$$

критичне відхилення за прийнятою навчальною моделлю

$$D \geq D_2.$$

Такий індекс не встановлює вид несправності. Він тільки показує ступінь сукупного відхилення вибраних параметрів від їхніх опорних значень.

Для реального діагностування необхідні нормативні межі для конкретного типу обладнання, історичні дані та підтвердження діагностичної значущості кожного показника.

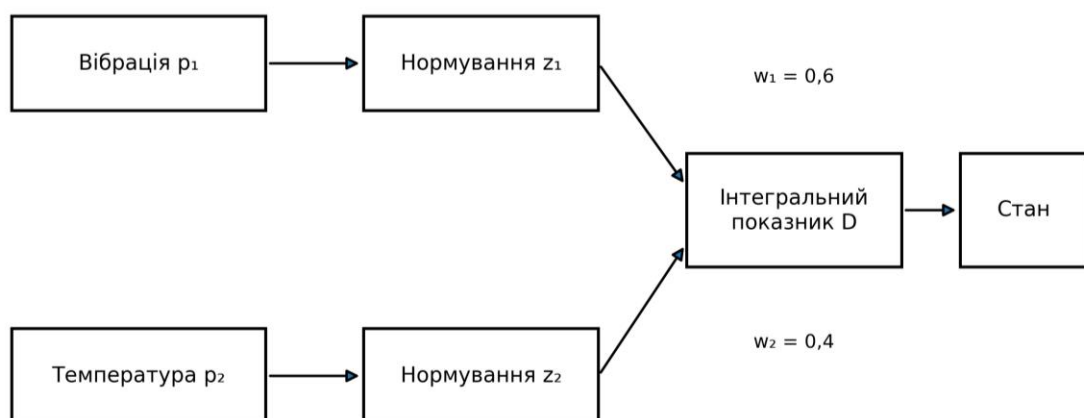


Рисунок 4.3. Структурна схема інтегрального оцінювання технічного стану виробничого об'єкта

3.9. Структура цифрового двійника в практичній роботі

Для виконання роботи цифровий двійник поділяється на шість функціональних компонентів.

1. Фізичний об'єкт

Містить технологічне обладнання та виконавчий механізм.

2. Вимірювальна підсистема

Отримує:

1. технологічну координату;
2. вібрацію;
3. температуру;
4. стан привода;
5. інші сигнали за потреби.

3. Комунікаційна підсистема

Передає дані від фізичного об'єкта до цифрового представлення. Її структура формується за результатами практичної роботи №3.

4. Динамічна модель

Розраховує прогнозовану поведінку фізичного об'єкта.

5. Підсистема оцінювання стану

Визначає:

1. залишкову похибку;
2. RMSE;
3. нормовані діагностичні параметри;
4. інтегральний показник DD.

6. Інформаційний сервіс

Передає результат до:

1. HMI;
2. SCADA;
3. системи архівування;
4. діагностичного сервісу;
5. системи планування технічного обслуговування.

Таким чином, цифровий двійник у роботі має не тільки модельний, але й інформаційний контур.

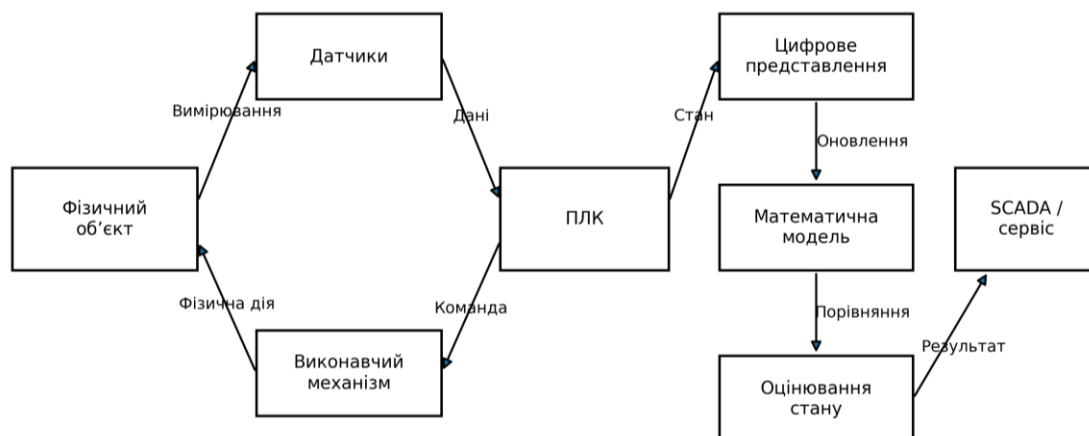


Рисунок 4.4 Структурна схема цифрового двійника кіберфізичного виробничого об'єкта

4. Методика виконання практичної роботи

Етап 1. Визначення об'єкта цифрового двійника

За своїм варіантом визначити:

1. фізичний об'єкт;
2. керувальний вплив;
3. технологічну координату;
4. основний виконавчий механізм;
5. контрольовані діагностичні параметри.

Етап 2. Визначення інформаційних джерел

Скласти таблицю:

№	Параметр	Джерело	Тип даних	Використання у цифровому двійнику
1	Керувальний сигнал	ПЛК	Цифрові дані	Вхід математичної моделі
2	Технологічна координата	Датчик	Вимірювання	Перевірка моделі
3	Вібрація	Датчик	Діагностичні дані	Оцінювання стану
4	Температура	Датчик	Діагностичні дані	Оцінювання стану
5	Стан привода	Привод	Дискретні / мережеві	Контекст режиму

Етап 3. Побудова структурної схеми цифрового двійника

Необхідно відобразити:

**фізичний об'єкт → датчики → мережа → цифрове представлення
→ модель → оцінювання стану → SCADA / діагностичний сервіс.**

Зворотний інформаційний або керувальний канал показується тільки тоді, коли він реально передбачений структурою.

Етап 4. Перевірка актуальності даних

Для отриманого повідомлення визначити $T_{\text{вік},k}$.

Порівняти його з допустимим значенням.

Якщо дані застаріли, вони не повинні використовуватися для оперативної корекції цифрового стану.

Етап 5. Розрахунок наступного стану моделі

За параметрами варіанта визначити

$$\hat{y}_{k+1}$$

за дискретною моделлю першого порядку.

Етап 6. Порівняння цифрового та фізичного об'єктів

За вимірним значенням визначити

$$\varepsilon_{k+1} = y_{k+1} - \hat{y}_{k+1}$$

Після цього розрахувати абсолютну похибку.

Етап 7. Перевірка точності моделі

Для послідовності вимірювань визначити E_{RMSE} .

Порівняти його із заданим $E_{\text{RMSE,доп}}$

Якщо умова не виконується, цифрову модель не можна вважати достатньо точною для заданої функції без повторної ідентифікації або зміни її структури.

Етап 8. Оцінювання діагностичних параметрів

За заданими значеннями розрахувати:

$$z_1 \text{ та } z_2$$

Етап 9. Розрахунок інтегрального показника

Визначити D .

Етап 10. Класифікація стану

За заданими навчальними межами визначити один із станів:

1. штатний;
2. такий, що потребує уваги;
3. критичне відхилення.

Етап 11. Формування цифрового запису стану

Цифровий двійник повинен сформувати набір:

Поле	Зміст
Ідентифікатор об'єкта	Обладнання, до якого належить запис
Часова мітка	Час отримання даних
Режим	Поточний режим роботи
uku_k	Керувальна дія
yku_k	Реальна технологічна координата
$\hat{y}_k$	Модельна координата
$\epsilon_k \backslash \text{varepsilon}_k$	Залишкова похибка
p1p_1	Діагностичний параметр 1
p2p_2	Діагностичний параметр 2
DD	Інтегральний показник
Стан	Результат класифікації

5. Таблиця варіантів

Технологічна координата та керувальний сигнал задані у відсотках робочого діапазону.

Запис для діагностичного параметра має форму:

поточне значення / опорне значення / масштаб нормування.

Вар.	Виробничий об'єкт	$\hat{y}_k, \%$	$u_k, \%$	K	$\tau, \text{с}$	$T_s, \text{с}$	$y_{k+1}, \%$	$p_1/p_{1,0}/\Delta p_1, \text{мм/с}$	$p_2/p_{2,0}/\Delta p_2, ^\circ\text{C}$	$E_{\text{RMSE,опт}}, \%$
1	Привод конвеєра	58	70	1,0	4,0	0,5	60,0	3,8 / 3,0 / 2,0	64 / 58 / 15	2,0
2	Насосний агрегат	62	75	1,0	5,0	0,5	63,5	4,4 / 3,2 / 2,0	68 / 60 / 15	2,0
3	Вентиляційний агрегат	55	68	1,0	3,5	0,5	57,1	4,8 / 3,0 / 2,0	72 / 60 / 15	2,5
4	Привод змішувача	64	72	1,0	4,5	0,5	64,9	5,0 / 3,0 / 2,0	75 / 60 / 15	2,0
5	Шпіндель верстата	70	80	1,0	3,0	0,2	70,9	5,4 / 3,2 / 2,0	76 / 62 / 15	1,5
6	Привод пакувального модуля	52	65	1,0	3,5	0,5	54,0	3,6 / 2,8 / 2,0	63 / 57 / 15	2,5
7	Привод дозатора	60	70	1,0	4,0	0,5	61,0	4,2 / 3,0 / 2,0	69 / 59 / 15	2,0
8	Привод роботизованої комірки	68	78	1,0	2,5	0,2	69,2	5,6 / 3,2 / 2,0	78 / 62 / 15	1,5
9	Електропривод сушильної установки	57	70	1,0	5,0	0,5	58,5	4,6 / 3,0 / 2,0	73 / 60 / 15	2,5
10	Привод сортувальної дільниці	63	74	1,0	3,5	0,5	64,0	5,2 / 3,0 / 2,0	77 / 60 / 15	2,0

Для всіх варіантів приймається:

$$w_1=0,6$$

$$w_2=0,4$$

$$D_1=1,0$$

$$D_2=1,5$$

5. Приклад розрахунку

5.1. Об'єкт

Електропривод витяжного вентилятора технологічної установки.

Приймаємо:

1. попереднє модельне значення – 60 %;
2. керувальний сигнал – 70 %;
3. $K=1,0$;
4. $\tau=4 \text{ с}$;
5. $T_s=0,5 \text{ с}$;
6. фактично виміряне наступне значення – 62 %.

Для оцінювання стану:

1. віброшвидкість
2. поточне значення – 5,2 мм/с;
3. опорне значення – 3,0 мм/с;
4. масштаб нормування – 2,0 мм/с;
5. температура вузла
6. поточне значення – 78 °C;

7. опорне значення – 60 °С;
8. масштаб нормування – 15 °С.

Допустима RMSE для прикладу:

$$E_{\text{RMSE, доп}} = 2\%$$

5.2. Розрахунок модельного значення

$$\hat{y}_{k+1} = 60 + \frac{0,5}{4}(1 \cdot 70 - 60) = 60 + 0,125 \cdot 10 = 61,25\%$$

Цифрова модель прогнозує наступне значення технологічної координати 61,25 %.

5.3. Розрахунок залишкової похибки

Фізичний об'єкт має значення 62 %.

$$\varepsilon_{k+1} = 62 - 61,25 = 0,75\%.$$

5.4. Абсолютна похибка

$$E_{a,k+1} = |0,75| = 0,75\%.$$

5.5. Оцінювання RMSE на послідовності даних

Для перевірки моделі приймаємо п'ять навчальних пар «вимірювання – результат цифрової моделі».

k	$y_k, \%$	$\hat{y}_k, \%$	$\varepsilon_k, \%$
1	60,0	59,2	0,8
2	62,0	61,25	0,75
3	63,5	62,7	0,8
4	65,0	64,1	0,9
5	66,2	65,4	0,8

Тоді

$$E_{\text{RMSE}} = \sqrt{\frac{0,8^2 + 0,75^2 + 0,8^2 + 0,9^2 + 0,8^2}{5}} =$$

$$= \sqrt{\frac{3,2925}{5}} = \sqrt{0,6585} = 0,8127\%$$

Порівняємо з прийнятою межею:

$$0,8127 < 2$$

Отже, для заданої навчальної вибірки модель задовольняє прийнятну умову точності.

Цей висновок стосується тільки наведених вихідних даних. П'яти точок недостатньо для підтвердження точності реальної промислової моделі у всьому діапазоні режимів.

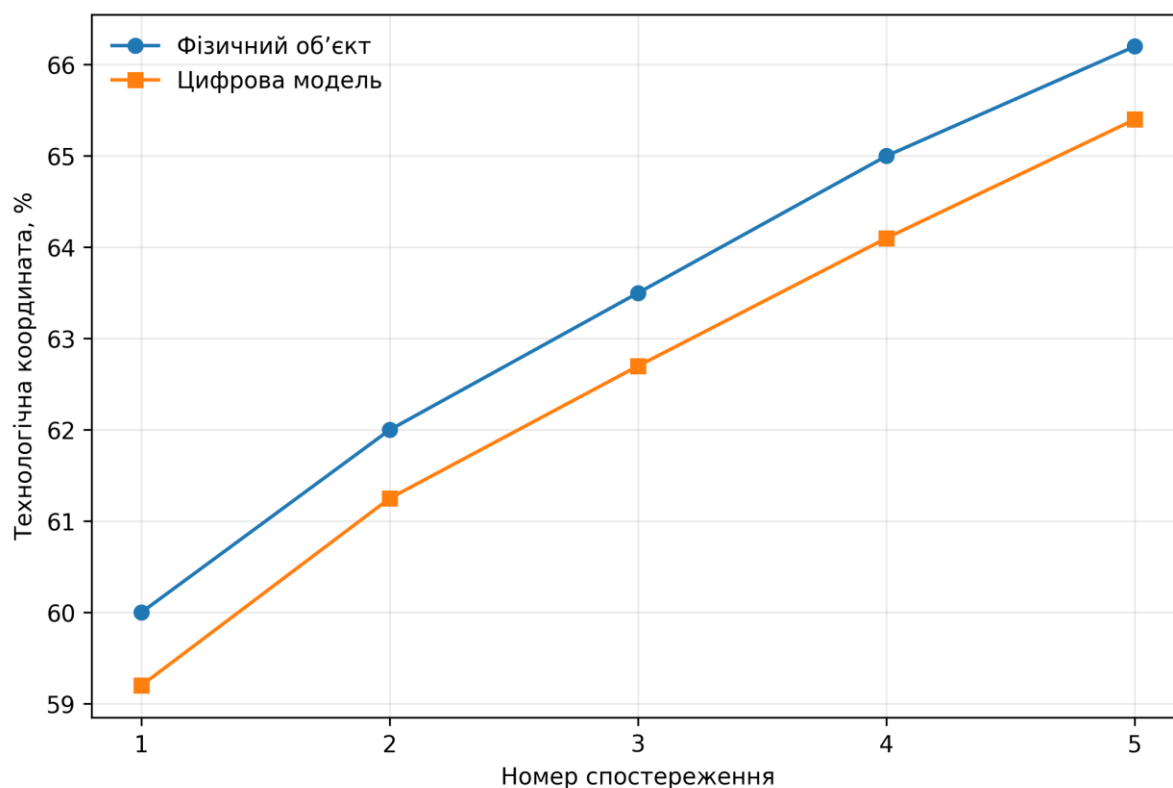


Рисунок 4.5. Порівняння технологічної координати фізичного об'єкта та цифрової моделі

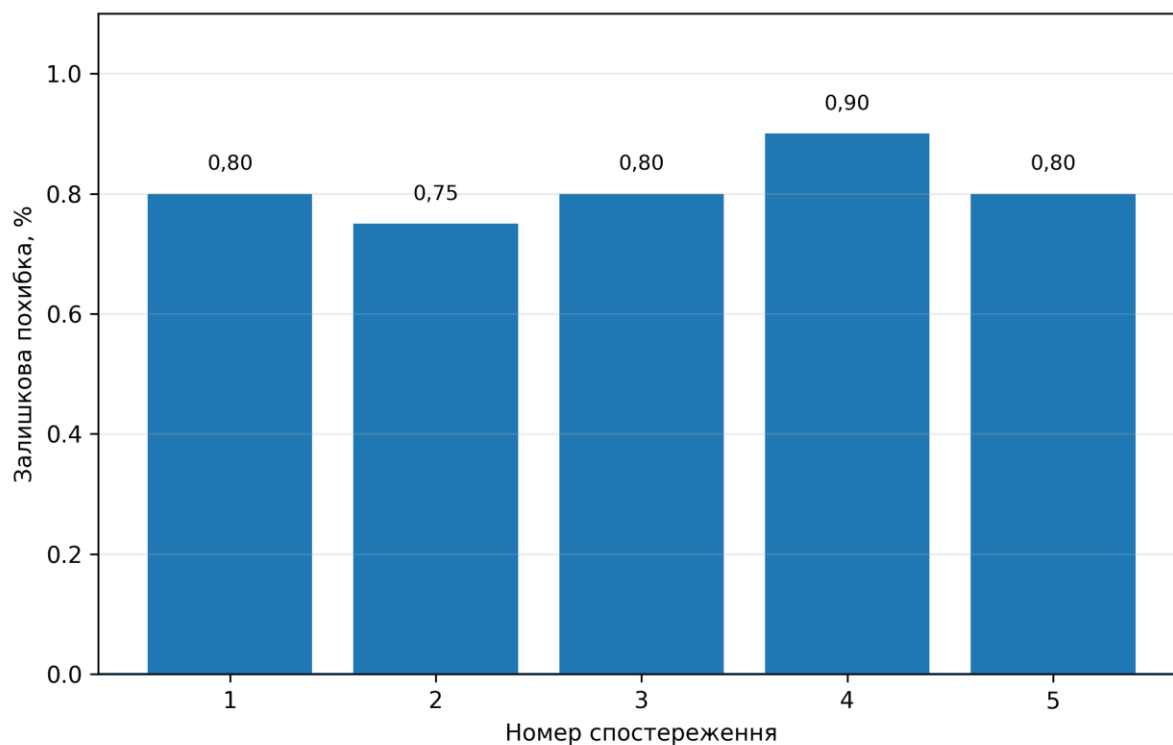


Рисунок 4.6. Залишкова похибка цифрової моделі для послідовності спостережень

5.6. Оцінювання стану за вібрацією

$$z_1 = \frac{|5,2 - 3,0|}{2,0} = 1,1.$$

Отже, відхилення вібраційного показника становить 1,1 прийнятого масштабу нормування.

5.7. Оцінювання стану за температурою

$$z_2 = \frac{|78 - 60|}{15} = 1,2.$$

5.8. Інтегральне оцінювання стану

$$D = \sqrt{0,6 \cdot 1,1^2 + 0,4 \cdot 1,2^2} = 1,141 < 1,5.$$

Порівнюємо:

$$1,0 \leq 1,141 < 1,5.$$

За прийнятою шкалою об'єкт належить до стану, що потребує уваги.

Цей результат не означає автоматичного встановлення несправності вентилятора. Він показує, що два вибрані діагностичні параметри разом відхилилися від опорного стану настільки, що виконано задану умову попередження.

Для визначення конкретної причини необхідний окремий діагностичний алгоритм.

5.9. Інтерпретація модельної та діагностичної інформації

Для прикладового варіанта отримано два різні результати.

Результат 1 – цифрова модель технологічної поведінки відповідає фізичному об'єкту.

$$E_{\text{RMSE}} \approx 0,813\% < 2\%$$

Результат 2 – діагностичний стан потребує уваги.

$$D = 1,141.$$

Ця ситуація цілком можлива.

Технологічна координата ще може добре відтворюватися моделлю, тоді як вібрація і температура вже зміщуються від опорного стану. Саме тому цифровий двійник доцільно будувати не на одному параметрі, а на сукупності технологічних та діагностичних даних.

Сучасні роботи з цифровими двійниками для condition monitoring також комбінують декілька джерел даних і моделей, оскільки одна технологічна координата не дає повного опису стану обладнання [32–35].

5.10. Алгоритм функціонування цифрового двійника

Для практичної роботи алгоритм повинен мати таку послідовність.

Крок 1. Отримання даних

З ПЛК та польових пристроїв надходять:

1. u_k ;
2. y_k ;
3. діагностичні параметри;
4. стан обладнання;
5. часова мітка.

Крок 2. Перевірка актуальності

Розраховується $T_{\text{вік},k}$.

Якщо отримане значення перевищує допустиме, дані позначаються як неактуальні.

Крок 3. Оновлення цифрового стану

Актуальні вимірювання записуються до поточного цифрового представлення.

Крок 4. Модельний розрахунок

Визначається прогнозоване значення $\hat{y}_{k+1}$.

Крок 5. Надходження нового фізичного вимірювання

Після наступного циклу отримується y_{k+1} .

Крок 6. Розрахунок залишку

Визначається ε_{k+1} .

Крок 7. Накопичення історичних даних

Зберігаються пари
вимірювання – модельне значення.

Крок 8. Оцінювання точності

Розраховується E_{RMSE} .

Крок 9. Оброблення діагностичних параметрів

Визначаються z_1 та z_2 .

Крок 10. Інтегральне оцінювання

Розраховується D .

Крок 11. Класифікація стану

Встановлюється цифрова ознака:

1. штатний;
2. потребує уваги;
3. критичне відхилення.

Крок 12. Передавання результату

Дані передаються до:

HMI / SCADA → архів → сервіс діагностування.

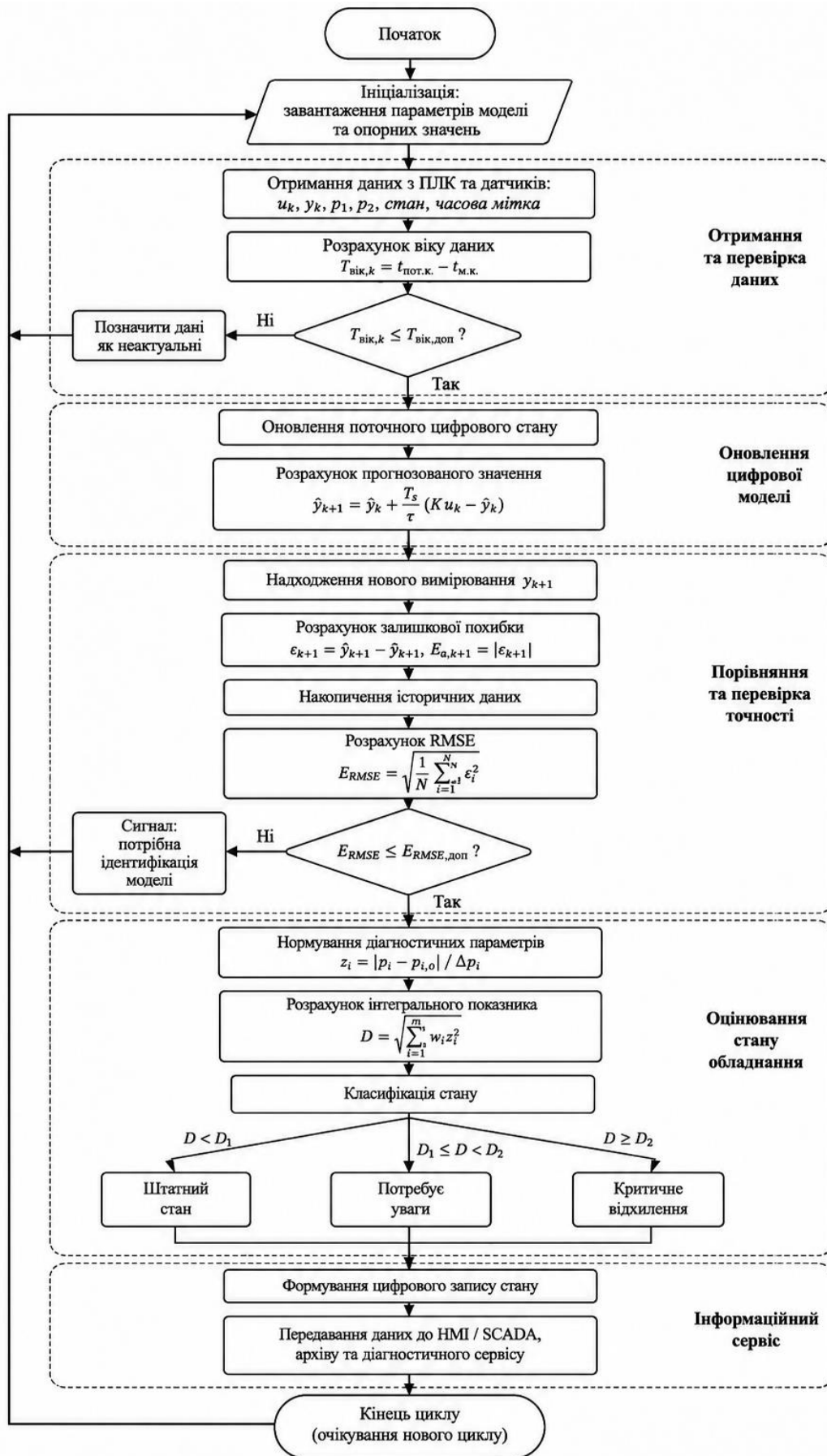


Рисунок 4.7 Алгоритм функціонування цифрового двійника

6. Зв'язок цифрового двійника з попередніми практичними роботами

Практичні роботи №1–4 утворюють одну послідовність проєктування.

Практична робота №1

Визначено: які компоненти входять до КФВС.

Практична робота №2

Визначено: як фізичний об'єкт керується.

Практична робота №3

Визначено: як передаються дані між компонентами.

Практична робота №4

Визначається: як за отриманими даними підтримується цифрове представлення фізичного об'єкта та оцінюється його стан.

Повний інформаційно-керувальний цикл можна подати як

фізичний процес → вимірювання → ПЛК → промислова мережа → цифровий двійник → оцінювання → інформаційний сервіс.

У разі використання результату цифрового аналізу для зміни технологічного режиму додається зворотний шлях:

цифровий сервіс → перевірене рішення → локальний алгоритм керування → виконавчий пристрій → фізичний процес.

Результат аналітичного сервісу не повинен безумовно підмінювати блокування та захисну логіку локального рівня.

7. Завдання до практичної роботи

За своїм номером варіанта необхідно:

1. описати фізичний об'єкт;
2. визначити його керувальну координату;
3. визначити технологічну координату;
4. визначити два діагностичні параметри;
5. скласти таблицю інформаційних джерел;
6. побудувати структурну схему цифрового двійника;
7. показати зв'язок із ПЛК та промисловою мережею;
8. визначити дані, які отримуються з практичної роботи №3;
9. описати принцип синхронізації фізичного і цифрового об'єктів;
10. розрахувати наступне модельне значення;
11. визначити залишкову похибку;
12. визначити абсолютну похибку;
13. сформулювати не менше п'яти пар «фізичне – модельне значення»;

14. розрахувати RMSE;
15. порівняти RMSE із заданим допустимим значенням;
16. розрахувати нормоване відхилення першого діагностичного параметра;
17. розрахувати нормоване відхилення другого параметра;
18. розрахувати інтегральний показник;
19. визначити стан об'єкта;
20. сформувати структуру цифрового запису;
21. розробити алгоритм оновлення цифрового двійника;
22. визначити дані, які передаються до SCADA;
23. визначити дані, що повинні архівуватися;
24. пояснити, як цифровий двійник пов'язаний із результатами практичних робіт №1–3;
25. сформулювати висновки.

8. Вимоги до оформлення звіту

Звіт з практичної роботи повинен містити:

1. тему роботи;
2. мету;
3. номер варіанта;
4. опис фізичного виробничого об'єкта;
5. перелік керувальних, технологічних та діагностичних параметрів;
6. структурну схему цифрового двійника;
7. таблицю джерел даних;
8. опис математичної моделі;
9. розрахунок модельного значення;
10. визначення залишкової похибки;
11. таблицю фізичних і модельних значень;
12. розрахунок RMSE;
13. перевірку точності моделі;
14. розрахунок $z1z_1$;
15. розрахунок $z2z_2$;
16. розрахунок DD;
17. класифікацію стану;
18. структуру цифрового запису стану;
19. алгоритм синхронізації двійника;
20. схему інформаційного обміну;
21. пояснення взаємозв'язку з КФВС;

9. Висновки

У практичній роботі цифровий двійник розглядається як інформаційно-модельна складова кіберфізичної виробничої системи. Його робота базується на зв'язку фізичного обладнання, вимірювальних каналів, промислової мережі, математичної моделі та засобів оброблення даних. Така структура відповідає підходу стандартів серії ISO 23247 до цифрового представлення виробничих елементів.

Математична модель використовується для прогнозування поведінки технологічної координати. Після надходження реального вимірювання результат моделі порівнюється з фізичним об'єктом. Залишкова та середньоквадратична похибки характеризують відповідність цифрового представлення реальному процесу. Якщо похибка систематично перевищує задану межу, необхідно перевірити параметри моделі, режим роботи, достовірність вимірювань і синхронізацію даних, а не автоматично трактувати розбіжність як несправність обладнання.

Оцінювання стану доповнює перевірку математичної моделі. Нормування діагностичних параметрів дає змогу привести фізичні величини різної природи до безрозмірного вигляду, після чого сформувати інтегральний показник. У цій роботі пороги та вагові коефіцієнти використовуються як навчальні параметри; застосовувати їх як нормативні критерії реального обладнання не можна.

Практична робота №4 завершує послідовне формування навчальної КФВС. У першій роботі визначається архітектура, у другій – алгоритм локального керування, у третій – мережевий інформаційний обмін, у четвертій – цифрове представлення та оцінювання стану. У результаті фізичний і цифровий контури розглядаються не ізольовано, а як взаємопов'язані складові однієї виробничої системи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

Міжнародні стандарти

1. ISO/IEC TR 63306-1:2020. Smart manufacturing standards map (SM2) – Part 1: Framework. Geneva : International Organization for Standardization, 2020.
2. IEC 63339:2024. Unified reference model for smart manufacturing. Geneva : International Electrotechnical Commission, 2024.
3. ISO/IEC 30141:2024. Internet of Things (IoT) – Reference architecture. Geneva : International Organization for Standardization, 2024.
4. ISO/IEC 20924:2024. Internet of Things (IoT) and digital twin – Vocabulary. Geneva : International Organization for Standardization, 2024.
5. ISO 23247-1:2021. Automation systems and integration – Digital twin framework for manufacturing – Part 1: Overview and general principles. Geneva : International Organization for Standardization, 2021.
6. ISO 23247-2:2021. Automation systems and integration – Digital twin framework for manufacturing – Part 2: Reference architecture. Geneva : International Organization for Standardization, 2021.
7. ISO 23247-3:2021. Automation systems and integration – Digital twin framework for manufacturing – Part 3: Digital representation of manufacturing elements. Geneva : International Organization for Standardization, 2021.
8. ISO 23247-4:2021. Automation systems and integration – Digital twin framework for manufacturing – Part 4: Information exchange. Geneva : International Organization for Standardization, 2021.
9. IEC 61131-3:2025. Programmable controllers – Part 3: Programming languages. 4th ed. Geneva : International Electrotechnical Commission, 2025.
10. IEC 62541-1:2025. OPC unified architecture – Part 1: Overview and concepts. Geneva : International Electrotechnical Commission, 2025.
11. IEC 62541-4:2025. OPC unified architecture – Part 4: Services. 4th ed. Geneva : International Electrotechnical Commission, 2025.
12. IEC 62541-14:2026. OPC unified architecture – Part 14: PubSub. 2nd ed. Geneva : International Electrotechnical Commission, 2026.
13. IEEE Std 802.1Q-2022. IEEE Standard for Local and Metropolitan Area Networks – Bridges and Bridged Networks. New York : IEEE, 2022. DOI: 10.1109/IEEESTD.2022.10004498.

14. IEC 62443-3-2:2020. Security for industrial automation and control systems – Part 3-2: Security risk assessment for system design. Geneva : International Electrotechnical Commission, 2020.

15. IEC 62443-2-1:2024. Security for industrial automation and control systems – Part 2-1: Security program requirements for IACS asset owners. 2nd ed. Geneva : International Electrotechnical Commission, 2024.

Наукові джерела

16. Головатенко І. А., Писаренко А. В. Методи моделювання кібер-фізичних систем. Адаптивні системи автоматичного управління. 2021. № 2 (39). С. 74–83. DOI: 10.20535/1560-8956.39.2021.247413.

17. Дубовой В. М., Юхимчук М. С. Дослідження стійкості та збіжності децентралізованої координації локальних систем управління розподіленими кібер-фізичними системами. Вісник Вінницького політехнічного інституту. 2021. № 4 (157). С. 62–69. DOI: 10.31649/1997-9266-2021-157-4-62-69.

18. Крижановський В. Г., Комаров В. Ф., Сергієнко С. П., Загоруйко Л. В. Використання дисперсії Аллана для ідентифікації нормальної роботи сенсорних вузлів. Вісник Вінницького політехнічного інституту. 2021. № 3 (156). С. 78–83. DOI: 10.31649/1997-9266-2021-156-3-78-83.

19. Ковалевський С. В., Ковалевська О. С., Сидюк Д. М. Інноваційні рішення для адаптації технологічних систем до сучасних вимог. Вісник Вінницького політехнічного інституту. 2023. № 5 (170). С. 85–92. DOI: 10.31649/1997-9266-2023-170-5-85-92.

20. Биков М. М., Грищук Т. В., Ковалюк О. О., Ковтун В. В., Юхимчук М. С. Модель експлуатації кіберфізичної системи в умовах впливу негативних зовнішніх факторів. Вісник Вінницького політехнічного інституту. 2023. № 6 (171). С. 30–38. DOI: 10.31649/1997-9266-2023-171-6-30-38.

Електронне навчальне видання комбінованого використання
Можна використовувати в локальному мережному режимі

Мезеря Андрій Юрійович

ПРОЕКТУВАННЯ ТА ТЕХНОЛОГІЇ КІБЕРФІЗИЧНИХ ВИРОБНИЧИХ СИСТЕМ

Методичні вказівки до проведення практичних занять
для здобувачів вищої освіти другого (магістерського) рівня
денної та заочної форм здобуття освіти за спеціальністю
G7 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка»

В авторській редакції

Підписано до розміщення 23.10.2025. Гарнітура Times New Roman.
Ум. друк. арк. 3,61. Обсяг 3,010 Мб. Зам. № 500/25.

Харківський національний університет імені В. Н. Каразіна,
61022, м. Харків, майдан Свободи, 4.
Свідоцтво суб'єкта видавничої справи ДК № 3367 від 13.01.2009
Видавництво ХНУ імені В. Н. Каразіна