

Міністерство освіти і науки України
Криворізький національний університет
Факультет інформаційних технологій
Кафедра автоматизації, комп'ютерних наук і технологій

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти – магістр
за освітньо-професійною програмою
«Кіберфізичні системи в промисловості, бізнесі та транспорті»

зі спеціальності

*174 – Автоматизація, комп'ютерно-інтегровані технології та
робототехніка*

тема роботи:

***«Адаптивне керування процесом нагрівом матеріалу на рухомому
конвеєрі з використанням технологій IoT»***

Виконав студент гр. АКІТР-23-1м _____ Колесник В. І.

Керівник _____ Рубан С. А.

Нормоконтроль _____ Маринич І. А.

В.о. завідувача кафедри _____ Рубан С. А.

КРИВОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет: інформаційних технологій

Кафедра: автоматизації, комп'ютерних наук і технологій

Ступінь вищої освіти: Бакалавр

Спеціальність: 174 - Автоматизація та комп'ютерно-інтегровані технології
та робототехніка

ЗАТВЕРДЖУЮ

В.о.зав. кафедрою: к.т.н. Рубан С.А.

« 5 » липня 2024 р.

ЗАВДАННЯ

на кваліфікаційну роботу бакалавра

студентові групи АКІТР-23-1м Колеснику Віталію Ігоровичу

1. Тема кваліфікаційної роботи: «Адаптивне керування процесом нагрівом матеріалу на рухомому конвеєрі з використанням технологій IoT»

затверджено наказом по університету № 595с від 04.07.2024 р.

2. Термін здачі кваліфікаційної роботи: 01.12.2024 р.

3. Склад кваліфікаційної роботи: Пояснювальна записка обсягом 72с., додатки, презентація у Microsoft PowerPoint (15 слайдів) в електронному та друкованому вигляді

4. Консультанти кваліфікаційної роботи:

Розділ 1-3

доц. Рубан С. А.

Нормоконтроль

доц. Маринич І. А.

5. Календарний план:

№	Етапи роботи	Термін виконання
1	<i>Вступ</i>	<i>10.07.24</i>
2	<i>Розділ 1</i>	<i>15.07.24</i>
3	<i>Розділ 2</i>	<i>18.08.24</i>
4	<i>Розділ 3</i>	<i>19.09.24</i>
5	<i>Висновки</i>	<i>15.10.24</i>
6	<i>Оформлення кваліфікаційної роботи</i>	<i>20.11.24</i>
7	<i>Підготовка презентації та графічного матеріалу</i>	<i>28.11.24</i>
8	<i>Підготовка доповіді до захисту</i>	<i>01.12.24</i>

6. Дата видачі завдання: 28.06.2024р.

Керівник _____ / Рубан С. А./

7. Запевнення: Я, Колесник Віталій Ігорович, запевняю, що ця кваліфікаційна робота виконана самостійно, не містить академічного плагіату, фабрикації, фальсифікації. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Із чинним Положенням про академічну доброчесність Криворізького національного університету ознайомлений.

Чітко усвідомлюю, що в разі виявлення у кваліфікаційній роботі умисних порушень робота не допускається до захисту або оцінюється незадовільно.

Здобувач _____ / Колесник В. І./

АНОТАЦІЯ

Колесник В. І. «Адаптивне керування процесом нагрівом матеріалу на рухомому конвеєрі з використанням технологій IoT»

Кваліфікаційна робота на здобуття ступеню вищої освіти магістр за освітньо-професійною програмою «Кіберфізичні системи в промисловості, бізнесі та транспорті» зі спеціальності 174 – Автоматизація, комп'ютерно – інтегровані технології та робототехніка– Криворізький національний університет, Кривий Ріг, 2024.

Об'єктом дослідження є система адаптивного керування температурними процесами, яка використовує сучасні технології, зокрема Інтернет речей (IoT), нечітку логіку та прогнозуюче керування.

У першому розділі проаналізовано традиційні підходи до керування температурою, зокрема використання ПД-регуляторів, та сучасні методи, які забезпечують підвищену ефективність і точність керування в умовах змінного середовища. Обґрунтовано переваги використання IoT і нечіткої логіки в адаптивних системах.

У другому розділі побудовано математичні моделі ПД-регулятора, розроблено алгоритм для визначення температури та реалізовано модель теплопередачі на основі рівняння теплопровідності Фур'є із застосуванням методу скінченних різниць. Алгоритм реалізовано мовою Python для інтеграції у системи автоматизованого контролю.

У третьому розділі описано апаратну складову системи, зокрема використання ПЛК S7-1200 і модуля Simatic IOT2040. Налаштовано базу даних PostgreSQL для зберігання фізичних властивостей матеріалів і забезпечено зв'язок між компонентами системи через Ignition Gateway і OPC.

Ключові слова:

АДАПТИВНЕ КЕРУВАННЯ, IoT, , ТЕПЛОПРОВІДНІСТЬ, МЕТОД СКІНЧЕННИХ РІЗНИЦЬ, АВТОМАТИЗАЦІЯ.

ANNOTATION

Kolesnyk V. I. «Adaptive Control of the Material Heating Process on a Moving Conveyor Using IoT Technologies».

Graduation master`s work for obtaining an educational degree «Master» for the educational and professional program « Cyber-physical systems in industry, business and transport » in specialty 174 – «Automation, computer-integrated technologies, and robotics». – Kryvyi Rih National University, Kryvyi Rih, 2024.

The object of the study is an adaptive control system for thermal processes that utilizes modern technologies, including the Internet of Things (IoT), fuzzy logic, and predictive control.

In the first chapter, traditional approaches to temperature control, such as PID controllers, and modern methods that provide increased efficiency and accuracy in variable environments are analyzed. The advantages of using IoT and fuzzy logic in adaptive control systems are substantiated.

In the second chapter, mathematical models of the PID controller are developed, an algorithm for determining temperature is designed, and a heat transfer model based on Fourier's heat conduction equation is implemented using the finite difference method. The algorithm is implemented in Python for integration into automated control systems.

In the third chapter, the hardware components of the system are described, including the use of the S7-1200 PLC and Simatic IOT2040 module. A PostgreSQL database is configured for storing physical material properties, and communication between system components through Ignition Gateway and OPC is established. A user-friendly interface for equipment control is implemented.

Keywords: ADAPTIVE CONTROL, IoT, HEAT CONDUCTION, FINITE DIFFERENCE METHOD, AUTOMATION.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 АНАЛІЗ ПІДХОДІВ ДО ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ПРОЦЕСУ	9
1.1 Традиційні методи для керування температурними процесами.....	9
1.2 Керування на основі нечіткої логіки.....	10
1.3 Модель прогнозуючого керування.....	11
1.4 Керування температурними процесами на основі нечіткої логіки.....	14
1.5 Перспекиви використання технологій IoT у системах адаптивного керування.	15
<i>Висновки за розділом.....</i>	19
РОЗДІЛ 2 ІДЕНТИФІКАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ ТА СИНТЕЗ КЕРУВАННЯ ПРОЦЕСОМ	21
2.1 Система керування процесом нагріву.....	21
2.1.1 Функціональна схема системи.....	21
2.1.2 Математична модель ПД-регулятора.....	22
2.2 Процес нагріву матеріалу.....	22
2.2.1 Математична модель нагріву матеріалу.....	23
2.2.2 Метод кінцевих різниць.....	24
2.2.3 Алгоритм вирішення.....	27
2.2.4 Реалізація алгоритму на Python.....	29
<i>Висновки за розділом.....</i>	30
РОЗДІЛ 3 ПРОГРАМНО-ТЕХНІЧНА РЕАЛІЗАЦІЯ СИСТЕМИ КЕРУВАННЯ ПРОЦЕСОМ.....	31
3.1 Загальний опис обладнання лабораторного стенду.....	31
3.1.1 Контролер Siemens серії S7-1200.....	32
3.1.2 Модуль SIMATIC IOT2040	32
3.1.3 Виконавчі механізми.....	33
3.1.4 Датчики для контролю технологічних параметрів.....	34
3.2 Системна архітектура	34
3.3 Програмна реалізація.....	36
3.3.1 Конфігурація програмного забезпечення модуля Simatic IOT2040	36

3.3.2	Встановлення зв'язку між Ignition Gateway та базою даних PostgreSQL	44
3.3.3	Встановлення зв'язку між ПЛК S7-1200 та Ignition Gateway з використанням технології OPC UA	49
3.3.4	Програма керування ПД-регулятором.....	54
3.3.1	Розробка панелі оператора з використанням Ignition Perspective	59
	<i>Висновки за розділом.....</i>	<i>67</i>
	ВИСНОВОК.....	68
	СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	70

ВСТУП

Адаптивне керування процесом нагріву матеріалу на рухомому конвеєрі з використанням технологій IoT є важливим напрямом розвитку сучасної промисловості, зокрема в контексті четвертої промислової революції (Industry 4.0). Основні завдання сучасної індустрії полягають у забезпеченні ефективності, економічності та екологічної безпеки виробничих процесів. Проблема контролю температури в промислових печах є критичною, оскільки правильна термічна обробка матеріалів впливає на їх якість, знижує енергоспоживання та мінімізує вплив на довкілля.

Традиційні методи керування мають обмеження, пов'язані зі змінами зовнішніх умов, зміною параметрів матеріалу або коливаннями швидкості конвеєра. У таких ситуаціях адаптивне керування на основі технологій IoT дозволяє реагувати на зміни в реальному часі, оптимізуючи процес і знижуючи людський фактор.

Практичне значення полягає в можливості впровадження інноваційних систем контролю та керування у виробничих умовах, що дозволяє суттєво підвищити ефективність роботи обладнання. В умовах підвищених вимог до якості продукції та скорочення витрат на енергоресурси, застосування IoT технологій дозволить автоматизувати процеси прийняття рішень, контролювати параметри роботи в режимі реального часу та інтегрувати дані з інших частин виробничого процесу для побудови оптимальних сценаріїв керування.

Об'єктом дослідження є процес термічної обробки матеріалів на рухомому конвеєрі у промисловій печі. Цей процес вимагає постійного контролю температури, часу перебування матеріалів у зоні нагріву, швидкості конвеєра та інших важливих параметрів, які впливають на якість кінцевого продукту.

Предметом дослідження є система адаптивного керування процесом нагріву матеріалів на основі технологій IoT. Основна увага магістерської роботи зосереджена на розробці методів і алгоритмів для автоматичного контролю та

оптимізації процесу нагріву. Предмет дослідження охоплює використання сенсорних мереж, аналіз даних у реальному часі та прийняття рішень щодо коригування параметрів процесу для забезпечення оптимальних умов термічної обробки.

Практичне значення результатів дослідження полягає в розробці методики адаптивного керування, яка може бути застосована у виробничих умовах для покращення якості матеріалів, що підлягають термічній обробці. Система адаптивного керування, запропонована у магістерській роботі, дає можливість:

- Зменшити енергетичні витрати завдяки точнішому контролю процесу нагріву, зокрема через мінімізацію перевитрати енергії у випадках, коли температура перевищує необхідну.

- Підвищити якість продукції за рахунок стабілізації температурного режиму та уникнення дефектів, пов'язаних із недостатнім або надмірним нагрівом матеріалу.

- Скоротити час на переналаштування обладнання при зміні типу матеріалів або режиму роботи, завдяки автоматичному збору й аналізу даних через IoT-сенсори.

- Покращити моніторинг і діагностику виробничого процесу через доступ до даних в режимі реального часу, що дозволяє оперативно реагувати на відхилення у роботі обладнання.

- Впровадження подібних систем може бути цікавим для підприємств важкої промисловості, металургійної галузі, де контроль нагріву матеріалів є критично важливим етапом виробничого циклу.

РОЗДІЛ 1

АНАЛІЗ ПІДХОДІВ ДО ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ПРОЦЕСУ

1.1 Традиційні методи для керування температурними процесами.

Традиційні підходи в керуванні системами, такі як пропорційно-інтегрально-диференційне (PID) регулювання, залишаються широко вживаними завдяки своїй простоті та надійності. Ці методи ґрунтуються на жорстко заданих налаштуваннях, які визначаються заздалегідь і працюють добре в стабільних умовах, коли характеристики системи не зазнають значних змін. Наприклад, PID-регулятори використовуються для керування температурою, тиском, швидкістю та іншими фізичними величинами в системах, де динаміка процесу відома або змінюється незначно. Основною перевагою традиційних підходів є їх передбачуваність і відносна простота у налаштуванні та реалізації. Однак вони можуть втрачати ефективність при роботі зі складними або нелінійними системами, де параметри змінюються з часом, що вимагає додаткових коригувань або переходу на більш гнучкі методи, такі як адаптивне керування [1].

У статті [1] описується система керування температурою електричного котла в реальному часі на базі STM32, що використовує PID-алгоритм. Модуль WiFi ESP8266 забезпечує з'єднання між нижніми та вищими рівнями ПК для збору даних віддалено, відображення в реальному часі та сигналізації про перевищення порогових значень. Система є стабільною, надійною та зручною у використанні, відповідаючи вимогам термостатичного контролю електричних котлів. У статті також розглядається управління температурою електричних котлів в режимі реального часу за допомогою PID-алгоритмів для електричних печей, а також наводяться приклади інших проектів систем контролю температури на основі PID.

У роботі J. H. Chang [2] та його команди основна увага приділяється дослідженню та оптимізації роботи печі з роликовим конвеєром за допомогою

числового моделювання. Для розрахунків енергоспоживання та продуктивності в моделі контролю температури вчені використовували PID-регулятор, щоб керувати потужністю нагрівача і підтримувати необхідний температурний діапазон. У порівнянні з іншими простими методами регулювання температури, PID-контроль дозволяє досягти більш точних результатів. Важливими параметрами були температура вставки, значення теплового потоку та керування перемиканням.

Основною темою цієї роботи [3] є новий підхід до системи керування температурою в печі опору. У дослідженні представлено нечіткий PID-регулятор, який базується на алгоритмі оптимізації рою частинок (PSO) і розробляє математичну модель печі опору на основі експериментальних даних. Для підвищення ефективності системи управління температурою цей підхід поєднує алгоритм PSO з нечітким PID-контролером. Також описується встановлення нечіткого PID-алгоритму на базі PSO з використанням контролера PIC16f та розробка плати для керування температурою. Для перевірки ефективності двох запропонованих контролерів для системи гартування в печі опору було створено модель системи в середовищі Matlab.

На завершення, PID-алгоритми керування є одним із найбільш широко використовуваних методів для регулювання температури в електричних печах, і вони неодноразово оптимізувалися та вдосконалювалися для досягнення високої точності й стабільності управління. Крім того, в цій галузі застосовуються більш передові методи керування, які в певних умовах можуть забезпечити ще вищу точність і стабільність контролю.

1.2 Керування на основі нечіткої логіки

Коли мова йде про методи контролю температури в електричних печах та їх застосування, нечітка логіка є ключовим аспектом. Цей метод керування, що ґрунтується на штучному інтелекті, краще підходить для нелінійних і змінних у часі систем, оскільки може адаптуватися в реальному часі до поточного стану

системи. Щоб підвищити точність і стабільність контролю температури в електричних печах, нечітка логіка часто використовується в поєднанні з PID-регуляторами.

У недавньому дослідженні V. Bharath Kumar та ін. представлено стратегію управління температурою на основі нечіткого PID-регулятора для промислових нагрівальних печей [4]. Автори пропонують систему контролю температури, яка використовує нечіткий PID-регулятор для управління температурою печі. Цей метод обробляє вхідні та вихідні сигнали за допомогою нечіткої логіки, після чого розраховує вихідні сигнали відповідно до заданого набору правил. Конкретно, автори використовують Нечіткий Логічний Контролер (FLC) для перетворення всіх вхідних і вихідних значень у діапазон від 0 до 1, а потім застосовують Структуру Нечіткого Висновку (FIS) для відображення трансформованих вхідних та вихідних даних, що можна зрозуміти в контексті наведеної ілюстрації. Після налаштування нечіткого PID-регулятора в реальному часі на основі даних про температуру, вихідний сигнал може бути розрахований для управління температурою печі відповідно до заздалегідь встановлених правил.

1.3 Модель прогнозуючого керування

Управління температурою в електричних печах є важливою складовою промислового виробництва, що сприяє підвищенню якості продукції та зниженню витрат. Протягом останніх кількох десятиліть PID-контрольні алгоритми залишалися одним із найпопулярніших методів регулювання температури електричних печей. Однак наявність мертвих зон у традиційних PID-алгоритмах ускладнює точне та стабільне регулювання температури. З розвитком комп'ютерного моделювання та математичних методів у останні роки все більше науковців почали досліджувати підходи, засновані на предиктивному моделюванні, щоб вирішити цю проблему.

У статті "Математична модель для прогнозування механічних властивостей і структури готової деталі та модель для керування електричною системою нагріву на основі PID-алгоритмів" представлено огляд літератури щодо методів контролю температури для електричних нагрівальних печей, а також приклади їх застосування [5]. У цьому дослідженні запропоновано і реалізовано стратегію предиктивного контролю для процесу термообробки сталі на основі математичних моделей та методів скінченних елементів. Основними компонентами технології є використання математичних моделей для прогнозування механічних характеристик і структурних властивостей матеріалу після термообробки, а також застосування принципів PID-контролю для регулювання температури в печі з метою досягнення оптимальних результатів термообробки.

У цій літературі використовується PID-алгоритм контролю, який поєднує алгоритм одноетапного прогнозування, що краще справляється з мертвим часом системи, ніж традиційні PID-алгоритми, що призводить до більш точної та стабільної регуляції температури. У статті також застосовується STEP ARX модель, метод скінченних елементів, для аналізу температури та теплового потоку компонента в залежності від часу, що дозволяє досягти кращого контролю температури. Ця технологія значно покращує механічні характеристики сталі та структурну однорідність, знижує витрати, пов'язані з експериментами, і підвищує ефективність виробництва. Крім того, цей метод може бути використаний для нагріву різноманітного електричного обладнання, включаючи промислові печі опору та установки індукційного нагріву.

У статті "Model-based vs data-driven adaptive control: An overview" [6] автори проводять порівняння двох основних напрямків адаптивного керування: моделюючого та на основі даних. Визначаючи ці два підполя, вони пропонують систематизацію існуючих монографій, оглядових статей та останніх досліджень, що свідчить про зростаючий інтерес до адаптивних контролерів. Використовуючи прості приклади, стаття ілюструє основні концепції кожного з підходів, що дозволяє читачеві краще зрозуміти їхні переваги та обмеження.

У статті "Adaptive Control Based Electric Furnace Temperature Control Using Simulink" [7] автори досліджують підвищення ефективності процесу термічної обробки в склоплавильній печі за допомогою адаптивного керування. Вони зазначають, що традиційні методи, такі як пропорційно-інтегрально-диференціальне (PID) або нечітке керування, неефективні при зміні умов навколишнього середовища. Натомість запропоновано використання адаптивного керування на основі методу Model Reference Adaptive Control (MRAC) для покращення динамічної відповіді системи. В статті також представлено реалізацію методу подачі прямого сигналу з використанням MIT-правила, що дозволяє ефективно справлятися зі змінами динаміки процесу та підвищувати ефективність роботи печі. Модель керування була реалізована та симульована за допомогою MATLAB/SIMULINK, що підтверджує ефективність підходу.

На завершення, предиктивне моделювання є багатообіцяючим підходом до управління температурою електричних печей. Поєднуючи математичне моделювання та комп'ютерне симулювання, можна досягти більш точної, надійної та ефективної регуляції температури, що, в свою чергу, підвищить економічні та соціальні переваги промислового виробництва.

У статті "Application of Adaptive Controller Neural Network Based on RBF NN for Temperature Control Electrical Resistance Furnace" [8] представлено адаптивний підхід до керування температурою електричної опору печі, який базується на нейронній мережі з радіально-базисними функціями (RBF NN). Через високу інерційність, нелінійність, велику затримку і змінні параметри печі, традиційні PI-контролери не забезпечують достатньої ефективності керування. Запропонований підхід використовує нейронну мережу для оцінки невідомих функцій і адаптивного закону на основі розв'язку рівняння Ляпунова, що дозволяє системі підлаштовуватися під зміни параметрів печі в реальному часі. Результати показали, що адаптивне керування на основі нейронної мережі значно покращує роботу замкненої системи, дозволяючи моделі коригуватися

онлайн і стабілізувати свої параметри, що робить цей підхід ефективним і простим для програмування в режимі реального часу.

1.4 Керування температурними процесами на основі нечіткої логіки

У статті "Design and Implementation of a Distributed Control Platform for a Smart Building Testbed" [9] описується створення і тестування кібер-фізичної системи для управління розумними будівлями. Автори розробили тестову платформу, яка дозволяє випробовувати алгоритми контролю, мережі зв'язку та інтерфейси користувача. Система використовує розподілені сенсори для збору даних про параметри будівлі (температура, освітлення, доступ) і передає їх через протокол MQTT до керуючої системи для подальшого аналізу та генерації оптимальних команд для виконавчих механізмів. Інтерфейс користувача побудований на платформі Node-RED, що дозволяє в реальному часі спостерігати за станом будівлі та взаємодіяти з системою автоматизації. Експериментальні результати підтвердили ефективність прототипу для досліджень у галузі моніторингу та керування розумними будівлями, підкреслюючи потенціал цієї технології для підвищення комфорту, безпеки і ефективності управління будівлями.

У статті автори пропонують нову модель для контролю температури в системі печі, яка базується на нечіткій логіці. Вони використовують систему нечіткої логіки (FLC) замість традиційного PID-контролю через недостатню ефективність управління температурою та складність параметричних налаштувань. Автори прагнуть зменшити відсоток перевищення, час підйому та час стабілізації, розробляючи нову функцію належності та декларацію правил. Запропонована модель демонструє здатність забезпечувати більш точні результати з високою стабільністю та надійністю в порівнянні з традиційними PID-регуляторами. Ефективність цього методу оцінюється за допомогою показників перевищення, швидкості реакції, часу стабілізації та похибки в стаціонарному стані. Запропонована модель показує, що нечіткий

самоналаштовуваний PID-регулятор є більш ефективним і відповідним для промислових систем печей [10].

1.5 Перспекиви використання технологій IoT у системах адаптивного керування.

Використання технологій Інтернету речей (IoT) в системах адаптивного керування відкриває нові можливості для оптимізації процесів у різних сферах, таких як промисловість, енергетика та сільське господарство. Завдяки численним сенсорам, що збирають дані в реальному часі, системи IoT можуть моніторити параметри, такі як температура, вологість, тиск та інші критично важливі показники. Ця інформація потім аналізується для прийняття рішень, що дозволяє адаптувати управлінські стратегії відповідно до змін у навколишньому середовищі. Наприклад, у виробничих процесах адаптивні системи можуть автоматично налаштовувати робочі параметри обладнання, реагуючи на коливання вхідних умов, що призводить до підвищення продуктивності та зниження витрат.

Крім того, інтеграція IoT у адаптивні системи керування сприяє розвитку смарт-інфраструктур. Наприклад, у розумних містах сенсори можуть відстежувати трафік, якість повітря та енергоспоживання, надаючи дані для оптимізації ресурсів і поліпшення життя громадян. Адаптивні алгоритми, підкріплені даними з IoT, можуть прогнозувати навантаження на мережі та автоматично регулювати їх для забезпечення стабільності. Таким чином, технології IoT не лише підвищують ефективність управлінських процесів, але й сприяють створенню більш стійких і екологічно чистих рішень.

Стаття [11] розглядає принципи використання технології Інтернету речей (IoT) в різних сферах техніки. Акцентується увага на технології міжмашинної взаємодії (M2M) та автоматизації рутинних операцій. Надано огляд архітектури Інтернету речей, що складається з чотирьох пов'язаних рівнів. Визначено Інтернет речей як інформаційно-технологічну концепцію, що базується на

обчислювальній мережі, що з'єднує фізичні об'єкти для комунікації між собою та із глобальною інфраструктурою. Детально описано застосування IoT в електроенергетиці, охороні здоров'я, сільському господарстві, моніторингу стану будівель, контролі якості повітря та логістиці. Зазначено вплив IoT на точне землеробство, управління транспортом, сільськогосподарські технології, логістичні процеси тощо.

У статті [12] розглядається проблема вибору протоколів комунікації для систем Інтернету речей (IoT) і визначається їх важливість. Зазначається, що вибір протоколу є складним завданням і залежить від характеру системи IoT та системи передачі даних. Вказується, що існує безліч протоколів комунікації, розроблених дослідниками в останнє десятиріччя. Презентуються шість різних протоколів: HTTP, MQTT, DDS, XMPP, AMQP і CoAP. Зазначається, що кожен з протоколів має свої характеристики та обмеження. Проводиться загальне порівняння протоколів для введення читача в їхні характеристики. Здійснюється детальний аналіз кожного протоколу для з'ясування їхніх переваг та обмежень.

Об'єктом дослідження статті [13] є протоколи передачі даних в умовах Інтернету речей (IoT). Визначена мета роботи – порівняння протоколів IoT на основі демонстраційного стенда. В якості апаратної платформи використовується модуль WeMosD1 з Wi-Fi для IoT-пристроїв. У ході експерименту оцінюються затримки та втрати пакетів залежно від їх розміру. Аналізується відношення службової інформації до корисної за одну транзакцію при використанні протоколів: MQTT, CoAP, HTTP/2. Основний акцент на порівнянні ефективності протоколів MQTT, CoAP та HTTP/2 для передачі даних в умовах IoT. Визначається, які з наведених протоколів є найбільш ефективними для пристроїв IoT на зазначеному демонстраційному стенді.

Стаття [14] розглядає перспективи та застосування Industrial Wireless Sensor Network в контексті Industry 4.0, представляючи конкретну реалізацію бездротової системи комунікації на основі Node-RED та протоколу Modbus для оптимізації промислових процесів. Описано бездротову промислову систему зв'язку, побудовану на платформі Node-RED з використанням протоколу Modbus

для реалізації концепції розумних фабрик. Зазначено використання технологій ПоТ, IWSN, Node-RED та протоколу Modbus для створення бездротової системи комунікації в промисловому середовищі. Описано загальні напрямки дослідження і вказано на можливості застосування створеної системи для розумних фабрик. Стаття висвітлює основні аспекти Internet of Things, включаючи його визначення, технічні здобутки, виклики та перспективи. Огляд статті надає загальний огляд понять та напрямків розвитку IoT. IoT розглядається як ключове слово, що охоплює різноманітні виклики, пов'язані з Інтернетом та вебом, що переходять у реальний фізичний світ.

Стаття [15] представляє ретельний аналіз та порівняння протоколів передачі даних для систем Інтернету речей з високими вимогами до передачі даних, зокрема в реальному часі. Результати та висновки надають читачеві важливий огляд та рекомендації щодо вибору протоколів для конкретних вимог системи. Розглянуті та проаналізовані протоколи MQTT, RTPS, JMS, та AMQP для визначення їхнього застосування та придатності для робототехнічних та автономних систем. Виділені стандарти, переваги та недоліки кожного протоколу. Розроблені власні сценарії тестування, які моделюють складні ситуації для оцінки характеристик передачі даних. Зроблено висновок, що RTPS є найкращим рішенням для систем реального часу з різним трафіком, а MQTT добре справляється з передачею коротких повідомлень.

Стаття [16] висвітлює важливість інтеграції IoT в промисловий сектор та пропонує приклади використання сучасних технологій для збору та обробки даних в індустріальних системах. Робота надає погляд на переваги та можливості, які надає ПоТ, і визначає певні технічні аспекти, такі як використання протоколів та інструментів для роботи з даними. Описано використання різних ПоТ протоколів комунікації, зокрема MQTT. Зазначено переваги використання інструменту Node-RED, розробленого IBM, для здійснення збору та обробки даних. Підкреслено легкість зберігання, аналізу та візуалізації даних з індустріальних систем за допомогою інноваційних IT-

розробок. Зазначено, що створена система надає корисні дані для подальшого розвитку.

Стаття [17] розглядає проблеми, пов'язані зі зростанням складності систем IoT та недостатньою кількістю експертів. Вона вирішує обмеження Node-RED, пропонуючи покращення для поліпшення розробки та відлагодження систем. Емпірична оцінка свідчить про позитивний вплив цих покращень на процес розробки.

Стаття [18] добре демонструє відповідність та практичність використання Raspberry Pi та Node-RED для навчання технологіям IoT. Вона розкриває методику проведення практичних лабораторних вправ, спрямованих на засвоєння студентами концепцій та навичок у сфері Інтернету речей.

У статті [19] досліджено проблематику енергоефективності в контексті сучасних викликів, таких як неолік ресурсів, урбанізація та екологічні регулятиви. Стаття приділяє особливу увагу потенціалу Інтернету речей (IoT) для автоматизації та оптимізації енергоспоживання. У глобальному контексті та із загальним технологічним прогресом дослідження Інтернету речей розглядаються як перспективний напрямок дослідження для ефективного використання енергетичних ресурсів. Аналізуються публікації, що стосуються розумних будівель в Україні, акцентуючи їхню споживчу роль в електроенергії та потенціал IoT для оптимізації енергоспоживання. "Зелений" IoT досліджує виклики та принципи енергоефективності для IoT-пристроїв. Проаналізовано дослідження можливостей Інтернету речей у секторі водопостачання та енергоефективності в будівельному секторі. Розглянуто потенціал технології Інтернету речей для зменшення енергоспоживання. Основні завдання дослідження включають: аналіз архітектури IoT для кращого розуміння її структури та можливостей; вивчення ключових областей застосування Інтернету речей в секторах енергозбереження; а також визначення основних принципів використання IoT для оптимізації енергоспоживання. Розглянуто архітектуру Інтернету речей для глибокого розуміння структури та динаміки систем Інтернету речей, які грають важливу роль у визначенні, як максимально

використовувати їх потенціал для зменшення використання енергії. Досліджено галузі застосування Інтернету речей у контексті енергозбереження, розумінні основних сфер в різних секторах, включаючи будівництво, промисловість та містобудування.

Висновки за розділом:

У цьому розділі було детально розглянуто різні підходи до підвищення ефективності керування температурними процесами, що є критично важливими для багатьох промислових галузей, зокрема у металургії, харчовій промисловості та виробництві побутової техніки.

Починаючи з традиційних методів, таких як пропорційно-інтегрально-диференційні (PID) контролери, можна констатувати, що вони мають певні обмеження, зокрема у вигляді труднощів в налаштуванні параметрів і недостатньої адаптивності до змін у навколишньому середовищі. Традиційні підходи можуть не завжди забезпечувати необхідну точність і швидкість реакції на зміни температури, що може призводити до перевищення заданих значень та втрат в енергії.

Наступним кроком було розглянуто методи на основі нечіткої логіки, які пропонують більш гнучкий підхід до обробки невизначеності в даних. Нечітка логіка дозволяє створювати системи, здатні адаптуватися до різноманітних умов, завдяки чому досягається покращена точність регулювання температури. Зокрема, комбінування нечіткої логіки з традиційними PID-контролерами може значно підвищити ефективність системи, забезпечуючи кращі показники, такі як зменшення перевищення та час стабілізації.

Далі було розглянуто модель прогнозуючого керування, яка використовує математичні моделі для передбачення поведінки системи у майбутньому. Цей підхід дозволяє проактивно регулювати параметри системи, що забезпечує ще більшу ефективність у порівнянні з традиційними методами. Прогнозуюче керування може суттєво знизити ризик появи помилок у контролі температури, оскільки система здатна враховувати потенційні зміни умов.

Розглянувши керування температурними процесами на основі нечіткої логіки, можна відзначити, що цей метод відкриває нові горизонти для адаптивних систем. Завдяки гнучкості нечітких систем, вони здатні вчитися з досвіду і підлаштовуватися під зміни в зовнішньому середовищі, що робить їх ідеальними для застосування у динамічних умовах.

Останнім пунктом аналізу стало обговорення перспектив використання технологій IoT у системах адаптивного керування. IoT надає можливості для збору і аналізу величезних обсягів даних у реальному часі, що дозволяє ще точніше контролювати температурні процеси. Інтеграція IoT з адаптивними системами керування забезпечує високий рівень автоматизації, що призводить до зменшення витрат і покращення ефективності виробництв.

У підсумку, аналіз різних підходів до керування температурними процесами показує, що сучасні технології, такі як нечітка логіка та IoT, здатні значно підвищити ефективність і точність цих систем. Використання новітніх методів обробки даних і адаптивних стратегій в управлінні може стати запорукою успішного розвитку промисловості в умовах швидко змінюваного середовища.

РОЗДІЛ 2

ІДЕНТИФІКАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ ТА СИНТЕЗ КЕРУВАННЯ ПРОЦЕСОМ

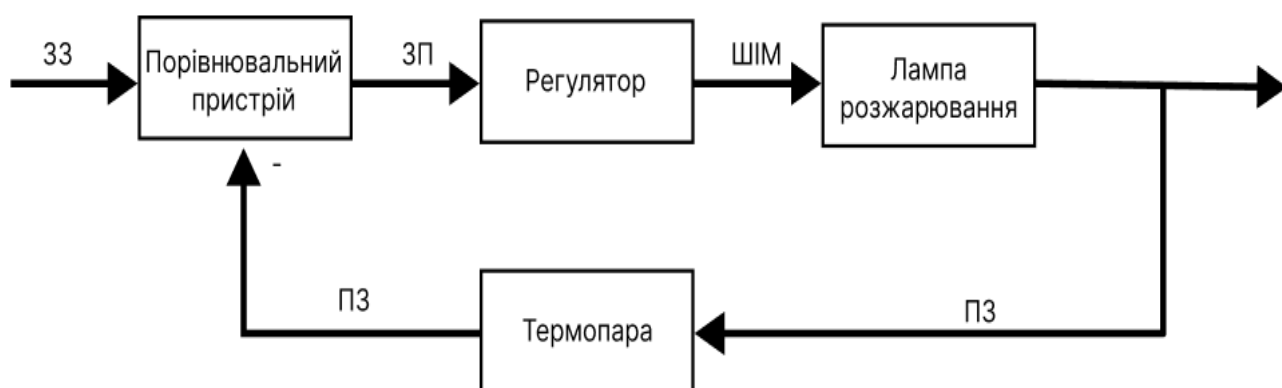
2.1 Система керування процесом нагріву

2.1.1 Функціональна схема системи

Система керування процесом нагріву складається з таких основних елементів:

- порівняльний пристрій – призначений для визначення значення помилки на основі даних про задаюче значення та поточне значення;
- регулятор – на основі значення помилки формує управляючий шім-сигнал;
- лампа розжарювання – нагрівальний елемент схеми (об'єкт управління);
- термопара – елемент для реалізації зворотного зв'язку.

Зв'язок між усіма елементами схем зображено на рис. 2.1.



ЗЗ – задаюче значення, ЗП – значення помилки, ШІМ – шім-сигнал, ПЗ – поточне значення

Рисунок 2.1 - Функціональна схема системи

2.1.2 Математична модель ПДД-регулятора

ПДД-регулятор формує управляюче значення на основі закону (2.1)

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (2.1)$$

де K_p – пропорційна складова;

K_i - інтегральна складова;

K_d - диференційна складова.

Для обмеження перевантаження виконавчого механізму управляюче значення $u(t)$ обмежено в проміжку:

$$u_{min} \leq u(t) \leq u_{max} \quad (2.2)$$

Управляюче значення $u(t)$ перетворюється у коефіцієнт заповнення D для генерації ШІМ-сигналу:

$$D = \frac{u(t) - u_{min}}{u_{max} - u_{min}} \quad (2.3)$$

де $D \in [0,1]$.

Для формування ШІМ-сигналу задається частота f_{PWM} , а також визначається час включення (T_{on}) та виключення T_{off} :

$$T_{on} = D * T_{PWM}, \quad T_{off} = T_{PWM} - T_{on} \quad (2.4)$$

$$\text{де } T_{PWM} = \frac{1}{f_{PWM}}$$

2.2 Процес нагріву матеріалу

2.2.1 Математична модель нагріву матеріалу

На рис. 2.2 зображено процес нагріву деталі у печі. З цього рисунку видно, що процес теплопередачі має 2 складові: конвекцію та теплопровідність.

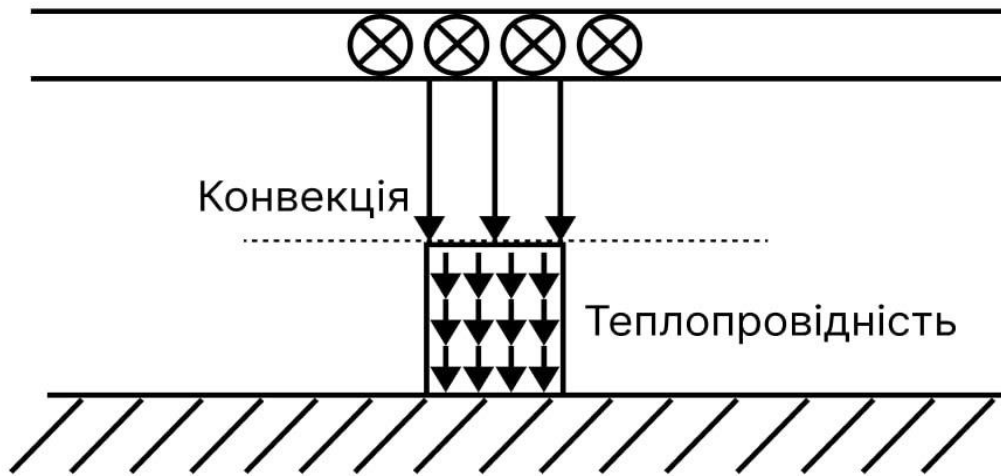


Рисунок 2.2 - Графічне зображення процесу нагріву

Для описання процесу передачі тепла використовується рівняння теплопровідності. Воно також відоме як рівняння теплопереносу або рівняння Фур'є. Це основне рівняння в теплопередачі, яке дозволяє визначати розподіл температури в тілі в залежності від часу і простору.

Рівняння Фур'є у загальному вигляді

$$\rho c \frac{dT}{dt} = \lambda \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) + Q(x, y, z, t) \quad (2.5)$$

де ρ – щільність матеріалу;

c – питома теплоємність;

λ – теплопровідність

$T(x, y, z, t)$ – температура у точці (x, y, z) в момент часу t ;

$Q(x, y, z, t)$ – функція джерела, яка пропорційна кількості тепла, що виділяється або поглинається у одиницю часу.

Відповідно до рис. 2.2 конвекційний перенос тепла є функцією $Q(x, y, z, t)$.

Дана функція є граничною умовою 3-го роду для рівняння теплопровідності (2.5), яка описана законом Ньютона – Ріхмана:

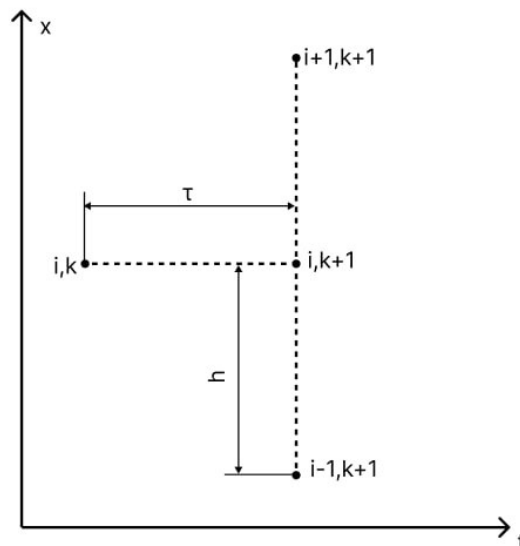
$$\alpha_{\text{кон}}(x, y, z, t); T_{\text{зовн.сер.}} \quad q = \alpha_{\text{кон}}(T_{\text{зовн.сер.}} - T_{\text{пов}}) \quad (2.6)$$

Оскільки процес передачі тепла відбувається у одному напрямку (Рисунок 2.2), то рівняння (2.5) може бути спрощене до виду:

$$\rho c \frac{dT}{dt} = \lambda \frac{d^2T}{dx^2} \quad (2.7)$$

2.2.2 Метод кінцевих різниць

Для чисельного розв'язання задачі застосуємо метод кінцевих різниць (рис. 2.3), дискретизуючи рівняння теплопровідності за часом та простором. У кожній точці сітки ми оновлюємо значення температури на кожному тимчасовому етапі, враховуючи як теплопровідність, так і конвективне перенесення тепла.



i – номер вузла, k – номер кроку за часом

Рисунок 2.3 - Метод кінцевих різниць

Розіб'ємо простір на елементи. Для цього визначимо розмір кроку у просторі h та за часом h за формулою (2.8).

$$h = \frac{L}{N}, \quad \tau = \frac{t_{\text{розр.}}}{K} \quad (2.8)$$

де L – товщина матеріалу;

N – число відрізків, при розбитті за координатою;

$t_{\text{розр.}}$ - час розрахунку;

K - число відрізків при розбитті за часом.

Апроксимуємо $\frac{dT}{dt}$ за формулою (2.9).

$$\frac{dT}{dt} \approx \frac{T_{i,k+1} - T_{i,k}}{\tau} \quad (2.9)$$

Тоді апроксимація $\frac{\partial^2 T}{\partial x^2}$ може бути представлена формулою (2.10).

$$\frac{\partial^2 T}{\partial x^2} = \frac{\partial}{\partial x} \frac{\partial T}{\partial x} \approx \frac{\frac{\partial T_{i+0.5,k+1}}{\partial x} - \frac{\partial T_{i-0.5,k+1}}{\partial x}}{h} = \frac{\frac{T_{i+1,k+1} - T_{i,k+1}}{h} - \frac{T_{i,k+1} - T_{i-1,k+1}}{h}}{h} = \frac{T_{i+1,k+1} - 2T_{i,k+1} + T_{i-1,k+1}}{h^2} \quad (2.10)$$

Підставимо рівняння (2.9) та (2.9) у (2.7).

$$\rho c \frac{T_{i,k+1} - T_{i,k}}{\tau} = \lambda \frac{T_{i+1,k+1} - 2T_{i,k+1} + T_{i-1,k+1}}{h^2} \quad (2.11)$$

Перепишемо рівняння (2.11) у формі:

$$\frac{\lambda}{h^2} T_{i+1,k+1} - \left(\frac{2\lambda}{h^2} + \frac{\rho c}{\tau} \right) T_{i,k+1} + \frac{\lambda}{h^2} T_{i-1,k+1} = -\frac{\rho c}{\tau} T_{i,k} \quad (2.12)$$

Введемо 4 коефіцієнти:

$$A = C = \frac{\lambda}{h^2}, \quad B = \left(\frac{2\lambda}{h^2} + \frac{\rho c}{\tau} \right), \quad F_i = -\frac{\rho c}{\tau} T_{i,k} \quad (2.13)$$

Підставимо коефіцієнти (2.13) у (2.12):

$$AT_{i+1,k+1} - BT_{i,k+1} + CT_{i-1,k+1} = F_i \quad (2.14)$$

Представимо $T_{i-1,k+1}$ та $T_{i,k+1}$ у вигляді рівнянь (2.15) та (2.16).

$$T_{i-1,k+1} = \alpha_{i-1} T_{i,k+1} + \beta_{i-1} \quad (2.15)$$

$$T_{i,k+1} = \alpha_i T_{i+1,k+1} + \beta_i \quad (2.16)$$

Підставимо праву частину рівняння (2.15) замість $T_{i-1,k+1}$ у рівняння (2.14).

$$AT_{i+1,k+1} - BT_{i,k+1} + C\alpha_{i-1}T_{i,k+1} + C\beta_{i-1} = F_i \quad (2.17)$$

З рівняння (2.17) виразимо $T_{i,k+1}$

$$T_{i,k+1} = \frac{A}{B - C\alpha_{i-1}} T_{i+1,k+1} + \frac{C\beta_{i-1} - F_i}{B - C\alpha_{i-1}} \quad (2.18)$$

З рівняння (2.18) отримаємо формули для розрахунку коефіцієнтів α_i та β_i :

$$\alpha_i = \frac{A}{B - C\alpha_{i-1}}, \quad \beta_i = \frac{C\beta_{i-1} - F_i}{B - C\alpha_{i-1}} \quad (2.19)$$

Визначимо прогоночні коефіцієнти на границях з урахуванням граничних умов 3-го роду.

Для правої границі:

$$x = 0: \quad -\lambda \frac{\partial T}{\partial x} = \alpha_{\text{кон}}(T_{\text{зовн.сер.}} - T_1) \quad 0 < t \leq t_{\text{розр.}} \quad (2.20)$$

Апроксимуємо рівняння:

$$-\lambda \frac{T_2 - T_1}{h} = \alpha_{\text{кон}}(T_{\text{зовн.сер.}} - T_1) \quad T_1 = \alpha_1 T_2 + \beta_1 \quad (2.21)$$

Виразимо T_1 :

$$T_1 = \frac{\lambda}{h\alpha_{\text{кон}} + \lambda} T_2 + \frac{h\alpha_{\text{кон}} T_{\text{зовн.сер.}}}{h\alpha_{\text{кон}} + \lambda} \quad (2.22)$$

З (2.22) отримаємо формули для розрахунку коефіцієнтів:

$$\alpha_1 = \frac{\lambda}{h\alpha_{\text{кон}} + \lambda}, \quad \beta_1 = \frac{h\alpha_{\text{кон}} T_{\text{зовн.сер.}}}{h\alpha_{\text{кон}} + \lambda} \quad (2.23)$$

2.2.3 Алгоритм вирішення

На основі рівнянь сформуємо алгоритм для розрахунку нагріву деталі. Представимо його у вигляді блок-схеми, як це показано на рис 2.4.

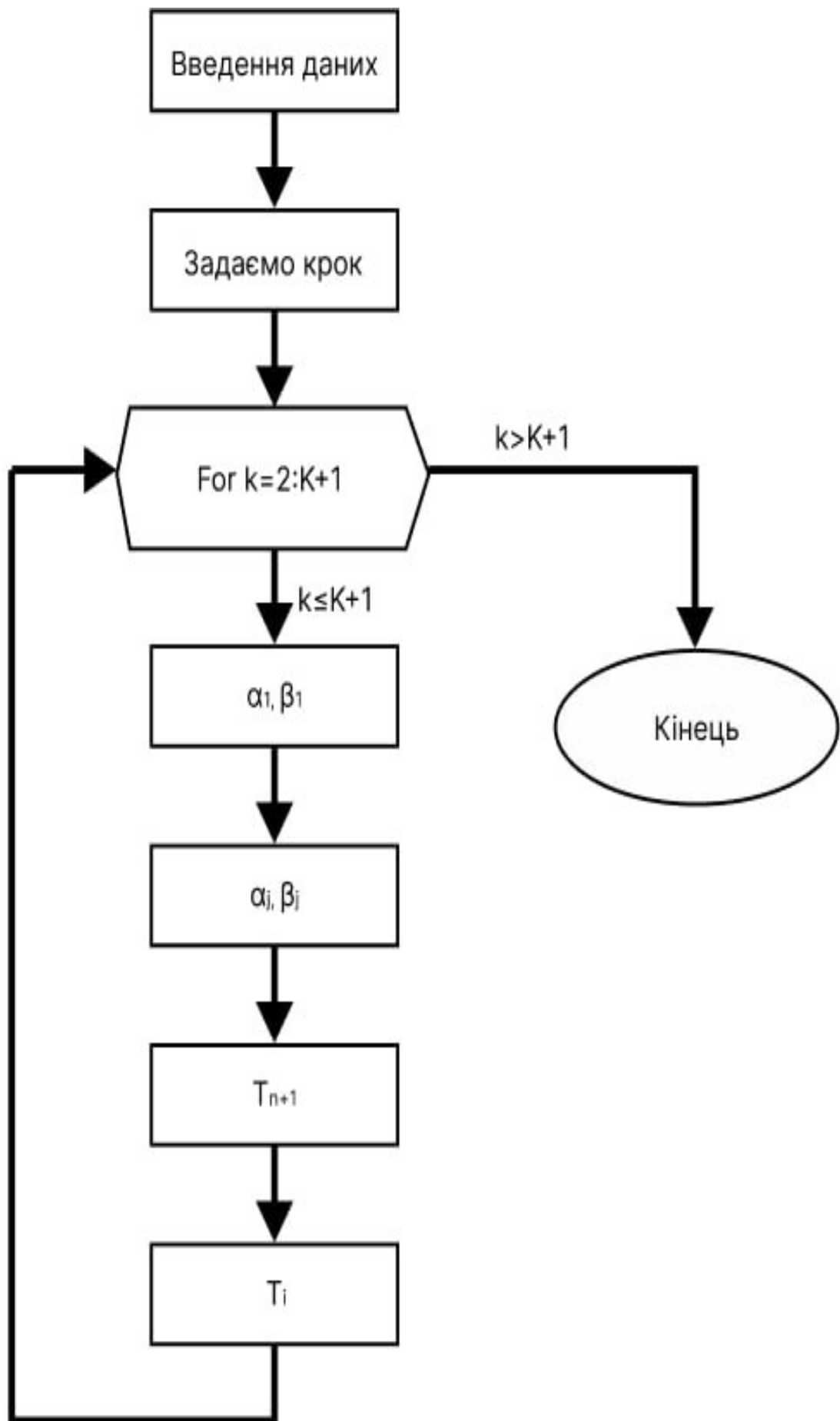


Рисунок 2.4 - Алгоритм для визначення температури

2.2.4 Реалізація алгоритму на Python.

Реалізуємо алгоритм (рис. 2.5) мовою Python

```

L = L / 1000
h = L / N
tau = t / K
hconv = 10
Tzovn = 25

T = np.ones((N+1, K+1)) * T0
time = np.zeros(K+1)
time_to_target = None

alfa = np.zeros(N+1)
beta = np.zeros(N+1)
alfa[0] = lambda_val/(h*hconv+lambda_val)
beta[0] = (h*hconv*Tzovn)/(h*hconv + lambda_val)
T[N, :] = Tg2

for k in range(1, K+1):
    time[k] = time[k-1] + tau

    A = np.zeros(N+1)
    B = np.zeros(N+1)
    C = np.zeros(N+1)
    F = np.zeros(N+1)

    for i in range(N+1):

```

Рисунок 2.5 - Алгоритм на мові python

Висновки за розділом:

У 2 розділі було зроблено наступне:

- Показано функціональну схему контуру керування температурою;
- Наведено математичну модель ПД-регулятора та розрахунок ШІМ-модульованого сигналу;
- Наведено математичну модель процесу нагріву матеріалу. Математична модель була представлена рівнянням теплопровідності Фур'є. Для отримання розв'язку рівняння було використано метод скінченних різниць.
- Побудовано алгоритму для визначення температури на основі математичної моделі процесу нагріву.
- Реалізовано алгоритм на мові Python.

РОЗДІЛ 3

ПРОГРАМНО-ТЕХНІЧНА РЕАЛІЗАЦІЯ СИСТЕМИ КЕРУВАННЯ ПРОЦЕСОМ

3.1 Загальний опис обладнання лабораторного стенду

Лабораторний стенд (рис. 3.1) призначений для дослідження системи керування процесом нагріву матеріалу на рухомому конвеєрі.

До його складу входять такі елементи:

- робоча станція;
- ПЛК з різними модулями;
- панель оператора;
- виконавчі механізми;
- датчики та сенсори;

На електричній принциповій схемі зображено усі елементи схеми та з'єднання між ними. Повна назва кожного елемента наведена у специфікації обладнання.

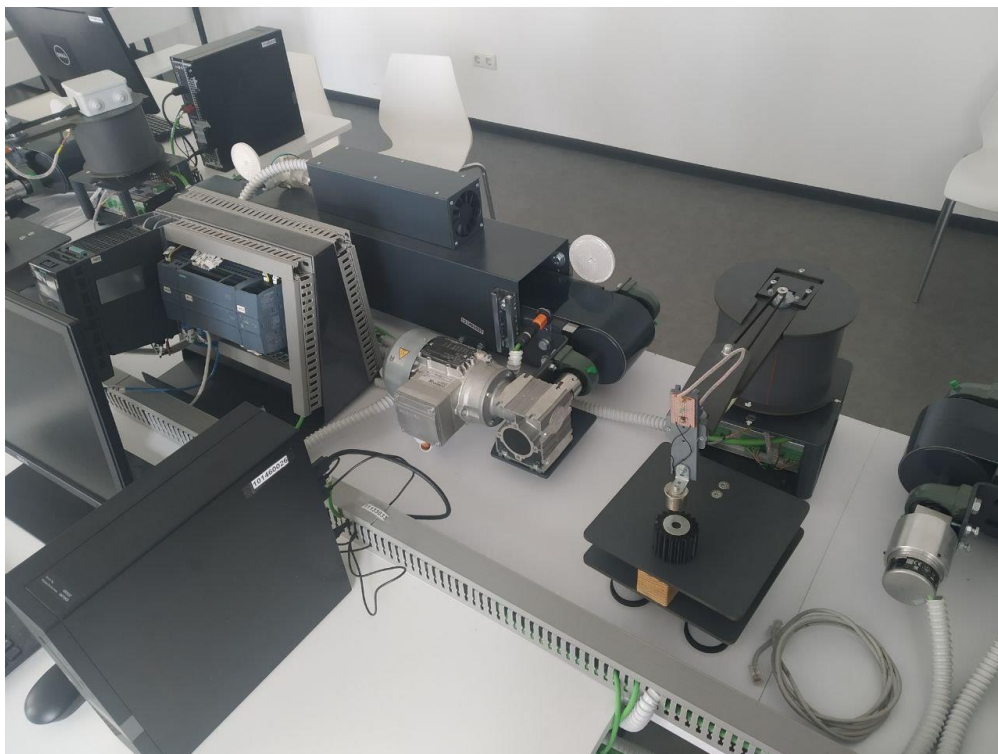


Рисунок 3.1 - Лабораторний стенд

3.1.1 Контролер Siemens серії S7-1200

Контролер Siemens S7-1200 (рис. 3.2) з процесорним модулем CPU 1215C — це програмований логічний контролер (ПЛК), призначений для управління процесом нагріву матеріалу на рухомому конвеєрі.



Рисунок 3.2 - ПЛК S7-1200

3.1.2 Модуль SIMATIC IOT2040

SIMATIC IOT2040 — це промисловий інтернет-шлюз (IoT Gateway), розроблений компанією Siemens для інтеграції, обробки та передачі даних у системах автоматизації та IoT-мережах. Він служить платформою для підключення пристроїв, збору даних, їх обробки та передачі в хмарні сервіси або інші системи.



Рисунок 3.3 IOT-модуль Simatic IOT2040

3.1.3 Виконавчі механізми

Виконавчі механізми (або актуатори) — це пристрої, які перетворюють керуючі сигнали від системи управління в механічний рух або іншу дію, що виконує певну задачу в процесі.

Серед виконавчих механізмів стенду можна виділити наступні:

- Електропривод - приводить в рух стрічку конвєсера, що переміщує матеріали або деталі.

- SCARA-робот, до складу якого входять крокові двигуни та електромагніт – призначений для переміщення матеріалу на стрічку конвеєра.
- Лампа розжарювання – нагрівальний елемент, тепло якого поширюється у нагрівальній камері.
- Вентилятор – здійснює обдув лампи розжарювання для зниження температури у камері нагріву.

3.1.4 Датчики для контролю технологічних параметрів.

Датчики призначені для збору інформації про технологічні параметри. Сигнали з датчиків надходять до ПЛК для контролю над технологічним процесом.

До складу стенду входять наступні датчики:

- Датчики положення – призначені для визначення місцеположення матеріалу на стрічці конвеєру. Розташовані з двох протилежних сторін поза нагрівальною камерою.
- Інкрементальний енкодер – використовується для позиціонування валу двигуна. За його допомогою можна визначити швидкість руху стрічки непрямым методом.
- Терморезистор – призначений для контролю рівня температури у нагрівальній камері.

3.2 Системна архітектура

Система складається з трьох основних пристроїв: АРМ (автоматизоване робоче місце), ПЛК S7-1200 та ІоТ-модуль SimaticIOT2040 (рис. 3.4)

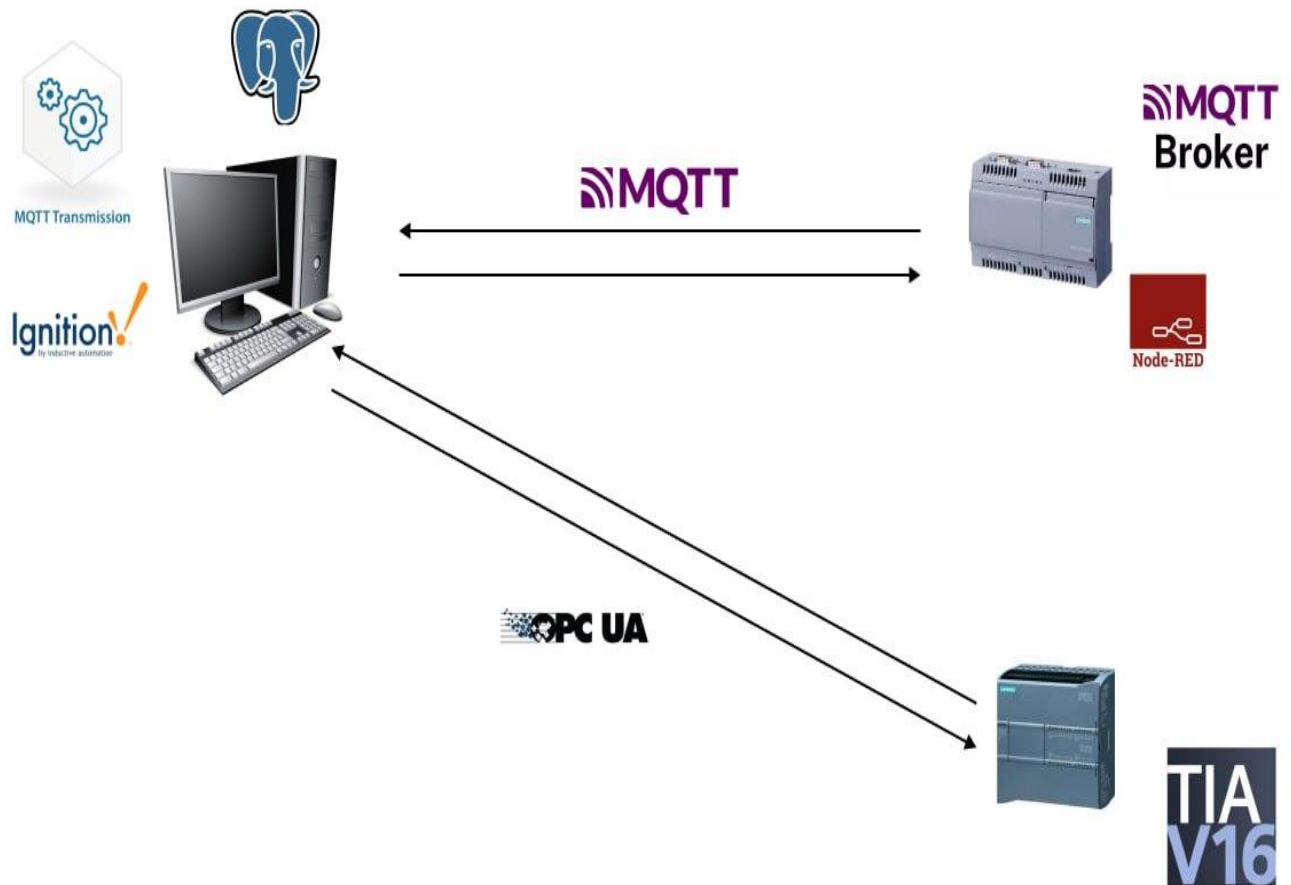


Рисунок 3.4 - Системна архітектура

На АРМ встановлено програмне забезпечення:

- База даних PostgreSQL, у якій зберігаються дані про фізичні властивості матеріалу, що гріється під час технологічного процесу.
- SCADA система Ignition Gateway. За допомогою модулів цієї системи (OPC Client та MQTT Transmission) АРМ здатен взаємодіяти з модулями S7-1200 та Simatic IOT2040. Модуль Designer потрібен для розробки НМІ-панелі, а також для її тестування.
- TIA Portal 16 – середовище розробки ПО для ПЛК S7-1200.

Но модулі Simatic IOT2040 встановлено середовище Node-RED та MQTT Broker. MQTT Broker є ключовим елементом для обміну даними за протоколом MQTT. Node-RED отримує дані через MQTT, виконує обчислення, та надсилає результат на брокера.

3.3 Програмна реалізація

3.3.1 Конфігурація програмного забезпечення модуля Simatic IOT2040

Для встановлення зв'язку з модулем Simatic IOT2040 потрібне пряме підключення через Ethernet-порт. Для цього з'єднаємо ПК та модуль Ethernet-кабелем.

Модуль має адресу 192.168.200.1. Задамо статичну адресу для робочого комп'ютера 192.168.200.2, аби пристрої були в одній підмережі (рис. 3.5).

Параметры IP

Назначение IP:	Вручную
IPv4-адрес:	192.168.200.2
Длина префикса подсети IPv4:	24
Шлюз IPv4:	192.168.200.1
DNS-серверы IPv4:	8.8.8.8
<input type="button" value="Редактировать"/>	

Рисунок 3.5 - встановлення статичної ір-адреси

Для підключення до модуля необхідно використати технологію віддаленого доступу.

Для цього через PuTTY встановимо зв'язок між ПК та модулем за протоколом SSH:

- Задамо ір-адресу – 192.168.200.1;
- Вкажемо порт – 22;
- Оберемо тип підключення – SSH.
- Процес підключення зображено на рис. 3.6

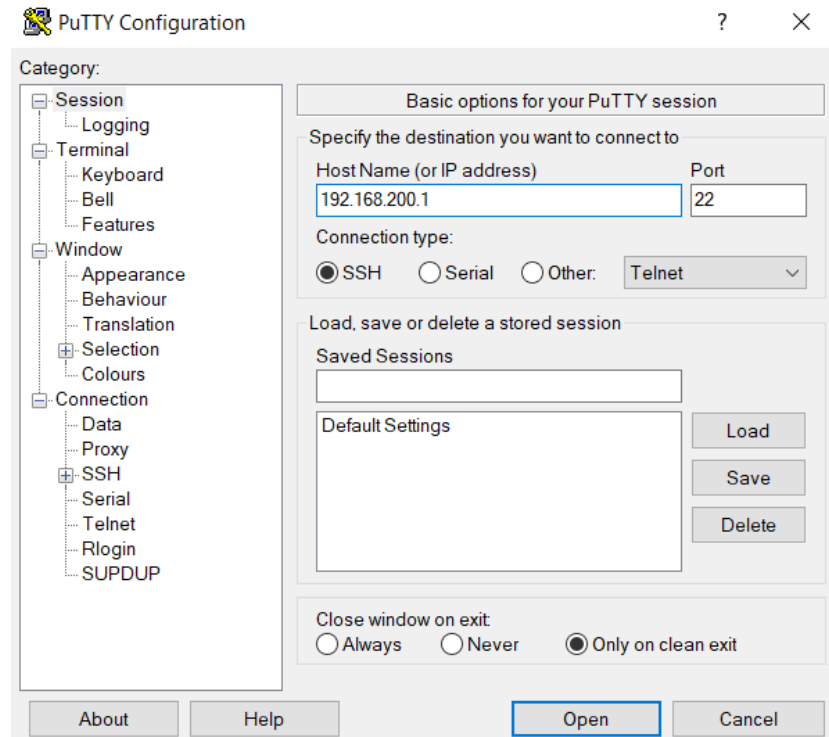


Рисунок 3.6 - встановлення зв'язку з модулем Simatic IOT2040

Після успішного виконання процедури з'явиться вікно з терміналом модуля Simatic IOT2040 (рис. 3.7)

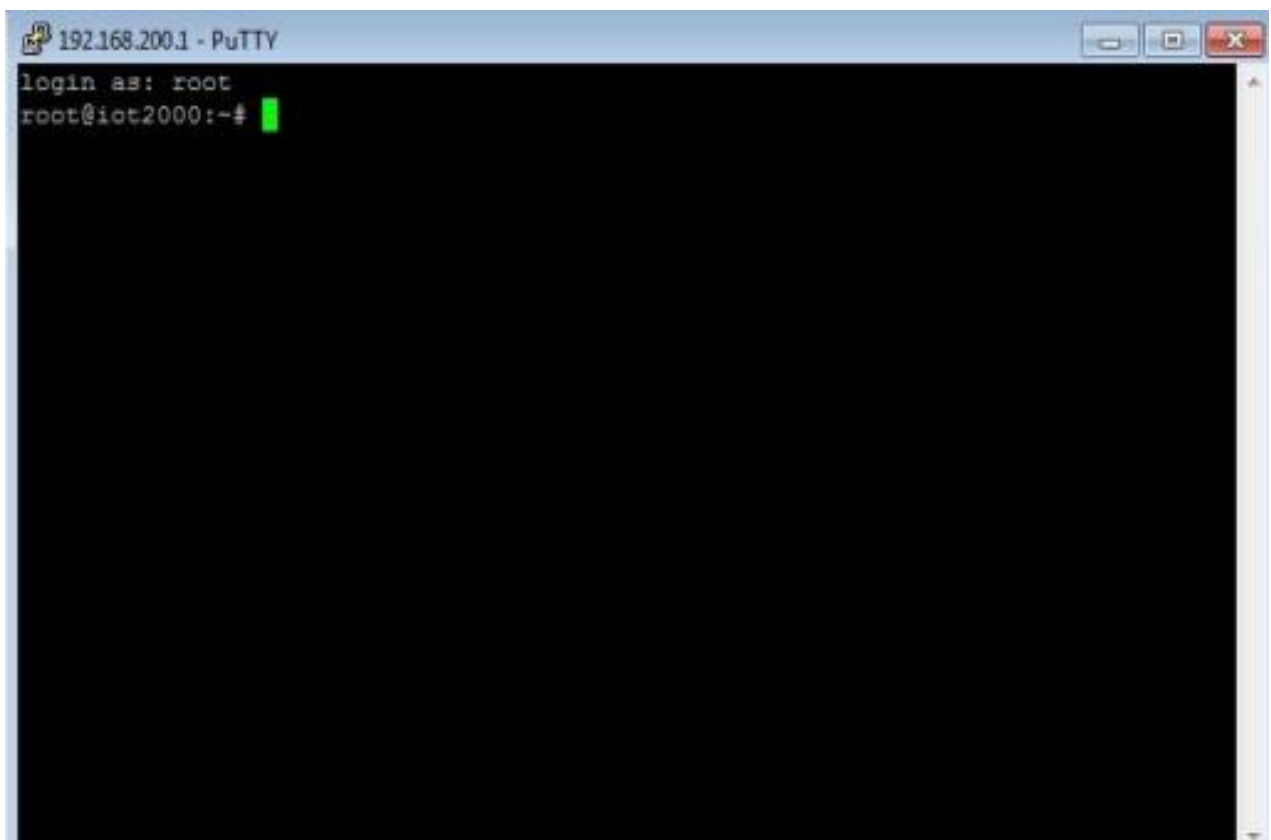


Рисунок 3.7 - Термінал модуля Simatic IOT2040

У терміналі введемо команду «iot2000setup» для запуску програми для конфігурації налаштування модуля.

Перейдемо до налаштувань автозапуску та оберемо сервіси (рис. 3.8), які будуть запущені при включенні модуля:

- Node-red;
- Mosquitto Broker.

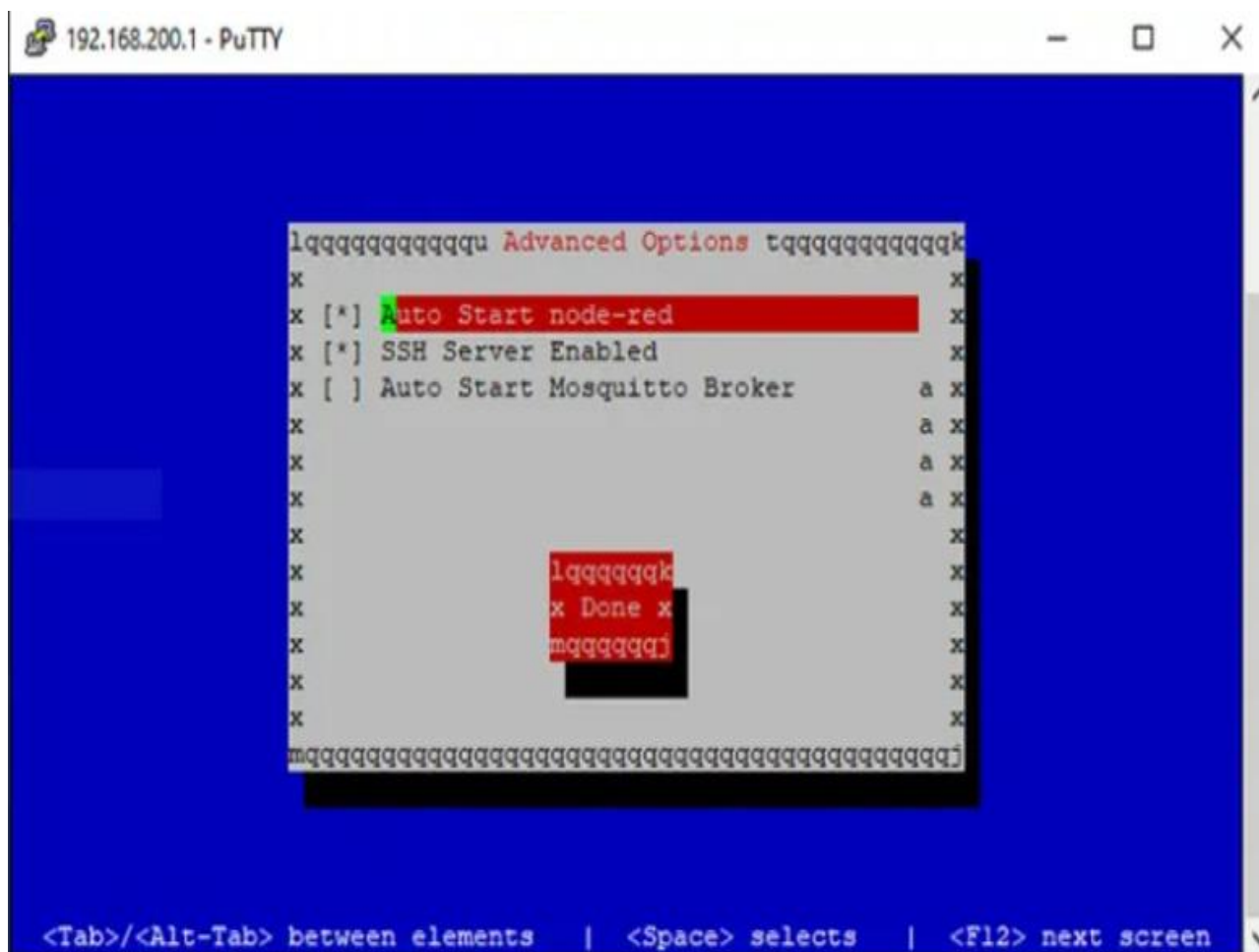


Рисунок 3.8 - Вікно налаштувань автозапуску

Підключимось до Node-RED. Для цього у браузері вкажемо URL <http://192.168.200.1:1880>.

Створимо програму для розрахунку часу нагріву матеріалу за алгоритмом з розділу 2.

Передача даних у середовище Node-RED здійснюється за протоколом MQTT. Налаштуємо блоки для «підписки» на дані та на «публікацію» даних.

Для «підписки» на дані використаємо блок «mqtt-in». Налаштування блоку наведено на рис. 3.9.

Свойства

Сервер: 192.168.200.1:1883

Action: Subscribe to single topic

Тема: SimaticIOT2040/params

QoS: 0

Выход: объект JSON

Имя: Имя

Свойства

Имя: Имя

Соединение | Безопасность | Сообщения

Сервер: 192.168.200.1 | Порт: 1883

Connect automatically

TLS

Protocol: MQTT V3.1.1

ID клиента: Оставьте пустым для автоматически сгенерированного

Keep-alive время (сек): 60

Session: Использовать чистую сессию

Рисунок 3.9 - Налаштування блоку mqtt-in

Процес налаштування блоку mqtt-out відбувається аналогічно рис. 3.9.

Дані з блоку «mqtt-in» надходять у форматі JSON. Ці дані будуть передані для python-скрипта у якості аргументу.

Процес передачі даних у якості аргументу виглядає так (рис. 3.10). Видно, що під час передачі аргументу виникає помилка, яка пов'язана з відсутністю подвійних кавичок.

```

10  data = json.loads(sys.argv[1])
11
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\VikondaUser9633>
* History restored

PS C:\Users\VikondaUser9633> python calc.py '{"key":"value"}'
Traceback (most recent call last):
  File "C:\Users\VikondaUser9633\calc.py", line 10, in <module>
    data = json.loads(sys.argv[1])
           ^^^^^^^^^^^^^^^^^^^^^^^^^
File "C:\Users\VikondaUser9633\AppData\Local\Programs\Python\Python312\Lib\json\__init__.py", line 346, in loads
    return _default_decoder.decode(s)
           ^^^^^^^^^^^^^^^^^^^^^^^^^
File "C:\Users\VikondaUser9633\AppData\Local\Programs\Python\Python312\Lib\json\decoder.py", line 337, in decode
    obj, end = self.raw_decode(s, idx=_w(s, 0).end())
               ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
File "C:\Users\VikondaUser9633\AppData\Local\Programs\Python\Python312\Lib\json\decoder.py", line 353, in raw_decode
    obj, end = self.scan_once(s, idx)
               ^^^^^^^^^^^^^^^^^^^^^
json.decoder.JSONDecodeError: Expecting property name enclosed in double quotes: line 1 column 2 (char 1)

```

Рисунок 3.10 - Передача даних у форматі JSON

Перепишемо код таким чином (рис. 3.11), щоб побачити, у якому вигляді надходять дані.

```

11  data = sys.argv[1]
12  print(data)

```

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

● PS C:\Users\VikondaUser9633> python calc.py '{"key":"value"}'
  {key:value}
○ PS C:\Users\VikondaUser9633> █

```

Рисунок 3.11 - Отримане значення аргументу

Видно, що під час передачі даних у якості аргумента через PowerShell рядок форматується.

Для вирішення даної проблеми змінимо процес передачі аргументу. Будемо передавати файл у форматі JSON, відкривати файл у режимі читання, та завантажувати з нього дані (рис. 3.12).

Видно, що передача даних здійснюється правильно.

```
1 import sys, json
2
3 json_file = sys.argv[1]
4 with open(json_file, 'r') as f:
5     json_data = json.load(f)
6 print(json_data)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
● PS C:\Users\VikondaUser9633> python calc.py data.json
  {'a': 5, 'b': 6, 'c': 7}
○ PS C:\Users\VikondaUser9633> █
```

Рисунок 3.12 - Зчитування даних з JSON-файлу

У середовищі Node-RED додамо блок «write file», який приймає дані у форматі JSON з блоку «mqtt-in».

Вкажемо назву файлу, у який буде здійснено запис. Також вкажемо дію, яку треба робити під час запису – перезаписувати файл (рис. 3.13).

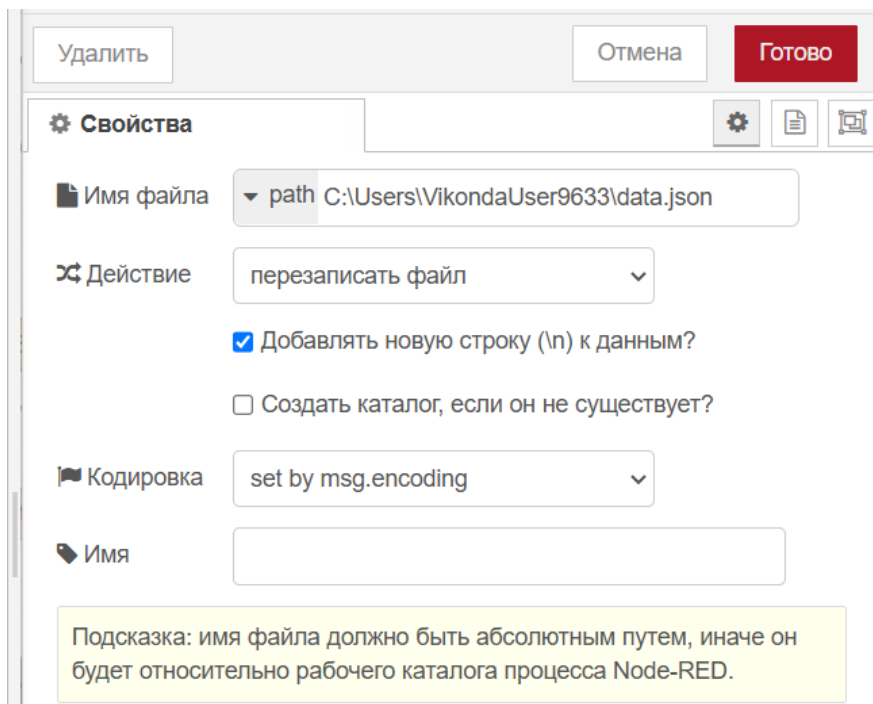


Рисунок 3.13 Налаштування блоку "write-file"

Для запуску python-скрипта використаємо блок «exec». У налаштуваннях блоку вкажемо команду «python calc.py data.json» (рис. 3.14).

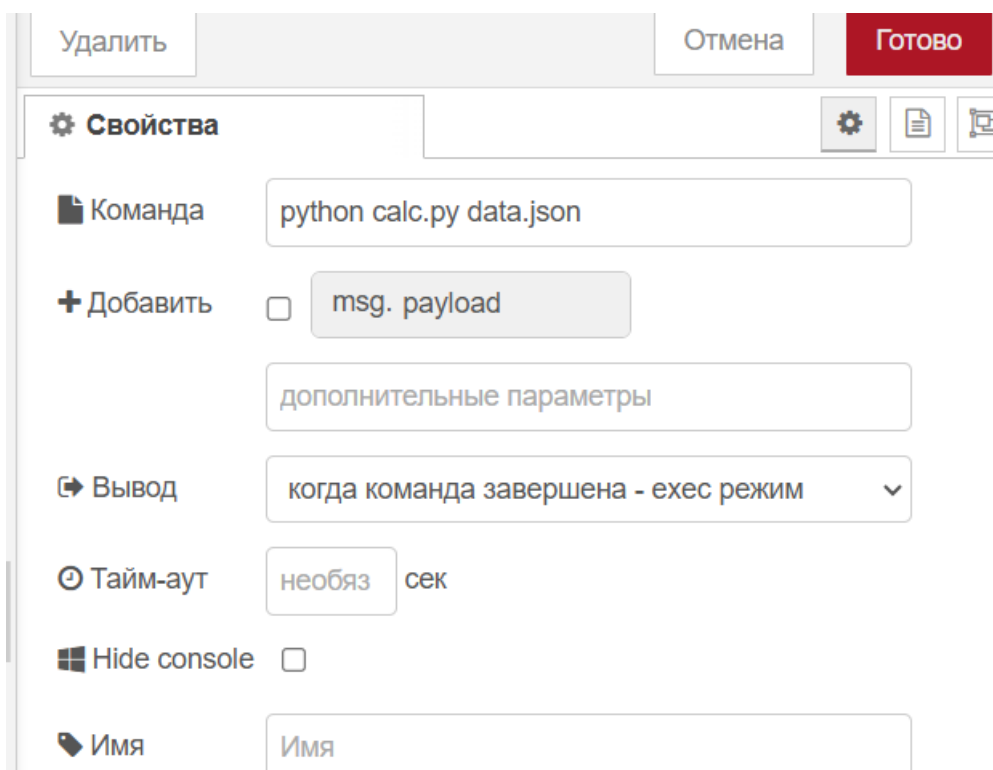


Рисунок 3.14 - Налаштування блоку "exec"

Об'єднаємо усі блоки в один потік (рис. 3.15).

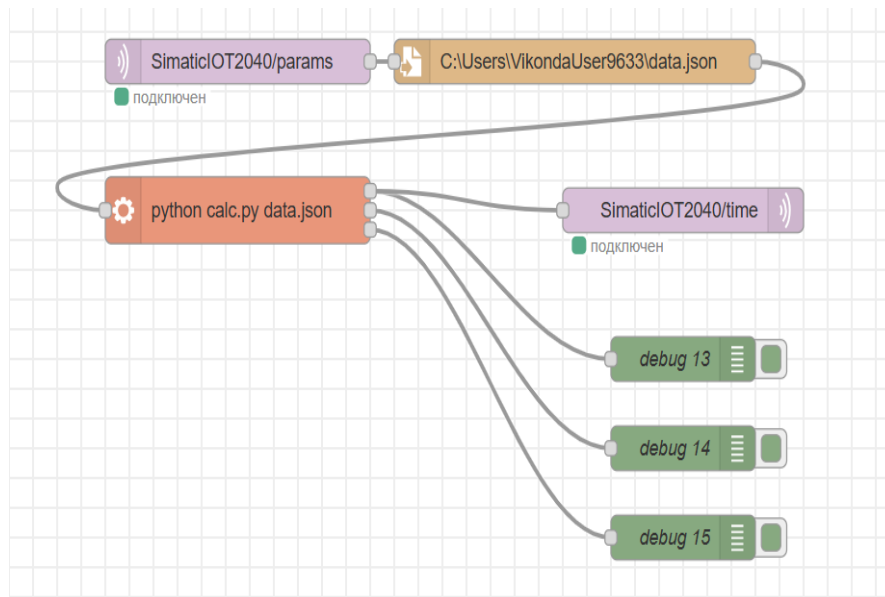


Рисунок 3.15 - Потік у Node-RED

Для перевірки роботи програми відправимо тестові дані з MQTT Explorer. Програма спрацювала коректно (рис. 3.16)

The screenshot shows the MQTT Explorer interface. On the left, the 'Topic' field is set to 'SimaticOT2040/params'. Below it, the 'Publish' section shows the message format set to 'json' and the payload as a JSON object:

```
{
  "a": 1,
  "b": 2,
  "c": 3
}
```

. A 'PUBLISH' button is visible. On the right, the 'отладка' (debug) console shows two log entries:

- 02.12.2024, 12:07:50 node: debug 13: SimaticOT2040/params : msg.payload : string[26] with a preview of the JSON string: `"{'a': 1, 'b': 2, 'c': 3}"`.
- 02.12.2024, 12:07:50 node: debug 15: SimaticOT2040/params : msg.payload : Object with a preview of the JSON object: `{ code: 0 }`.

Рисунок 3.16 - Тестування роботи програми у MQTT Explorer

Також у MQTT Explorer бачимо, що дані успішно публікуються через блок «mqtt-out» (рис. 3.17).

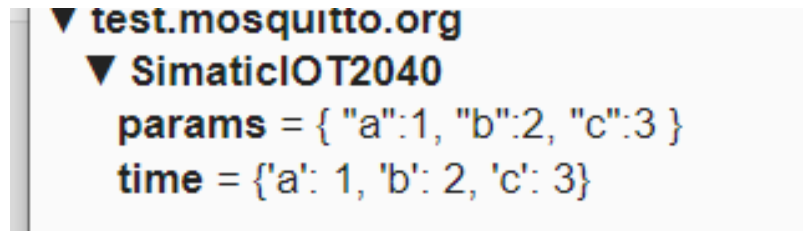


Рисунок 3.17 - Успішна публікація даних

3.3.2 Встановлення зв'язку між Ignition Gateway та базою даних PostgreSQL

У базі даних PostgreSQL створимо таблицю, у якій будемо зберігати дані про фізичні властивості матеріалів. Також наповнимо таблицю тестовими даними (рис. 3.18).

The screenshot shows a query editor with the following SQL code:

```

1 DROP TABLE IF EXISTS material_definition;
2
3 CREATE TABLE material_definition(
4     id serial PRIMARY KEY,
5     name text,
6     termal_conductivity integer,
7     specific_heat_capacity integer,
8     density integer,
9     height float
10 );
11
12
13 INSERT INTO material_definition(name, termal_conductivity, specific_heat_capacity,
14     density, height) VALUES
15 ('Мідь', 390, 385, 8960, 0.03),
16 ('Алюміній', 237, 900, 2700, 0.03),
17 ('Сталь', 50, 500, 7850, 0.03),
18 ('Залізо', 80, 459, 7870, 0.03)
19

```

Below the query editor, the 'Messages' tab is active, showing the following output:

```

CREATE TABLE

Query returned successfully in 48 msec.

```

Рисунок 3.18 - Створення таблиці та її наповнення

Виконаємо SQL-запит для перевірки даних у таблиці (рис. 3.19).

Query		Query History	
1	▼	SELECT *	FROM public.material_definition
2		ORDER BY id	ASC

Data Output		Messages		Notifications		
	id	name	termal_conductivity	specific_heat_capacity	density	height
	[PK] integer	text	integer	integer	integer	double precision
1	1	Мідь	390	385	8960	0.03
2	2	Алюміній	237	900	2700	0.03
3	3	Сталь	50	500	7850	0.03
4	4	Залізо	80	459	7870	0.03

Рисунок 3.19 - SQL-запит

Перейдемо на веб-сторінку Ignition Gateway за посиланням <http://localhost:8088/web/config>. У лівому меню оберемо пункт «Connections» у блоці «Databases» (рис. 3.20).

Рисунок 3.20 - Ignition Gateway

Натиснемо «Create new Database Connection» для створення нового зв'язку з базою даних (рис. 3.21).

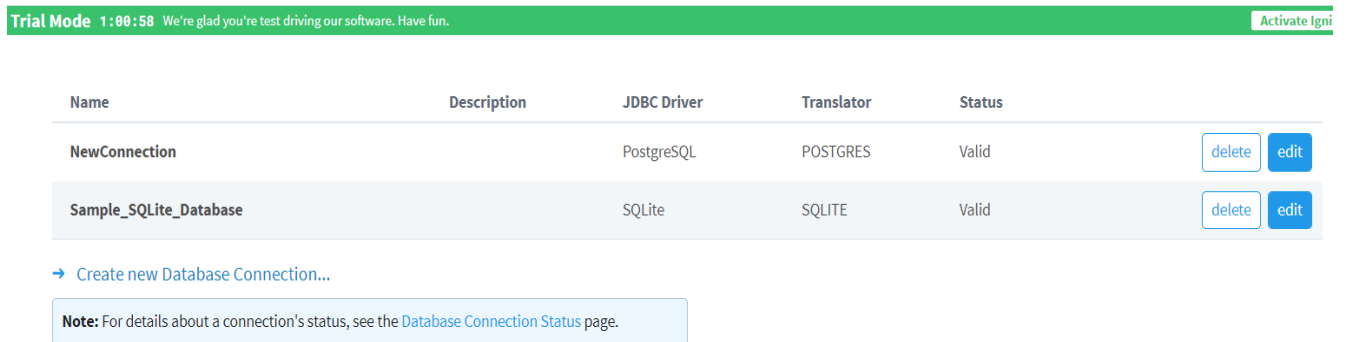


Рисунок 3.21 - Підключені бази даних

Із запропонованого списку оберемо драйвер бази даних PostgreSQL (рис. 3.22).

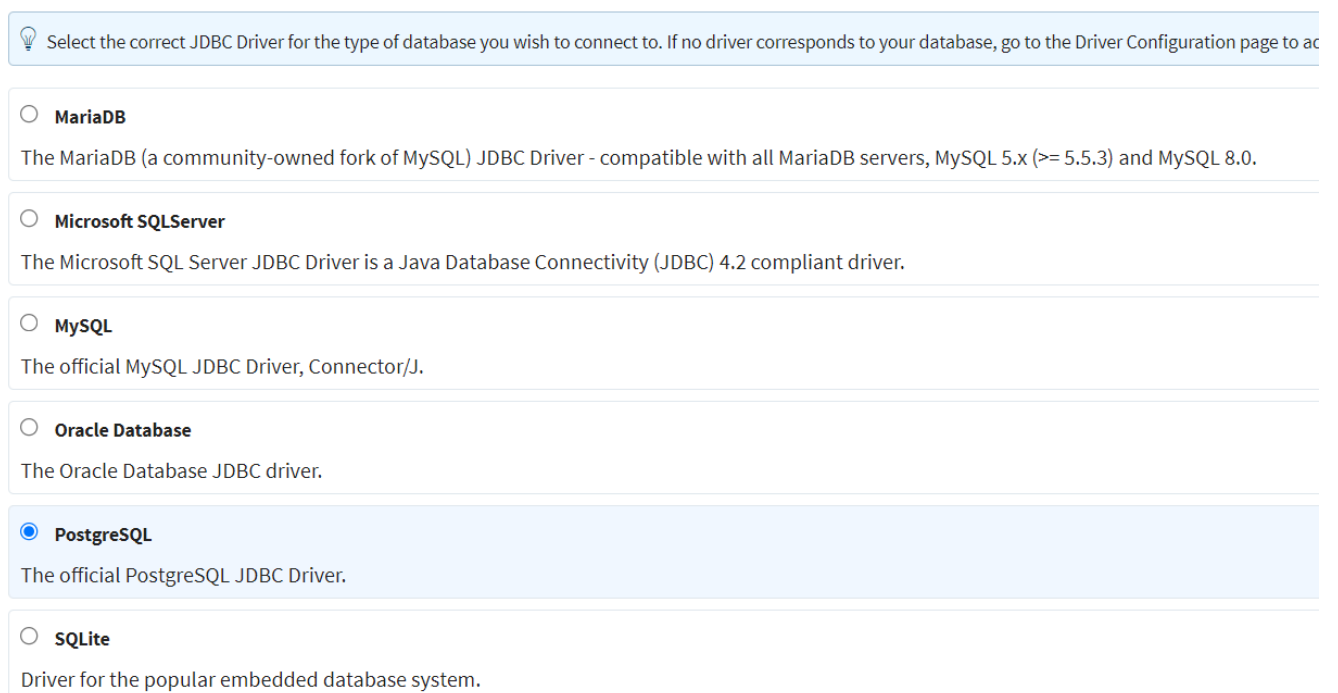


Рисунок 3.22 - Список драйверів баз даних

У полі «name» вкажемо назву для бд, у полі «Connect URL» - адресу бази даних (рис. 3.23).

Main Properties	
Name	<input type="text" value="materialDB"/> Choose a name for this database connection.
Description	<input type="text"/>
JDBC Driver	PostgreSQL The JDBC driver dictates the type of database that this connection can connect to. It cannot be changed once created.
Connect URL	<input type="text" value="jdbc:postgresql://localhost:5432/material"/> The Connect URL is JDBC-driver specific. It usually contains the address of the machine that the database is running on. The format of the PostgreSQL connect URL is: <input type="text" value="jdbc:postgresql://host:port/database"/> With the three parameters (in bold) host : The host name or IP address of the database server. port : The port that the database server is running on. PostgreSQL default port is 5432 . database : The name of the logical database that you are connecting to on the PostgreSQL server.
Username	<input type="text" value="postgres"/>
Password	<input type="password" value="....."/>
Password	<input type="password" value="....."/> Re-type password for verification.

Рисунок 3.23 - Властивості бази даних

В результаті бачимо, що базу даних PostgreSQL було підключено до Ignition Gateway (рис. 3.24).

✓ Successfully created new Database Connection "materialdb"

Name	Description	JDBC Driver	Translator	Status	
NewConnection		PostgreSQL	POSTGRES	Valid	<input type="button" value="delete"/> <input type="button" value="edit"/>
Sample_SQLite_Database		SQLite	SQLITE	Valid	<input type="button" value="delete"/> <input type="button" value="edit"/>
materialdb		PostgreSQL	POSTGRES	Valid	<input type="button" value="delete"/> <input type="button" value="edit"/>

Рисунок 3.24 - Додана база PostgreSQL

Запустимо Ignition Designer та оберемо пункт «Named Queries» (рис. 3.25).

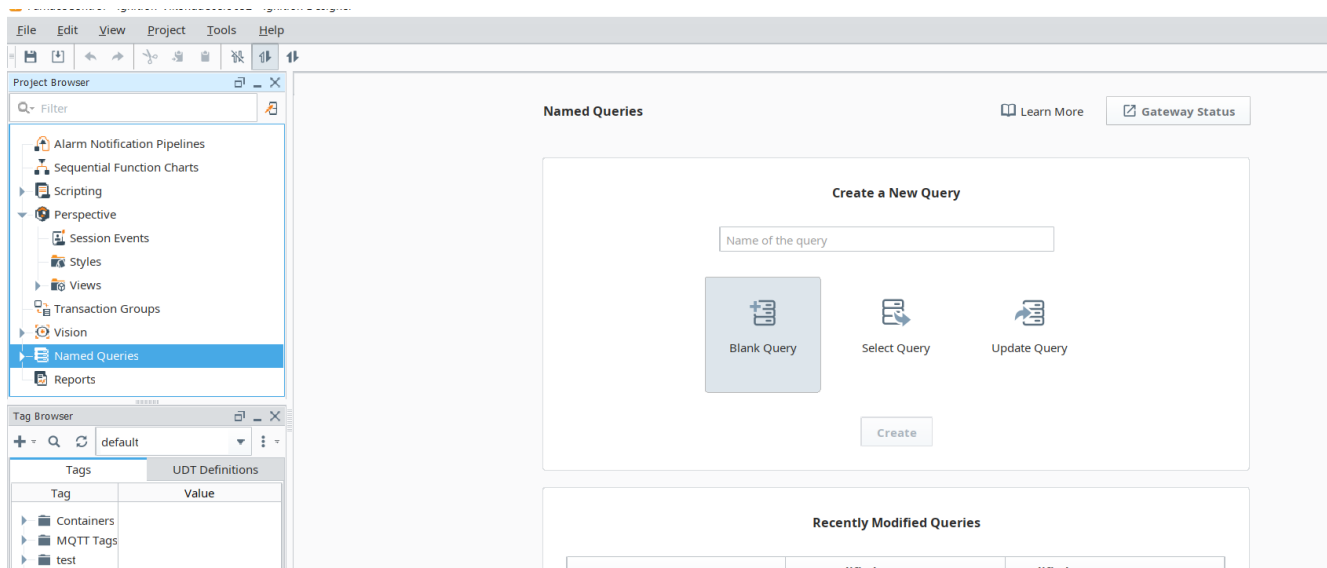


Рисунок 3.25 - Gateway Designer

Створимо SQL-запит. У налаштуваннях запиту оберемо пункт «Authoring». У ньому вкажемо підключення до бази даних Postgres (рис. 3.24) та тип запиту.

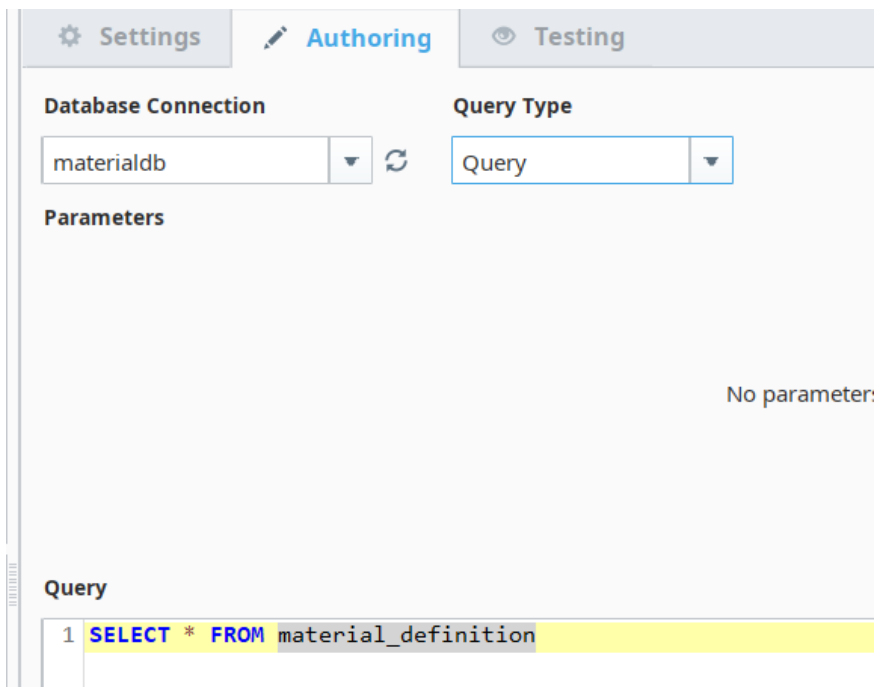


Рисунок 3.26 - Налаштування запиту

У вкладці «Testing» виконаємо SQL-запит для перевірки його роботи (рис. 3.27).

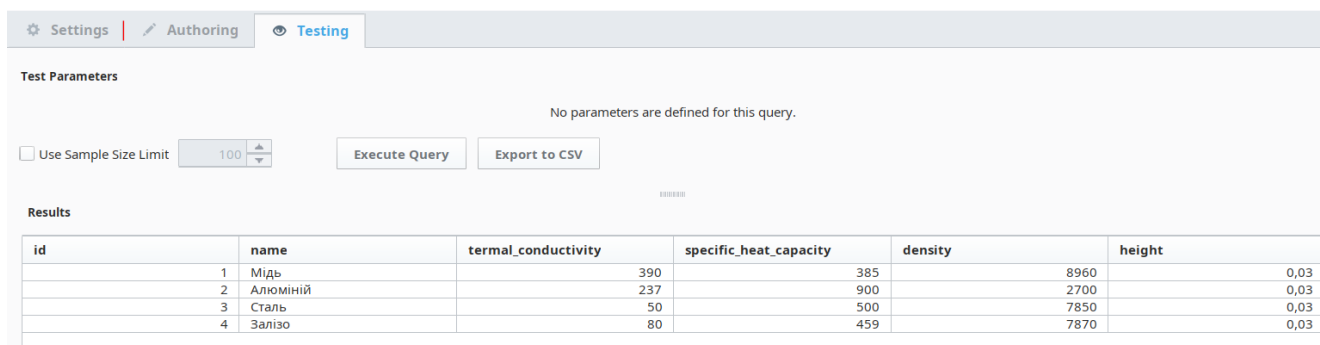


Рисунок 3.27 - Перевірка роботи SQL-запиту

3.3.3 Встановлення зв'язку між ПЛК S7-1200 та Ignition Gateway з використанням технології OPC UA

У Ignition Gateway оберемо пункт «OPC Connection». На екрані побачимо OPC-підключення (рис. 3.28).

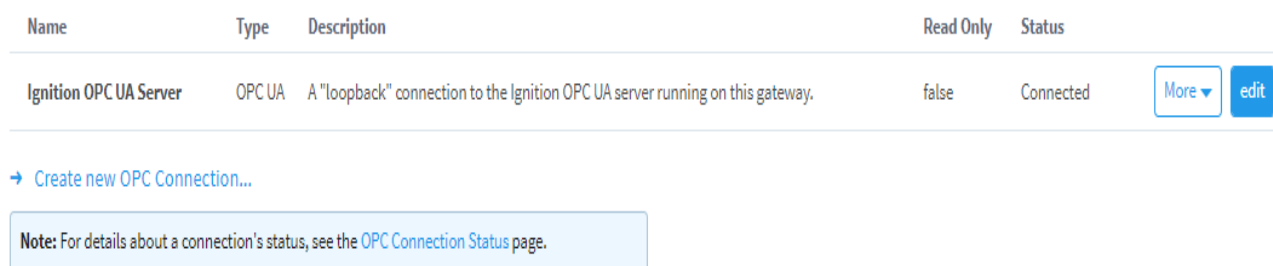


Рисунок 3.28 - Список OPC-підключень

Додамо нове OPC-UA підключення (рис. 3.29).

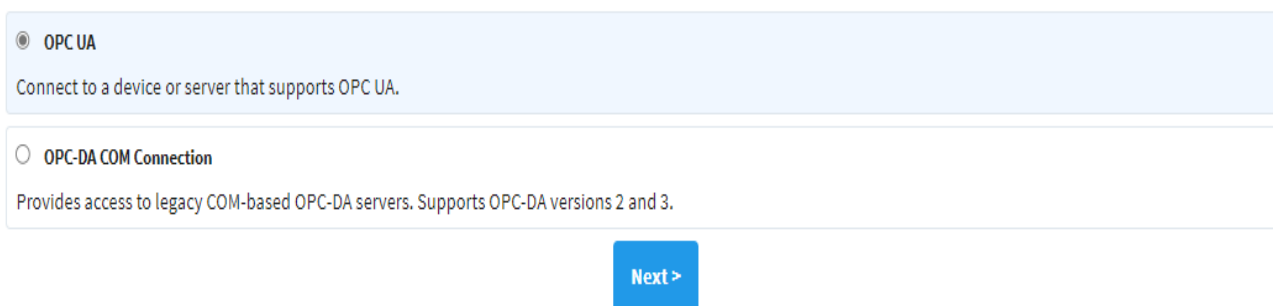


Рисунок 3.29 - Вибір типу підключення

Далі необхідно отримати URL-адресу сервера, до якого будемо підключатись. Для цього відкриємо середовище TIA Portal та скопіюємо адресу (рис. 3.30).

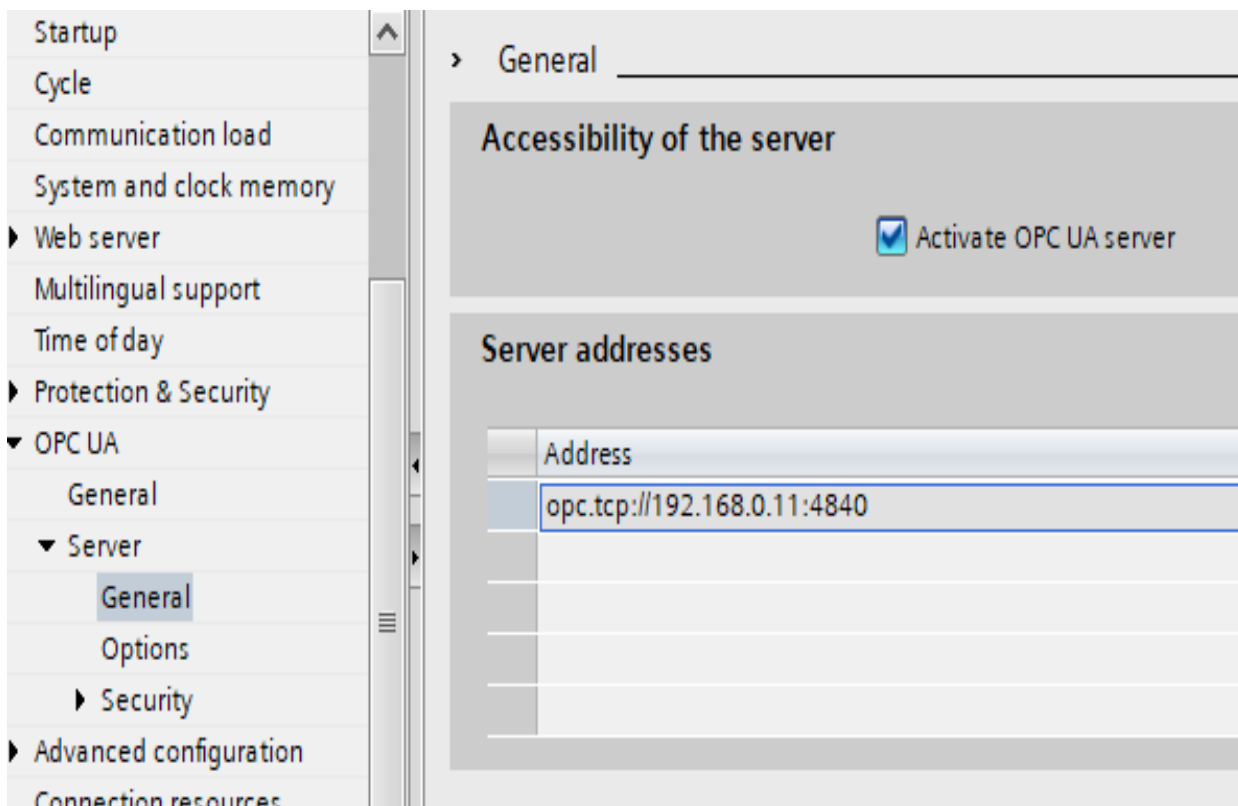


Рисунок 3.30 - Адреса для підключення

Вставимо скопійовану адресу у відповідне поле (рис. 3.31).

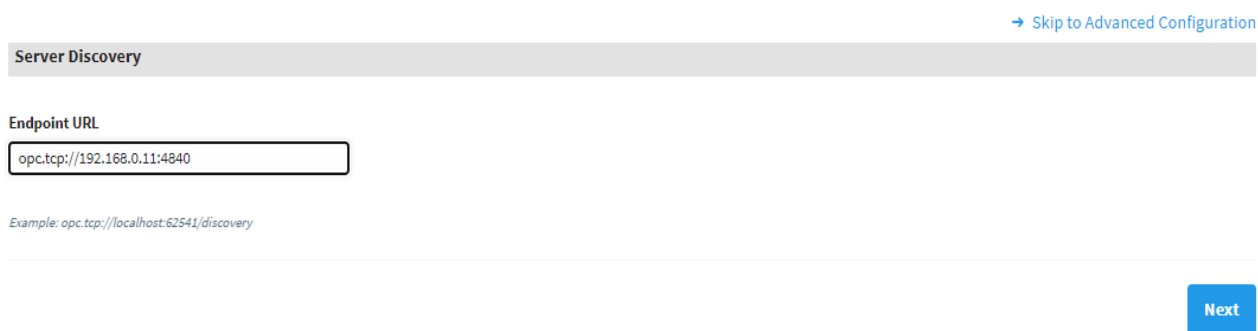


Рисунок 3.31 - Поле для адреси

На наступних кроках підтверджуємо вибір натисканням на кнопку «Next» (рис. 3.32).

The screenshot shows a four-step configuration wizard for an OPC-UA connection:

- Choose a Server:** Shows a selected server: SIMATIC_S7-1200.OPC-UA.Applications:PLC_2 with endpoint opc.tcp://192.168.0.11:4840.
- Select Endpoint:** Shows three options for security policy and mode. The first option (None/None) is selected.
- Manage Certificate:** Shows certificate details for PLC-2/OPCUA-1, including organization (Siemens), country (DE), and expiration date (Sat Nov 21 00:00:00 EET 2037). The "Trust certificate?" option is set to "No".
- Confirm Connection Settings:** Shows the final configuration: Discovery URL: opc.tcp://192.168.0.11:4840, Endpoint URL: opc.tcp://192.168.0.11:4840, Endpoint Host Override: (empty), Endpoint Security Mode: None, Endpoint Security Policy: None.

Рисунок 3.32 - Підтвердження вибору

В результаті було створено нове OPC-UA підключення (рис. 3.33).

✓ Successfully created new OPC Connection "SIMATIC.S7-1200.OPC-UA.Application:PLC_2"

Name	Type	Description	Read Only	Status	
Ignition OPC UA Server	OPC UA	A "loopback" connection to the Ignition OPC UA server running on this gateway.	false	Connected	More edit
SIMATIC.S7-1200.OPC-UA.Application:PLC_2	OPC UA		false	Connected	More edit

→ [Create new OPC Connection...](#)

Note: For details about a connection's status, see the [OPC Connection Status](#) page.

Рисунок 3.33 - Створене OPC-UA підключення

У TIA Portal створимо інтерфейс сервера, у який додамо теги, значення яких будемо зчитувати або записувати (рис. 3.34).

Browse name	Node type	Access level	Local data	Data type
Diploma interface	Interface	---		
Axis_1_ActualPosition	REAL	RD	"Axis_1"."ActualPosition"	
Axis_2_ActualPosition	REAL	RD	"Axis_2"."ActualPosition"	
Axis_3_ActualPosition	REAL	RD	"Axis_3"."ActualPosition"	
Axis_1_Error	BOOL	RD	"Axis_1"."StatusBits"."Error"	
Axis_2_Error	BOOL	RD	"Axis_2"."StatusBits"."Error"	
Axis_3_Error	BOOL	RD	"Axis_3"."StatusBits"."Error"	
Data_block_1_OnloffD1	BOOL	RD/WWR	"Data_block_1"."On/off D1"	
Data_block_1_OnloffD2	BOOL	RD/WWR	"Data_block_1"."On/off D2"	
Data_block_1_OnloffD3	BOOL	RD/WWR	"Data_block_1"."On/off D3"	
Data_block_1_angle D1	INT	RD/WWR	"Data_block_1"."angle D1"	
Data_block_1_angle D2	INT	RD/WWR	"Data_block_1"."angle D2"	
Data_block_1_angle D3	INT	RD/WWR	"Data_block_1"."angle D3"	
Data_block_1_home D1	BOOL	RD/WWR	"Data_block_1"."home D1"	
Data_block_1_home D2	BOOL	RD/WWR	"Data_block_1"."home D2"	
Data_block_1_home D3	BOOL	RD/WWR	"Data_block_1"."home D3"	
Data_block_1_move D1	BOOL	RD/WWR	"Data_block_1"."move D1"	
Data_block_1_move D2	BOOL	RD/WWR	"Data_block_1"."move D2"	
Data_block_1_move D3	BOOL	RD/WWR	"Data_block_1"."move D3"	
Data_block_1_reset D1	BOOL	RD/WWR	"Data_block_1"."reset D1"	
Data_block_1_reset D2	BOOL	RD/WWR	"Data_block_1"."reset D2"	
Data_block_1_reset D3	BOOL	RD/WWR	"Data_block_1"."reset D3"	
Data_block_1_stop D1	BOOL	RD/WWR	"Data_block_1"."stop D1"	
Data_block_1_stop D2	BOOL	RD/WWR	"Data_block_1"."stop D2"	
Data_block_1_stop D3	BOOL	RD/WWR	"Data_block_1"."stop D3"	
RightSensor	BOOL	RD/WWR	"RightSensor"	
LeftSensor	BOOL	RD/WWR	"LeftSensor"	
Start_AUT	BOOL	RD/WWR	"Start_AUT"	
Stop_AUT	BOOL	RD/WWR	"Stop_AUT"	
Switch_AUTIMAN	BOOL	RD/WWR	"Switch_AUTIMAN"	

Рисунок 3.34 - Інтерфейс сервера у TIA Portal

У Ignition Designer перейдемо до браузера тегів (рис. 3.35) для того, щоб додати теги з рис. 3.34.

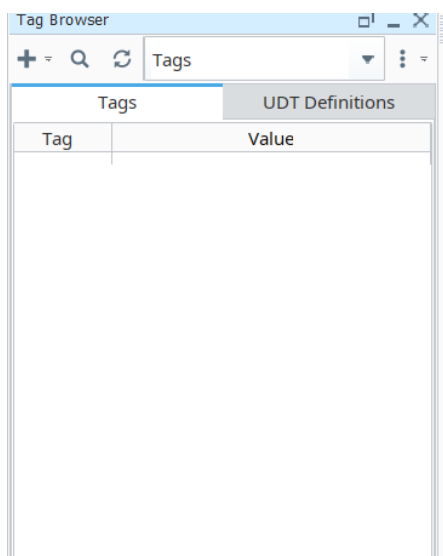


Рисунок 3.35 - Браузер тегів

Для додавання тегу виконаємо наступні дії (Рисунок 3.36):

- У полі «Name» вкажемо ім'я тегу;
- У полі «Value Source» вкажемо тип тегу «OPC»;
- У полі «Data Type» вкажемо тип даних тегу відповідно до рис. 3.34;
- У полі «OPC Server» вкажемо назву серверу (Рисунок 3.33);
- У полі «OPC Item Path» вкажемо назву тегу відповідно до рис. 3.34.

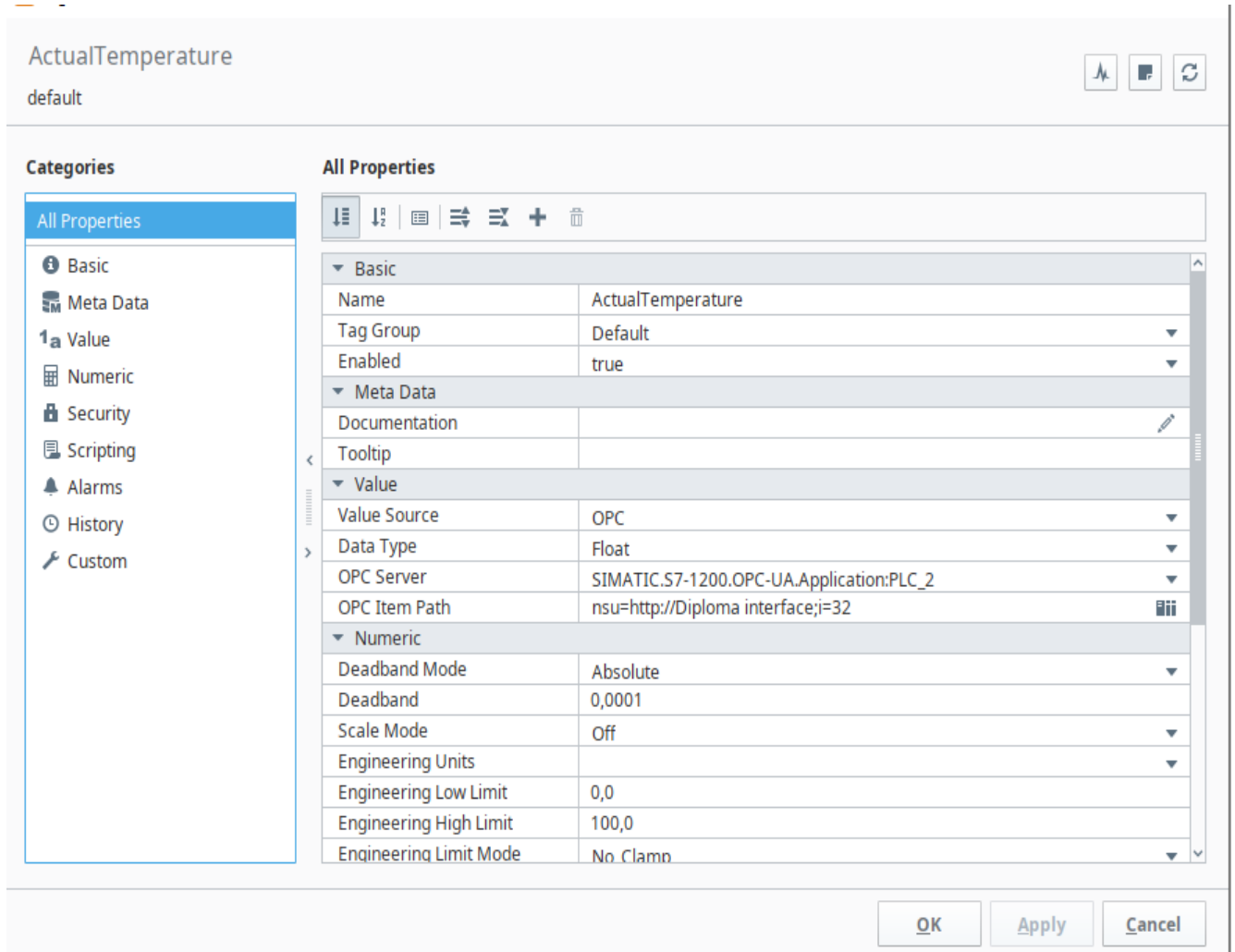


Рисунок 3.36 - Налаштування тегу перед його додаванням

Після додавання тегів пересвідчимось, що з ними є зв'язок. Для цього подивимось на тег «ActualTemperature». Значення цього тегу – це поточна температура у печі нагрівання (рис. 3.37). У разі, якщо зв'язок з тегом не встановлено, замість значення буде напис «ErrorConfiguration».

Tag	Value
Stend2	
ActualTemperature	17,44
base	<input type="checkbox"/>
conveyor	<input type="checkbox"/>
cooling	<input type="checkbox"/>
heating	<input type="checkbox"/>
magnit	<input type="checkbox"/>
set temperature	0
start aut	<input type="checkbox"/>
stop aut	<input type="checkbox"/>
switch autman	<input type="checkbox"/>

Рисунок 3.37 – Значення тегу "ActualTemperature"

3.3.4 Програма керування ПІД-регулятором

У TIA Portal ПІД-регулятор рекомендовано використовувати у блоці типу Cyclic interrupt (рис. 3.38). Це пов'язано з наступними причинами:

- Стабільність алгоритму PID. PID-регулятор обчислює вихідний сигнал на основі похибки (різниці між заданим значенням і вимірним значенням) з використанням коефіцієнтів P, I, D. Розрахунки залежать від часу між викликами функції регулятора. Якщо час між викликами змінний, інтегральна та диференціальна складові можуть працювати некоректно. Рівномірний інтервал часу дозволяє регулятору правильно інтегрувати та обчислювати похідні.

- Циклічний виклик. Циклічний інтервал у перериванні гарантує, що PID-регулятор виконується з однаковою частотою (наприклад, кожні 100 мс). Алгоритм отримує однакові часові проміжки для виконання обчислень, що критично важливо для точного налаштування регулятора.

- Синхронізація з іншими процесами. Використання cyclic interrupt дозволяє регулятору працювати синхронно з іншими задачами автоматки. Ви можете визначити пріоритет виклику PID-регулятора у порівнянні з іншими програмами.

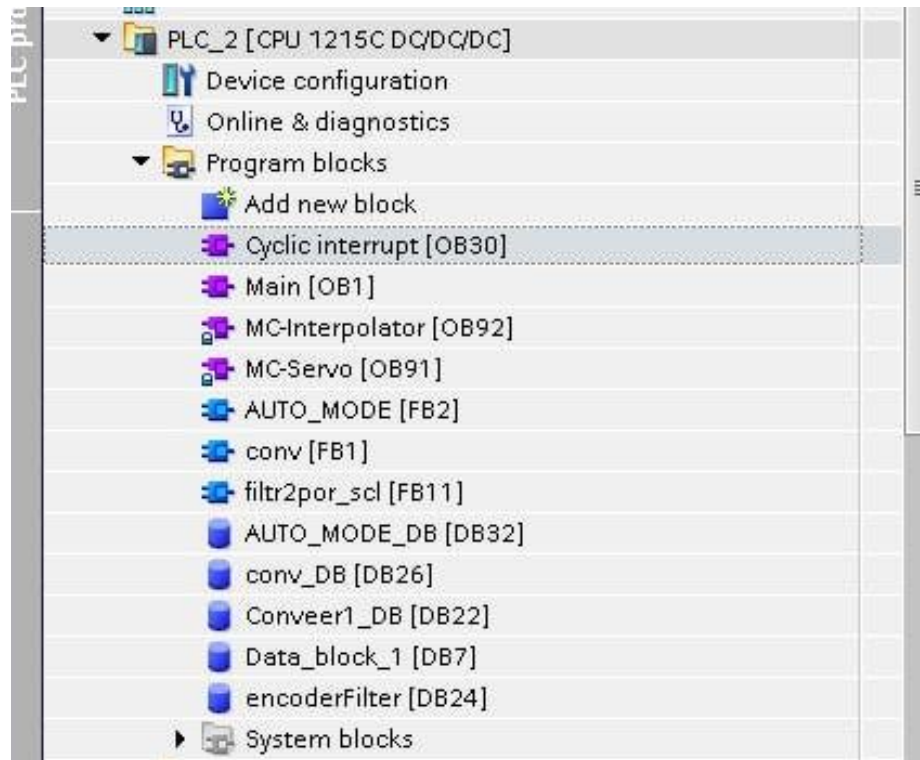


Рисунок 3.38 - Блок Cyclic interrupt

Програма керування ПІД-регулятора зображена на рис.3.39.

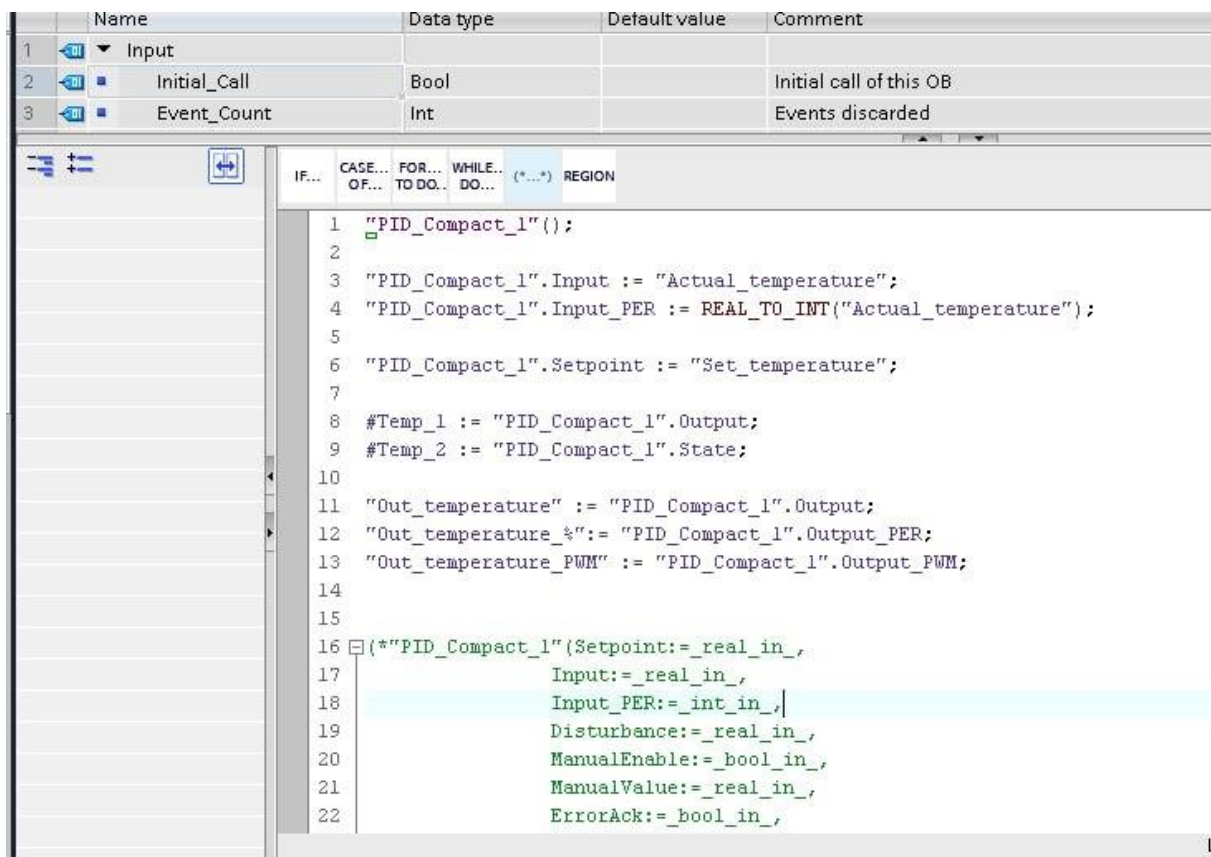


Рисунок 3.39 - Програма керування ПІД-регулятором

Програма працює наступним чином:

- Відбувається процес ініціалізації ПІД-регулятора;
- Задається значення уставки;
- На вхід ПІД-регулятора надходить поточне значення температури.
- На вихід ПІД-регулятора надходить ШІМ-сигнал;

Перед тим, як сигнал потрапить на вхід регулятора, сигнал оброблюється блоками нормування та масштабування (рис. 3.40).

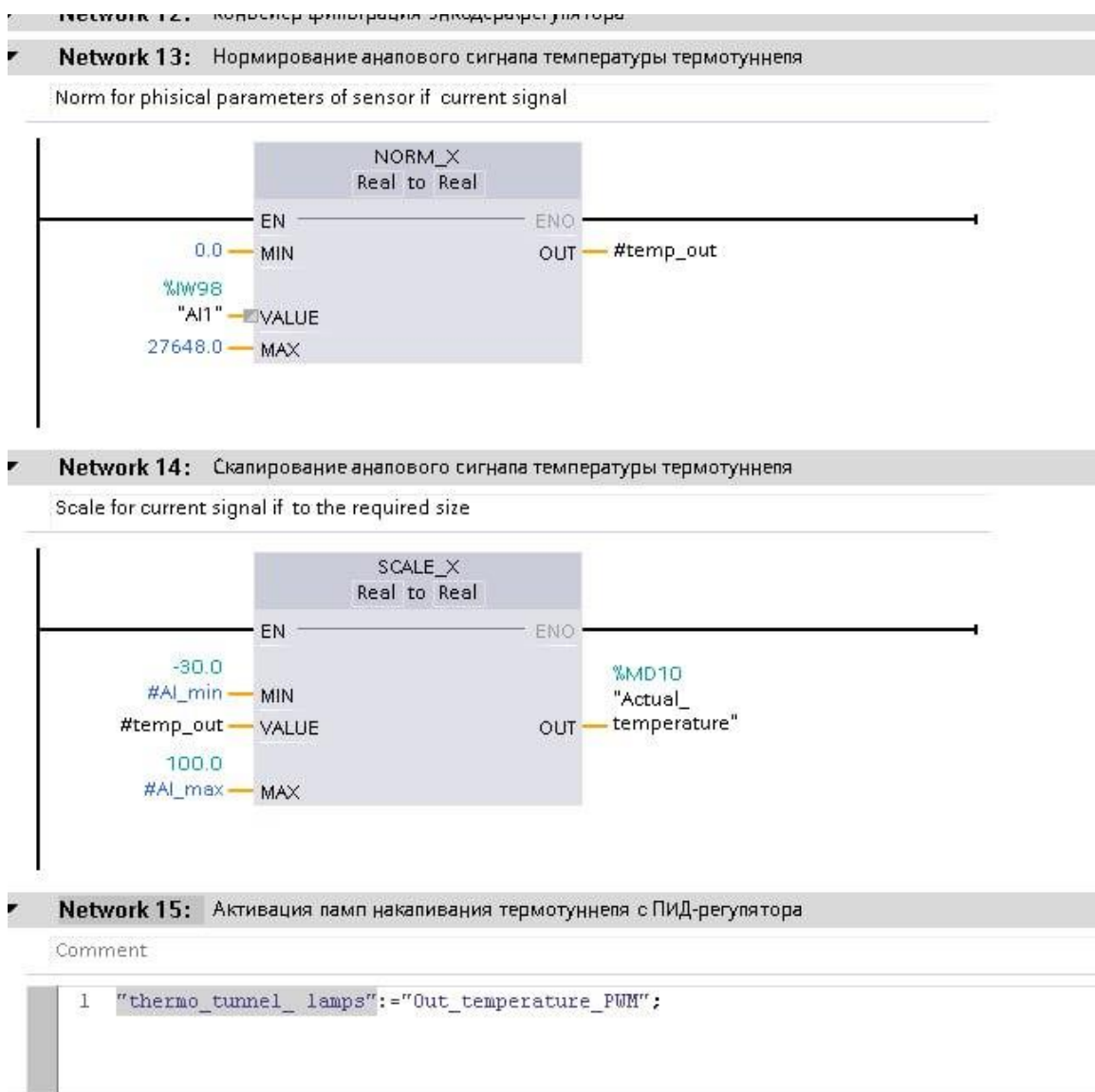


Рисунок 3.40 - Блоки для нормування та масштабування сигналу

Для налаштувань ПІД-регулятора у дереві проєкті перейдемо у папку «Technology objects», розгорнемо об'єкт «PID_compact_1» та перейдемо у вкладку «Configuration» (рис. 3.41).

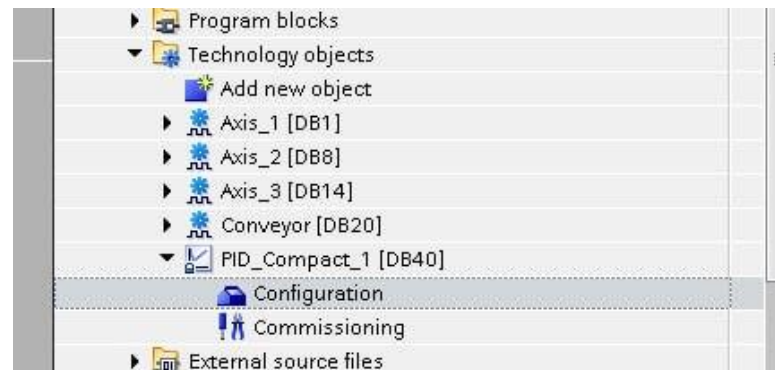


Рисунок 3.41 - Вкладка для конфігурації ПІД-регулятора

У базових налаштуваннях оберемо фізичну величина для регулювання та вкажемо одиниці вимірювання. Регулятор буде працювати у автоматичному режимі.

Також вкажемо параметри входу та виходу, а також значення уставки (рис. 3.42).

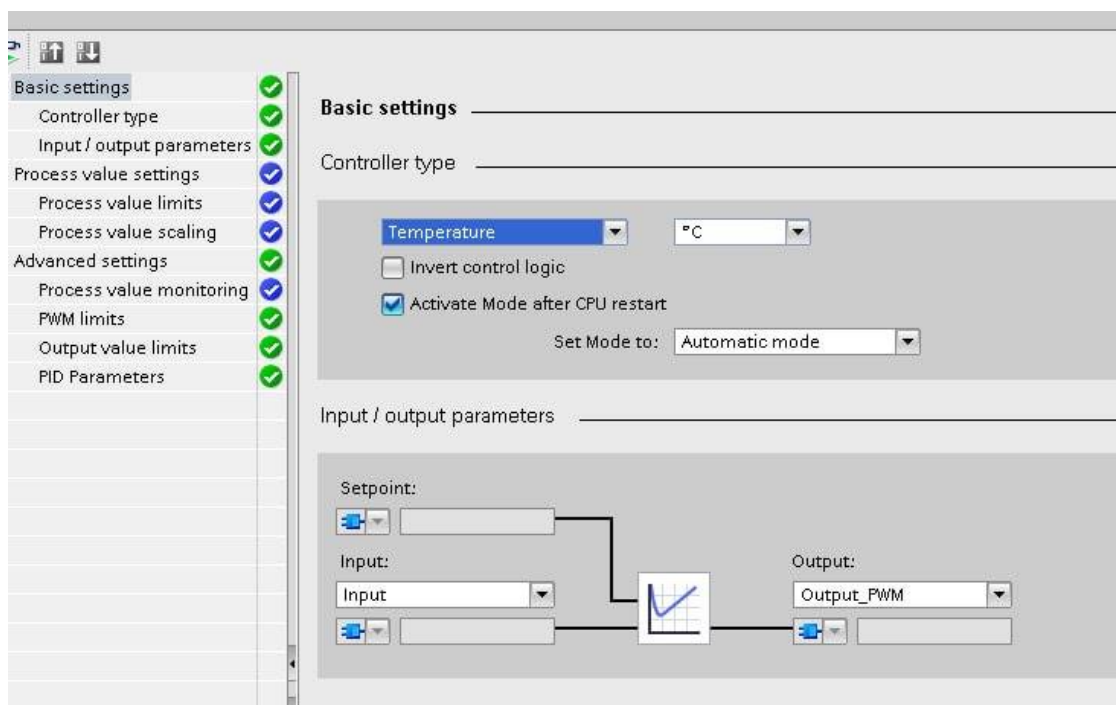


Рисунок 3.42 - Базові налаштування

У налаштуваннях значення вимірюваної величини задамо верхню та нижню границі. У разі, якщо вимірювана величина буде за межами, ПІД-регулятора повідомить про помилку (рис. 3.43).

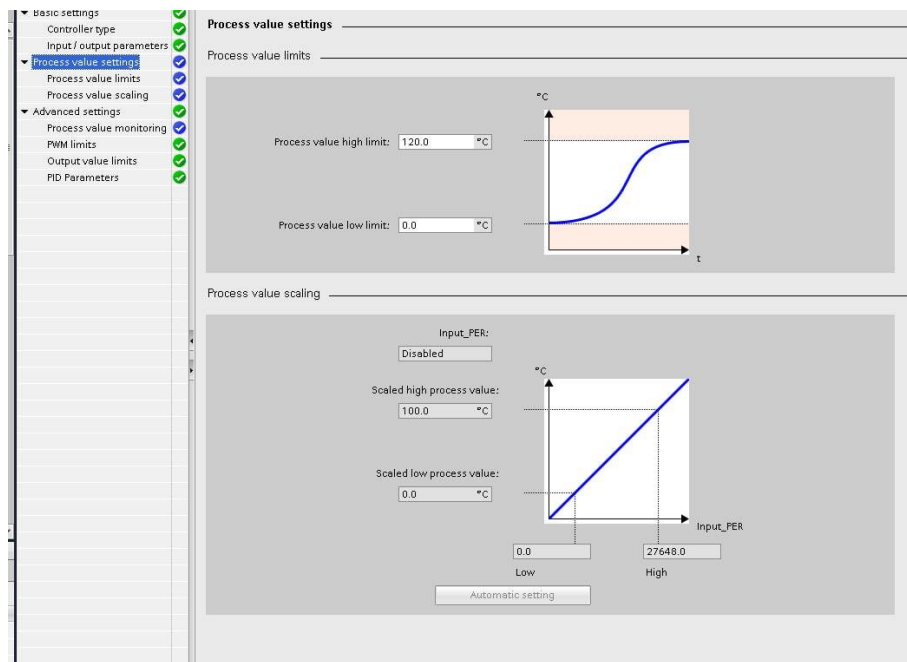


Рисунок 3.43 - Налаштування вимірюваної величини

Задамо мінімальний час включення/виключення для ШІМ-модуляції (рис. 3.44).

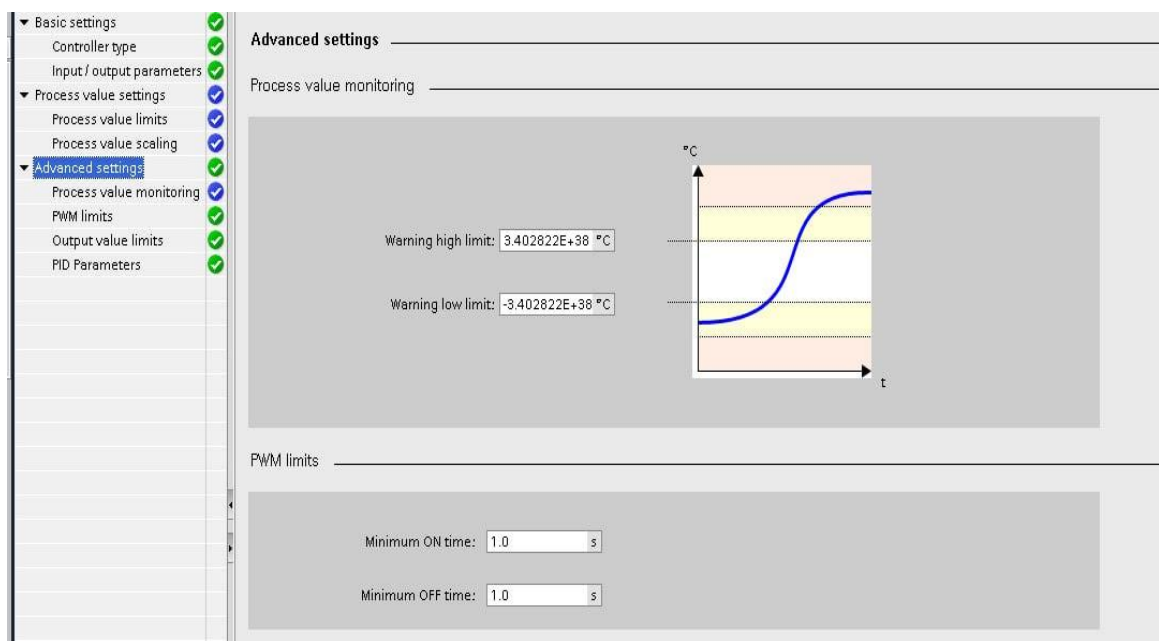


Рисунок 3.44 - Налаштування ШІМ-модуляції

У налаштуваннях вихідного сигналу задамо межі та реакції на значення помилки (). Також вкажемо структуру ПІД-регулятора (рис. 3.45).

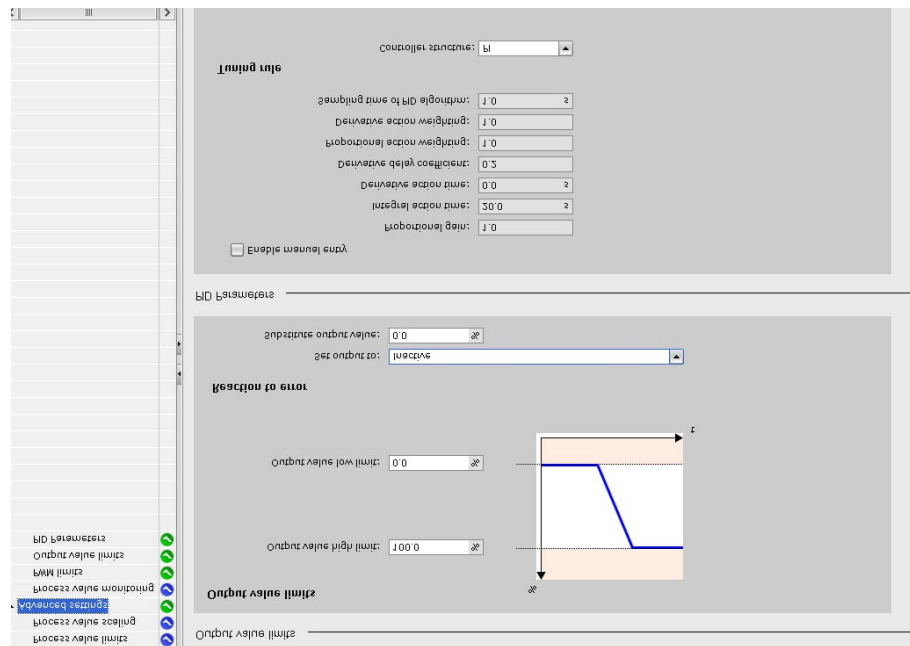


Рисунок 3.45 - Налаштування вихідної величини та параметрів ПІД-регулятора

3.3.1 Розробка панелі оператора з використанням Ignition Perspective

Панель оператора НМІ (Human-Machine Interface) – це апаратно-програмний засіб, який забезпечує взаємодію між людиною (оператором) і машиною (обладнанням, системою автоматизації чи виробничими процесами).

Розроблена панель оператора зображена на рис. 3.46. Опис основних елементів наведено у табл. 3.1.

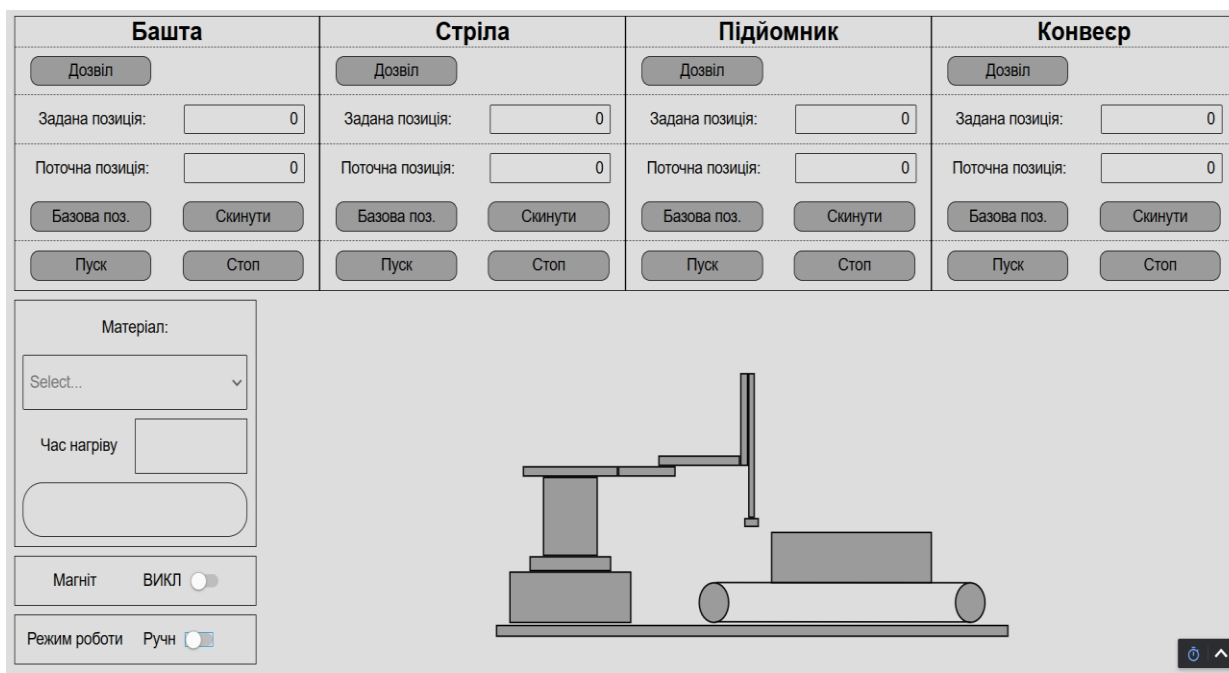


Рисунок 3.46 - Панель оператора

Таблиця 3.1 - Опис елементів

Елемент	Опис
Кнопка «Дозвіл»	Призначена для вмикання елементів
Кнопка «Базова поз.»	Призначена для встановлення поточної позиції як базової
Кнопка «Скинути»	Призначена для скидання базової позиції
Кнопка «Пуск»	Призначена приведення виконавчих механізмів в дію
Кнопка «Стоп»	Призначена для зупинки руху виконавчих механізмів
Поле «Задана позиція»	Призначене для задання кінцевої позиції для виконавчого механізму
Поле «Поточна позиція»	Відображає поточне положення виконавчого механізму
Список «Матеріал»	Призначений для вибору матеріалу для нагріву
Поле «Час нагріву»	Відображає час, необхідний для нагріву матеріалу

Числова шкала	Відображає, скільки часу пройшло з моменту початку нагріву
Перемикач для магніту	Вмикає або вимикає магніт
Перемикач режиму роботи	Для перемикання між ручним та автоматичним режимами.

Для вибору матеріалу оператору необхідно натиснути на список. Із запропонованого матеріалу необхідно обрати той матеріал, який буде прогріватись у печі.

Після вибору матеріалу у полі «Час нагріву» з'явиться числове значення часу у секундах.

На рис. 3.47 показано процес вибору матеріалу.

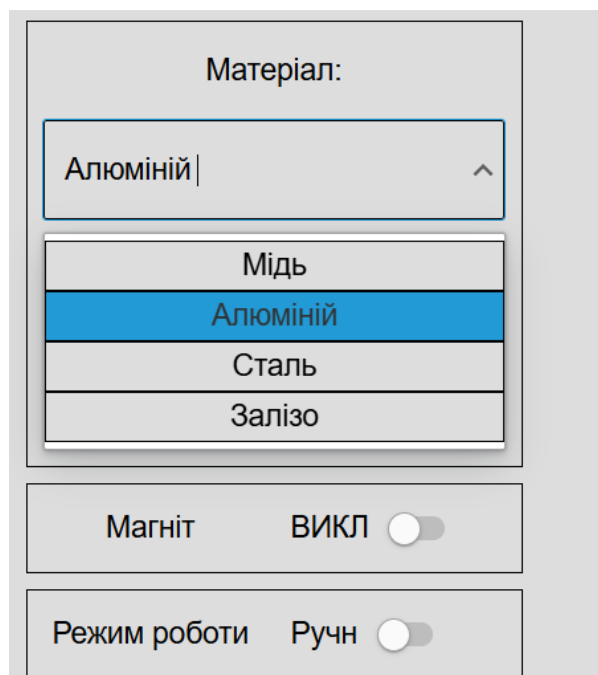


Рисунок 3.47 - Меню для вибору матеріалу

Для того, аби дані з'явились у списку, зроблено прив'язку даних до властивості списку «options» (рис. 3.48).

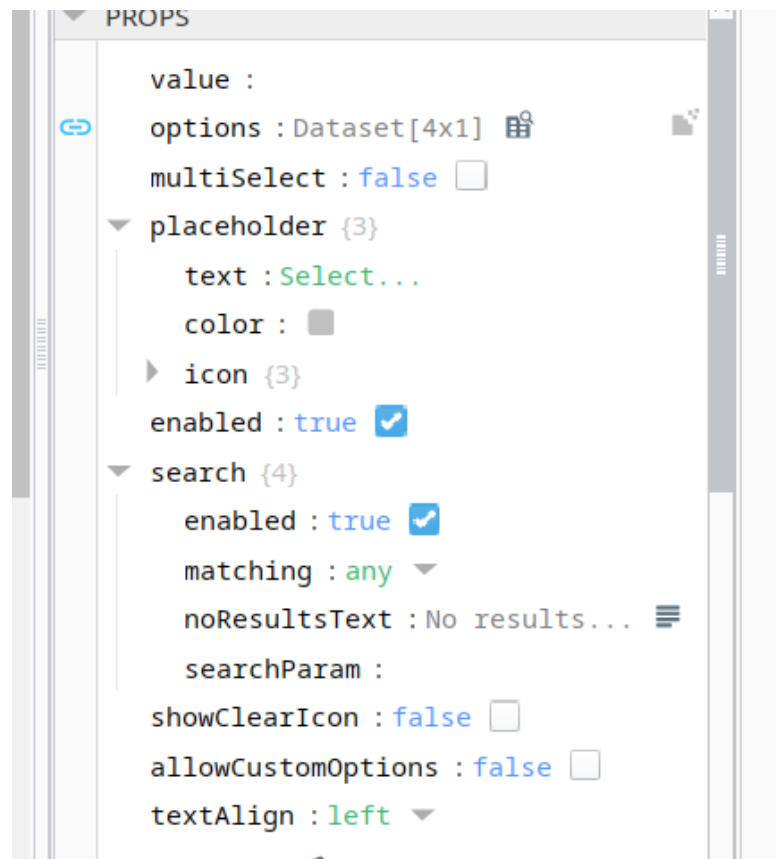


Рисунок 3.48 - Прив'язка даних до властивості елемента

У налаштуваннях прив'язки обираємо тип «Query» (SQL-запит) та вказуємо запит, який повертає назви матеріалу з бази даних (рис. 3.49). В результаті отримуємо набір даних (тип «Dataset»).

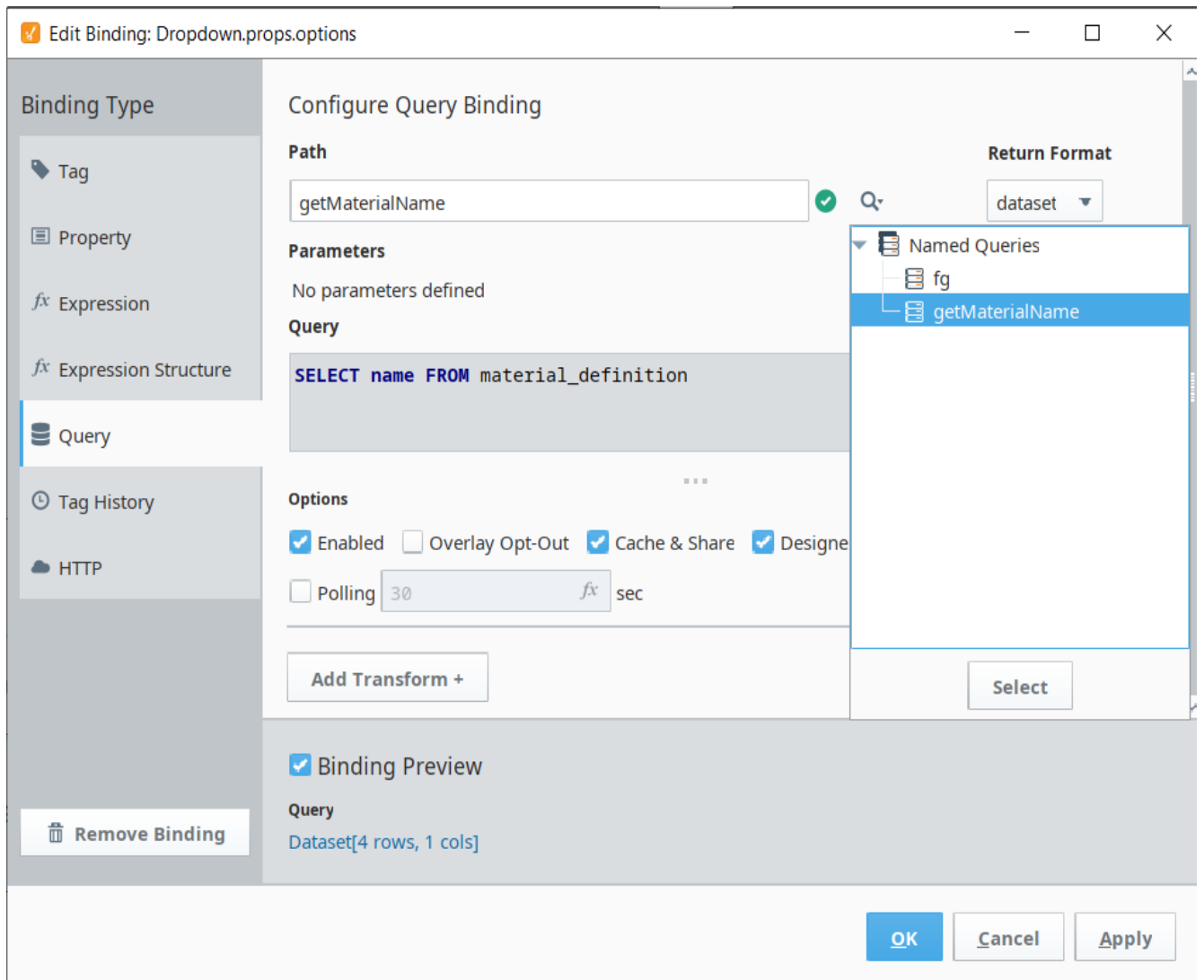


Рисунок 3.49 - Запит до бази даних

На рис. 3.50 показано елементи для керування рухом робота.

Башта	Стріла	Підйомник	Конвеєр
Дозвіл	Дозвіл	Дозвіл	Дозвіл
Задана позиція: <input type="text" value="0"/>	Задана позиція: <input type="text" value="0"/>	Задана позиція: <input type="text" value="0"/>	Задана позиція: <input type="text" value="0"/>
Поточна позиція: <input type="text" value="0"/>	Поточна позиція: <input type="text" value="0"/>	Поточна позиція: <input type="text" value="0"/>	Поточна позиція: <input type="text" value="0"/>
Базова поз. <input type="button" value="Скинути"/>	Базова поз. <input type="button" value="Скинути"/>	Базова поз. <input type="button" value="Скинути"/>	Базова поз. <input type="button" value="Скинути"/>
<input type="button" value="Пуск"/> <input type="button" value="Стоп"/>	<input type="button" value="Пуск"/> <input type="button" value="Стоп"/>	<input type="button" value="Пуск"/> <input type="button" value="Стоп"/>	<input type="button" value="Пуск"/> <input type="button" value="Стоп"/>

Рисунок 3.50 - Частина екрану для керування виконавчими механізмами робота

Видно, що елементи виглядають однаково. Для їх створення використано окреме вікно під назвою «Елемент робота» (рис. 3.51).

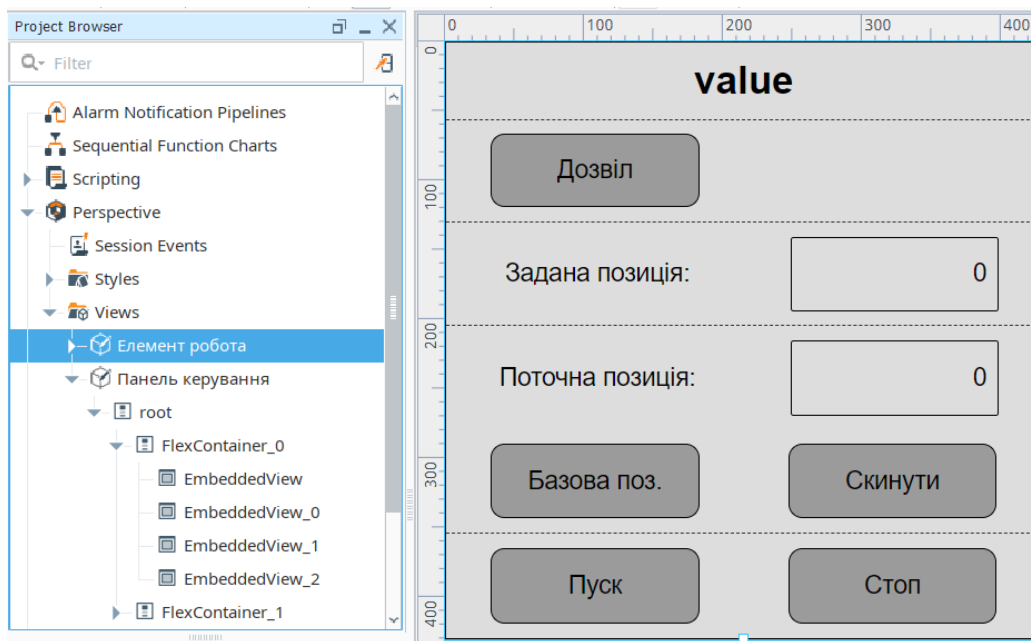


Рисунок 3.51 - Вікно "Елемент робота"

Для того, щоб використати це вікно у основному вікні «Панель керування» використано елемент «EmbeddedView». Цей елемент призначений для вбудовування вікон (рис. 3.52).

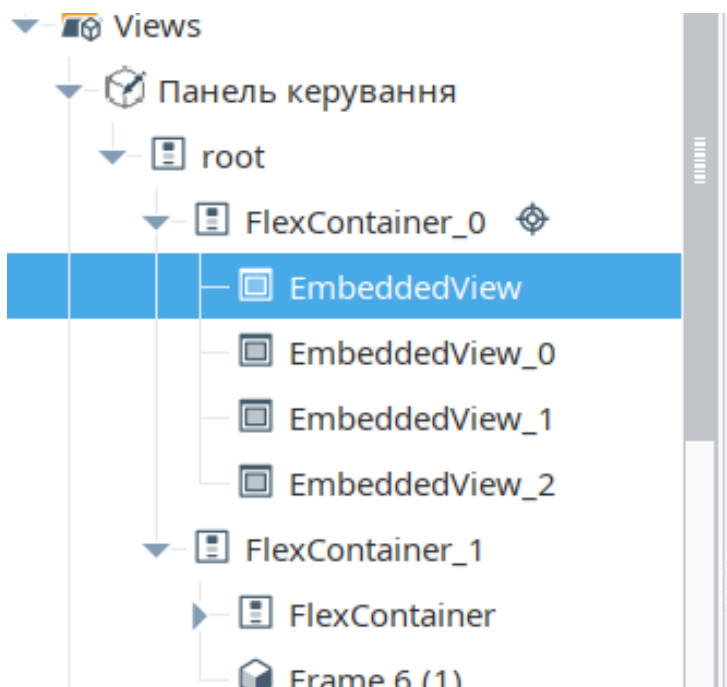


Рисунок 3.52 - Елемент "EmvdeddedView"

У властивостях елемента вказуємо шлях вікна, яке потрібно «вбудувати», а також передаємо параметри. На рис. 3.53 показано передачу значення «Стріла» параметру «element» вікна «Елемент робота».

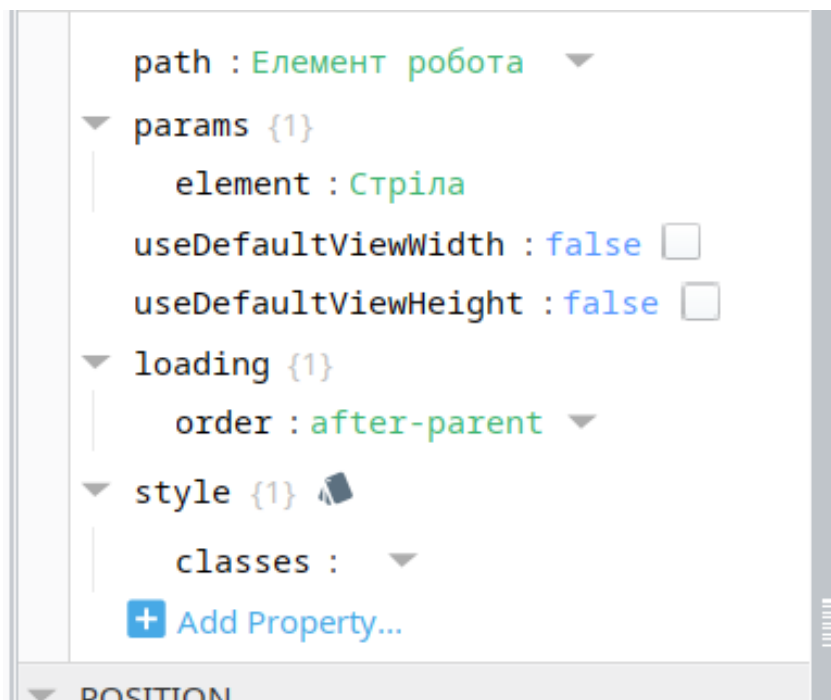


Рисунок 3.53 - Передача параметру "елемент" у вікно "Елемент роботи"

На Рисунок 3.54 показано процес запису значення у тег. Для цього використовується скрипт, який спрацьовує при активації події «onActionPerformed».

Для цього у скрипт передається шлях до тегу, а потім здійснюється запис за допомогою системної функції (рис. 3.54).

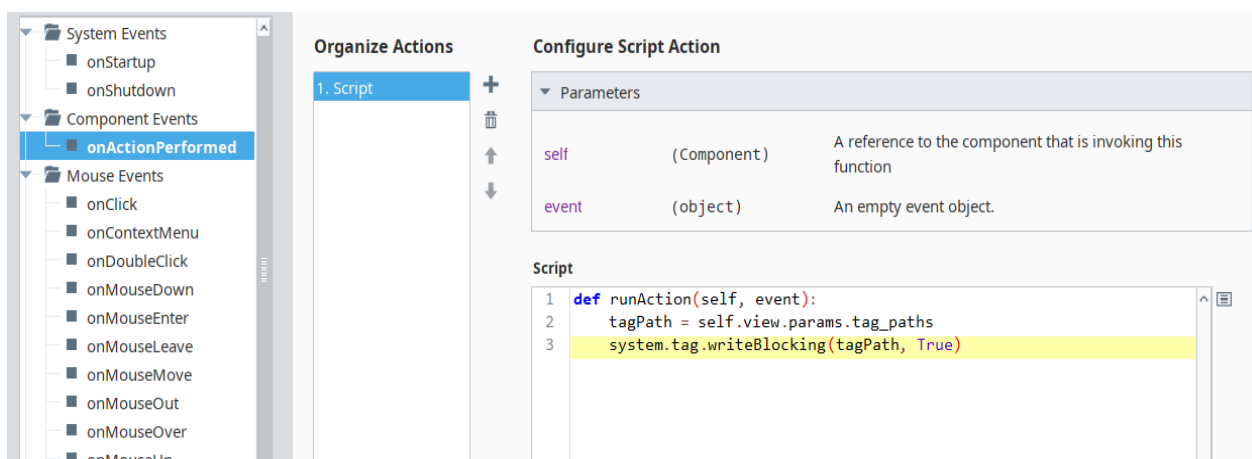


Рисунок 3.54 - Запис значення у тег

Для написів було застосовано такі налаштування тексту (рис. 3.55).

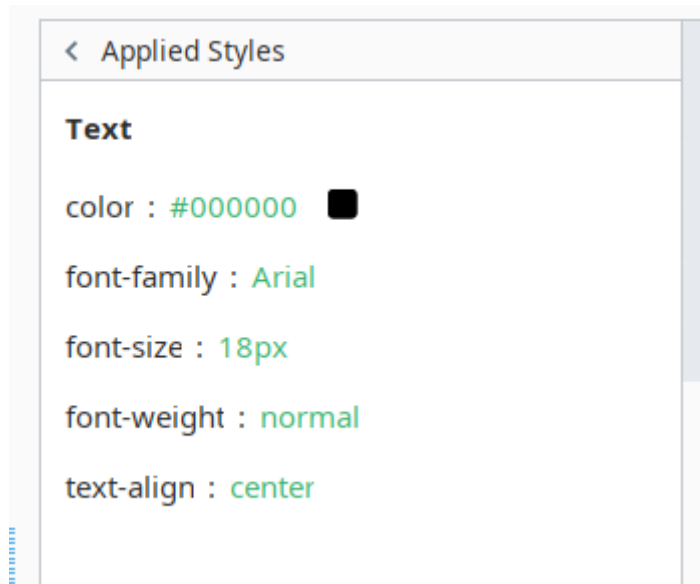


Рисунок 3.55 - Налаштування тексту

Для фону було використано такий колір (рис. 3.56).

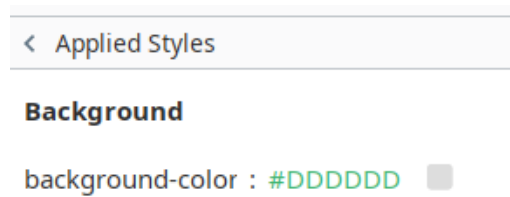


Рисунок 3.56 - Фоновий колір

Для кнопок було використано такий колір (рис. 3.57).

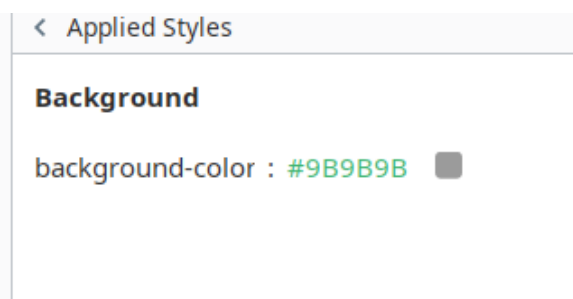


Рисунок 3.57 - Колір кнопок

Висновки за розділом:

У 3 розділі було надано опис обладнання, яке були використано для виконання роботи. 2 основні елементи – це ПЛК S7-1200 та модуль Simatic IOT2040.

Було схематично показано системну архітектуру. На схемі було відображено пристрої, які входять до системи, програмне забезпечення, яке на них встановлено, та зв'язки між цими пристроями.

Було налаштовано модуль SimaticIOT2040 за допомогою підключення до нього через SSH. У меню автозапуску модуля було включено MQTT Eclipse Broker та Node-RED.

Було створено програму у середовищі Node-RED, яка приймає дані через MQTT протокол, оброблює їх та передає так само через MQTT. Процес передачі даних було відтестовано.

Було створено базу даних PostgreSQL, у якій зберігаються дані про фізичні властивості матеріалу. Вони потрібні для роботи алгоритму, який визначає час для нагріву деталі. Було показано процес підключення бази до Ignition Gateway.

Було встановлено OPC-зв'язок між контролером S7-1200 та Ignition Gateway.

У середовищі TIA Portal було сконфігуровано програму для ПД-регулятора.

За допомогою Ignition Perspective було створено панель керування обладнанням стенду.

ВИСНОВКИ

У рамках цієї кваліфікаційної роботи було здійснено комплексне дослідження та реалізацію системи адаптивного керування температурними процесами, що використовує сучасні технології, зокрема нечітку логіку, Інтернет речей (IoT) та прогнозує керування. У процесі роботи були розглянуті традиційні підходи до керування температурою, зокрема використання ПД-регуляторів, а також новітні методи, які дозволяють підвищити ефективність та точність регулювання в умовах швидко змінюваних середовищ. Використання IoT і нечіткої логіки в контексті адаптивних систем керування значно розширює можливості системи, забезпечуючи підвищену точність та гнучкість в управлінні. Враховуючи ці аспекти, запропоновані методи можуть бути корисні для впровадження в різноманітні галузі промисловості, де критичним є точне та адаптивне керування температурними процесами.

У другому розділі роботи було здійснено глибокий аналіз та побудовано математичні моделі для ПД-регулятора, що дозволяють моделювати і точно контролювати температурні зміни. Для розрахунку теплопередачі була використана модель рівняння теплопровідності Фур'є, і для її розв'язання застосовано метод скінченних різниць. Це дозволило створити високоточну модель процесу нагріву матеріалу, яка може бути використана для розрахунку необхідного часу нагріву за заданими умовами. Розроблений алгоритм для визначення температури дозволяє отримувати надійні результати навіть при змінних зовнішніх умовах. Реалізація цього алгоритму на мові Python дозволяє інтегрувати його в більші системи автоматизованого контролю, що є важливим кроком для впровадження в реальні виробничі процеси.

У третьому розділі роботи було детально описано обладнання, що використовувалося для реалізації системи. Основними елементами системи є ПЛК S7-1200 та модуль Simatic IOT2040, які забезпечують необхідну взаємодію між різними пристроями та програмним забезпеченням. Встановлення та

налаштування цих пристроїв через протокол SSH дозволило забезпечити стабільну та надійну роботу системи. Було здійснено налаштування бази даних PostgreSQL, що дозволяє зберігати фізичні властивості матеріалу, необхідні для роботи алгоритму нагріву. Після налаштування зв'язку між Ignition Gateway та базою даних, а також між контролером S7-1200 та Ignition через OPC, було реалізовано зручний інтерфейс для керування обладнанням. Завдяки цьому вдалося забезпечити високий рівень автоматизації та інтеграції між різними елементами системи, що значно покращує ефективність роботи.

Результатом виконаної роботи стала розробка повноцінної та ефективної системи керування температурними процесами, яка використовує передові технології для точного контролю та адаптації до змінних умов. Використання таких інноваційних рішень, як IoT, нечітка логіка і прогнозуюче керування, дозволяє значно підвищити ефективність роботи виробничих процесів, зменшити енергетичні витрати та покращити стабільність роботи обладнання. Застосування цих технологій може знайти своє місце у широкому спектрі промислових підприємств, що займаються високотехнологічними процесами. Отримані результати можуть стати основою для подальших досліджень і впроваджень в галузі автоматизації промислових процесів, забезпечуючи їх більш високий рівень інтелектуалізації та автоматизації.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Yuan G., Xu J., Cheng M L. (2019) Design of an STM32-based temperature control system for electric boilers. *Electronic Testing*, 416(11): 10–13.
2. Chang J., Lee H. (2020) A study on the optimization of bottom insulation for injection molding machine platens. *International Journal of Heat and Mass Transfer*, 160: 120222.
3. Choudhary M. (1985) Three-dimensional mathematical model for flow and heat transfer in electric glass furnaces. *Heat Transfer Engineering*, 6(4): 55–65.
4. Kumar V B., Rao K S., Charan G., Kumar Y V P. (2021) Industrial heating furnace temperature control system design through Fuzzy-PID controller. In 2021 International Conference on Emerging Trends in Information Technology and Engineering (ICETITE) IEEE, 34: 1–6.
5. Oancea M., Popa F., Plotog I. (2018) Mathematical model for predicting mechanical properties and structure of the finished part and model for controlling electrical heating system based on PID adjustment algorithms. *Journal of Electrical Engineering*, 69(6): 377–382.
6. Benosman M. Model-based vs data-driven adaptive control: An overview. *Int J Adapt Control Signal Process.* 2018; 32: 753–776. <https://doi.org/10.1002/acs.2862>
7. Samad, Syed Abdul; Selvaraj, Senthil Kumar, Adaptive Control Based Electric Furnace Temperature Control Using Simulink. *Turkish Online Journal of Qualitative Inquiry*, 2021, Vol 12, Issue 3, p5226
8. KADA, Benyekhleif; ELKEBIR, Abdelkader; NEGADI, Karim; BELHADJ, Hafida; CHAOUCH, Djamel Eddine, Application of Adaptive Controller Neural Network Based on RBF NN for Temperature Control Electrical Resistance Furnace. *Przeгляд Elektrotechniczny*, 2020, Vol 96, Issue 6, p33

9. Puryear, Nathan & Zaman, Mostafa & Eini, Roja & Abdelwahed, Sherif. (2021). Design and Implementation of a Distributed Control Platform for a Smart Building Testbed. 10.1109/HPCC-DSS-SmartCity-DependSys53884.2021.00216.
10. M. Al-Amin and M. S. Islam, "Design of an Intelligent Temperature Controller of Furnace System using the Fuzzy Self-tuning PID Controller," 2021 International Conference on Electronics, Communications and Information Technology (ICECIT), Khulna, Bangladesh, 2021, pp. 1-4, doi: 10.1109/ICECIT54077.2021.9641467.
11. SIDNA, Jeddou, et al. Analysis and evaluation of communication protocols for IoT applications. In: Proceedings of the 13th international conference on intelligent systems: theories and applications. 2020. p. 1-6.
12. MALOKHVII, E.; MOLCHANOV, H. ДОСЛІДЖЕННЯ ПРОТОКОЛІВ ПЕРЕДАЧІ ДАНИХ В УМОВАХ ІНТЕРНЕТУ РЕЧЕЙ. Системи управління, навігації та зв'язку. Збірник наукових праць, 2022, 1.67: 66-74.
13. ТАБАА, Mohamed, et al. Industrial communication based on modbus and node-RED. Procedia computer science, 2018, 130: 583-588.
14. ISRA, Gourav, et al. Internet of things (iot)—a technological analysis and survey on vision, concepts, challenges, innovation directions, technologies, and applications (an upcoming or future generation computer communication system technology). American Journal of Electrical and Electronic Engineering, 2016, 4.1: 23-32.
15. GAVRILOV, Andrei, et al. Using IoT Protocols in Real-Time Systems: Protocol Analysis and Evaluation of Data Transmission Characteristics. Journal of Computer Networks and Communications, 2022, 2022.
16. FERENCZ, Katalin; DOMOKOS, József. Using Node-RED platform in an industrial environment. XXXV. Jubileumi Kandó Konferencia, Budapest, 2019, 52-63.
17. TORRES, Diogo, et al. Real-time feedback in node-red for iot development: An empirical study. In: 2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications (DS-RT). IEEE, 2020. p. 1-8.

18. CHACZKO, Zenon; BRAUN, Robin. Learning data engineering: Creating IoT apps using the node-RED and the RPI technologies. In: 2017 16th International Conference on Information Technology Based Higher Education and Training (ITNET). IEEE, 2017. p. 1-8.

19. САПОЖНИК, Д. О.; ПЛЕЧИСТИЙ, Д. Д. Використання інтернету речей для економії енергетичних ресурсів. Вісник Вінницького політехнічного інституту, 2023, 4: 39-45.

20. Тронь В. В., Маринич І. А. Методичні вказівки до виконання магістерської кваліфікаційної роботи для студентів спеціальності 174 Автоматизація, комп'ютерно-інтегровані технології та робототехніка". Кривий Ріг: Видавничий центр КНУ, 2022. 50 с.

21. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура і правила оформлення. Київ, ДП «УкрННЦ», 2015. 26с.(Інформація та документація).

22. ДСТУ 8302:2015. Бібліографічне посилання. Загальні вимоги та правила складання Київ, ДП «УкрННЦ», 2016. 16 с.(Інформація та документація).

23. ДСТУ 3582:2013. Бібліографічний опис. Скорочення слів і словосполучень в українській мові. Загальні вимоги та правила. Київ, ДП «УкрННЦ», 2013. 23 с.(Інформація та документація).

24. ДСТУ 3651.0-97 Метрологія. Одиниці фізичних величин. Основні одиниці фізичних величин Міжнародної системи одиниць. Основні положення, назви та позначення Київ, Держстандарт України, 1998. 27 с.(Інформація та документація).