

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет радіоелектроніки та телекомунікацій
(повне найменування інституту, факультету)

Кафедра мікро- та наноелектроніки
(повне найменування кафедри)

Пояснювальна записка
до магістерської роботи(роботи)
магістр
(ступінь вищої освіти)

на тему Розробка SCADA системи автоматичного регулювання
технологічних процесів на основі Arduino

Виконав: студент ІІ курсу, групи РТ-311М

Спеціальності 153 Мікро- та наносистемна техніка
(код і найменування спеціальності)

Освітня програма (спеціалізація)
Мікро- та наноелектронні прилади і пристрої

Довгаль А.О.

(прізвище та ініціали)

Керівник Погосов В.В.

(прізвище та ініціали)

Рецензент Сніжної В.Л.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Інститут, факультет ФРЕТ
 Кафедра Кафедра мікро- та наноелектроніки
 Ступінь вищої освіти магістр
 Спеціальність 153 Мікро- та наносистемна техніка
 (код і найменування)
 Освітня програма (спеціалізація) Мікро- та наноелектронні прилади і пристрої
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри Генадій СІЖНОЙ
 « _____ » _____ 20 _____ року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

Довгаля Артура Олексійовича

(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Розробка SCADA системи автоматичного регулювання технологічних процесів на основі Arduino

керівник проєкту (роботи) Погосов Валентин Вальтерович, д. ф.-м. н., професор

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затвержені наказом закладу вищої освіти від «21» листопада 2022 року № 393

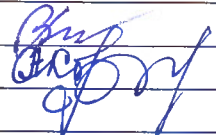
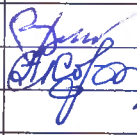
2. Строк подання студентом проєкту (роботи) 10.12.2022

3. Вихідні дані до проєкту (роботи) одноконтурна система автоматичного регулювання температурою, межі керування: від 25°C до 100 °C; SCADA-система MasterSCADA; відстань зв'язку до 1,2 км.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Визначити етапи та види забезпечення системи автоматичного регулювання технологічних процесів, проаналізувати основні структурні компоненти SCADA системи, розробити методику проєктування системи автоматичного регулювання на базі MasterScada.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Консультанти розділів проєкту (роботи)


Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконав завдання
1-3	Погосов В.В., професор каф. МіНЕ		
Нормоконтроль	Коротун А.В., доц. каф. МіНЕ		

7. Дата видачі завдання « _____ » _____ 20__ року.

КАЛЕНДАРНИЙ ПЛАН

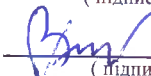
№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Збір і аналіз інформації	26.10.22 – 31.10.22	виконано
2	Систематизація літературних даних	01.11.22 – 05.11.22	виконано
3	Складання і затвердження наукового та технічного завдання	06.11.22 – 11.11.22	виконано
4	Формування та уточнення наукового завдання	12.11.22 – 14.11.22	виконано
5	Розрахункова робота	15.11.22 – 20.11.22	виконано
6	Аналіз розрахункової роботи	21.11.22 – 27.11.22	виконано
9	Оформлення ПЗ	28.11.22 – 30.11.22	виконано

Студент


(підпис)

Артур ДОВГАЛЬ
(прізвище та ініціали)

Керівник проекту (роботи)


(підпис)

Валентин ПОГОСОВ
(прізвище та ініціали)

АНОТАЦІЯ

ПЗ: 63 с., 30 рис., 1 додаток, 27 джерел.

Мета роботи: розробка SCADA системи автоматичного регулювання процесів в печах на основі мікроконтролера Arduino UNO.

Методи дослідження: лінгвістичні, технічні та інформаційні.

Методи проєктування: програмні.

Результати: У результаті дослідження написана програма, що забезпечує обмін даними між мікроконтролером і датчиком температури, автоматичне регулювання технологічним процесом на основі ПІ-регулювання і ШІМ, а також здійснено обмін даними між досліджуваною системою і OPC сервером з допомогою протоколу Modbus.

САР, SCADA, OPC, МІКРОКОНТРОЛЕР ARDUINO, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, , MODBUS, РЕГУЛЮВАННЯ, ШІМ, ФІЛЬТРАЦІЯ.

ЗМІСТ

ПЕРЕЛІК ВИКОРИСТАНИХ СКОРОЧЕНЬ.....	7
ВСТУП.....	8
1. ЛІТЕРАТУРНИЙ ОГЛЯД. ЗАГАЛЬНІ ВІДОМОСТІ ПРО АВТОМАТИЧНЕ РЕГУЛЮВАННЯ.....	9
1.1 Поняття автоматичного регулювання.....	9
1.2 Закони регулювання.....	10
1.3 SCADA-системи.....	11
1.4 Системи зв'язку для збирання технологічних параметрів.....	13
1.5 Апаратні засоби SCADA-систем.....	14
1.6 Протокол ModBus.....	15
1.7 Інтерфейси передачі у SCADA-системах.....	17
1.8 Зближення керуючих систем типу SCADA,DCS.....	18
1.9 Сучасні SCADA-системи.....	19
1.10 Огляд сучасної SCADA-системи.....	23
1.11 Вразливості SCADA-систем.....	25
1.12 Засоби захисту системи SCADA.....	27
1.13 Стандарт OPC.....	28
2. РОЗРОБКА МЕТОДИКИ СИСТЕМИ АВТОМАТИЧНОГО РЕГУЛЮВАННЯ ТЕМПЕРАТУРИ.....	30
2.1 САР температури.....	30
2.2 Оцінка якості САР.....	30
2.3 Принцип роботи САР.....	31
2.4 Пошук оптимального закону регулювання.....	35
2.4.1 Визначення передавальної функції системи.....	35
2.4.2 Вибір закону регулювання.....	38
2.5 Налаштування ПІ-регулятора.....	39
2.5.1 Метод Зіглера-Нікольса.....	39
2.5.2 Метод CHR.....	42

2.5.3	Метод ручного налаштування	45
2.6	Забезпечення коректної роботи реле та фільтрації сигналу	46
2.6.1	Широтно-імпульсна модуляція	46
2.6.2	Первинна обробка сигналу	59
3.	РОЗРОБКА МЕТОДИКИ ПРОЄКТУВАННЯ САР НА БАЗІ MASTERSCADA	51
3.1	Основні структурні компоненти SCAD-системи	51
3.2	Встановлення зв'язку між мікроконтролером та SCADA.....	52
3.3	Відображення технологічного процесу в режимі реального часу ..	54
3.4	Інтерфейс САР температури у SCADA-системі	56
	ВИСНОВКИ.....	57
	ПЕРЕЛІК ПОСИЛАНЬ.....	58
	ДОДАТОК А.....	61

ПЕРЕЛІК ВИКОРИСТАНИХ СКОРОЧЕНЬ

OPC (OLE for Process Control – Object Linking and Embedding) – набір правил і методів, що забезпечує обмін даними в системах управління та контролю.

CAP – система автоматичного керування

SCADA – програмний пакет, що забезпечує збирання, обробку та відображення даних у режимі реального часу

Мікроконтролер – мікросхема, що виконує різні функції та виконана в вигляді міні-комп'ютера.

ОУ – об'єкт управління

PLC – програмований логічний контролер (англ. Programmable Logic Controller)

RTU (ПЗО) – пристрій зв'язку з об'єктом (англ. Remote Terminal Unit)

ШИМ - широтно-імпульсна модуляція.

ТП – технологічний процес

ПЗ – програмне забезпечення

ОС – операційна система

ВСТУП

Актуальність цього проєкту полягає у розробці SCADA системи автоматичного регулювання процесів в печах на основі мікропроцесорної плати Arduino UNO, що дозволить підтримувати в системі що досліджується необхідний режим роботи без прямого втручання людини, обов'язком якого залишається тільки налаштування регулятора на певний режим та його запуск.

Arduino UNO - мікроконтролер, котрий має компактні габаритні розміри, з можливістю простого підключення різного роду датчиків і може передавати дані, використовуючи протокол Ethernet, засобом плати розширення – Shield. Має просте і зрозуміле середовище програмування для мікроконтролерів, мова програмування подібна мовами C/C++. Низька вартість та можливість прийому аналогових і цифрових сигналів від різних датчиків, підключених до платформи такого роду, управління виконавчими пристроями є основними перевагами програмованої плати. За рахунок цього мікроконтролер Arduino UNO набув широкого поширення у сфері моделювання об'єктів автоматизації, а також використовується з метою вивчення особливостей технологічного процесу в лабораторних умовах.

Метою дослідження даної роботи стоїть розробка і налаштування регулятора на базі компактного і високошвидкісного мікроконтролера Arduino UNO для управління технологічними процесами, а також зв'язок SCADA системи, яка дозволить здійснити процес збору інформації в режимі реального часу з досліджуваної системи та забезпечить процес дистанційного керування системою, на основі отриманих даних.

1 ЛІТЕРАТУРНИЙ ОГЛЯД. ЗАГАЛЬНІ ВІДОМОСТІ ПРО АВТОМАТИЧНЕ РЕГУЛЮВАННЯ

1.1 Поняття автоматичного регулювання

Автоматичне регулювання – це область автоматики, яка включає комплекс методів і засобів, що дозволяють у певний відрізок часу забезпечити заданий стан технологічного процесу з необхідною точністю, або зумовити хід цього процесу відповідно до заздалегідь заданим законом, що залежить від певних умов процесів.

Системи з використанням такого роду регулювання отримали широке практичне застосування ще наприкінці минулого століття, удосконалені їх прототипи застосовуються в різноманітних галузях промисловості, в таких як металургія, нафтогазова, машинобудівна, енергетична, також для регулювання температури та тиску.

Регулятор - автоматичний керуючий пристрій, що працює по замкненому циклу і призначений для стабілізації будь-якого параметра. [1]. За різновидом регульованого параметра їх можна розділити на регулятори тиску, рівня, температури, потужності, напруги і інші..

Автоматичний регулятор представляє собою сукупність пристроїв, приєднаних до керованого об'єкта для регулювання його вихідної величини. До виходу об'єкта приєднується вимірювальний елемент, що контролює вихідну величину, а до регульованого органу об'єкта - виконавчий елемент, при відхиленні регульованого параметра об'єкта від заданого значення регулятор відповідно до закладеного в нього закону, формує керуючий вплив на регульований орган, щоб зменшити відхилення [1].

Системи, що реалізуються на виробництві, мають широкий спектр регульованих параметрів, що ускладнює сам процес регулювання і їх розробку, тому в даній роботі проводиться дослідження над спрощеною моделлю системи.

1.2 Закони регулювання

Закон регулювання, який формує регулятор, представляє залежність між регульованим параметром і положенням регулюючого органу. Регулятори прийнято називати відповідно тому закону регулювання, що він реалізує [1].

Коротка характеристика основних законів регулювання:

1. Пропорційний (П-регулятор):

- Переваги: швидкодія;
- Недоліки: залишкове відхилення регульованого параметра;
- Особливості: може використовуватися для регулювання об'єктів без самовирівнювання і при частих відхиленнях навантаження.

2. Інтегральний (І-регулятор):

- Переваги: в що встановилося режимі значення регульованого параметра залишається постійним незалежно від зміни навантаження об'єкта;
- Недоліки: великий час регулювання;
- Особливості: не застосовується на об'єктах без самовирівнювання.

3. Пропорційно-інтегральний (ПІ-регулятор):

- Особливості: може застосовуватися як для об'єктів з самовирівнюванням, так і без самовирівнювання у тих випадках, коли необхідна велика точність регулювання при великих, але плавних змінах навантаження.

4. Пропорційно-інтегрально-диференціальний (ПІД- регулятор):

- Особливості: застосовується на об'єктах, що не допускають статичної помилки, навантаження яких змінюється часто, а також є запізнення. Регулюючий орган ПІД-регулятора переміщається

пропорційно відхиленню, інтегралу і швидкості відхилення регульованого параметра.

На виробництві широке застосування отримали релейні регулятори, у яких регулюючий орган переміщається стрибком з одного крайнього положення в інше кожен раз, коли регульований параметр досягає порогового значення. Існує два типу такого регулятора: двопозиційний, трипозиційний. Особливості двопозиційного регулятора:

- Застосовується для управління об'єктами з малим запізненням та з великою постійною часу: наприклад, для регулювання рівня води в резервуарах, для регулювання температури і вологості.
- Підтримує середнє задане значення регульованого параметра, при цьому сам параметр змінюється, роблячи незатухаючі коливання, амплітуда та період коливань залежать від зони нечутливості регулятора.

1.3 SCADA-системи

Інтерфейс між людиною та системою називають людино- машинним інтерфейс (HMI). В окремому випадку, коли HMI призначений для взаємодії з автоматизованим технологічним процесом, його називають SCADA- системою (Supervisory Control And Data Acquisition). Ця система виконує безліч функцій.

Приклад застосування OPC для мережного доступу до даних у пристроях автоматизації, можна розділити на наступні групи:

- налаштування на конкретне завдання;
- диспетчерське керування;
- автоматичне керування;

- зберігання історії процесу;
- виконання функцій безпеки;
- виконання загальносистемних функцій.

Однак основною її ознакою є наявність інтерфейсу з користувачем. Якість рішень прийнятих оператором часто впливає не лише на якість вироблених функцій, а й на житті людей, тому зрозумілість інтерфейсу, наявність підказок, блокування явних помилок оператора, є дуже важливими властивостями SCADA-системи. У розробку інтерфейсу на SCADA входить створення графічного інтерфейсу (мнемосхем, графіків, таблиць, спливаючих вікон, елементів введення команд), програмування та налагодження роботи алгоритмів системи автоматизації, налаштування систем комунікації, підключення бази даних. Застосування SCADA у системах віддаленого доступу вимагає розмежування доступу категорій користувачів до системи, захист інформації (шляхом шифрування та забезпечення секретності протоколів обміну даними).

SCADA-системи вирішують наступні завдання:

- Обмінюватися даними з «об'єктними комунікаційними пристроями» (наприклад, промисловими контролерами та платами введення/виведення) у режимі реального часу через драйвери.
- Обробка інформації в режимі реального часу.
- Логічне керування.
- Відображення інформації на екрані в зрозумілій формі для людини.
- Ведення в режимі реального часу, бази даних з технологічною документацією.
- Управління аварійними сигналами та повідомленнями тривоги.
- Підготовка та створення звітів про хід технологічного процесу.
- Забезпечення зв'язку із зовнішніми програмами. У системі управління підприємством такими переважно рівня MES.

1.4 Системи зв'язку для збирання технологічних параметрів

У ранніх SCADA-системах, що використовувалися на підприємствах водопостачання та збору стічних вод, застосовувалися орендовані телефонні пари, по одній парі на один сигнал/аларм. Однак підведення телефонних ліній до віддалених станцій влітало в копійчину, та й орендна плата була високою; до того ж, телекомунікаційні компанії неохоче погоджувалися на фіксацію окремих з'єднань у своїх комутаторах. Це спонукало SCADA-операторів на пошук інших рішень. У 1970 роках багато хто спробував перейти на радіозв'язок і негайно зіткнувся з низкою проблем: смуги частот тоді були значно вже, ніж на початку XXI століття, а правила ліцензування частот у містах по всьому світу були такими, що часто перетворювали SCADA-системи на базі радіо у нездійсненну мрію.

Ситуація спростилася після того, як у 70 роках минулого століття почався перехід з аналогової телеметрії, що функціонує за принципом частотної маніпуляції (Frequency Shift Keying/FSK), до цифрової телеметрії. Перші цифрові рішення були приватними фірмами, потім з'явилися системи на базі COTS-продуктів (Commercial Off The Shelf / готові комерційні продукти з полиці). Мікропроцесори разом з розробленими в НАСА технологіями стиснення та кодування (метод Боуза – Чоудхурі та ін.) дозволили організувати передачу на одній радіочастоті (або по одній орендованій лінії в тих випадках, коли використовувати радіо не можна було) відразу кілька алармів та аналогових величин.

Зрештою SCADA-системи стали давати своїм користувачам те, про що завжди мріяли: актуальну інформацію про те, що відбувається в масштабах всього підприємства, плюс можливість управління всім підприємством з єдиного центру. І все ж таки перші цифрові рішення були не такі надійні, як

реле, підключені до одного FSK-каналу. SCADA-системи з радіозв'язком періодично виявлялися недоступні, що могло бути обумовлено різними причинами, аж до спалахів на Сонці. Орендовані телефонні лінії також були панацеєю: їх могли порвати будівельники. Та й друковані плати сорок років тому були зовсім не такі надійні, як сьогодні – відмови компонентів та прорахунки в конструкції були звичайнісінькою справою.

В силу всього вищесказаного ранні SCADA-системи проектувалися таким чином, щоб зберегти за віддаленими станціями якнайбільше керуючих функцій. Були розроблені спеціальні дистанційні термінали (Remote Terminal Unit/RTU), здатні зберігати обмежені обсяги даних та підтримувати роботу віддалених станцій у періоди відсутності зв'язку з головною станцією. Типова SCADA-система першого покоління залишалася підключеною до імітаційної стіни, і дуже часто для обслуговування такої системи були потрібні «оператори на колесах».

1.5 Апаратні засоби SCADA-систем

Спочатку в SCADA-подібних управляючих системах приватним було все. Дистанційні термінали являли собою шасі з однією або декількома платами частнофірмової конструкції, і для організації зв'язку з головною станцією використовувалися частнофірмові технології.

Поява програмованих логічних контролерів змусило SCADA-інженерів замислитися над перевагами комерційно готового устаткування (COTS). Технологія Modbus перевела ці роздуми в практичну площину: ПЛК з підтримкою шини Modbus стали доступною та гідною заміною для приватних дистанційних терміналів. ПЛК, промислові шини та віртуальні людино-машинні інтерфейси забезпечили доступ на ринок SCADA комерційно готових продуктів та технологій, які підходять для використання як на віддалених, так і на головних станціях. Чим досконаліше ставали

програмовані логічні контролери, тим паче складну функціональність дистанційних терміналів у яких можна було реалізовувати. Чим краще, швидше та потужніше робилися комп'ютери на базі ОС Windows, тим краще, швидше та потужніше ставали програмні продукти класу SCADA/HMI.

На початку 1990 років завдяки появі комерційного програмного забезпечення для управління базами даних та збільшення обсягів пам'яті з'явилася можливість організації збору, зберігання та швидкого аналізу величезних обсягів робочих даних на базі ПЛК та ПК. Останніми шматками мозаїки стали однотегові бази даних та програмовані контролери автоматизації (ПКА, ProgrammableAutomationController), що йдуть на зміну простоватим і недостатньо гнучким ПЛК. Контролери типу ПКА розробляються спеціально під однотегові БД, що створює умови для безшовної інтеграції на технологічній платформі SCADA та забезпечення цілісності даних рівня пристроїв до архівного сховища.

1.6 Протокол ModBus

Цей протокол було розроблено у 1980-х роках фірмою Modicon для своїх контролерів і зараз він широко поширений у системах промислової автоматизації. Незважаючи на його вік, він не лише не застарів, а й демонструє попит на нього у системах автоматизації. За моделлю OSI він займає 3 рівня: фізичний, каналний та прикладний. Modbus RTU забезпечує транзакції типу «клієнт – сервер» між провідним пристроєм (master) та веденими (slaves) пристроями, пов'язаними будь-якою фізичною лінією передачі. Передбачається застосування фізичного рівня стандартів RS-232, RS-422 та RS-485. Швидкість передачі повинна витримуватися в передавачі з точністю не гірше 1%, а приймач повинен приймати дані при відхиленні швидкості передачі до 2 %. Максимальна довжина магістрального кабелю при швидкості передачі 9600 біт/с та перерізі жил понад 0,13 мм становить 1 км.

Відведення від магістрального кабелю не повинні бути довгими ніж 20 метрів. Хвильовий опір кабелю обирають по можливості більше 100 Ом, особливо при швидкості обміну більше 19200 біт/с. Згідно з протоколом, на лінії має знаходитися лише один провідний пристрій, який не має мережеву адресу. Всі ведені пристрої мають однобайтову адресу в діапазоні від 1 до 247 (адреси від 248 до 255 зарезервовані, адреса 0 – широкомовна). Таким чином, протокол дозволяє пов'язати до 247 відомих пристроїв з провідним. Ведучий віддає інструкції веденим в двох передбачених протоколом режимах – широкомовний і одноадресний.

Повідомлення Modbus передаються у вигляді кадрів, для яких відомі початок та кінець. Ознакою початку та кінця кадру є інтервал тиші, довжиною 3,5 шістнадцяткових символу (14 біт). Кадр передається безперервно. Якщо при передачі кадру виявляється пауза, довжиною понад 1,5 шістнадцяткових символу (6 біт), то приймальний пристрій вважає, що кадр містить помилку і відхиляє його. У Modbus RTU крім контролю парності виявлення помилок використовується обчислення контрольної суми CRC-16, що використовується завжди, незалежно від використання контролю парності. Значення CRC-16 обчислюється передавачем. Приймач також обчислює CRC-16 і порівнює значення зі значенням у відповідному полі кадру. Якщо дані значення збігаються, то приймач вважає, що кадр прийнято без помилки. Стопові, стартові і біти парності не беруть участь в обчисленні контрольної суми.

Поле адреси завжди містить адресу веденого, навіть у відповідях на команду, надіслану ведучим. Завдяки цьому провідний пристрій знає, від якого веденого надійшла відповідь. Поле "Код функції" говорить веденому пристрою, яку функцію потрібно виконати. Поле «Дані» може містити до 252 байт даних. Блок PDU – це основний блок кадру Modbus, який не змінюється під час використання разом із іншими протоколами транспортного і канального рівнів моделі OSI. ADU – елемент даних програми. Протоколом Modbus встановлені три категорії кодів функцій: встановлені стандартним,

що задаються користувачем і зарезервовані. Коди є числами в діапазоні від 1 до 127. Коди в діапазоні від 62 до 70 та від 100 до 110 відносяться до функцій, що задаються користувачем. Коди від 128 до 255 (старший біт дорівнює логічній одиниці) використовуються для пересилання кодів помилок у відповідних кадрів.

Код помилки	Значення
01	Помилковий код функції
02	Адреса даних, вказаний в запит, недоступний
03	Неприпустиме значення у полі даних запиту
04	Невідома помилка під час виконання запиту
05	Ведений пристрій вимагає більше часу для виконання запиту
06	Ведений пристрій вже зайнятий обробкою іншої команди
07	Використовується з кодами функцій 13 і 14
08	Пристрій при зчитуванні пам'яті виявив помилку парності

Рисунок 1.1 – Коди помилок Modbus

Генерація помилки у повідомленні відповідного пристрою генерується наступним чином: у полі «Код функції» значення старшого біта змінюється на одиницю, а в поле «Дані» вставляється один з кодів, зображених на рис. 1.1.

1.7 Інтерфейси передачі даних у SCADA-системах

У перетворенні SCADA-систем з повністю приватних фірм, якими вони були в 1970 роках, в майже повністю відкриті, якими вони стали на початку XXI століття, велика заслуга належить відкритим мережевим протоколам. Першим таким протоколом став Modbus, потім у корпоративному світі розвинувся сектор ІТ, де придумали спосіб об'єднання окремих комп'ютерів у мережі з архітектурою «клієнт-сервер». Поява технології Ethernet та її

зрощення зі стеком протоколів TCP/IP дозволило організувати переміщення величезних обсягів даних на великі відстані з використанням виключно COTS-продуктів та відкритих нечасто фірмових технологій.

Крім того, наполегливість, з якою компанія Microsoft прагнула створення універсального механізму, який повинен був дозволити додаткам різних постачальників взаємодіяти між собою, призвела до появи в індустрії SCADA ряду промислових стандартів: спочатку DCOM і OLE, потім OPC – спеціальної версії OLE для автоматизації технологічних процесів.

У сучасних керуючих системах типу SCADA зв'язок з польовими пристроями та корпоративним рівнем реалізується за допомогою Ethernet або бездротових мереж на базі технологій OPC та TCP/IP, які не прив'язані жорстко до конкретних комунікаційних протоколів та середовищ. У нових системах застосовуються сервіси Microsoft .NET і стандарт XML, які розширюють можливості технології OPC і традиційних мережевих комунікацій.

1.8 Зближення керуючих систем типу SCADA, DCS

Концепція SCADA була вироблена у пошуках способів організації управління на розподілених підприємствах, що займаються водопостачанням та збиранням стічних вод, транспортуванням нафти та газу, доставкою електроенергії тощо. Аналогами SCADA-систем для обробної промисловості, що з'явилися під впливом подібних причин і покликаними вирішувати подібні завдання, є рішення типу DCS.

Існують різні думки щодо того, як слід розшифровувати аббревіатуру DCS, проте більшість фахівців схиляються до варіанта Distributed Control System – розподілена система керування. DCS-рішення завжди були майже повністю приватними і продовжують залишатися такими до цього дня. У цьому відношенні керуючі системи DCS сильно відрізняються від керуючих

систем SCADA, які також належать до розподілених, але будуються з використанням COTS-продуктів та відкритих технологій.

Проте з часом дистанція між рішеннями двох типів скорочується. Керуючі системи на базі SCADA вбирають все більше оригінальної DCS-функціональності, включаючи локальне управління зі зворотним зв'язком, роботу з алармами, оптимізацію технологічних процесів і аналіз даних. У свою чергу, DCS-постачальники пропонують системи, що важко відрізняються від цих SCADA-аналогів, але як і раніше називаються DCS. Якщо забути про деякі особливо критичні функції, потрібні в нафтохімічних додатках, типова сьогодення DCS-система, яку пропонує класичний постачальник промислових керуючих рішень, нічим не відрізняється від типової сьогодення SCADA-системи, над якою зачарував інтегратор.

1.9 Сучасні SCADA-системи

Сучасні керуючі системи типу SCADA мають такі можливості, про які піонери SCADA-напрямки 50 років тому не могли і мріяти. рекомендацій EEMUA та ASM. Операторам доступно все, що необхідно для побудови кінцевого рішення SCADA на базі комерційного SCADA-пакета з об'єктним конфігуруванням, включаючи різні інструменти, шаблони і підказки. Використовуючи високошвидкісні з'єднання Ethernet і TCP/IP, оператори можуть працювати буквально з тисячами віддалених статусних точок, а за достатньої пропускної спроможності каналів навіть отримувати з віддалених локацій відео. У багатьох сьогоденних SCADA-системах, що використовуються в нафтовій та газовій галузях, канали зв'язку організуються на основі волоконної оптики, що забезпечує максимально можливу пропускну здатність та швидкість передачі даних.

Сьогодні оператори можуть бачити та аналізувати дані, що стосуються обслуговування та оптимізації, керувати алармами та активами та змінювати

робочі характеристики керуючої системи, не залишаючи диспетчерської. Сучасні керуючі рішення типу SCADA є відкритими та підтримують підключення веб-клієнтів, що надає їм додаткової гнучкості. У той же час, сучасні SCADA-системи повинні бути захищені як від внутрішніх, так і від зовнішніх загроз, для чого в них передбачені налаштування безпеки.

Сьогодні концепція SCADA виходить на глобальний рівень: багато користувачів мають по кілька керуючих систем у різних куточках земної кулі, де оператори говорять іншими мовами, ніж творці цих систем. Сучасні SCADA-системи повинні представляти дані в якомога простішій і зрозумілішій формі з мінімальними змінами у версіях для різних мов.

Архівне зберігання даних у сучасних керуючих рішеннях типу SCADA – це функції БД зі структурованими запитамі. Підсистема зберігання повинна вміти надавати допомогу оператору в аналізі даних, які система збирає і відображає. До складу сучасних SCADA-пакетів включаються високорівневі графічні засоби роботи з даними, що підтримують аналіз першопричин, порівняння процесів та груп, візуалізацію алармів та послідовностей подій.

Крім підтримки візуальної роботи з даними, сучасні програмні продукти класу SCADA повинні містити повністю інтегровані інструменти, що дозволяють персоналу створювати докладні звіти про те, що відбувається на польовому рівні. Сучасні SCADA-системи не тільки допомагають інженерам та операторам складати звіти, але й самі здатні генерувати звіти та спрямовувати їх за потреби на корпоративний рівень аж до зали засідань ради директорів.

Нарешті, сучасні керуючі рішення типу SCADA повинні мати повний набір функцій для управління алармами, використовуючи які інженери та оператори можуть конфігурувати аварійні повідомлення таким чином, щоб ефективно ізолювати та ідентифікувати збої в системі. Сьогодні від SCADA-пакету очікується підтримка роботи з аналоговими алармами, статусними алармами, алармами статистичного контролю виробничих процесів

(StatisticalProcessControl/SPC), а також можливість адаптації алармів та визначення алармів користувача.

Бізнес компаній, що пропонують програмні SCADA-пакети, залежить від існування невеликих інженерних фірм, що виступають у ролі інтеграторів систем керування. Найбільш далекоглядні представники спільноти інтеграторів утворили асоціацію Control System Integrators Association, покликану допомагати інтеграторам обмінюватися досвідом, порівнювати ефективність їх рішень та вирішувати питання сертифікації. Інтегратор керуючих систем – це нова професія, що має особливі риси, що відрізняють її від професій інженера та конструктора. Доступність готового COTS-обладнання для оснащення віддалених станцій та програмного забезпечення типу SCADA для створення людино-машинних інтерфейсів та реалізації іншої високорівневої функціональності призвела до того, що компанії-інтегратори все рідше працюють «по-старому» з одним-двома постачальниками. Сьогодні інтегратори можуть вибирати для своїх клієнтів найкращі продукти різних постачальників. Без широкого вибору програмних людино-машинних інтерфейсів і SCADA-пакетів ринок інтеграції управляючих систем у його нинішньому вигляді було б існувати.

Сучасні продукти класу SCADA дозволяють створювати закінчені інтегровані системи управління для роботи з усіма даними, що надходять каналами введення-виведення. Такі системи повинні підтримувати призначення глобальних тимчасових міток для всіх даних, ведення глобальної історії даних та аналіз даних, а також імпорт даних з різних БД та представлення імпортованих даних таким чином, начебто всі вони зберігалися в одній БД. Цілісність даних – це один із ключових показників якості функціонування сучасного SCADA-пакету.

Для COTS-обладнання тривалі життєві цикли нехарактерні, але SCADA-системи повинні мати тривалий життєвий цикл. На практиці це означає, що апаратні та програмні засоби керуючих систем повинні бути легко оновлюваними, оскільки тоді ці системи можна буде інтегрувати різні

технічні новинки в міру їх появи. Сучасні SCADA-системи здатні витримувати щонайменше чотири-п'ять повних модернізацій протягом життєвого циклу віддалених станцій і підконтрольних виробничих процесів, засобів та програмного забезпечення. З цього негайно випливає, що для сучасних SCADA-систем готовність до модернізації можна назвати природним станом.

Оскільки керуючим системам властиво розширюватися, а компаніям, що використовують ці системи, змінюватися і розвиватися, дуже важливо, щоб SCADA-рішення спочатку мали масштабовану архітектуру: тоді нарощування їх можливостей позначатиме лише додавання нових компонентів у існуючу структуру, а не переробку всього проекту з нуля .

Незважаючи на те, що керуючі системи типу SCADA зародилися в індустрії розподільчих мереж, а потім включили в зону свого впливу сегменти автоматизації технологічних процесів та серійного виробництва, вони успішно використовуються також для автоматизації дискретних виробництв. У зв'язку з цим сучасні SCADA-рішення повинні бути дуже надійними і передбачати резервування як введення-виводу, так і мережі.

Історично SCADA-системи мали справу лише з даними, забезпечуючи їх надходження до оперативних центрів управління або на головні станції розподільчих мереж. В останні 10 років глобалізація та конкуренція, а також пошуки нових, більш ефективних способів оптимізації виробництва, спільної розробки родовищ, управління основними фондами та витратами привели до необхідності прямого підключення SCADA-рішень до корпоративних систем організації виробництва (Manufacturing Execution System/MES). Програмні продукти класу MES перетворюють SCADA-дані у зручну форму та забезпечують їх доставку всім тим людям, яким за обов'язком служби необхідно мати уявлення про роботу їхньої системи SCADA. Ці продукти повинні відповідати стандартам ANSI/ISA88 та ANSI/ISA95 та витримати випробування на сумісність із високорівневими бізнес-системами (SAP та іншими.).

1.10 Огляд сучасної SCADA-системи

MasterSCADA - система для АСУТП, MES, завдань обліку та диспетчеризації об'єктів промисловості. Основні переваги MasterSCADA :

- єдине середовище розробки всього проекту;
- роздільне конфігурування структури системи та логічної структури об'єкта;
- відкритість та дотримання стандартів;
- інтуїтивна легкість освоєння;
- потужна тривимірна графіка та мультимедіа;
- необмежена гнучкість обчислювальних можливостей;
- об'єктний підхід;
- галерея мнемосхем з об'єктів;
- можливість стикування різних пристроїв системи управління;
- перерозподіл сигналів або алгоритмів їх обробки по окремих пристроях;
- створення розподілених за пристроями алгоритмів контролю та управління;
- доступ з будь-якого робочого місця до будь-якої інформації, що є в системі;
- роздільне конфігурування структури системи та логічної структури об'єкта;
- взаємодія з іншими програмами за допомогою сучасних технологій

(OPC, OLE, DCOM, ActiveX, OLE DB, ODBC та ін);

- використання в операторському інтерфейсі системи документів будь-якого типу та обмін даними з ними;
- необмежене розширення функціональності MasterSCADA продуктами сторонніх розробників;
- зв'язок з АСУ виробництвом;
- відкриті інтерфейси для створення користувачем будь-яких базових елементів;
- реалізація більшості дій користувача методом «перетягни і кинь»;
- контроль допустимості інформації, що вводиться;
- можливість розробки проекту в зручному порядку;
- можливість повної налагодження проекту без зв'язку з об'єктом;
- бібліотека об'ємних елементів із вбудованим індикатором рівня заповнення;
- правильні врізання одних елементів до інших виходячи з співвідношення їх діаметрів і кутів нахилу;
- об'ємний трубопровід будь-якої складності;
- імпорт зображень у будь-яких графічних форматах
- візуальне створення схеми обчислень на мові функціональних блоків (FBD);
- програмування алгоритмів технологічною процедурною мовою ST;
- бібліотека з понад 150 функціональних блоків, включаючи контроль та керування;
- Первинна обробка кожного сигналу з автоматичним контролем до 6 кордонів;
- формульні обчислення значень та подій з великою бібліотекою функцій;
- автоматична та користувальницька обробка ознак якості значень
- Автоматична індикація значень всіх обчислених сигналів

імітаційний режим з індивідуальним вибором функцій імітації сигналів;

- Можливість створення користувальницьких функціональних блоків і макроблоків.

1.11 Вразливості SCADA-системи

Вразливості можуть простежуватися на рівнях програмного забезпечення технологічних процесів (управління різними системами збору інформації за допомогою радіолокаційних систем). Для супроводу роботи різних технічних пристроїв застосовуються програмовані контролери під управлінням SCADA-систем із підтримкою промислових протоколів обміну даними. Здебільшого це протоколи CAN, HART, MODBUS. Кожен з них у поєднанні з технологіями XML, FDT та з урахуванням відомих особливостей моделі OSI можна використовувати для порушення сталого з'єднання підсистем. Для визначення вразливостей, здатних вплинути на функціонал автоматизованого управління, доцільно розглянути структуру систем інформаційного супроводу підрозділів підприємств морської спрямованості у межах прийнятої класифікації підсистем промислових підприємств.

Попередньо необхідно визначити абстрактну архітектуру системи інформаційного супроводу, яку застосовують підприємства морської спрямованості. Центральним компонентом системи є сервер OPC, який вирішує завдання управління та забезпечує трансляцію даних службового характеру. До нього підключаються на рівні клієнтських додатків інтерфейси користувача, а також підсистеми збору і зберігання даних. З іншого боку, до OPC-сервера підключені контролери управління технічними пристроями (датчики, механізми). Усі наведені частини функціонують у межах прийнятої повсюдно архітектури сервер-клієнт. Це вже створює можливість реалізації

широко поширених мережових загроз, оскільки за роки експлуатації систем з цією архітектурою нагромаджено великий досвід у сфері порушення роботи серверних та клієнтських модулів.

Особливий інтерес під час розгляду проблем безпеки промислових систем викликає протокол HART (Highway Addressable Remote Transducer). В даний час стандарт HART підтримується міжнародною організацією HCF (HART Communication Foundation). У специфікації протоколу вказані вимоги до апаратури каналів зв'язку контролерів, датчиків, систем, що управляють, а завдання його реалізації пов'язані із забезпеченням працездатності правил взаємодії мережових пристроїв на промислових об'єктах.

Установка пристроїв під керуванням протоколу, що розглядається, дає можливість спільного застосування HART-сумісних контролерів і спеціалізованого програмного забезпечення. Технологія HART передбачає, що починає обмін даними контролер. Він запитує у вигляді цифрового сигналу, який приймається датчиком. Сигнал передається в аналоговій та цифровій формі. Він фільтрується і перетворюється програмою з урахуванням похибки під час передачі в послідовність, що складається з двійкових одинадцятибітних слів, що розділяються стартовими і стоповими бітами. Задана послідовність передається через модем інтерфейсний блок.

Функціональність та масштаб систем HART може розширюватись за допомогою підключення мереж на основі інших промислових протоколів (Modbus, Profibus, Ethernet). Доступ до SCADA-систем інформаційного супроводу та управління здійснюється через підключення, орієнтоване на обробку протокольних даних сервера (HART OPC сервера).

Документи з кодом XML, що використовуються в системах стандарту HART, застосовуються як керуючі сценарії, а також для проміжного зберігання даних. Взаємодія системи з датчиками та контролерами забезпечує можливість створення запиту на зміну поля у протокольному файлі. Під час обміну інформаційними пакетами в поля подібного файлу можуть вноситись сторонні дані, які можуть містити посилання на

позасистемний код. Технологія HART дозволяє використовувати зовнішні посилання для обробки XML-документа. Це вразливість SCADA-системи, оскільки поява стороннього коду призведе систему до відмови. Таким чином, суть атаки полягає у внесенні змін до XML-коду файлів SCADA-системи.

Створимо дерево вразливостей, погроз та контрзаходів, щоб визначити найпроблемніші області SCADA-системи на адміністративному рівні.

Для побудови дерева використовуються такі параметри:

- 1) V_i - вразливості
- 2) $T_{i,j}$ - загрози, що експлуатують уразливості
- 3) $L_{i,j}$ - нестача контрзаходів для загрози

Значення параметрів:

$V1$ – відсутність захисту протоколу TCP/IP

$V2$ – відсутність політики безпеки

$T1,1$ – атаки на серверні служби АС

$T2,1$ – НСД

$T2,2$ – використання шкідливого ПЗ

$T2,3$ – витік конфіденційної інформації каналами зв'язку

1.12 Засоби захисту системи SCADA

Для усунення вразливостей системи SCADA необхідно ввести заходи протидії.

Першою необхідною дією для забезпечення безпеки даних є модернізація політики безпеки системи SCADA. Дискреційна політика дозволяє диференційовано розподіляти права доступу та обмеження, що необхідно за існуючого рівня інтеграції елементів SCADA, оскільки вона має відносно просту реалізацію відповідних механізмів захисту, і на відміну від мандатної політики з меншою ймовірністю створить колізії при відпрацюванні правил політики безпеки.

Другою дією є включення до системи антируткіту, у разі порушення цілісності політики безпеки (перевірка за контрольним правилом). Існує інший спосіб застосування антируткіт-засобів. Якщо політика безпеки не була порушена, антируткіт може включатися з періодичністю 10 хвилин. Прикладами таких програм є Gmer 1.0, AVG Anti-Rootkit, Rootkit Unhooker.

І третім є включення антивірусу для можливості перехоплення вірусу одразу після його проникнення в систему (наприклад Kaspersky, TOTAL Security).

1.13 Стандарт OPC

Стандарт OPC розроблений міжнародною організацією OPC Foundation, членами якої є понад 400 фірм, які працюють у галузі засобів автоматизації. Головна мета цього стандарту – забезпечення інтероперабельності засобів автоматизації, працюючих на різних апаратних платформах, в різних промислових мережах, створених різними виробниками. У справжній момент практично всі SCADA-системи проектуються як OPC- клієнти. Даний стандарт складається з кількох частин:

- OPC DA (OPC Data Access) – Специфікація обміну даними в реальному часі між клієнтським програмним і апаратним забезпеченням;
- OPC A&E – специфікація для звітування про події та аварійні сигнали клієнту;
- OPC HDA (Historical Data Access) – специфікація доступу до збережених даних щодо процесів;

Стандарт постійно розширюється і з'являються його нові специфікації.

OPC DA сервер забезпечує синхронний, асинхронний або підписний режим обміну даними із клієнтом. Передача даних організована за допомогою тегів. Кожен тег є змінною, що зберігає стан якогось певного параметра системи. У синхронному режимі клієнт надсилає запит тега і чекає на відповідь від сервера. В асинхронному режимі клієнт після запит відразу приступає до вирішення інших завдань, а відповідь від сервера викликає переривання роботи клієнта для обробки нових даних. В режимі підписки клієнт повідомляє серверу імена тегів, які сервер повинен буде передавати лише у разі зміни їхнього значення. Режим дозволяє мінімізувати навантаження на мережу. Клієнтська програма та OPC DA сервер можуть бути встановлені на один комп'ютер або бути рознесені в мережі (рис 1.16). У такій системі будь-який OPC-клієнт з будь-якого комп'ютера може запитувати дані з будь-якого OPC-сервера, що працює на віддаленому комп'ютері у тому ж сегменті промислової мережі. Часто OPC-сервер монополює на комп'ютері COM-порт, оскільки часто виконує актуалізацію даних. Для збільшення кількості COM-портів часто застосовують перетворювачі інтерфейсів (USB - RS-232).

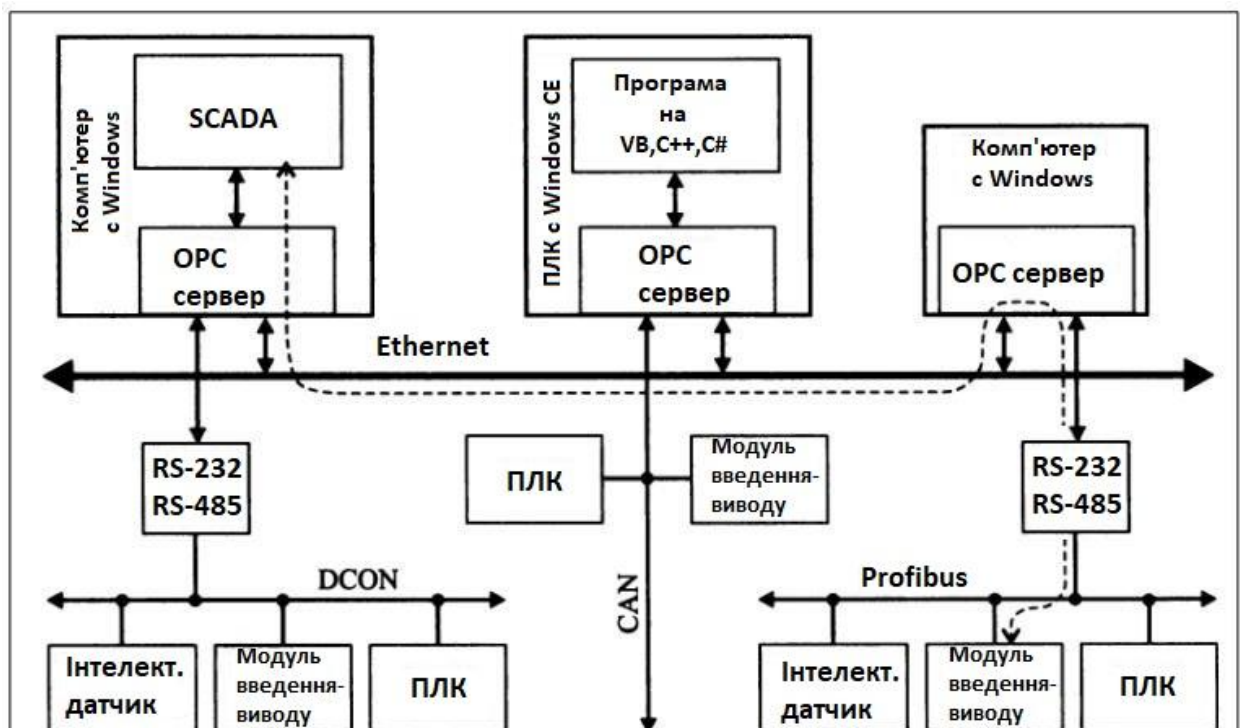


Рисунок 1.2 – Приклад застосування OPC для мережного доступу до даних в пристроях автоматизації

До недоліків технології OPC відносять її зв'язок із технологією DCOM, вихідний код якої є закритим, доступність тільки на операційних системах Windows (знову ж таки у зв'язку з DCOM), неточні повідомлення від DCOM про переривання зв'язку, непристосованість DCOM для обміну даними через мережу Інтернет.

2 РОЗРОБКА МЕТОДИКИ СИСТЕМИ АВТОМАТИЧНОГО РЕГУЛЮВАННЯ ТЕМПЕРАТУРИ

2.1 САР температури

Наочним прикладом системи автоматичного регулювання може служити модель САР температури яка зображена на рис. 2.1, де регульованим об'єктом є електропіч, вимірювальним елементом - датчик температури, перетворюючим елементом – реле, а виконавчим - мікроконтролер.

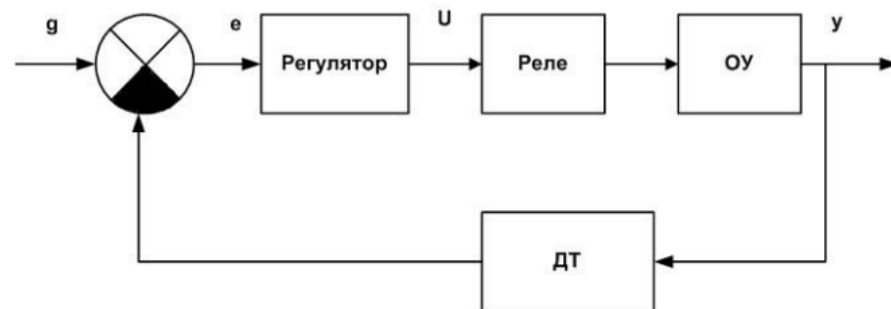


Рисунок 2.1 – Загальна структурна схеми САР температури, де ОУ - об'єкт управління, ДТ - датчик температури, g - керуючий вплив, e – сигнал неузгодженості, U – вихідна величина регулятора, y - регульована величина.

2.2 Оцінка якості САР

Оцінка якості процесу регулювання є основною складовою аналізу якості роботи стійких САР, тому для різних методів регулювання використовуються такі показники:

1. Встановлене відхилення, під яким розуміється значення регульованої величини після закінчення скільки завгодно великого відрізка часу після початку перехідного процесу, тобто $y_{уст} = \lim_{t \rightarrow \infty} (y)$, де величина y характеризує точність системи у встановленому режимі.

2. Час регулювання t_n

3. Максимальне перерегулювання

$$\sigma = \frac{y_{\max} - y_{уст}}{y_{уст}} \cdot 100\% \quad (2.1)$$

де y_{\max} - максимальне відхилення регульованої величини. Величина σ характеризує точність системи в перехідному процесі і зазвичай не повинна перевищувати 40%.

4. Число перерегулювань δ , тобто кількість максимумів функції $y(t)$ за час перехідного процесу, більших за значенням, ніж $y_{уст} + \Delta$.

2.3 Принцип роботи САР

Принцип роботи системи полягає в наступному: аналоговий датчик температури Texas Instruments LM335 формує напругу, пропорційну температурі об'єкта управління ПЕВР-25, що надходить на вхід мікроконтролера Arduino UNO, який програмно перетворює його в градуси Цельсія і здійснює регулювання температури відповідно до заданого закону регулювання. Для охолодження об'єкту передбачений вентилятор рис. 2.2.

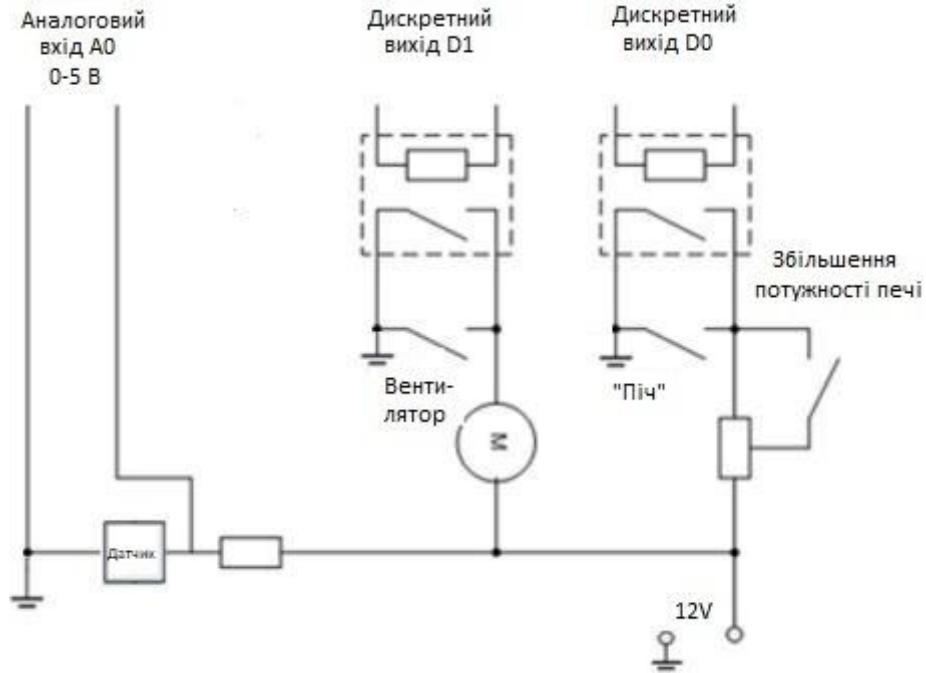


Рисунок 2.2 – Принципова схема

Технічні характеристики перерахованих компонентів системи:

1. Мікроконтролер Arduino UNO зображений на рис. 2.3.

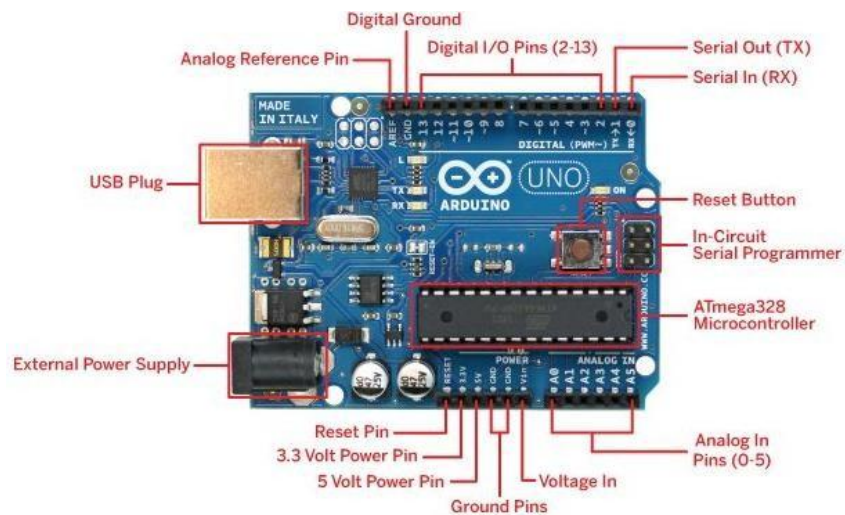


Рисунок 2.3 – Плата Arduino UNO

Технічні характеристики Arduino UNO зображені на рис. 2.4.

Основні критерії	Значення
Мікроконтролер	ATmega328
Робоча напруга	5В
Напруга живлення (рекомендований)	7-12В
Напруга живлення (граничне)	6-20В
Цифрові входи / виходи	14 (з них 6 можуть використовуватися в якості ШІМ-виходів)
Аналогові входи	6
Максимальний струм одного виведення	40мА
Максимальний вихідний струм виводу 3.3V	50мА
Flash-пам'ять	32 КБ (ATmega328) з яких 0.5 КБ використовуються завантажувачем
SRAM	2 КБ (ATmega328)
EEPROM	1 КБ (ATmega328)
Тактова частота	16МГц

Рисунок 2.4 – Технічні характеристики Arduino UNO

2. Резистор ПЕВР-25 (дротяний, емальований, вологостійкий, регульований).

Технічні характеристики ПЕВР-25 зображені на рис. 2.5.

- Межі номінальних опорів: 10 Ом... 510 Ом;
- Номінальна потужність: 25Вт;
- Гранична напруга: 1400 В;
- Допустимі відхилення опору від номінальної: $\pm 5\%$; $\pm 10\%$;
- Діапазон температур: $-60 \dots +155 \text{ }^\circ\text{C}$;
- Мінімальна напрацювання: 1500.

Рисунок 2.5 – Технічні характеристики ПЕВР-25

3. Датчик температури ТІ LM335.

Технічні характеристики датчика температури зображені на рис. 2.6.

- Діапазон робочих температур: $-40 \dots +100 \text{ }^\circ\text{C}$;
- Напруга живлення: 5В;
- Інтерфейс: аналоговий вивід;
- Локальна точність: $\pm 6^\circ\text{C}$.

Рисунок 2.6 – Технічні характеристики датчика температури ТІ LM335

Датчик температури ТІ LM335 зображений на рис. 2.7.



Рисунок 2.7 – Датчик температури ТІ LM335

4. Модуль з двома електромеханічними реле

Технічні характеристики двоканального релейного модуля зображені на рис. 2.8.

- Напруга живлення: 5 В
- Споживаний струм: 30 мА ... 40 мА
- Сигнал увімкнення: 0 В (низький рівень)
- Оптична ізоляція: є
- Кількість реле: 2 шт.
- Тип реле: електромеханічне
- Номінальний струм навантаження: 10 А

Рисунок 2.8 – Технічні характеристики двоканального релейного модуля

Двоканальний релейний модуль зображений на рис. 2.9

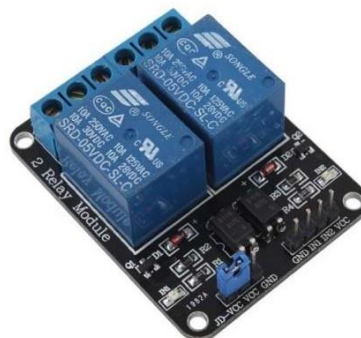


Рисунок 2.9 – Двоканальний релейний модуль

2.4 Пошук оптимального закону регулювання

2.4.1 Визначення передавальної функції системи

Для визначення закону регулювання, необхідного у досліджуваній системі, слід визначити передавальну функцію (ПФ) об'єкта управління. Для отримання перехідної характеристики процесу проведено експеримент, електрична схема якого наведена на рис. 2.10.

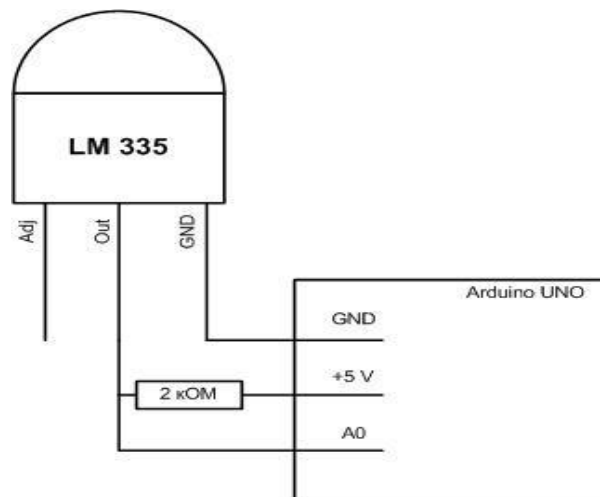


Рисунок 2.10 – Електрична схема експерименту

У експерименті мовою програмування C++ реалізовано нагрівання резистора ПЕВР-25, з інтервалами в 2 секунди з допомогою аналогового датчика ТІ LM335 зафіксовані свідчення температури протягом 70 секунд, які виводяться на монітор послідовного порту комп'ютера (рис. 2.11), до якого підключена палата Arduino UNO, і можуть бути відстежені в режимі реального часу. Програмна реалізація експерименту представлена у додатку А.

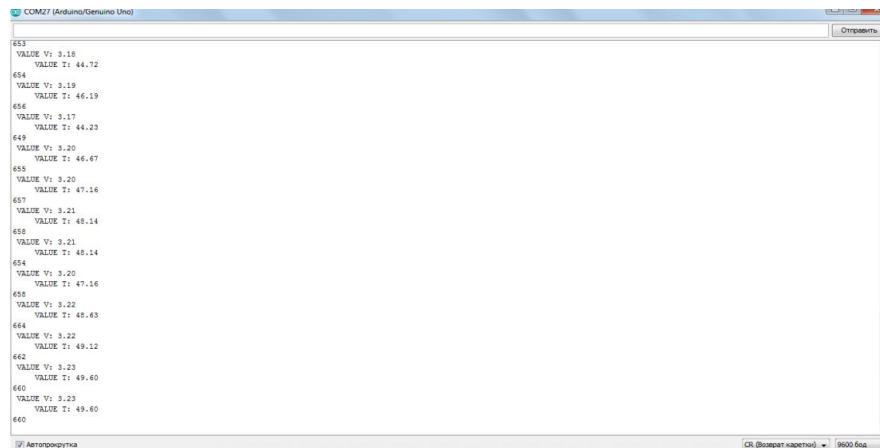


Рисунок 2.11 – Монітор послідовного порту плати Arduino UNO

На основі знятих показань з датчика температури в математичному пакеті MathCAD побудований графік перехідного процесу, який був зображений на рис. 1.12.

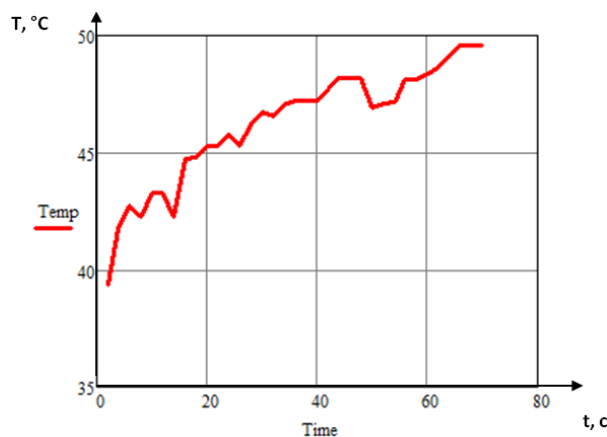


Рисунок 2.12 – Графік експерименту

На основі графіка можна зробити висновок про те, що перехідний процес відповідає аперіодичній ланці першого порядку, але має перешкоди, тому за допомогою вбудованої функції апроксимації даних було отримано графік ідеального перехідного процесу (рис. 2.13), за допомогою якого було визначено передатну функцію об'єкта управління. При цьому для того, щоб перехідна характеристика починалася в нулі, поставимо початкові умови експерименту: кімнатна температура дорівнює 38°C .

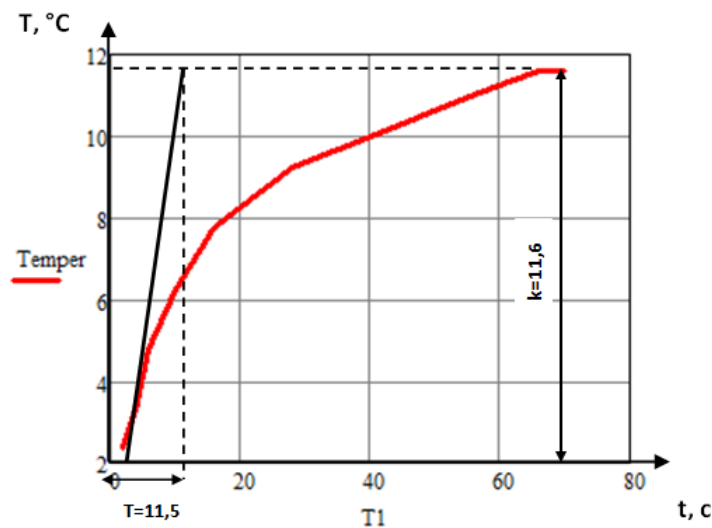


Рисунок 2.13 – Графік перехідного процесу експерименту

Значення температури що встановилося при постійно включеній печі складає 120°C , початок перехідного процесу при якому температура досягає 50°C представлена на рис. 2.13, оскільки не надається можливість вивести параметри ОУ на граничний рівень через технічні особливості установки, то при ідентифікації моделі ОУ необхідно врахувати ці параметри і на їх основі визначити передавальну функцію системи.

Функція аперіодичної ланки першого порядку має наступний вигляд:

$$W(s) = \frac{k}{T \cdot s + 1} \quad (2.2)$$

За графіком перехідного процесу визначається на основі коефіцієнта посилення k та постійної інтегрування T , що характеризує інерційність ланки.

У програмному пакеті MATLAB побудовано перехідну характеристику системи, на основі якої отримано передатну функцію системи і перевірено її достовірність (рис. 2.14).

$$W(s) = \frac{120}{150 \cdot s + 1} \quad (2.3)$$

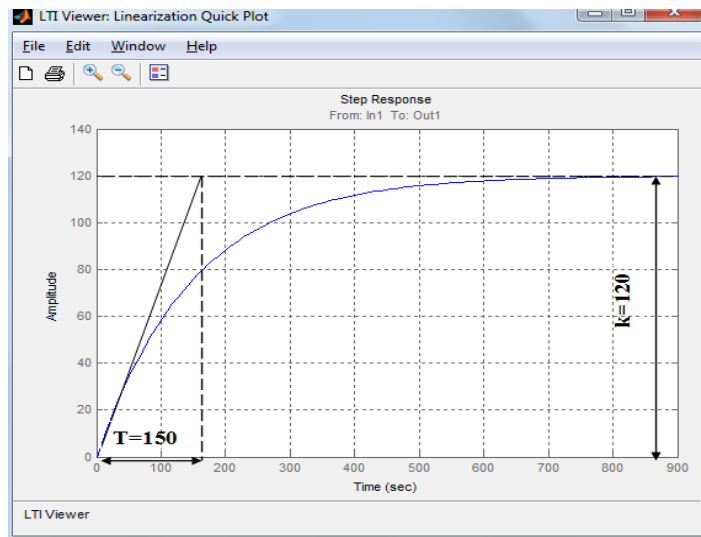


Рисунок 2.14 – Графік перехідного процесу САР температури у програмному пакеті MATLAB

2.4.2 Вибір закону регулювання

САР температури доповнена функцією включення, відключення вентилятора та нагрівального елемента, а також реалізована компенсація напруги між всією системою і мікропроцесором з допомогою двопозиційного регулювання. У досліджуваній системі програмно реалізовані фільтрація вхідного сигналу і ПІ-регулювання для стабілізації вимірних значень. Удосконалена система представлена на рис. 2.15.

У ході вивчення роботи системи та на основі аналізу характеристик законів регулювання був обраний закон ПІ-регулювання, так як він може забезпечити велику точність регулювання при великих, але плавних змінах навантаження. П-регулятор має на тлі ПІ-закону регулювання суттєвий недолік - залишкове відхилення регульованого параметра, яке впливає на якість системи в цілому. І-регулятор має великий час регулювання, що неприпустимо для систем подібного роду.

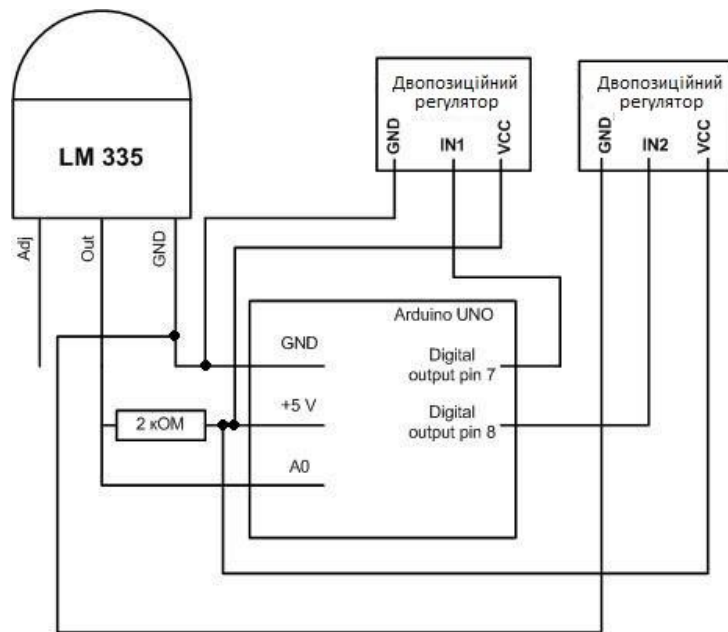


Рисунок 2.15 – Електрична схема САР температури

2.5 Налаштування ПІ-регулятора

2.5.1 Метод Зіглера-Нікольса

Дана система має один чутливий елемент і здійснює обробку одного вхідного сигналу на регулятор, отже вона є одноконтурною системою регулювання. У таких системах застосовуються десятки методів параметрів налаштування регуляторів. Розрахунок коефіцієнта передачі регулятора k_p , постійною інтегрування T_i , та постійною диференціювання T_d і є безпосередньо налаштуванням регулятора.

Для налаштування ПІ-регулятора використано кілька методів, одним із яких є метод Зіглера-Нікольса, котрий ґрунтується на пошуку параметрів відгуку об'єкта на одиничний стрибок, якими є a і L , зображені на рис. 2.16 [2]. На основі цих параметрів розраховуються коефіцієнти різних законів регулювання, формули для розрахунку яких наведені в табл. 2.1.

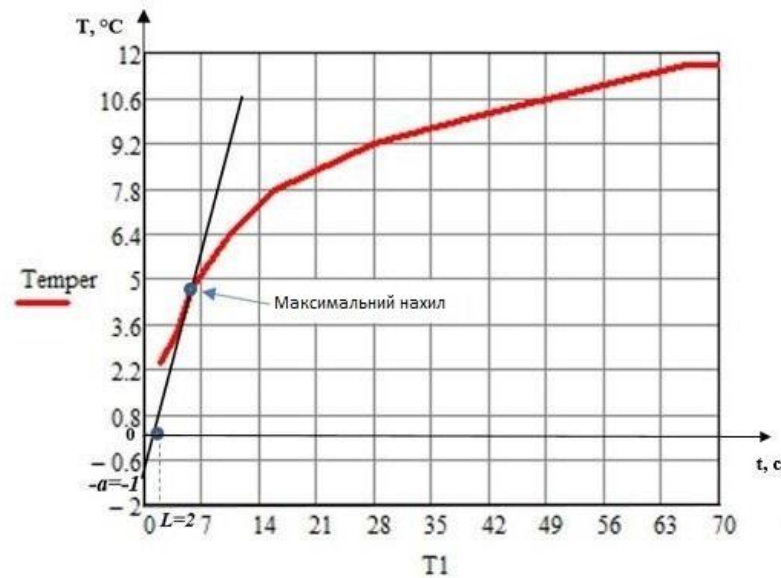


Рисунок 2.16 – Перехідна характеристика об'єкта з параметрами відгуку об'єкта на одиничний стрибок a та L

Розрахунок налаштувань для ПІ-регулятора за методом Зіглера-Нікольса:

$$k_p = \frac{0,9}{a} = \frac{0,9}{1} = 0,9 \quad (2.4)$$

$$T_i = \frac{3 \cdot L}{k_p} = \frac{3 \cdot 2}{0,9} = 6,66 \quad (2.5)$$

Щоб оцінити якість системи для знайдених параметрів регулювання, у програмному пакеті MATLAB була змодельована структурна схема системи (рис. 2.17), на основі якої була отримана перехідна характеристика системи з ПІ-регулюванням (рис. 2.18), за якою визначено основні показники якості САР температури.

Таблиця 2.1 – Формули для розрахунку коефіцієнтів регулятора методом Зіглера-Нікольса по відгуку на стрибок

Регулятор	k_p	T_i	T_d
П	$\frac{1}{a}$		
ПІ	$\frac{0,9}{a}$	$\frac{3 \cdot L}{k_p}$	
ПІД	$\frac{1,2}{a}$	$\frac{0,9 \cdot L}{k_p}$	$\frac{0,5 \cdot L}{k_p}$

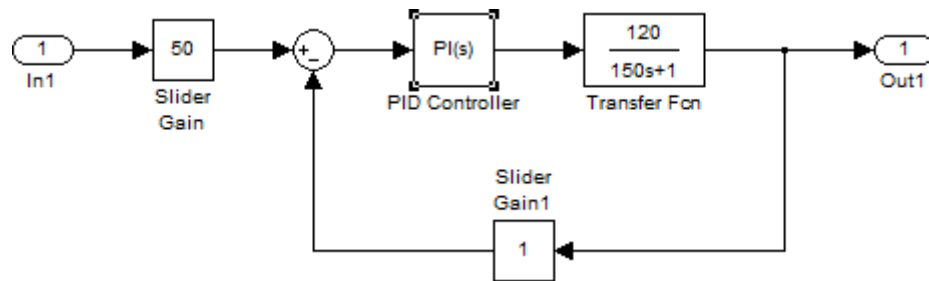


Рисунок 2.17 – Структурна схема САР температури

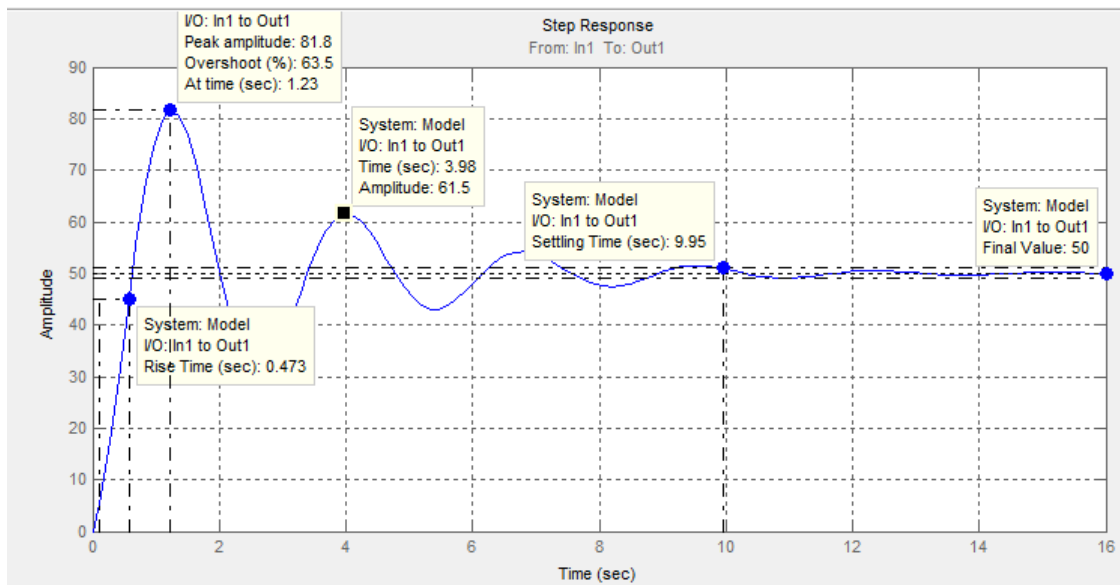


Рисунок 2.18 – Перехідна характеристика системи з налаштуванням коефіцієнтів ПІ-регулятора методом Зіглера-Нікольса

У ході аналізу перехідної характеристики системи з налаштуванням параметрів ПІ-регулятора методом Зіглера-Нікольса отримані наступні показники якості системи на основі графіка перехідного процесу в пакеті MATLAB (рис. 1.18):

- Відхилення $y_{уст}=50$ при заданій уставці в 50;
- Час регулювання $t_n=9,95$ с;
- число перерегулювань $\delta=3$;
- максимальне значення регульованої величини $y_{max}=81,8$.

Даний аналіз дозволяє зробити висновок про те, що налаштування регулятора, отримані методом Зіглера-Нікольса, впливають на характер перехідного процесу наступним чином: коливання в системі мають повільний загасаючий характер, перерегулювання перевищує допустимі норми, що знижує точність системи в загалом, але дані результати виправдані тим, що сам метод ґрунтується тільки на двох параметрах системи.

2.5.2 Метод CHR

Критерієм якості другого досліджуваного методу налаштування регулятора CHR (Chien, Hrones, Reswick) є критерієм максимальної швидкості наростання за відсутності перерегулювання або за наявності не більше чим 20 відсоткового перерегулювання, даний метод дозволяє отримати більший запас стійкості, ніж попередній [2].

Метод CHR дає дві різні системи параметрів регулятора. Одна з них отримана при спостереженні відгуку зміни уставки (табл. 1.2), а друга - при спостереженні відгуку на зовнішні обурення. Вибір системи параметрів залежить від пріоритетів, які ставить розробник для певного технологічного процесу: якість регулювання при зміні уставки або ослаблення зовнішніх впливів [2].

Розглянутий метод використовує ті ж параметри a і L , що і метод Зіглера-Нікольса, але має менші коефіцієнти пропорційності в формулах. Оскільки якість регулювання при зміні уставки є більш пріоритетним фактором для досліджуваної системи, то пошук коефіцієнтів налаштування регулятора був обмежений першою системою параметрів регулятора за умовою 20-відсоткового перерегулювання.

Таблиця 2.2 - Формули для розрахунку коефіцієнтів регулятора методом CHR по відгуку на зміну уставки

Регулятор	Без перерегулювання			З 20-відсотковим перерегулюванням		
	k_p	T_i	T_d	k_p	T_i	T_d
II	$0,3/a$	-	-	$0,7/a$	-	-
III	$0,3/a$	$1,2 \cdot L / k_p$	-	$0,6/a$	$1 \cdot L / k_p$	-
IIIД	$0,6/a$	$1 \cdot L / k_p$	$0,5 \cdot L / k_p$	$0,95/a$	$1,4 \cdot L / k_p$	$0,47 \cdot L / k_p$

Розрахунок налаштувань для III-регулятора за методом CHR:

$$k_p = \frac{0,6}{a} = \frac{0,6}{1} = 0,6 \quad (2.6)$$

$$T_i = \frac{1 \cdot L}{k_p} = \frac{1 \cdot 2}{0,9} = 3,33 \quad (2.7)$$

У ході аналізу перехідної характеристики системи з налаштуванням параметрів III-регулятором методом CHR отримані наступні показники

якості системи на основі графіка перехідного процесу в пакеті MATLAB (рис. 2.19).

- Відхилення $y_{уст}=50$ при заданій уставці в 50;
- Час регулювання $t_n=15,7$ с;
- максимальне перерегулювання $\sigma= 65\%$;
- число перерегулювань $\delta= 3$;
- максимальне значення регульованої величини $y_{max}=82,5$.

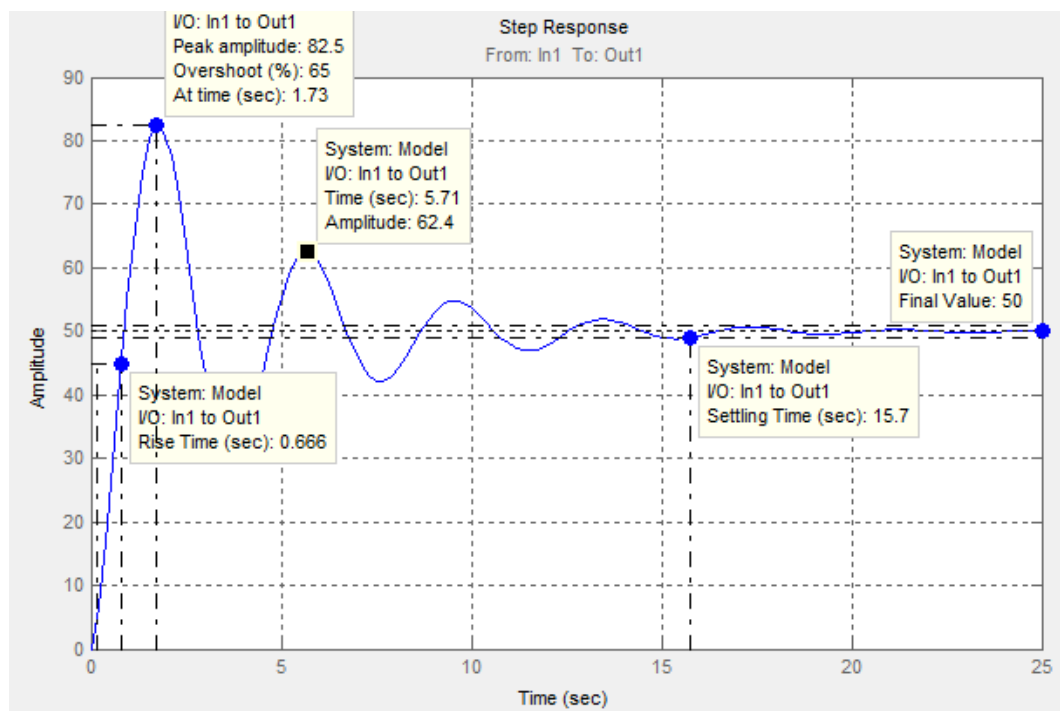


Рисунок 2.19 – Перехідна характеристика системи з налаштуванням коефіцієнтів ПІ-регулятора методом CHR

Даний аналіз дозволяє зробити висновок про тому, що налаштування регулятора, отримані методом CHR, впливають на характер перехідного процесу наступним чином: коливання в системі, як і в попередньому методі, мають повільний загасаючий характер, перерегулювання також перевищує допустимі норми, час регулювання збільшилося, що знижує не тільки точність системи, але та її якість.

2.5.3 Метод ручного налаштування

Слід відзначити, що розрахунок коефіцієнтів налаштування регулятора не дав очікувані результати, оскільки якість регульованої системи не задовольняє встановленим вимогам. Оскільки розглянуті методи ґрунтуються на спрощених моделях об'єкта, самі моделі використовують параметри, ідентифіковані з деякою похибкою, отже, після розрахунку знайдених параметрів необхідно здійснити їх підстроювання на основі правил, використовуваних при ручному налаштуванні:

- збільшення пропорційного коефіцієнта збільшує швидкодію та знижує запас стійкості;
- зі зменшенням інтегральної складової помилка регулювання з часом зменшується швидше;
- зменшення постійної інтегрування зменшує запас стійкості;
- збільшення диференціальної складової збільшує запас стійкості та швидкодію [2].

Застосування цих правил, можливо, тільки при попередньому налаштуванні регулятора за формулами, знайдені коефіцієнти справедливі лише в межах оптимального налаштування регулятора.

Дотримуючись вище наведеним правилам, регулятор після підстроювання має такі параметри: $k_p=3,5$, $T_i=1,5$ та перехідну характеристику, зображену на рис.1.20.

Система має такі показники якості:

- Відхилення $y_{уст}=50$ при заданій уставці в 50;
- Час регулювання $t_n=5,12$ с;
- максимальне перерегулювання $\sigma=9,42\%$;
- коливання у системі відсутні;
- максимальне значення регульованої величини $y_{max}=54,7$.

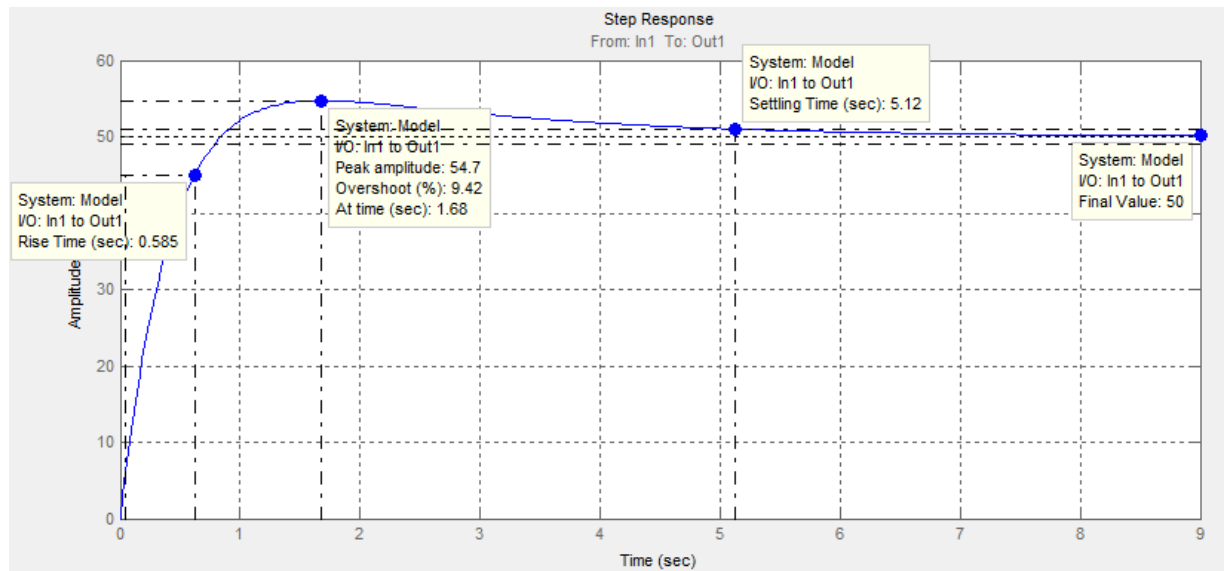


Рисунок 2.20 – Перехідна характеристика системи після ручного підстроювання ПІ-регулятора

2.6 Забезпечення коректної роботи реле та фільтрації сигналу

2.6.1 Широтно-імпульсна модуляція

З метою забезпечення коректною роботою реле-печі і реле-вентилятора застосовано широтно-імпульсну модуляцію, яка забезпечує кодування аналогового вхідного сигналу за рахунок зміни його тривалості (ширини) прямокутних імпульсів несучої частоти, що дозволяє здійснювати перемикання між мінімальним та максимальним значеннями. Широтно-імпульсна модуляція формується на основі двох сигналів - модулюючого, у даній роботі яка формується на основі показань, що прийшли з датчика температури, і допоміжного, має пилкоподібну форму сигналу. Амплітуда пилкоподібного сигналу обрана таким чином, щоб вона відповідала діапазону зміни сигналу керуючого ПІ-регулятора, котрий лежить в межах від -255 до

255. Вибір частоти визначено в ході експерименту на основі наступних принципів: період допоміжного сигналу не може мати мале значення,

оскільки це вплине на частоту перемикання реле, що може призвести до зносу контактів та швидкого виходу з ладу; велике значення періоду не дозволить забезпечити коректне перемикання реле в рамках технологічного процесу. Сигнал моделює напругу, у цьому випадку максимальна напруга становить 5В, а мінімальне 0В, що визначається технічними характеристиками мікроконтролера. У процесі модуляції змінюється ширина імпульсу, тобто тривалість часу вмикання та відключення пристрою, в даному експерименті – реле, що відповідає за роботу нагрівального елемента та вентилятора. На основі загальної структурної схеми для систем з використанням модуляторів такого роду, зображеного на рис. 2.21, в пакеті MATLAB побудована модель САР з застосуванням широтно-імпульсної модуляції і одночасно з ПІ-регулюванням (рис. 2.22). Програмна реалізація на мікроконтролері представлена у додатку А.

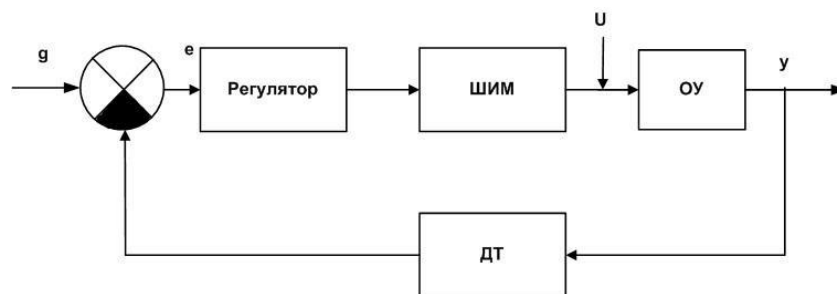


Рисунок 2.21 – Типова структурна схема САР із застосуванням широтно-імпульсної модуляції та ПІ-регулюванням

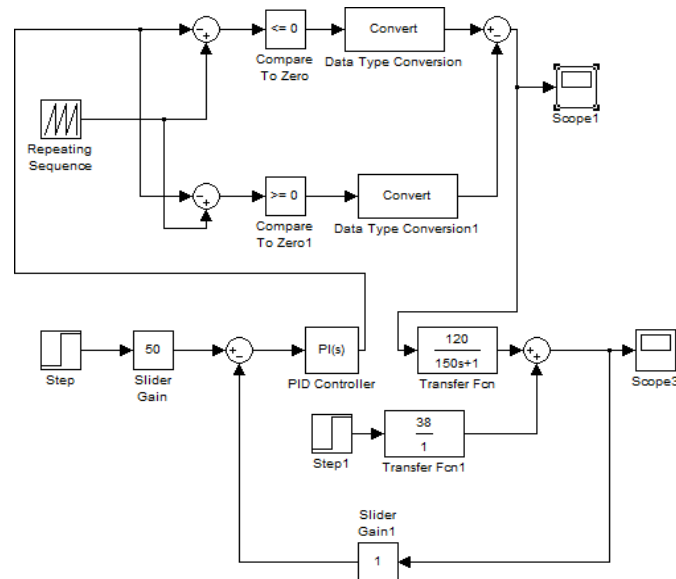


Рисунок 2.22 – Структурна схема САР із застосуванням широтно-імпульсної модуляції і ПІ-регулюванням, в пакеті MATLAB

За характером графіка перехідного процесу, що вийшов (рис. 2.23), можна зробити висновок про те, що отримані налаштування для регулятора, а також перемикавання реле, що здійснюються на основі ШІМ, забезпечують необхідний принцип роботи досліджуваної системи. При регулюванні такого роду, температура досягає максимального значення рівного 55°C і вагається в районі заданого значення, маючи відхилення в $(2-4)^{\circ}\text{C}$.

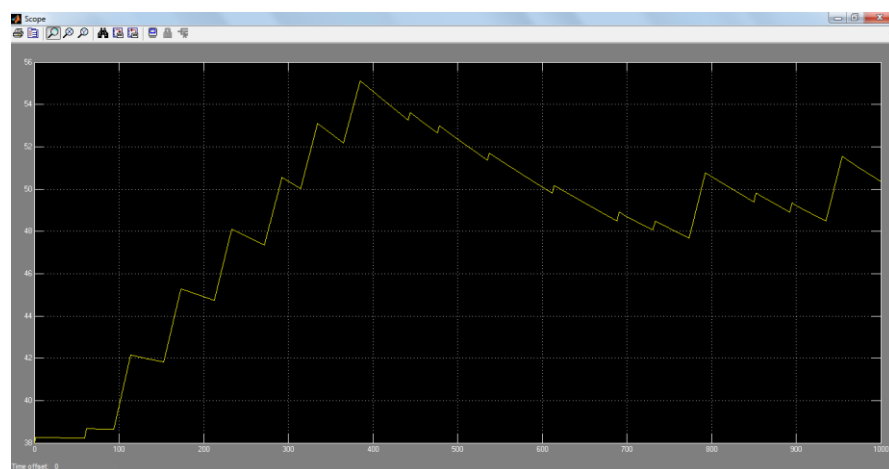


Рисунок 2.23 – Перехідна характеристика процесу із застосуванням широтно-імпульсної модуляції та ПІ-регулювання

2.6.2 Первинна обробка сигналу

Оскільки отримані результати характерні для ідеалізованої системи, а її сигнал, знятий в ході експерименту (рис. 2.12), має імпульсні перешкоди, репрезентують собою порівняно короткий і потужний сплеск напруги в каналі, а час дії імпульсної перешкоди виявляється досить малим, то існує необхідність в фільтрації вхідного сигналу [4].

Для первинної обробки вимірних значень застосована фільтрація, що забезпечує виділення корисного сигналу з отриманої інформації з перешкодою. Найбільш поширеними є фільтри низьких частот, до яких відноситься фільтр експоненційного згладжування, в основі якого лежить розрахунок експонентних ковзних середніх рядів, членами якого є вимірні значення. Основною перевагою такого роду згладжування є здатність адаптуватись до нового рівня процесу без значного реагування на випадкові відхилення [6]. Процедура простого експоненційного згладжування здійснюється по наступним формулам:

$$S_t = \alpha \cdot X_{t-1} + (1 - \alpha) \cdot S_{t-1} \quad (2.8)$$

де S_t – значення експоненційного середнього в момент t , X_{t-1} – фізичне спостереження в момент $t-1$, α – параметр згладжування.

Чим менше α , тим більшою мірою пригнічуються коливання вихідного ряду, але при дуже малих значеннях може статися спотворення корисного сигналу.

У прикладному пакеті MathCAD написана програма для експоненційного згладжування, з допомогою якої експериментально було визначено значення $\alpha=0,5$ (рис. 2.24).

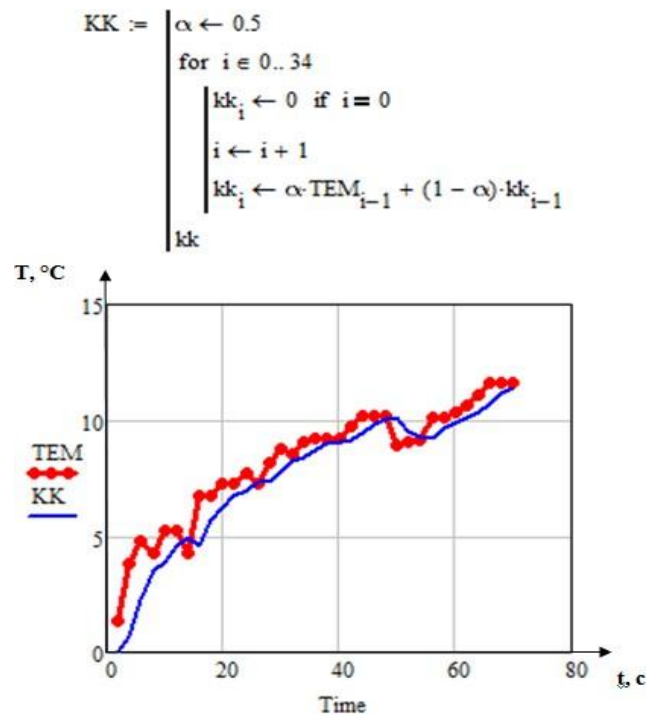


Рисунок 2.24 – Експериментальне визначення параметра згладжування у пакеті MathCAD

3 РОЗРОБКА МЕТОДИКИ ПРОЄКТУВАННЯ САР НА БАЗІ MASTERSCADА

3.1 Основні структурні компоненти SCADA-системи

З метою забезпечення високої якості регулювання системи, а також візуалізації перехідного процесу аналізованої системи, використана SCADA система, що дозволяє здійснити процес збору інформації в режимі реального часу з віддалених об'єктів для управління ними та забезпечує аналіз отриманих даних [7].

Сучасні SCADA системи включають до свого складу три основні компоненти, структура яких зображена на рис. 3.1, і виконують наступні функції:

- Видалений термінал здійснює обробку завдання у режимі реального часу
- Диспетчерський пункт здійснює обробку даних і управління високого рівня в режимі реального часу; забезпечення інтерфейсу між людиною-оператором та системою.
- Комунікаційна система необхідна для передачі даних з віддалених об'єктів, терміналів на центральний інтерфейс оператора-диспетчера і передачі сигналів управління на диспетчерський пункт [7].



Рисунок 3.1 – Основні структурні компоненти SCADA системи

Віддаленим терміналом в цій досліджуваній системі є мікроконтролер Arduino UNO, він здійснює обробку завдання засобами написаної програми мовою програмування C++. Комунікаційною системою, яка здійснює обмін даними між віддаленим терміналом та диспетчерським пунктом є Modbus Universal MasterOPC сервер, який використовує для роботи найпоширеніший протокол обміну у промисловості Modbus [8]. Комп'ютер, що виконує функції диспетчерського пункту, оснащений пакетом програм MasterSCADA, котрий побудований на клієнт-серверній архітектурі з можливістю функціонування у локальних та глобальних мережах. Прийом та передача в даному пакеті відбувається на основі стандартів OPC, таких як HTML, OLE, ODBC, COM/DCOM, ActiveX і інші. Істотною перевагою обраного SCADA-пакету є його забезпечення всіх необхідних можливостей для стикування із зовнішніми програмами та системами, а також надання користувачам безкоштовної ліцензійної версії у розмірі 32-точок виконання на офіційному сайті inSAT, що є достатнім для роботи з досліджуваною системою.

3.2 Встановлення зв'язку між мікроконтролером та SCADA

Оскільки основним завданням використання пакета SCADA в даній роботі є регулювання температури в печі ПЕВР-25, то додаток що ми розроблюємо повинен здійснювати налаштування коефіцієнтів для ПІ-закону регулювання; корекцію уставки температури; відображення стану відключення та включення реле-печі та реле-вентилятор; відображення вхідних значень температури, вихідного сигналу з ПІ-регулюванням і ШИМ-модуляцією в вигляді трендів.

Для встановлення зв'язку між платою Arduino і MasterSCADA в розроблюваному коді і в MasterOPC сервері, для комунікаційного стандарту, підтримуючого взаємодію між польовими пристроями, контролерами і додатками пакету Master, виділено наступні змінні, які представлені в

таблиці 3.1. Встановлено зв'язок із портом, до якого підключений мікроконтролер, швидкість опитування рівна 11520 оскільки реєстрація показань на платі відбувається з такою ж самою частотою, час відповіді та період опитування, рівні 500 мс, таким чином, завершена налаштування OPC сервера.

З метою перевірки працездатності написаної програми, код завантажений в мікроконтролер, конфігурація для досліджуваної системи запущена, а в пакеті MasterOPC можна спостерігати процес обміну даними між датчиком температури та OPC сервером через мікропроцесорну плату на основі зміни прийнятих значень та якості передачі. (рис. 3.2)

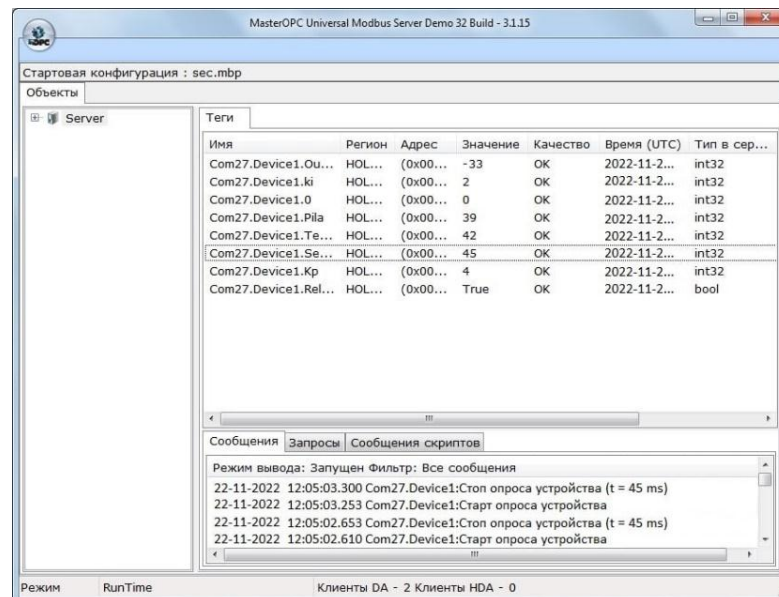


Рисунок 3.2 – Процес обміну даними між мікроконтролером та OPC сервером по засобам програми MasterOPC

Таблиця 3.1 - Список змінних, відповідальних за подання інформації пакеті SCADA

Назва змінної	Реєстр контролера	Тип	Тип доступу	Призначення
Temperature	HOLDING_ REGISTERS[4]	Real	ReadOnly	Зберігання показань з датчика температури
Pila	HOLDING_ REGISTERS[3]	Double (float)	ReadOnly	Формування ШІМ-сигналу
k_i	HOLDING_ REGISTERS[1]	Double (float)	ReadOnly	Коефіцієнт постійного струму
Output	HOLDING_ REGISTERS[2]	Double (float)	ReadOnly	Сигнал ПІ-регулювання
Setpoint	HOLDING_ REGISTERS[5]	Double (float)	WriteOnly	Уставка температури
k_p	HOLDING_ REGISTERS[6]	Double (float)	WriteOnly	Коефіцієнта передачі регулятора
Relay3State	HOLDING_ REGISTERS[6]	Bool (boolean)	WriteOnly	Стан реле-печі

3.3 Відображення технологічного процесу в режимі реального часу

Оскільки на виробництві безпека управління по засобам SCADA систем здійснюється за рахунок видалення оператора від пульта технологічного процесу, розмежуванням прав доступу, шифруванням,

секретністю протоколів зв'язку, а також застосуванням міжмережевих протоколів, то невід'ємним завданням для розробника таких систем залишається детальна візуалізація технологічного процесу, яка допоможе оператору коректно керувати системою.

Основними засобами візуалізації в пакетах SCADA є тренди, графічні індикатори, параметри та відображення значення будь-яких величин, а також кнопки, що дозволяють переміщатися між об'єктами мнемосхем. З допомогою тренда, котрий в режимі реального часу дозволяє відстежити зміну значень температури, був візуалізований ШІМ-сигнал, а також процес зміни перехідного процесу в залежності від зміни коефіцієнтів регулятора, уставок, які задаються оператором, а також стан працездатності печі і вентилятора.

На основі конфігураційного файлу, що зв'язує OPC сервер та мікропроцесорну плату Arduino UNO, в пакеті MasterSCADA встановлено змінні в розроблений проект, які містяться в таблиці 3.1, та за допомогою елементів палітри, описаних вище, був розроблений проект, який виконує функції регулювання і візуалізацію технологічного процесу, які зображені на рис. 3.3.

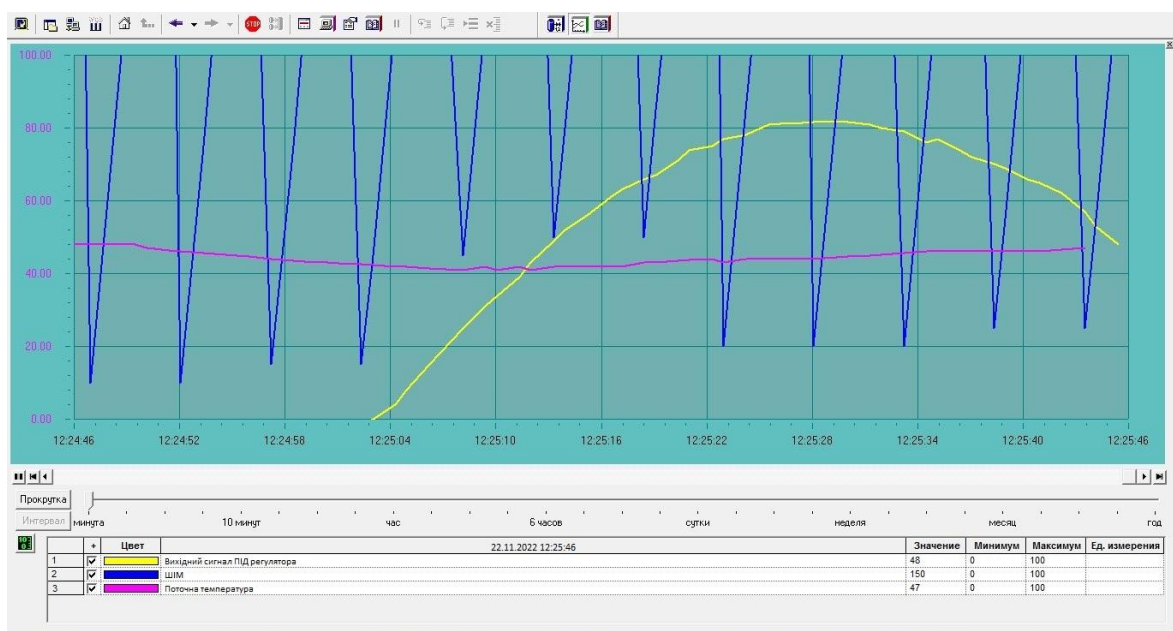


Рисунок 3.3 – Тренд сигналу ПІД-регулятора, ШІМ-сигналу та поточного значення температури САР

3.4 Інтерфейс САР температури у SCADA-системі

З метою комфортного дистанційного керування системою створено дружній інтерфейс САР температури на мнемосхемі об'єкта управління (рис 3.4) у ньому містяться три параметра: уставки, коефіцієнт передачі регулятора, коефіцієнт постійною інтегрування; відображається стан увімкнення реле-печі у вигляді індикатора стану; представлено поточне значення температури, вихідного сигналу з ПІД-регулятора, ШІМ-сигналу; кнопка, що забезпечує перехід до тренду, зображеному на рис 3.3.

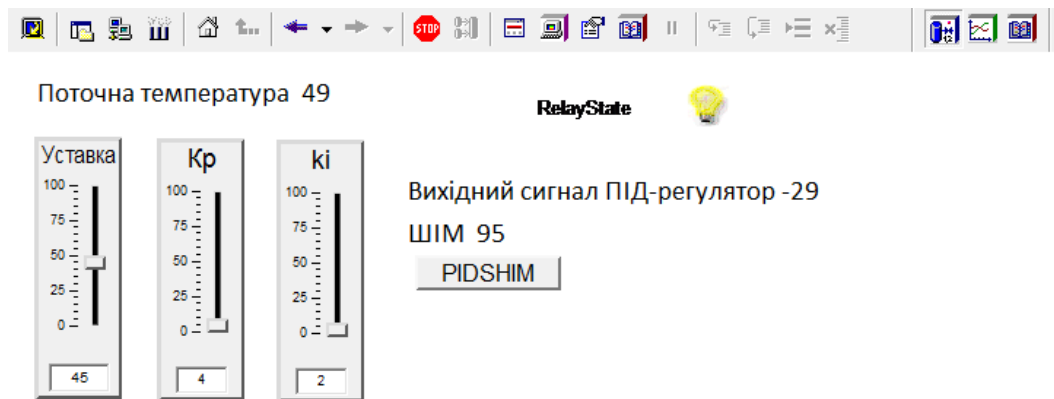


Рисунок 3.4 – Мнемосхема об'єкта управління САР температури

ВИСНОВКИ

В даній роботі розроблено SCADA система для автоматичного регулювання температури в печі, яка включає функції дистанційного керування системою на базі мікроконтролера Arduino UNO, на основі розрахованих показників П-закону регулювання, а також представляє наочну візуалізацію технологічного процесу оператору системи.

Досліджувана система може послужити як недорогога, виконуюча основні функції, прикладна модель, яка дозволить провести аналіз основних характеристик технологічного процесу та налаштувань необхідних параметрів системи без значних витрат та зусиль.

Побудова прикладної моделі у MasterSCADA-системі різко скорочує набір необхідних знань у галузі класичного програмування для проєктування складних систем автоматичного регулювання, дозволяючи концентрувати зусилля на засвоєнні знань у прикладній області.

ПЕРЕЛІК ПОСИЛАНЬ

1. **Бобух А. О.** Автоматизовані системи керування технологічними процесами [Текст]: навч. посібник / А. О Бобух; ХНАМГ. – Харків: ХНАМГ, 2006. – 65с. - Режим доступу: https://eprints.kname.edu.ua/9106/1/АСК_ТП.pdf [ukr.]
2. **Паламар, Михайло.** Проектування комп'ютеризованих вимірювальних систем і комплексів. [Текст] / Михайло Паламар, Михайло Стрембіцький, Андрій Паламар; Тернопільський національний технічний університет імені Івана Пулюя, 2018. – 150 с.
3. ARDUINO. Arduino Uno SMD. [Електронний ресурс]. – Режим доступу: <http://arduino.cc/en/Main/arduinoBoardUno>
4. Temperature sensor. [Electronic resource]. – Access mode: <https://www.automationdirect.com/adc/shopping/catalog/sensorszencoders/temperatureensorsatransmitters>
5. Введення в систему SCADA. [Електронний ресурс]. - Режим доступу: <https://uk.answersexpress.com/an-introduction-scada-systems-99185>[ukr.]
6. Programmable Controllers. [Electronic resource]. – Access mode: <http://velocio.net/>
7. **Samiec, T** Comparison of communication nets and programming methods fo plc simatic: тези / Т. Samiec ; BUOT. – BRNO: BUOT, 2010. – 53 с.
8. SCADA системи [Електронний ресурс]. - Режим доступу: [http://www.scadasystems.net/\[eng.\]](http://www.scadasystems.net/[eng.])
9. Як вибрати SCADA систему [Електронний ресурс]. - Режим доступу: [http://readonline.com.ua/items/47871-yak-vibratiscada-sistemu/\[ukr.\]](http://readonline.com.ua/items/47871-yak-vibratiscada-sistemu/[ukr.])
10. Certifications: Industrial control systems. [Electronic resource]. – Access mode: <https://www.giac.org/certifications/ics>

11. **Conte, D. L.** Designing and Developing Virtual and Real Testbeds for Industrial Control Systems Education and Training [Text]: monograph. / D.L. Conte; Eskom Peaking Engineering Head Office. – New Jersey: 2019. – 94 с.

12. Интегрированные системы проектирования и управления: SCADA-системы : учебное пособие / И. А. Елизаров, А. А. Третьяков, А. Н. Пчелинцев и др. – Тамбов : Изд-во ФГБОУ ВПО «ТГТУ», 2015. – 160 с. – 400 экз.

13. **Чернов, Д. В.** Применение диаграмм Эйлера-Венна при решении задачи выбора мер защиты АСУ ТП [Электронный ресурс] / Д. В. Чернов // Естественные и Технические Науки. — 2021. — Вып. №07. — С. 127-131. — Режим доступа: <https://doi.org/10.37882/2223-2966.2021.07.37>.

14. Датчик температуры LM335. [Электронный ресурс]. - Режим доступа: [https://www.ti.com/product/LM335\[eng.\]](https://www.ti.com/product/LM335[eng.])

15. FBD [Электронный ресурс]. - Режим доступа: [https://flprog.ru/chto-takoe-flprog/\[rus.\]](https://flprog.ru/chto-takoe-flprog/[rus.])

16. Види середовищ програмування. [Электронный ресурс]. - Режим доступа: [https://foktver.ru/uk/internet/sreda-programmirovaniya-arduino-blokami-programmirovanie-arduino-s/\[ukr.\]](https://foktver.ru/uk/internet/sreda-programmirovaniya-arduino-blokami-programmirovanie-arduino-s/[ukr.])

17. **Петров И. В.** ПЗО Программируемые контроллеры. Стандартные языки и приемы прикладного проектирования / В. П. Дьяконова. – М.: СОЛОН-Пресс, 2004. – 256 с : ил. - (Серия «Библиотека инженера»)

18. **Елизаров, И. А.** Интегрированные системы проектирования и управления [Текст]: SCADA-системы : учебное пособие / И. А. Елизаров, А. А. Третьяков, А. Н. Пчелинцев и др. – Тамбов : Изд-во ФГБОУ ВПО «ТГТУ», 2015. – 160 с. – 400 экз.

19. Procedure for substantiated development of measures to design secure software for automated process control systems. [Electronic resource]. - Access mode: <https://ieeexplore.ieee.org/abstract/document/7491660>

20. Автоматизація виробничих процесів [Текст] : підручник / І. В. Ельперін, О. М. Пупена, В. М. Сідлецький, С. М. Швед ; Нац. ун-т харч. технол. – 2-ге вид., випр. – К. : Ліра-К, 2015. – 378 с.

21. Промислові мережі та інтеграційні технології в автоматизованих системах [Текст] : навч. посіб. / О. М. Пупена, І. В. Ельперін, Н. М. Луцька, А. П. Ладанюк. — К. : Ліра-К, 2011. – 552 с.

22. **Schmutz, W. F.** Development Of The Control And Automation System For Mini Hydro Turbines : monograph. / W.F. Schmutz; ЕРЕНО – Cape Town : 1995. – 122 с.

23. A Configurable Dependency Model of a SCADA System for Goal-Oriented Risk Assessment. [Electronic resource] – Access mode: <https://doi.org/10.3390/app12104880>.

24. SCADA system to investigate an experimental nonlinear control system for an electric discharge. [Electronic resource] – Access mode: <https://doi.org/10.21685/2587-7704-2021-6-2-8>.

25. SCADA communication and security. [Electronic resource] – Access mode: <https://doi.org/10.1002/sec.698>.

26. Scada Security. [Electronic resource] – Access mode: <https://doi.org/10.1115/1.2012-jun-1>.

27. Turc, T. Using WEB Services in SCADA Applications. [Electronic resource] – Access mode: <https://doi.org/10.1016/j.protcy.2015.02.083>.

ДОДАТОК А

Програмна реалізація експерименту:

```
int ledpin=13;
// підключення світлодіода для перевірки працездатності програми int lm335 = 0;
// оголошення типу змінних для показань з датчика та його підключення до
аналогового входу A0
void setup()
{
Serial.begin(9600); pinMode(ledpin, OUTPUT);
}
void loop()
{
double val = analogRead(lm335); // зчитування свідчень з датчика double voltage =
val*5.0/1024;
// Перетворення показань в одиниці вимірювання напруги (В) Serial.print("
VALUE V: ");
Serial.println(voltage); //виведення значень поточної напруги double temp =
voltage*100 - 273.15;
//перетворення показань на одиниці вимірювання температури (°C) Serial.print("
VALUE T: ");
Serial.println(temp); // Висновок значень поточної температури delay (2000); //
інтенсивність фіксування свідчень
}
```

Програмна реалізація налаштування ПІ-регулятора:

```
#include <PID_v1.h> // Підключення стандартної бібліотеки ПІД- регулятора
double Setpoint, Input, Output; double Kp=15, Ki = 0.5, Kd=0;

PID myPID(&Input, &Output, &Setpoint, Kp, Ki, Kd, DIRECT); // вбудована
функція ПІД - регулювання
boolean ZPID;

int SampleTime=100; void setup()
{Serial.begin(115200);

Input = analogRead(lm335)/13; //зчитуємо свідчення температури з датчика
Setpoint = 50;// завдання значення температури в якості уставки

myPID.SetOutputLimits(-255, 255); //Межі вихідного сигналу для ПІД-регулятора
myPID.SetMode(AUTOMATIC); //встановлює автоматичне включення ПІ-
регулятора
myPID.SetSampleTime(SampleTime);} //задає частоту рахунку вихідного
```

сигналу

```
void loop()
```

```
{ZPID=myPID.Compute();} //зчитуємо вихідний сигнал ПІД- регулятора
```

Програмна реалізація ШІМ-сигналу:

```
double Pila = 0; // Змінна для зберігання значення unsigned long now;
```

```
unsigned long lastPilaChange = 0; unsigned long lastSerialOutput = 0; bool
```

```
Relay3State = false;
```

```
bool Relay4State = false;
```

```
bool isPWMChanged = false; // зміна пили void setup()
```

```
{
```

```
Serial.begin(115200); pinMode(ledpin, OUTPUT); pinMode(Relay4, OUTPUT);
```

```
pinMode(Relay3, OUTPUT); } void loop()
```

```
now = millis();
```

```
if (now - lastPilaChange >= 100)
```

```
{ Pila=Pila+3; if (Pila> = 255)
```

```
{Pila=0;
```

```
isPWMChanged = false;} lastPilaChange = now; }
```

```
if (now - lastSerialOutput >= 1000) { lastSerialOutput = now;
```

```
if (Pila >= Output)
```

```
{ if (Relay3State == LOW)
```

```
{digitalWrite(Relay3, HIGH); // вимикаємо грубку isPWMChanged = true;
```

```
Relay3State = HIGH;}}
```

```
if ((Pila<Output) && (isPWMChanged == false))
```

```
{ if (Relay3State == HIGH)
```

```
{digitalWrite(Relay3, LOW); // Включаємо грубку } Relay3State = LOW;}}
```

```
if (Pila < Output)
```

```
{ if (Relay4State == LOW) {
```

```
digitalWrite(Relay4, HIGH); // вимикаємо грубку isPWMChanged = true;
```

```
Relay3State = HIGH;}}
```

```
if ((Output<=0) && (-Pila>Output) && (isPWMChanged == false))
```

```
{digitalWrite(Relay3, HIGH); digitalWrite(Relay4, LOW);
}
```

Програмна реалізація фільтра експоненційного згладжування:

```
double gamma = 0.002; double y1 = 0;
void loop()

{

double val = (double)analogRead(A0)*500.0/1024.0-273.15; y1 = gamma * val + (1
- gamma) * y1;
Input = y1;}

```

Програмна реалізація передачі даних між OPC сервером і мікроконтролером Arduino UNO:

```
#include <SimpleModbusSlave.h> // підключення стандартної бібліотеки OPC
сервера
HOLDING_REG_SIZE=8 //вводимо у каналів

unsigned int HOLDING_REGISTERS[HOLDING_REG_SIZE];

// задається масив розміру регістра

modbus_configure(&Serial, 115200, SERIAL_8N2, 1, 2, HOLDING_REG_SIZE,
HOLDING_REGISTERS);
//вводимо швидкість опитування, вага даних, що передаються дорівнює 8 біт,
один з яких старт-біт, а два стоп-біти
modbus_update_comms(115200, SERIAL_8N2, 1); modbus_update();
HOLDING_REGISTERS[0]=Relay3State; HOLDING_REGISTERS[1]=Input;
HOLDING_REGISTERS[2]=Output; HOLDING_REGISTERS[3]=Pila;
HOLDING_REGISTERS[4]=val; Setpoint = HOLDING_REGISTERS [5]; Kp =
HOLDING_REGISTERS [6]; HOLDING_REGISTERS[7]= Relay3State;
```