

БАКАЛАВРСЬКА РОБОТА

БР.АКП-62.00.00.000 ПЗ

Група АКП-23-2К

Вадим Стефурак

2025

Міністерство освіти і науки України
Івано-Франківський національний технічний університет нафти і газу
Факультет автоматизації та енергетики
Кафедра автоматизації та комп'ютерно-інтегрованих технологій

Стефурак Вадим Любомирович
(прізвище, ім'я, по батькові)

УДК 681.53
(індекс)

БАКАЛАВРСЬКА РОБОТА

Автоматизація реакторного блоку з використанням ІІоТ-технологій
(назва роботи)

Автоматизація та комп'ютерно-інтегровані технології
(назва освітньої програми)

174 - Автоматизація, комп'ютерно-інтегровані технології та робототехніка
(шифр і назва спеціальності)

Робота містить результати власних досліджень, використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Нормоконтроль

Доцент О.В. Кучмистенко
(посада) (підпис) (дата) (ініціали та прізвище)

Рецензент

Доцент В.С. Борин
(посада) (підпис) (дата) (ініціали та прізвище)

Здобувач освітнього ступеня

АКП-23-2К В.Л. Стефурак
(шифр групи) (підпис) (дата) (ініціали та прізвище)

Науковий керівник

Доцент М.І. Когутяк
(посада) (підпис) (дата) (ініціали та прізвище)

Допущено до захисту

Завідувач кафедри

Доцент А.І. Лагойда
(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківський національний технічний університет нафти і газу

(повне найменування закладу вищої освіти)

Факультет автоматизації та енергетики

Кафедра автоматизації та комп'ютерно-інтегрованих технологій

Освітній рівень перший (бакалаврський)

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

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри АКІТ

(А.І. Лагойда).

«___» _____ 20__ року

**З А В Д А Н Н Я
НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ**

Стефураку Вадиму Любомировичу

(прізвище, ім'я, по батькові)

1. Тема роботи Автоматизація реакторного блоку з використанням
ІоТ-технологій

керівник роботи Когутяк Мирослав Іванович, к.т.н., доцент

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

затверджені наказом закладу вищої освіти від « 07 » 05 20 25 року № 52/8

2. Строк подання студентом роботи 13.06.2025

3. Вихідні дані до роботи Матеріали практики, реалізація алгоритму
управління, технологічний регламент, автоматичне управління процесом
нагріву, методичні вказівки, інтернет ресурси

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно
розробити) _____

1 Аналіз процесу пастеризації та реакторних блоків

2. Математичне моделювання і ідентифікація об'єкта керування

3 Синтез та аналіз системи автоматичного керування

4. Розробка проектної документації реакторного блоку та його реалізація

5. Розробка схем та підбір ТЗА

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

Лист 1 – Структурна схема об'єкта – БР.АКП-62.00.00.000. 01

Лист 2 – Функціональна схема об'єкта – БР.АКП-62.00.00.000. 02

Лист 3 – Електрична-принципова схема підключень – БР.АКП-62.00.00.000. 03

Лист 4 – Дизайн передньої панелі – БР.АКП-62.00.00.000. 04

Лист 5 – Блок-схема роботи програми – БР.АКП-62.00.00.000. 05

Лист 6 – Математична модель – БР.АКП-62.00.00.000. 06

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання 06.11.2024

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Термін виконання етапів роботи	Примітка
1	Аналіз процесу пастеризації та реакторних блоків	15.01.2025 р.	
2	Математичне моделювання і ідентифікація об'єкта керування	23.02.2025 р.	
3	Синтез та аналіз системи автоматичного керування	31.03.2025 р.	
4	Розробка проектної документації реакторного блоку та його реалізація	05.04.2025 р.	
5	Розробка схем та підбір ТЗА	09.05.2025 р.	
6	Висновки по роботі	11.06.2025 р.	
7	Оформлення ПЗ	13.06.2025 р.	

Студент _____
(підпис)

Стефурак В.Л.
(ініціали та прізвище)

Керівник роботи _____
(підпис)

Когутяк М.І.
(ініціали та прізвище)

АНОТАЦІЯ

Бакалаврська робота містить: 54 сторінки друкованого тексту, 21 рисуноків, 1 таблиця, 11 посилань на джерела.

Тема: «Автоматизація реакторного блоку з використанням ІоТ-технологій».

Об'єкт дослідження: реакторний блок пастеризації фруктового пюре, що включає резервуари, клапани, помпи, нагрівальний елемент і систему контролю.

Мета роботи: створення автоматизованої системи керування пастеризаційним процесом з використанням технологій Інтернету речей (ІоТ), мікроконтролерів NodeMCU та Arduino Pro Mini, а також хмарного сервісу Adafruit ІО для дистанційного моніторингу та управління.

Методи дослідження: математичне моделювання, побудова структурних та функціональних схем, імітаційне моделювання у середовищі MATLAB/Simulink, розробка програмного забезпечення для мікроконтролерів, застосування MQTT-протоколу, синтез ПІ-регулятора.

Результати бакалаврської роботи: розроблено діючий макет автоматизованої системи керування пастеризацією, побудовано цифровий двійник об'єкта в середовищі MATLAB, реалізовано програмний код для контролерів, створено проект передньої панелі з функціональними зонами керування, забезпечено хмарну інтеграцію системи для віддаленого моніторингу параметрів процесу.

Ключові слова: пастеризація, ІоТ, NodeMCU, Arduino Pro Mini, автоматизація, температурний контроль, Adafruit ІО, цифровий двійник, MQTT, керування процесом.

ANNOTATION

Bachelor's thesis contains: 54 pages of printed text, 21 figures, 1 table, 11 references to sources.

Topic: "Automation of a reactor unit using IIoT technologies".

Object of research: reactor unit for pasteurization of fruit puree, including tanks, valves, pumps, heating element and control system.

Purpose of work: creation of an automated control system for the pasteurization process using Internet of Things (IIoT) technologies, NodeMCU and Arduino Pro Mini microcontrollers, as well as the Adafruit IO cloud service for remote monitoring and control.

Research methods: mathematical modeling, construction of structural and functional diagrams, simulation modeling in the MATLAB/Simulink environment, development of software for microcontrollers, application of the MQTT protocol, synthesis of a PI controller.

Results of the bachelor's thesis: a working model of an automated pasteurization control system was developed, a digital twin of the object was built in the MATLAB environment, program code for controllers was implemented, a front panel design with functional control zones was created, and cloud integration of the system for remote monitoring of process parameters was provided.

Keywords: pasteurization, IIoT, NodeMCU, Arduino Pro Mini, automation, temperature control, Adafruit IO, digital twin, MQTT, process control.

ЗМІСТ

ПЕРЕЛІК ОСНОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ	9
ВСТУП	10
1. АНАЛІЗ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ АВТОМАТИЗАЦІЇ РЕАКТОРНОГО БЛОКУ З ВИКОРИСТАННЯМ ІОТ-ТЕХНОЛОГІЙ ЯК ОБ'ЄКТА АВТОМАТИЧНОГО КОНТРОЛЮ І КЕРУВАННЯ.....	12
1.1. Характеристика реакторних блоків.....	12
1.2. Обґрунтування вибору тематики	14
1.3. Технологічний процес пастеризації фруктового пюре	15
1.4. Об'єкт і предмет дослідження.....	16
1.5. Аналіз закордонних та вітчизняних рішень	16
1.6. Вибір апаратної платформи	18
1.7. Вибір протоколів зв'язку	19
1.8. Вибір хмарного сервісу	20
1.9. Постановка задачі	20
Висновки до розділу.....	21
2. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ І ІДЕНТИФІКАЦІЯ ОБ'ЄКТА КЕРУВАННЯ	22
2.1. Вибір об'єкта керування та визначення його структурної схеми	22
2.2. Вимоги до системи керування.....	22
2.3. Математична модель.....	24
Висновки до розділу.....	26

					БР.АКП-62.00.00.000 ПЗ			
Змн.	Лист	№ докум.	Підпис	Дата	Автоматизація реакторного блоку з використанням Іот-технологій	Літ.	Арк.	Акрушів
Розроб.	Стефурак В.Л.							
Перевір.	Когутяк М.І.						7	55
Реценз.	Борин В.С.					АКП-23-2К ІФНТУНГ		
Н. Контр.	Кучмистенко О.В.							
Затверд.	Лагойда А.І.							

3. СИНТЕЗ ТА АНАЛІЗ САК.....	27
3.1. Вибір та обґрунтування алгоритму керування. Визначення параметрів налаштування регуляторів	27
3.2. Синтез в середовищі MatLab/Simulink	28
3.3. Аналіз результатів	29
Висновки до розділу.....	30
4. РОЗРОБКА ПРОЕКТНОЇ СКЛАДОВОЇ САК	31
4.1. Вибір технічних засобів.....	31
4.2. Структурна схема системи	38
4.3. Функціональна схема системи	40
4.4. Електрична-принципова схема.....	42
4.5. Проектування дизайну передньої панелі	44
4.6. Код програми для NodeMCU8266	47
4.7. Код програми Arduino Pro Mini.....	49
Висновки до розділу.....	50
ЗАГАЛЬНІ ВИСНОВКИ	51
ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА	53

ПЕРЕЛІК ОСНОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

ІоТ – Індустріальний Інтернет речей (Industrial Internet of Things).

PI – Пропорційно-інтегральний регулятор.

ПІ – Пропорційно-інтегральний регулятор (український аналог PI).

ТЕН – Тепловий електронагрівач.

MQTT – Протокол телеметрії з чергою повідомлень (Message Queuing Telemetry Transport).

DS18B20 – Цифровий датчик температури з інтерфейсом OneWire.

UART – Уніфікований асинхронний прийомопередавач (послідовний інтерфейс для передачі даних).

NodeMCU – Мікроконтролерна плата на базі ESP8266 з вбудованим Wi-Fi-модулем.

Arduino Pro Mini – Компактна мікроконтролерна плата на базі ATmega328P.

Adafruit IO – Хмарна платформа для зберігання та візуалізації даних з ІоТ-пристроїв.

LCD – Рідкокристалічний дисплей (Liquid Crystal Display).

PWM – Широтно-імпульсна модуляція (Pulse Width Modulation).

I2C – Інтерфейс міжмікросхемного зв'язку (Inter-Integrated Circuit).

SPI – Послідовний периферійний інтерфейс (Serial Peripheral Interface).

CAN – Контролерна мережа (Controller Area Network) для промислових систем.

SCADA – Система диспетчерського контролю і збору даних (Supervisory Control and Data Acquisition).

PLC – Програмований логічний контролер (Programmable Logic Controller).

PID – Пропорційно-інтегрально-диференціальний регулятор.

pH – Показник кислотності середовища (логарифмічна міра концентрації йонів H⁺).

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

ВСТУП

Актуальність теми. У сучасній харчовій промисловості ключовим викликом є забезпечення високої якості продукції при мінімальних енергетичних і ресурсних витратах. Одним із найважливіших етапів обробки фруктового пюре є пастеризація — процес, що вимагає точного контролю температури та часу для знищення мікроорганізмів без втрати корисних властивостей продукту. Впровадження технологій Інтернету речей (ІоТ) у цю сферу дозволяє реалізувати віддалений моніторинг, автоматизоване керування і збір даних у реальному часі. Це, своєю чергою, сприяє підвищенню ефективності, гнучкості й безпеки технологічного процесу. Розробка ІоТ-рішень для автоматизації реакторного блоку пастеризації є актуальною задачею, яка відповідає сучасним вимогам цифровізації виробництва.

Зв'язок роботи з науковими програмами, планами і темами. Робота узгоджується з напрямками розвитку інтелектуальних систем автоматизації, цифрового моделювання технологічних об'єктів і впровадження ІоТ у промисловість. Проект підтримує ініціативи щодо енергоефективного виробництва, екологічної безпеки та інтеграції хмарних сервісів у керування харчовими технологіями.

Мета і задачі дослідження. Метою роботи є створення автоматизованої системи керування реакторним блоком пастеризації фруктового пюре з використанням ІоТ-технологій.

Для досягнення мети передбачено вирішення таких задач:

- аналіз технологічного процесу пастеризації та вимог до автоматизації;
- вибір апаратної платформи та засобів зв'язку;
- побудова структурної і функціональної схеми системи;
- розробка цифрового двійника реакторного блоку;
- реалізація алгоритмів керування на основі NodeMCU та Arduino Pro Mini;

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

- налаштування хмарного сервісу (Adafruit IO) для віддаленого моніторингу й керування;
- тестування системи на макеті та оцінка її ефективності.

Об'єкт дослідження. Об'єктом дослідження є реакторний блок пастеризації фруктового пюре, який включає в себе резервуари, нагрівальний елемент, клапани, помпи та систему змішування.

Предмет дослідження. Предметом дослідження є система автоматизованого керування об'єктом пастеризації на базі IoT, що включає сенсори, виконавчі пристрої, контролери та засоби віддаленого доступу.

Методи дослідження. У роботі застосовано методи математичного моделювання, побудови структурних схем, лінеаризації об'єктів керування, синтезу ПІ-регуляторів, а також використання інструментів мікроконтролерного програмування та обміну даними по MQTT-протоколу.

Наукова новизна. Наукова новизна полягає у створенні функціонального макету реакторного блоку з цифровим двійником, що дозволяє здійснювати автоматичне керування на базі дешевого, відкритого обладнання з підтримкою хмарних сервісів.

Практичне значення. Розроблена система може бути впроваджена у невелике виробництво або використана як навчальний приклад сучасної IoT-автоматизації. Вона забезпечує дистанційний контроль температури, рН, стану виконавчих механізмів та дозволяє здійснювати гнучке налаштування параметрів без потреби фізичної присутності оператора.

Структура роботи. Бакалаврська робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел і додатків. У роботі наведено структурні, функціональні та електричні схеми, реалізовано програмне забезпечення для контролерів, а також описано процес моделювання та тестування розробленої системи.

					БР.АКП-62.00.00.000 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

1. АНАЛІЗ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ АВТОМАТИЗАЦІЇ РЕАКТОРНОГО БЛОКУ З ВИКОРИСТАННЯМ ІОТ-ТЕХНОЛОГІЙ ЯК ОБ'ЄКТА АВТОМАТИЧНОГО КОНТРОЛЮ І КЕРУВАННЯ

1.1. Характеристика реакторних блоків

Реакторні блоки періодичної дії (Batch reactors). Процес проводиться окремими партіями. Після завантаження сировини виконуються усі технологічні операції (змішування, нагрівання, витримка), після чого готовий продукт вивантажується, і цикл повторюється.

Переваги:

- Простота реалізації;
- Гнучкість у налаштуваннях рецептур;
- Можливість ручного або автоматичного керування.

Недоліки:

- Тривалий час обробки;
- Вища трудомісткість;
- Можливі температурні коливання.

Підходять для невеликих виробництв, експериментальних установок або коли важливо часто змінювати склад суміші. У пастеризації пюре такий блок зручний при роботі з обмеженим обсягом або змінною рецептурою (наприклад, різні види фруктів).

Реакторні блоки безперервної дії (Continuous flow reactors). Сировина безперервно подається в реактор, обробляється в русі, після чого готовий продукт постійно виводиться. Уся система працює в режимі сталої динаміки.

Переваги:

- Висока продуктивність;
- Енергоефективність;
- Простота автоматизації (наприклад, за допомогою ПІД-регуляторів і ІоТ).

Недоліки:

- Менша гнучкість у рецептурі;

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ док.ум.	Підпис	Дата		12

- Складніша система налаштувань;
- Вища залежність від надійності датчиків і виконавчих механізмів.

Ці блоки є основою великих виробничих ліній. У пастеризації пюре дозволяють точно контролювати температуру та час витримки, що особливо важливо для збереження органолептичних властивостей.

Реакторні блоки з рециркуляцією (Recirculating reactors). Продукт багаторазово прокачується через реакційну зону до досягнення заданих параметрів. Такий принцип забезпечує кращу однорідність та ефективність обробки.

Переваги:

- Рівномірність обробки;
- Можливість гнучкого контролю;
- Підвищення ефективності теплообміну.

Недоліки:

- Вища енерговитратність (через повторну циркуляцію);
- Потреба в потужних насосах та точних зворотних клапанах;
- Складніша логіка керування.

Доцільні для густих продуктів, таких як фруктове пюре, де потрібна гарантована термічна обробка всієї маси без локального перегріву. Рециркуляція дозволяє точно витримати час пастеризації та уникнути застійних зон.

Модульні реакторні блоки (Modular reactors). Система збирається з окремих модулів: подачі сировини, попереднього нагріву, реакційної камери, охолодження, тощо. Кожен модуль має окремий блок керування, а взаємодія здійснюється через спільну мережу (наприклад, MQTT або HTTP з використанням NodeMCU/Arduino та Adafruit IO).

Переваги:

- Масштабованість;
- Простота обслуговування;
- Легке впровадження IoT-рішень;

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докum.	Підпис	Дата		13

- Можливість віддаленого моніторингу.

Недоліки:

- Вища вартість (на старті);
- Необхідність налаштування взаємодії між модулями;
- Залежність від стабільності зв'язку.

Ідеальні для побудови інтелектуальних систем керування. Наприклад, у твоєму випадку — з використанням NodeMCU для дистанційного керування ТЕНами, клапанами та моніторингу температури у реальному часі через Adafruit IO.

Кожен тип реакторного блоку має своє оптимальне поле застосування. Вибір залежить від виду продукту, обсягів виробництва та вимог до гнучкості системи. В умовах впровадження ІоТ-автоматизації особливої актуальності набувають модульні та безперервні системи, які забезпечують високу точність, адаптивність і можливість інтеграції з хмарними платформами.

Таким чином, правильне проектування та автоматизація реакторного блоку — це не лише про ефективність, а й про надійність, масштабованість та готовність до цифрової трансформації виробництва.

1.2. Обґрунтування вибору тематики

У сфері переробки харчової сировини особливої уваги потребують процеси, що забезпечують мікробіологічну стабільність продукції при збереженні її поживної цінності та смакових характеристик. Пастеризація фруктового пюре, як один із ключових етапів технологічного циклу, виконує ці завдання шляхом термічного впливу з контрольованими параметрами температури та часу.[1]

Проте традиційні системи керування пастеризаційними установками, що базуються на централізованій логіці з жорстко закладеною структурою, виявляються технічно складними в масштабуванні, модернізації та віддаленому обслуговуванні. В умовах сучасних виробничих викликів доцільно орієнтуватися на децентралізовані підходи, зокрема ті, що ґрунтуються на

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

технологіях промислового Інтернету речей (IIoT).

IIoT-рішення дозволяють інтегрувати мікроконтролерні пристрої, системи хмарної аналітики та мобільні засоби інтерфейсу в єдину адаптивну екосистему. Це відкриває можливості для гнучкого моніторингу, енергозберігаючого керування та оптимізації виробничих режимів у реальному часі з мінімальними витратами. [2]

1.3. Технологічний процес пастеризації фруктового пюре

Виробничий процес отримання фруктового пюре з використанням пастеризації є ключовим етапом у харчовій галузі, оскільки він дозволяє отримати продукт із тривалим терміном придатності та стабільними якісними характеристиками. Основна функція технологічної установки полягає в поетапному змішуванні фруктової сировини з водою й харчовими компонентами, термічній обробці суміші для знешкодження патогенних мікроорганізмів та подальшій стабілізації продукту з метою його зберігання чи розфасовки.

Наочним прикладом може слугувати виробничий цикл приготування яблучного пюре, що є базовим інгредієнтом у виготовленні сокової продукції, десертів і дитячого раціону. Процес умовно поділяється на два ключові блоки: змішування інгредієнтів і пастеризація.

Установку умовно складають наступні елементи:

- Перший резервуар, призначений для змішування: до нього подаються фруктове пюре, питна вода та необхідні добавки. Ємність оснащена мішалкою.
- Другий резервуар, де здійснюється пастеризація суміші. Він обладнаний нагрівальним елементом і сенсором температури.
- Група насосів і клапанів забезпечує транспортну функцію між ємностями.
- Контрольна система на базі мікроконтролера відповідає за виконання послідовності технологічних операцій, а також регулює

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

параметри температури та часу.

- Додаткові вимірювальні модулі, включають рН-метр, витратоміри, датчики рівня, а також, за необхідності, теплообмінник.

Така архітектура установки дозволяє організувати технологічний процес у напівавтоматизованому або повністю автоматичному режимі з гарантованим забезпеченням стабільної якості кінцевого продукту.

1.4.Об'єкт і предмет дослідження

Об'єкт дослідження — термічний модуль пастеризації фруктового пюре, що є частиною умовно неперервного технологічного процесу.

Предмет дослідження — технічні засоби автоматизованого керування та алгоритми, що забезпечують ефективну взаємодію апаратної платформи з цифровим середовищем передачі та обробки даних.

1.5. Аналіз закордонних та вітчизняних рішень

Серед визнаних світових виробників обладнання для переробки фруктів і овочів варто виділити італійську компанію Bertocchi S.r.l., яка понад чотири десятиліття розробляє технології для виробництва пюре, соків, нектарів, концентратів та джемів. Однією з найінноваційніших ліній Bertocchi є серія UCX Turbo Extractors, що реалізує процес холодної екстракції сировини при кімнатній температурі.[4]

Ці установки орієнтовані на збереження максимальної харчової цінності кінцевого продукту. Застосовується фірмова технологія Rotorpulse®, яка забезпечує м'яке подрібнення без термічного впливу, що дозволяє уникнути руйнування вітамінів, ферментів і ароматичних сполук. Сфера застосування включає яблука, груші, персики (без кісточок), абрикоси, ягоди, тропічні фрукти та різні овочі.

Основні характеристики Bertocchi UCX:

- Продуктивність: від 1000 кг/год до 24 тонн/год залежно від моделі;

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

- Тип обробки: холодна екстракція без нагріву;
- Форма кінцевого продукту: фруктово-овочеve пюре високої якості;
- Автоматизація: передбачено застосування PLC, сенсорного керування (HMI) та можливість інтеграції з SCADA-системами;
- Гігієнічність: спрощена система очищення та доступ до всіх критичних вузлів обслуговування.

Таке рішення дозволяє реалізувати безперервне виробництво з контролем у реальному часі.

Сучасні українські виробники також здійснюють кроки в напрямку автоматизації виробничих процесів. Яскравим прикладом є фермерське господарство «Гадз» [5], розташоване в Тернопільській області, яке входить до числа найбільших виробників яблук в Україні. Для переробки сировини підприємство використовує обладнання італійської компанії Rossi, що дозволяє виробляти пюре, соки та супутні продукти.[6]

Установлена технологічна лінія має дві незалежні секції: одну — для отримання соку прямого віджиму, іншу — для виробництва фруктового пюре. Керування процесом реалізується за допомогою автоматизованих систем управління, що мінімізують втручання персоналу, підвищують точність дотримання параметрів та загальну ефективність виробництва. Завдяки впровадженню сучасного іноземного обладнання, підприємство розширило свій асортимент — сьогодні крім основної продукції, воно випускає джеми, дитяче харчування, фруктові пасти та муси.\

Порівняння двох рішень — UCX-серії Bertocchi та лінії Rossi, використаної у ФГ «Гадз», — дозволяє виявити спільні риси й відмінності. Обидва варіанти демонструють високий рівень автоматизації, що відповідає сучасним вимогам до гігієнічності, ефективності та якості продукції. Проте Bertocchi акцентує увагу на холодній екстракції як способі зберегти біоактивні компоненти у фруктовому пюре, тоді як Rossi — на універсальності та термічній стабілізації.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		17

Таблиця 1.1 - Порівняльна характеристика

Параметр	Bertocchi UCX Turbo Extractors	ФГ «Гадз» (лінія Rossi)
Продуктивність	1–24 т/год	близько 5 т/год
Тип обробки	Холодна екстрація	Гаряча обробка
Рівень автоматизації	Високий (PLC, SCADA, HMI)	Високий (автоматизовані контролери)
Збереження харчової цінності	Максимальне	Високе
Гнучкість налаштування	Висока	Середня
Вимоги до обслуговування	Низькі	Середня

Досвід ФГ «Гадз» свідчить про успішне впровадження європейських технологій у вітчизняні умови. Такий підхід може стати прикладом для подальшої модернізації підприємств галузі, особливо з урахуванням потреб у виробництві дитячого харчування та продуктів з підвищеною біологічною цінністю.

1.6. Вибір апаратної платформи

Для розробки системи автоматизованого керування пастеризаційним процесом у межах реакторного блоку було прийнято рішення використовувати підхід, базований на технологіях промислового інтернету речей (Industrial Internet of Things, IIoT). Основними критеріями, які вплинули на вибір апаратної частини, стали: підтримка бездротового зв'язку, енергоефективність, можливість інтеграції з хмарними платформами, відкритість програмного середовища та масштабованість системи.

У якості центрального та допоміжного вузлів обрано:

- NodeMCU (ESP8266) — як головний контролер, що відповідає за підключення до мережі та обмін даними з хмарою;
- Arduino Pro Mini — як периферійний модуль для зчитування параметрів із сенсорів та управління виконавчими елементами.

NodeMCU (ESP8266)

Цей мікроконтролер був обраний завдяки вбудованому Wi-Fi-модулю,

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

який дозволяє безпосередньо під'єднувати пристрій до мережі без додаткових засобів зв'язку. Також він підтримує MQTT-протокол, що є ключовим для IoT-рішень. Плата має достатню обчислювальну потужність (до 160 МГц), оперативну пам'ять і стабільно працює з численними бібліотеками Arduino IDE, що значно спрощує розробку.

Arduino Pro Mini

Компактний мікроконтролер з низьким енергоспоживанням, який забезпечує зчитування даних з аналогових та цифрових датчиків. У системі він виконує функцію локального виконавчого пристрою та працює в парі з NodeMCU, обмінюючись інформацією за допомогою послідовного інтерфейсу (UART).

Функціональні задачі Arduino Pro Mini:

- Моніторинг показників (температура, рН, рівень рідини);
- Керування реле або клапанами;
- Передача зібраних даних на головний вузол.

Вибір такої конфігурації дає змогу реалізувати розподілену систему управління, де основна логіка контролю реалізована в NodeMCU, а низькорівневе зчитування параметрів виконує Arduino Pro Mini.

1.7. Вибір протоколів зв'язку

Внутрішня взаємодія: UART

Для організації обміну даними між NodeMCU і Arduino Pro Mini застосовано послідовний інтерфейс UART, який дозволяє передавати значення з датчиків (температура, рівень, рН) у простому текстовому форматі. Цей протокол є енергоефективним, має невелике навантаження на ресурси мікроконтролерів і легко реалізується програмно.

Зовнішня комунікація: MQTT

Для зв'язку з хмарним середовищем (Adafruit IO) було обрано протокол MQTT (Message Queuing Telemetry Transport). Це полегшений протокол публікації/підписки, який ідеально підходить для пристроїв із обмеженими

					БР.АКП-62.00.00.000 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

ресурсами.

Основні переваги MQTT:

- Низьке енергоспоживання та трафік;
- Підтримка QoS (Quality of Service), що дозволяє забезпечити гарантовану доставку;
- Робота в асинхронному режимі (непостійне з'єднання);
- Простота реалізації через бібліотеку Adafruit_MQTT.

Завдяки MQTT, NodeMCU може не лише надсилати телеметричні дані, але й приймати команди від користувача через інтерфейс Adafruit IO, реалізуючи зворотний зв'язок.

1.8. Вибір хмарного сервісу

Серед багатьох хмарних платформ для роботи з IoT (Blynk, Firebase, ThingSpeak) було обрано Adafruit IO як найбільш сумісне рішення з NodeMCU, орієнтоване на MQTT-комунікації.

Переваги Adafruit IO:

- Гнучка підтримка MQTT/HTTP;
- Графічне налаштування dashboard для відображення значень датчиків та інтерактивного керування;
- Можливість зберігання історії даних (графіки, лог-файли);
- Вбудована система авторизації через API-ключ;
- Безкоштовний базовий обліковий запис (до 30 днів збереження даних і 30 feed'ів);
- Просте підключення через бібліотеки Adafruit_MQTT та WiFiClient.

У разі втрати зв'язку NodeMCU з мережею передбачено буферизацію даних локально, з подальшим відновленням синхронізації, що підвищує надійність роботи всієї системи.[7]

1.9. Постановка задачі

Метою дослідження є розробка прикладного рішення для автоматизації

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

реакторного модуля пастеризації пюреподібних харчових продуктів із використанням інструментарію ІоТ. У фокусі — застосування мікроконтролерів NodeMCU та Arduino Pro Mini для збору даних, керування процесом та передавання інформації до хмарної платформи Adafruit ІО.

У межах поставленої мети передбачено реалізацію наступних завдань:

- дослідити специфіку термічного оброблення фруктового пюре;
- здійснити порівняльний аналіз доступних технічних рішень для автоматизації;
- побудувати структурну модель розподіленої системи керування;
- реалізувати демонстраційний прототип із функцією телеметрії;
- створити математичну модель динаміки процесу та виконати її моделювання;

Висновки до розділу

У першому розділі проведено всебічний аналіз технологічного процесу пастеризації фруктового пюре як об'єкта автоматичного керування. Розглянуто класифікацію та функціональні особливості реакторних блоків, визначено об'єкт і предмет дослідження. Було обґрунтовано доцільність використання ІоТ-технологій для автоматизації та моніторингу процесу. На основі аналізу сучасних рішень обрано апаратну платформу (NodeMCU та Arduino Pro Mini), протоколи зв'язку (MQTT) та хмарний сервіс (Adafruit ІО), що забезпечують гнучкість, масштабованість і віддалений доступ до системи керування.

					БР.АКП-62.00.00.000 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

2. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ І ІДЕНТИФІКАЦІЯ ОБ'ЄКТА КЕРУВАННЯ

2.1. Вибір об'єкта керування та визначення його структурної схеми

Для цього курсового проектування було обрано об'єкт керування, що складається з двох резервуарів, де відбувається змішування, нагрівання та перекачування рідини. До першого резервуару підводяться два клапани, через які подається сировина, а змішувач забезпечує необхідну однорідність суміші. Перекачування суміші з резервуару 1 до резервуару 2 здійснюється за допомогою насоса. У резервуарі 2 відбувається нагрівання рідини, після чого вона перекачується в інгу ємність. Для вимірювання рівня рідини в резервуарах застосовується датчик рівня, розташований посередині між ними. Цей об'єкт є типовим для технологічних процесів, де необхідно здійснювати контроль та управління рівнем, температурою та потоками рідини.

Структурна схема об'єкта керування включає:

- два резервуари,
- клапани для подачі сировини,
- змішувач,
- насос для перекачування рідини,
- датчик рівня, що контролює рівень рідини в резервуарах,
- систему нагріву для другого резервуару.

2.2. Вимоги до системи керування

Ефективність та якість виробництва фруктового пюре з пастеризацією значною мірою залежать від точності підтримання технологічних параметрів на всіх етапах процесу. Для забезпечення стабільної якості продукції та відповідності вимогам безпеки харчових продуктів необхідно обрати та обґрунтувати ключові параметри контролю та управління.

Контроль температури пастеризації

Параметр: температура продукту під час пастеризації.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ док.ум.	Підпис	Дата		22

Номінальне значення: 85–90 °С.

Обґрунтування: Це температурне вікно забезпечує ефективне знищення патогенних мікроорганізмів і ферментів, що можуть викликати псування. Зниження температури призведе до неповної пастеризації, тоді як перевищення може спричинити втрату смакових властивостей та руйнування вітамінів.

Тип контролю: безперервний (датчик температури, PID-регулятор, керування через ПЛК).

Час витримки при пастеризації

Параметр: тривалість перебування продукту при цільовій температурі.

Номінальне значення: 2–3 хвилини.

Обґрунтування: Цей проміжок часу забезпечує необхідний ступінь пастеризації без надмірного нагрівання. Надто короткий час не гарантує мікробіологічної безпеки, а надто довгий — призведе до зниження якості.

Тип контролю: таймер з прив'язкою до температурного контролю, реалізується за допомогою ПЛК з відповідним програмним забезпеченням.

Температура зберігання

Параметр: температура у зоні охолодження або зберігання.

Номінальне значення: 0...+4 °С (для охолодженого пюре), до +25 °С (для пастеризованого в герметичній тарі).

Обґрунтування: Збереження стабільної температури попереджає вторинне зараження продукту та забезпечує тривалий термін придатності.

Тип контролю: термостат, температурний датчик з сигналізацією відхилень, передача сигналу на ПЛК для реакції системи охолодження.

Рівень рН пюре

Параметр: кислотність суміші.

Оптимальне значення: рН 3.5–4.0.

Обґрунтування: Визначає стабільність продукту та ефективність пастеризації. При вищому рН зростає ризик розвитку мікроорганізмів.

Тип контролю: рН-метр, періодичний контроль оператором або автоматичний моніторинг із передачею даних на ПЛК.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ док.ум.	Підпис	Дата		23

Усі вищезгадані параметри можуть бути інтегровані в єдину систему автоматичного керування на базі програмованого логічного контролера (ПЛК), що забезпечує надійність, гнучкість і можливість адаптації до змін технології. ПЛК дозволяє реалізувати логіку контролю, аварійну сигналізацію, архівацію даних та віддалений моніторинг процесу.

2.3. Математична модель

Математичне моделювання об'єкта автоматизації передбачає перехід від опису фізичних процесів до формалізованої моделі у вигляді диференціального рівняння або, що більш зручно для системного аналізу, — до передавальної функції. Передавальна функція є відношенням зображення вихідного сигналу до зображення вхідного сигналу у частотній області (зазвичай у просторі Лапласа), за нульових початкових умов. Виведення передавальної функції проводиться за результатами аналізу реакції об'єкта на тестовий сигнал, зазвичай — одиничний стрибок (ступінчаста функція Хевісайда). У межах даної роботи такий стрибок було реалізовано подачею постійної потужності на нагрівальний елемент (ТЕН) реакторного блоку, а результатом спостереження стала температурна реакція середовища.

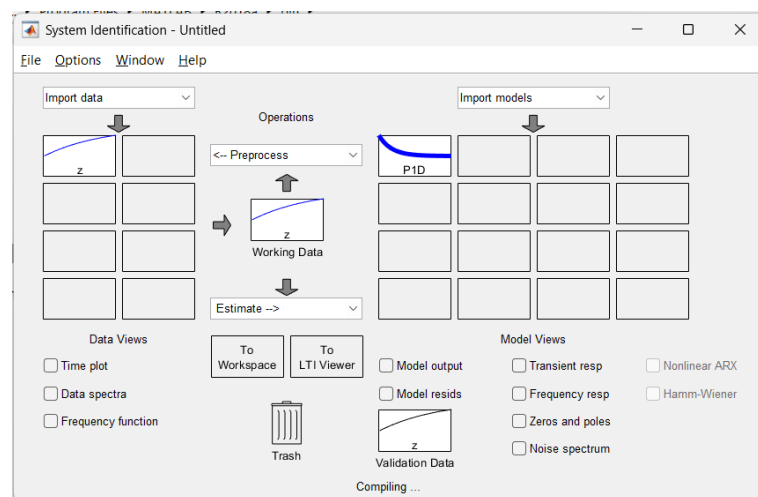


Рисунок 2.1 - Інструмент System Identification Toolbox

Отримані експериментальні (або теоретично апроксимовані) дані були оброблені в середовищі MATLAB, зокрема за допомогою інструменту System Identification Toolbox рис. 2.1, який дозволяє побудувати передавальні функції

					БР.АКП-62.00.00.000 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

об'єкта на основі наявних часових рядів.

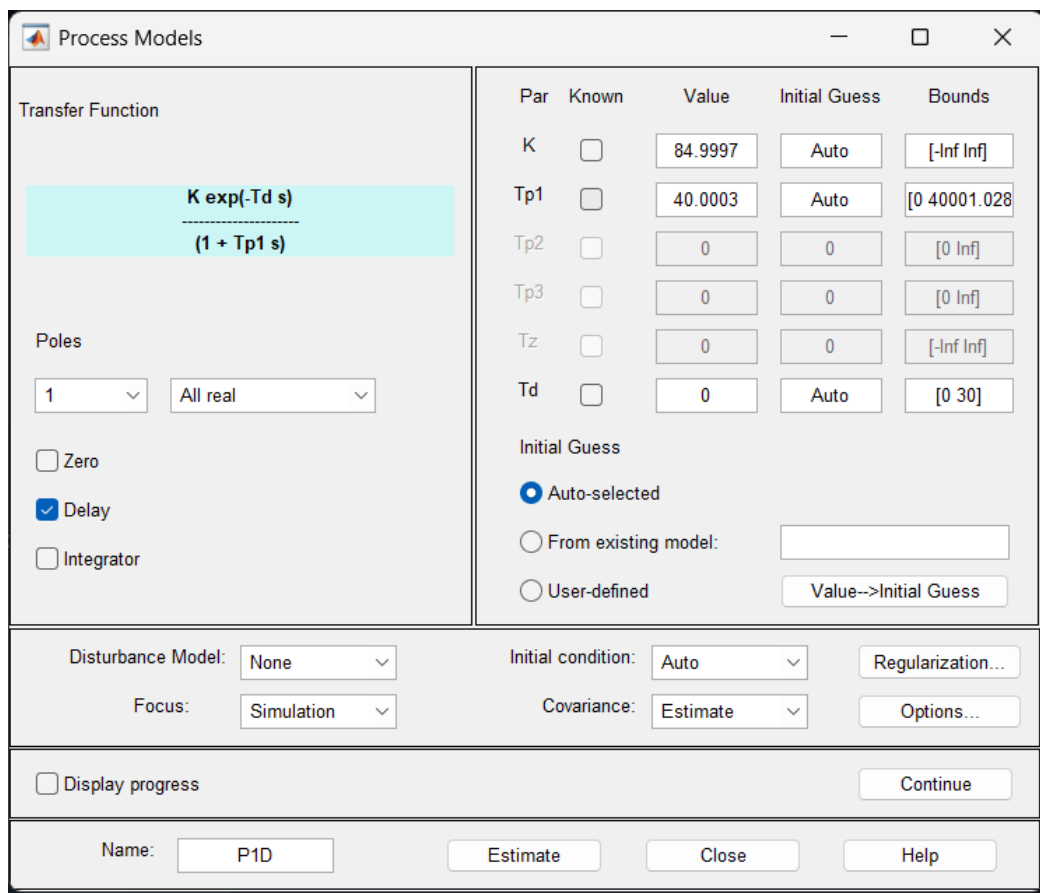


Рисунок 2.2 - Вікно налаштування моделі

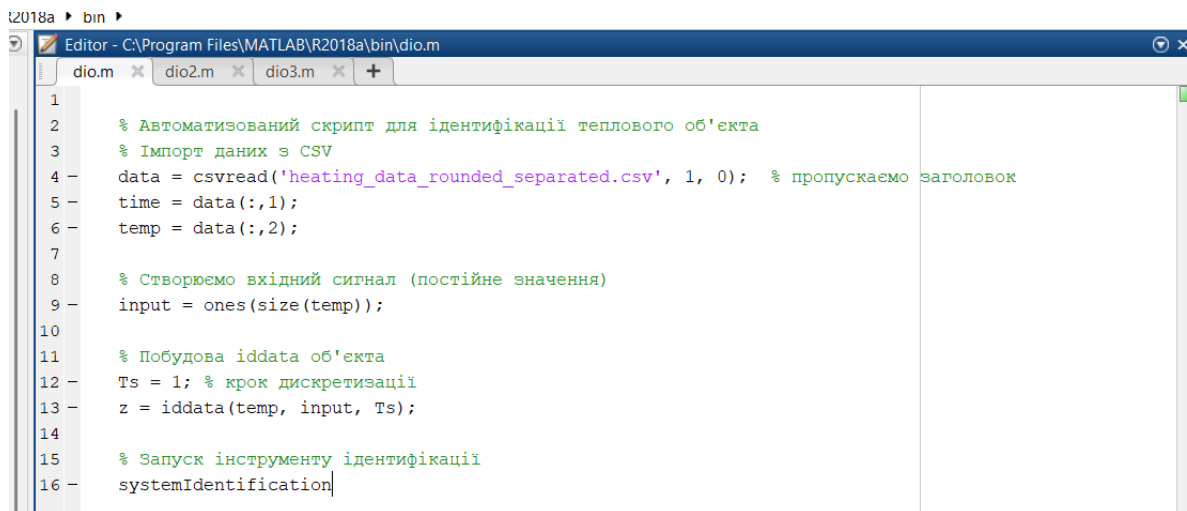


Рисунок 2.3 - Програма для оброблення експериментальних даних

Після імпортування даних було обрано тип моделі — об'єкт першого порядку з затримкою (P1D) рис. 2.2, після чого виконано автоматичне налаштування параметрів. В результаті ідентифікації отримано передавальну функцію виду:

									Арк.
									25
Змн.	Арк.	№ докум.	Підпис	Дата					

$$G(s) = \frac{84.9997}{40.0003s + 1} \quad (3.1)$$

де чисельник відповідає статичному коефіцієнту підсилення системи (K), а знаменник містить сталу часу (τ), яка відображає інерційність теплового процесу. Така модель забезпечує високу точність апроксимації — понад 95% відповідності реальним даним, що свідчить про її придатність для подальшого синтезу системи автоматичного регулювання. На рисунку X представлено код програми.

Висновки до розділу

У цьому розділі було побудовано структурну модель реакторного блоку пастеризації, визначено вимоги до системи керування та здійснено математичне моделювання об'єкта з урахуванням теплових процесів. Ідентифіковано передавальну функцію системи на основі експериментальних даних, що забезпечує основу для подальшого синтезу регулятора. Отримані результати дозволяють ефективно адаптувати керуючу систему до особливостей процесу пастеризації.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

3. СИНТЕЗ ТА АНАЛІЗ САК

3.1. Вибір та обґрунтування алгоритму керування. Визначення параметрів налаштування регуляторів

Синтез системи автоматичного керування передбачає не лише побудову моделі об'єкта, а й вибір оптимального алгоритму регулювання, який забезпечуватиме виконання вимог до точності, стійкості, швидкодії та простоти реалізації. У випадку термічних процесів, які відзначаються значною інерційністю, найчастіше доцільним є використання пропорційно-інтегрального регулятора (ПІ). На відміну від повного ПІД-регулятора, ПІ-регулятор не містить диференціальної складової, що суттєво спрощує як його технічну реалізацію, так і забезпечує підвищену стабільність у системах, де присутні затримки та повільні динамічні зміни, як-от у теплових установках.

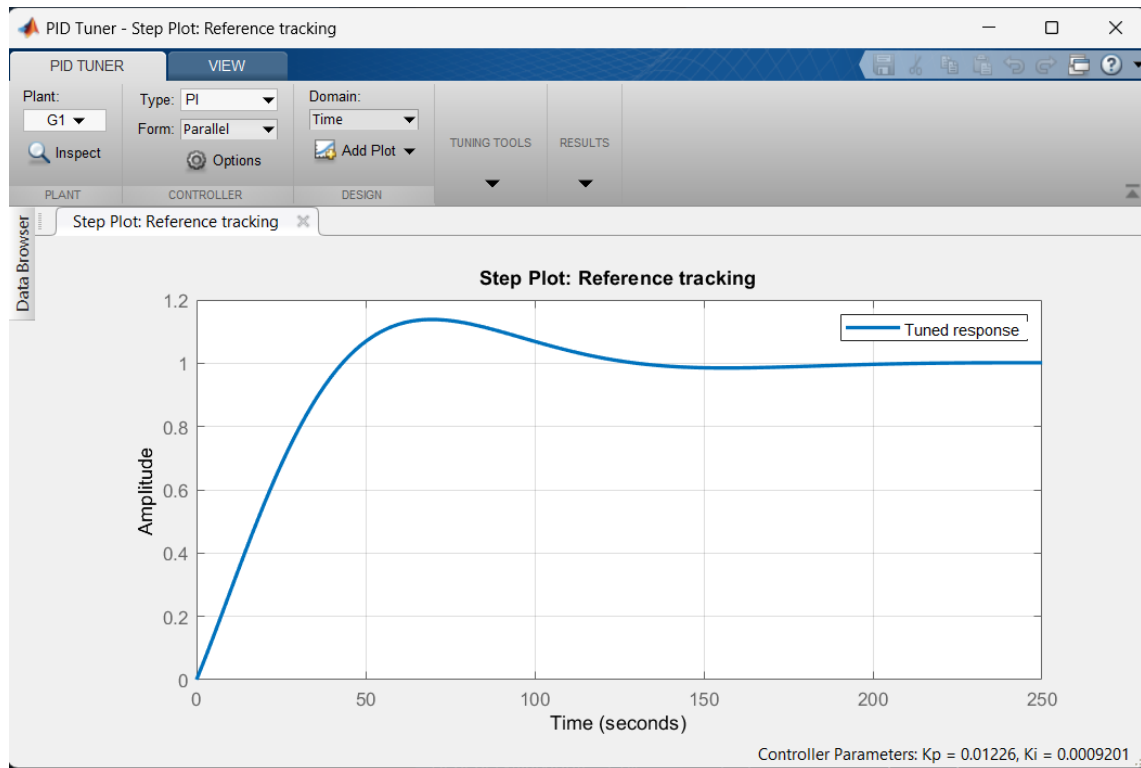


Рисунок 3.1 - Вікно програми PID Tuner

Перевагою ПІ-регулятора є його здатність забезпечувати не лише пропорційне реагування на похибку (миттєве відхилення від заданого значення), а й її повне нівелювання у сталому режимі завдяки інтегральній складовій. Це дозволяє усувати статичну похибку та досягати точного дотримання референсного значення. Для налаштування ПІ-регулятора у даній роботі було

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		27

використано вбудований інструмент PID Tuner у середовищі MATLAB/Simulink. З його допомогою виконано автоматичний підбір коефіцієнтів K_p і K_i на основі ідентифікованої динамічної моделі об'єкта. Регулятор було налаштовано так, щоб забезпечити збалансовану поведінку системи — мінімальне перерегулювання при достатньо короткому часі встановлення, що підтверджено подальшим моделюванням.

Таким чином, застосування ПІ-регулятора у даному контексті є виваженим компромісом між складністю, стабільністю та ефективністю, що забезпечує надійне автоматичне регулювання температури в умовах реального виробничого процесу. Нижче на рис. 3.2 код програми.

```

1 % ПЕРЕДУМОВА: модель G1 вже має бути в Workspace, наприклад:
2 % G1 = tf(85, [40 1]);
3
4
5 % Перевіримо наявність G1
6 if ~exist('G1', 'var')
7     error('Модель G1 не знайдено в Workspace. Експортуй її з System Identification.');
```

Рисунок 3.2 - Запуск PID Tuner

3.2. Синтез в середовищі MatLab/Simulink

Побудовано імітаційну модель у середовищі Simulink, яка реалізує зворотний зв'язок: Step → Sum → PI → Об'єкт → Scope, де об'єкт описано передавальною функцією $G(s)$.

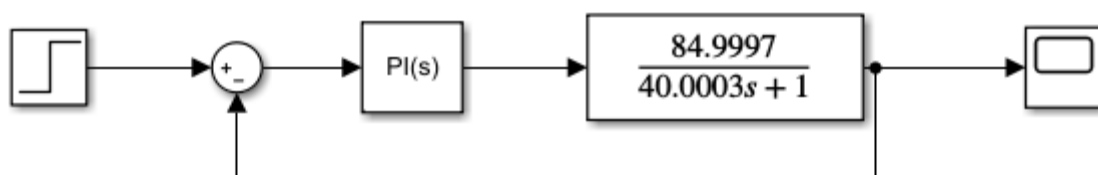


Рисунок 3.3 - Побудована модель у Simulink

Після запуску моделі в середовищі Simulink з тривалістю 300 секунд, було отримано перехідну характеристику системи з PI-регулятором.

Осцилограма демонструє:

- швидке зростання вихідного сигналу;
- перерегулювання $\approx 15\%$;
- стабілізацію біля значення 1 після ~ 200 секунд;
- відсутність коливань у сталому режимі.

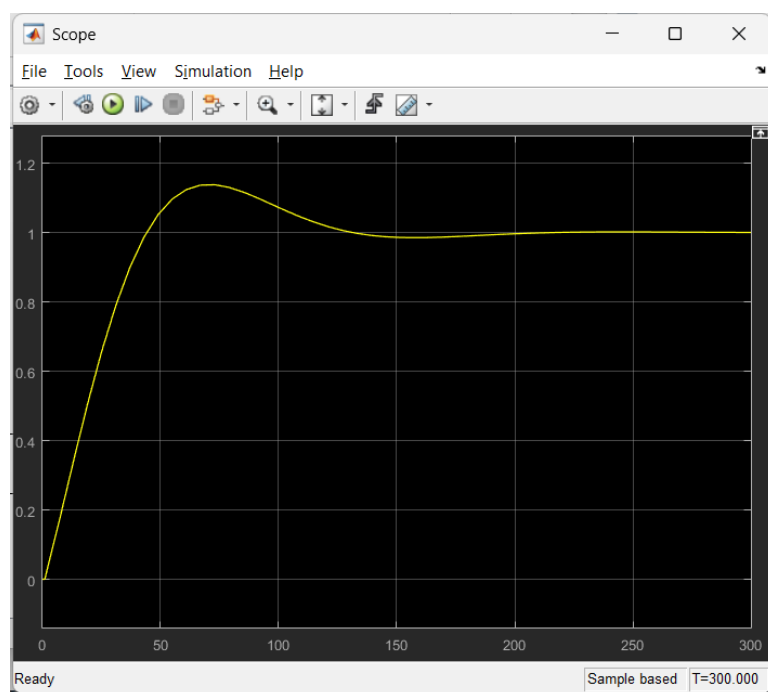


Рисунок 3.4 - Результат запуску моделі

3.3. Аналіз результатів

Проведене моделювання дозволило зробити низку важливих висновків щодо динамічної поведінки системи автоматичного регулювання температури. Система показала стійке регулювання з прийнятною швидкодією, відсутністю значного перерегулювання та коливань у сталому режимі.

Порівняно з теоретичною температурною кривою без регулятора (пасивне нагрівання), використання ПІ-регулятора дозволяє суттєво прискорити вихід на задану температуру та уникнути накопичення похибки. Застосований підхід дає змогу не тільки досягти бажаної температури, а й утримувати її з мінімальними

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		29

відхиленнями, що є критично важливим для процесів пастеризації, де навіть невеликі коливання температури можуть призвести до погіршення якості продукції.

Отже, ПІ-регулятор продемонстрував високу ефективність в умовах даного об'єкта керування, що дозволяє рекомендувати його до впровадження в реальну систему автоматизації пастеризаційного процесу.

Висновки до розділу

У третьому розділі виконано синтез системи автоматичного керування з використанням ПІ-регулятора, параметри якого були налаштовані в середовищі MATLAB. Побудовано імітаційну модель у Simulink, яка підтвердила ефективність регулювання температури за допомогою запропонованого алгоритму. Система показала стабільну роботу без коливань у сталому режимі, що дозволяє рекомендувати її до впровадження в реальну систему автоматизації.

					БР.АКП-62.00.00.000 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

4. РОЗРОБКА ПРОЕКТНОЇ СКЛАДОВОЇ САК

4.1. Вибір технічних засобів

NodeMCU V3 — це мікроконтролерна плата, створена для реалізації проектів у сфері Інтернету речей (IoT). В її основі лежить модуль ESP-12E на базі мікросхеми ESP8266, яка поєднує центральний процесор та Wi-Fi-модуль. Це дозволяє реалізовувати бездротову передачу даних без потреби в додаткових адаптерах.

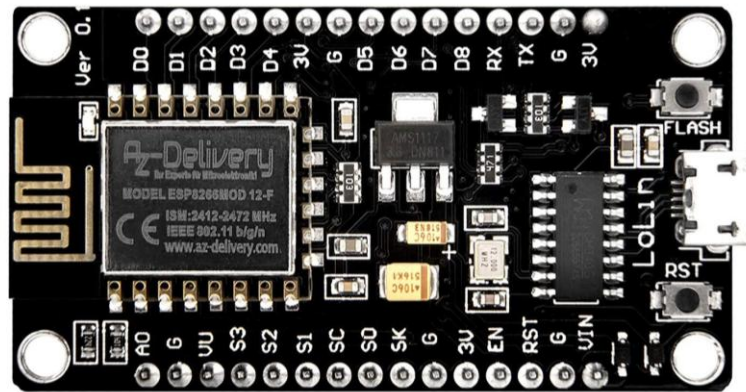


Рисунок 4.1 - NodeMCU V3

Плата функціонує на 32-бітному процесорі Tensilica з тактовою частотою 80 МГц, з можливістю збільшення до 160 МГц. Обсяг оперативної пам'яті становить 96 КБ, а флеш-пам'яті — 4 МБ, що дозволяє зберігати прошивку і дані користувача.

Для живлення передбачений вхід VIN (допустимий діапазон — від 4,5 до 9 В), а логічний рівень на входах/виходах становить 3,3 В. У NodeMCU доступні цифрові входи/виходи (GPIO), більшість з яких підтримують PWM, I2C, SPI, а також 1-Wire. Присутній також один аналоговий вхід з діапазоном вимірювання до 1 В.

Для комунікації з іншими пристроями передбачені UART-інтерфейси, а передача даних до хмари або локальної мережі здійснюється через модуль Wi-Fi з підтримкою стандартів 802.11 b/g/n.

Програмування NodeMCU здійснюється через порт Micro-USB, за допомогою Arduino IDE або мови Lua. Завдяки наявності вбудованого

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докum.	Підпис	Дата		31

стабілізатора напруги та готових бібліотек, плата є надзвичайно зручною для розробки розумних систем моніторингу та керування.[8][10]

Технічні характеристики:

- Мікроконтролер: ESP8266 (Tensilica L106 32-бітний RISC процесор) з тактовою частотою 80 МГц (можливе підвищення до 160 МГц).
- Пам'ять:
 - Оперативна пам'ять (RAM): 96 КБ.
 - Флеш-пам'ять: 4 МБ для зберігання прошивки та даних.
- Живлення:
 - Вхідна напруга (VIN): 4,5В – 9В (максимум 20В).
 - Робоча напруга вводу/виводу: 3,3В.
- Інтерфейси вводу/виводу:
 - Цифрові GPIO: 17 пінів, з яких всі підтримують PWM, I2C, SPI та 1-Wire, окрім D0.
 - Аналоговий вхід: 1 пін (A0) з діапазоном 0 – 1В.
- Комунікація:
 - Wi-Fi: 802.11 b/g/n з підтримкою режимів станції (STA), точки доступу (AP) та одночасного STA+AP.
 - UART: 2 порти.
 - SPI та I2C: програмно реалізовані.
- Інші особливості:
 - USB-інтерфейс: Micro-USB для живлення та програмування.
 - Розміри: близько 60 мм x 31 мм x 13 мм.
 - Вбудований стабілізатор напруги: 3,3В для живлення модуля та периферії.

Arduino Pro Mini — це компактна мікроконтролерна плата, спеціально розроблена для застосування у вбудованих системах, де критичними є габарити та енергоспоживання. Вона є зменшеною версією Arduino Uno, побудованою на тому ж мікроконтролері — ATmega328P, і має майже ідентичні технічні

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

- На пін I/O: 20 мА
- Тактова частота: 8 МГц (для 3.3 В) або 16 МГц (для 5 В)
- Розміри:
 - Довжина: 33 мм
 - Ширина: 18 мм
 - Вага: ~2 г
- Інші особливості:
 - Відсутній USB-інтерфейс — для прошивки потрібен окремий USB-UART адаптер
 - Присутній ICSP-роз'єм для прямого програмування

Завдяки своїй малій вазі, габаритам та енергоефективності, Arduino Pro Mini є ідеальним рішенням для компактних пристроїв, де використання повнорозмірної плати є недоцільним. Вона активно використовується в носимих пристроях, автономних сенсорах, мобільній робототехніці та системах дистанційного збору даних.

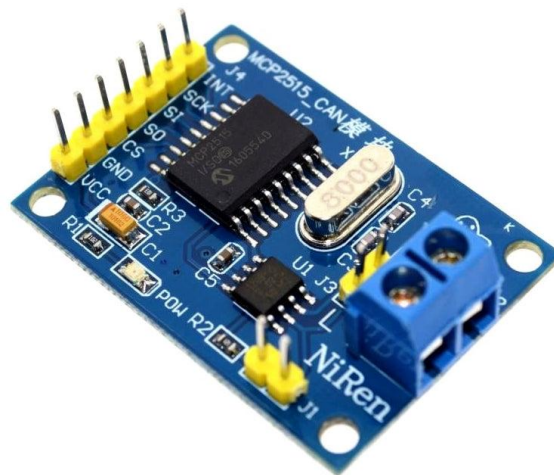


Рисунок 4.3 - CAN-шина та модуль MCP2515

У контексті промислових систем керування особливу цінність має протокол CAN (Controller Area Network), який дозволяє надійно передавати дані між мікроконтролерами та периферійними пристроями в умовах високого електромагнітного шуму.[11]

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34

Для реалізації підтримки CAN-шини з Arduino-сумісними мікроконтролерами використовується модуль MCP2515, що базується на однойменному контролері компанії Microchip. Цей модуль реалізує повний стек CAN-протоколу (версія 2.0B) і передає дані до мікроконтролера через інтерфейс SPI.

Основні характеристики MCP2515:

- Підтримка швидкостей до 1 Мбіт/с.
- Працює за стандартом CAN 2.0B.
- Підключення через SPI (4 дроти + живлення).
- Має вбудований буфер повідомлень, фільтри і маски.
- Сумісний із зовнішнім трансивером (наприклад, TJA1050), що забезпечує фізичний рівень.

Принцип роботи:

1. Обмін через SPI: Мікроконтролер (наприклад, Arduino Pro Mini або ESP8266) надсилає команди до MCP2515 через SPI-шину.
2. Передача CAN-повідомлень: MCP2515 кодує повідомлення згідно зі стандартом CAN і передає їх через зовнішній трансивер на CAN-шину.
3. Прийом повідомлень: Отримані з шини пакети проходять фільтрацію, після чого потрапляють до приймального буфера та надсилаються на контролер.

Схема підключення до Arduino:

- MOSI, MISO, SCK, CS — SPI-лінії.
- INT — вивід переривання для повідомлення про нові дані.
- VCC, GND — живлення (5В або 3.3В залежно від плати).
- CANH, CANL — вихід на фізичну CAN-лінію.

У рамках розширення системи автоматизації пастеризації модуль MCP2515 може бути використаний для підключення декількох блоків управління в загальну CAN-мережу. Це дозволить масштабувати систему, забезпечити синхронізацію між кількома технологічними вузлами та зменшити

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35

загальну кількість з'єднань.



Рисунок 4.4 - LCD 1602 I2C

LCD 1602 з I2C-модулем — дворядковий текстовий дисплей із роздільною здатністю 16×2 символи. Робоча напруга становить 5 В. I2C-плата на базі PCF8574 дозволяє зменшити кількість необхідних пінів до двох: SDA та SCL. Споживання струму зазвичай не перевищує 2 мА. Може регулювати яскравість підсвітки за допомогою змінного резистора. Застосовується для зручного виведення інформації: температури, стану процесу, повідомлень.



Рисунок 4.5 - Датчик температури DS18B20 (в ізоляції)

Датчик температури DS18B20 — цифровий температурний сенсор із протоколом OneWire. Має три виводи: живлення, земля та дані. Може працювати в режимі «паразитного живлення». Діапазон вимірювання — від $-55\text{ }^{\circ}\text{C}$ до $+125\text{ }^{\circ}\text{C}$, точність — $\pm 0.5\text{ }^{\circ}\text{C}$ у межах $-10\dots+85\text{ }^{\circ}\text{C}$. Роздільність налаштовується від 9

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

до 12 біт. Підтримує підключення декількох сенсорів на один пін, що дозволяє створювати мережі вимірювання температури.

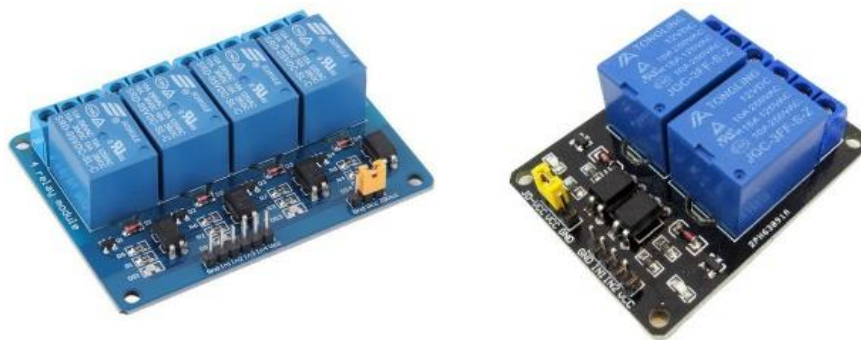


Рисунок 4.6 – Модулі реле

4-канальний та 2-канальний інверторні модулі реле — плати реле з оптопарою та логікою активного LOW-сигналу.[9] Релейні модулі призначені для комутації зовнішніх навантажень змінного (до 250 В) або постійного струму (до 30 В) при максимальному струмі до 10 А. Оснащені світлодіодами індикації стану. Працюють при живленні 5 В, мають вбудований драйвер на транзисторі та можливість гальванічної розв’язки (в деяких версіях). Керування відбувається подачею низького рівня на цифровий вхід (IN1–IN4).

Ці компоненти забезпечують фізичну реалізацію логіки, описаної в програмі, дозволяють здійснювати контроль та індикацію технологічного процесу.



Рисунок 4.7 - Електромагнітний клапан KAISER 17476000001170

Цей елемент є електромеханічним пристроєм, який забезпечує автоматичне відкривання або перекривання потоку води в системах водопостачання побутової техніки. Він використовується переважно в посудомийних машинах, зокрема у моделях виробника KAISER, для контролю

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		37

за надходженням води у внутрішній контур.

Клапан складається з корпусу, соленоїда та внутрішнього механізму, що активується під дією електромагнітного поля. При подачі живлення на котушку соленоїда (зазвичай 220–240 В змінного струму), клапан відкривається й пропускає воду; при відсутності живлення — автоматично закривається.

Основні технічні характеристики:

- Призначення: Подача води в посудомийну машину
- Тип: Односекційний електромагнітний клапан
- Робоча напруга: ~220–240 В АС
- Тип з'єднання: Штуцерний (вхід – з'єднання до водопроводу, вихід – до гідросистеми)
- Матеріал корпусу: Полімер (термостійкий пластик)
- Тип керування: Нормально закритий (відкривається при подачі живлення)
- Сумісність: Моделі посудомийних машин KAISER (та аналогічні)
- Робоче середовище: Холодна та помірно тепла вода (до 90 °С)

Клапан відзначається надійною роботою, компактними розмірами й простотою монтажу. Він ідеально підходить для застосування в системах, де необхідне автоматичне дозування подачі води під керуванням електроніки.

4.2. Структурна схема системи

На рисунку 4.8 зображено структуру автоматизованої системи керування, яка реалізує процес пастеризації за допомогою рішень промислового Інтернету речей (IIoT). Основу системи становлять два мікроконтролери: Arduino Nano та NodeMCU ESP8266, які обмінюються даними через послідовний інтерфейс UART.

Такий підхід забезпечує логічний поділ функцій: Arduino відповідає за керування виконавчими механізмами на місці, тоді як NodeMCU виконує роль мережевого модуля, що забезпечує передавання інформації до хмари через Wi-Fi.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		38

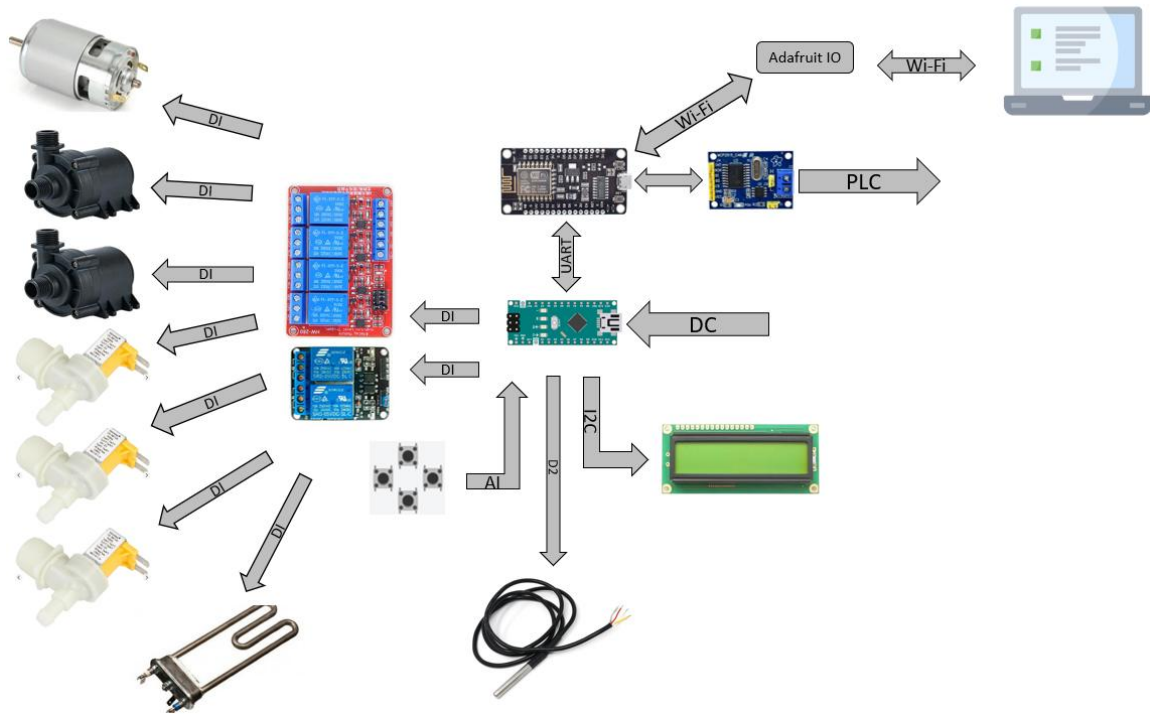


Рисунок 4.8 - Структурна схема системи реакторного блоку

Керування виконавчими пристроями здійснюється через релейні модулі (на чотири та два канали), що підключені до цифрових виходів Arduino. За допомогою цих реле відбувається вмикання і вимикання таких компонентів:

- мішалки з електродвигуном;
- двох pomp для рециркуляції;
- трьох електромагнітних клапанів;
- електричного нагрівача (ТЕН).

Зчитування температури реалізовано через цифрові входи Arduino за допомогою датчика DS18B20. Для введення команд оператором передбачено кнопки, підключені до аналогових входів. Відображення інформації здійснюється за допомогою LCD-дисплея з інтерфейсом I²C, що мінімізує використання виводів мікроконтролера.

Мікроконтролер NodeMCU виконує функцію комунікаційного вузла: отримані з Arduino дані він передає до віддаленого серверу, забезпечуючи дистанційний моніторинг параметрів у режимі реального часу.

4.3. Функціональна схема системи

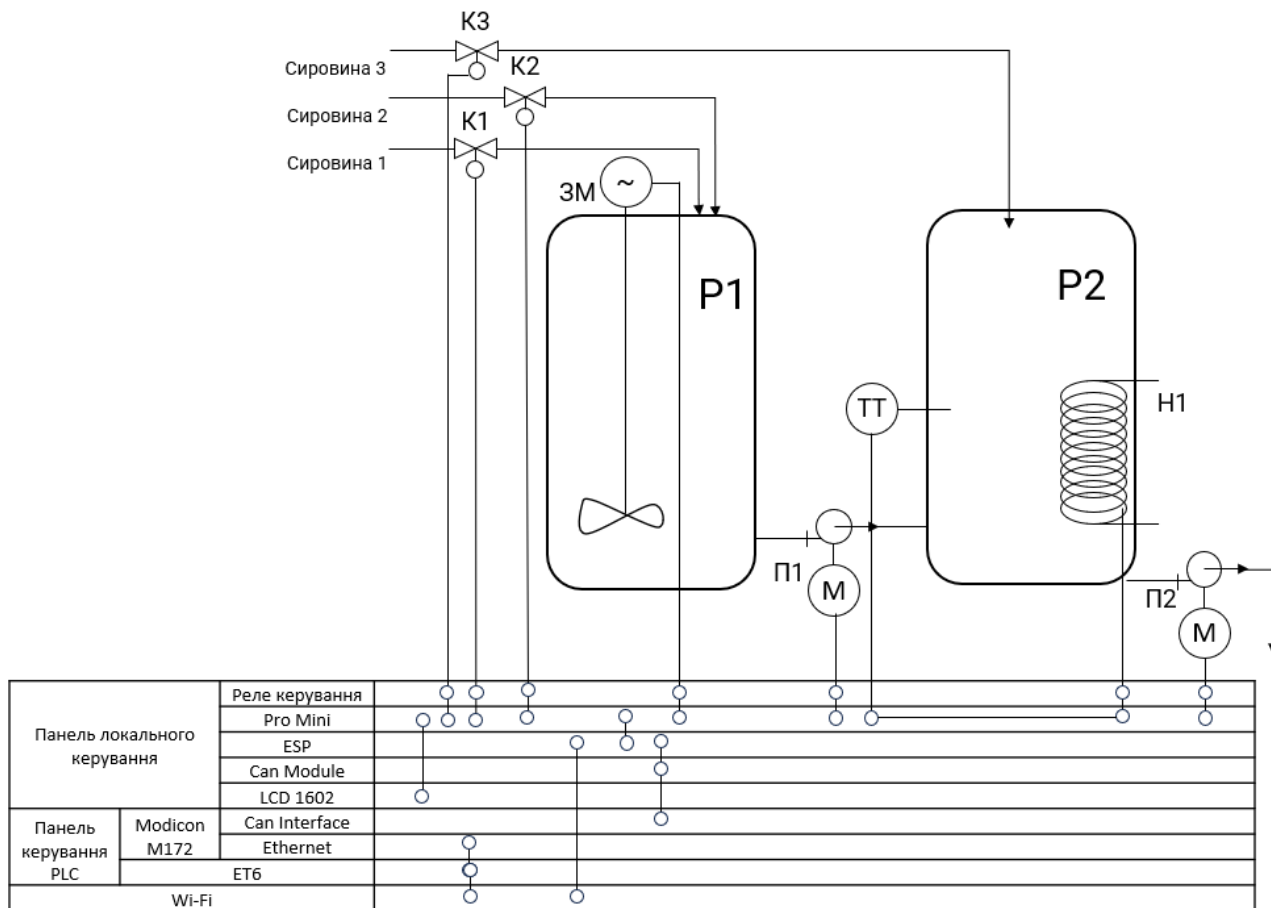


Рисунок 4.9 - Функціональна схема об'єкта

Функціональна схема автоматизації охоплює логічно взаємопов'язані компоненти, що забезпечують повний цикл обробки фруктового пюре (або іншого процесу) — від початкового змішування інгредієнтів до пастеризації готової суміші. Система виконує контроль ключових технологічних параметрів, таких як температура а при необхідності встановлення додаткових (кислотність середовища – рН), із залученням датчиків, виконавчих пристроїв і мікроконтролерного керування.

Секція 1 – Змішування компонентів

Клапани К1 і К2 регулюють надходження сировинних компонентів у змішувальний резервуар (ЗМ). Вони відкриваються за командами від контролера, що дозволяє здійснювати дозовану подачу, наприклад, пюре, води або добавок.

У змішувачі (ЗМ) відбувається поєднання інгредієнтів до утворення однорідної маси. Залежно від технологічних вимог змішування може проводитися з попереднім нагріванням або без нього.

Помпа П1 транспортує суміш зі змішувача до наступного етапу — резервуару для пастеризації (резервуар 2). Її робота координується з контролем рівня в резервуарі, щоб запобігти переливу.

Секція 2 – Пастеризація

Клапан К3 забезпечує подачу підготовленої суміші з попереднього етапу в резервуар 2 для термічної обробки.

Температурний сенсор (ТЕ) визначає поточну температуру продукту в резервуарі. Ці дані використовуються системою для керування нагрівальним елементом.

Нагрівач Н1 відповідає за підвищення температури до рівня, необхідного для пастеризації. Керування його вмиканням/вимиканням здійснюється контролером відповідно до показань температурного сенсора.

Після досягнення встановлених умов пастеризації (температури та часу витримки), помпа П2 виконує перекачування готової продукції на наступний етап — зберігання або фасування. Робота П2 синхронізується з контролем температури, щоб уникнути випуску недостатньо обробленої суміші.

Алгоритм роботи системи

Відкриваються клапани К1 та К2, здійснюється подача компонентів у змішувач.

Сформована суміш переміщується помпою П1 у резервуар 2 через відкритий клапан К3.

В резервуарі відбувається контроль температури, за необхідності активується нагрівальний елемент Н1.

Після завершення пастеризації, коли досягнуто встановлені параметри, включається П2, і суміш надходить до вихідного етапу.

і					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		41

4.4.Електрична-принципова схема

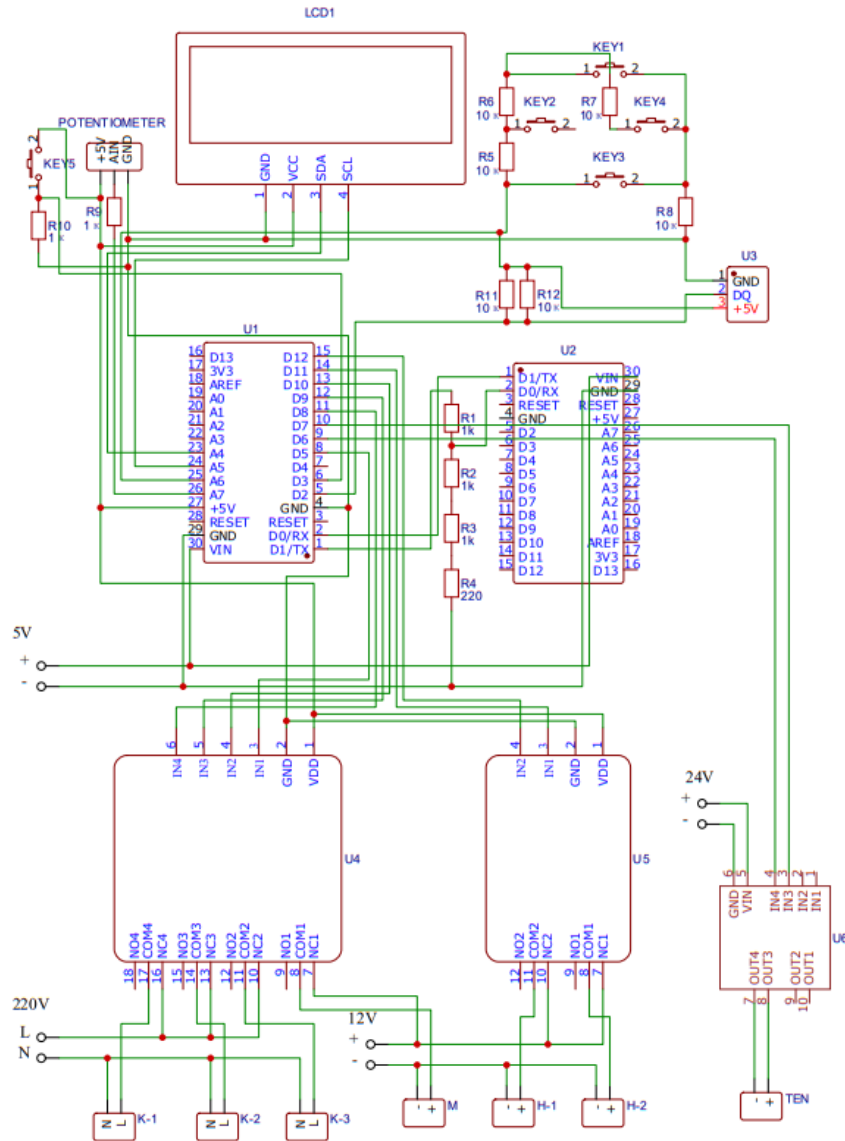


Рисунок 4.10 - Електрична-принципова схема об'єкта

У складі автоматизованої ПоТ-системи керування процесом пастеризації фруктового пюре передбачено використання низки виконавчих пристроїв, а також двох мікроконтролерів, що відповідають за керування та обмін даними. Система побудована таким чином, щоб забезпечити ефективне керування потоками сировини, термічною обробкою та моніторингом у режимі реального часу.

Електромагнітні клапани К-1, К-2 та К-3 відповідають за дозовану подачу сировини на різних етапах процесу:

Клапан К-1 відкриває доступ до резервуара з основною фруктовною масою;

										Арк.
										42
Змн.	Арк.	№ докум.	Підпис	Дата						

Клапан К-2 регулює подачу додаткового компонента (наприклад, води, сиропу або регулятора кислотності);

Клапан К-3 керує виходом готової суміші до пастеризатора або накопичувальної ємності.

Кожен з клапанів є пристроєм з нормально закритим положенням і приводиться в дію при подачі напруги 220 В. Керування здійснюється через релейний модуль, підключений до цифрових виходів контролера.

Насоси Н-1 і Н-2 відповідають за транспортування сировини між резервуарами та пастеризаційною секцією:

Н-1 подає суміш зі змішувального резервуара до блоку пастеризації;

Н-2 використовується для рециркуляції або перекачування пюре після термічної обробки у тару.

Насоси живляться від мережі постійного струму 12В та керуються окремими реле або напівпровідниковими ключами, які забезпечують електричну розв'язку з мікроконтролером.

Змішувач (позначено як М) забезпечує гомогенізацію компонентів усередині ємності. Він може бути реалізований як мотор-редуктор з лопатевим або спіральним мішалкою. Його функція полягає у створенні рівномірної консистенції суміші перед подачею на пастеризацію. Активується за потреби, відповідно до заданого алгоритму (наприклад, після відкриття клапанів К-1 та К-2).

Основним елементом теплової обробки у системі виступає нагрівальний елемент (TEN), що працює від джерела постійного струму 9 В. Тип елемента — керамічний резистивний нагрівач, який монтується у пастеризаційну ємність. Нагрівання контролюється за допомогою температурного датчика, а також системи керування ШІМ-сигналом через потужний MOSFET або реле. Це дозволяє реалізувати плавне регулювання потужності для уникнення перегріву.

Для вимірювання температури пастеризованої суміші використовується цифровий термометр DS18B20, позначений як U3. Цей сенсор підключається до одного цифрового піну Arduino Nano через інтерфейс 1-Wire, що дозволяє

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

зчитувати температурні значення з високою точністю (до 0,5 °C). DS18B20 працює в діапазоні температур від -55 до +125 °C, має водонепроникне виконання з нержавіючої сталі та ідеально підходить для агресивних середовищ. Його дані використовуються для керування нагрівальним елементом, а також передаються на хмарний сервіс для віддаленого моніторингу процесу.

Для реалізації розподіленого керування обрано два контролери:

U1 – Arduino Nano відповідає за низькорівневе керування виконавчими елементами (реле, датчики, ШІМ на ТЕН), а також обробку сигналів з аналогових пристроїв. Nano обрано за простоту реалізації, доступність та сумісність з великою кількістю модулів.

U2 – NodeMCU ESP8266 виконує функцію центрального вузла зв'язку з хмарною платформою (Adafruit IO) та забезпечує дистанційний моніторинг і керування. Через UART або інший протокол (наприклад, I²C або SPI) NodeMCU обмінюється даними з Arduino Nano, приймаючи або передаючи команди та параметри роботи.

Поділ функцій між мікроконтролерами дозволяє зменшити навантаження на кожен з них, підвищити стабільність та розділити локальні та мережеві задачі.

4.5. Проектування дизайну передньої панелі

Основним завданням при проектуванні панелі було забезпечити логічне, інтуїтивно зрозуміле розміщення всіх необхідних елементів керування та індикації, відповідно до функціонального поділу на зони. Особливу увагу приділено ергономіці – зручності доступу до кнопок, перемикачів, дисплея та інших компонентів у процесі роботи оператора.

Конструктивно панель складається з прозорого оргскла, на яке нанесено кольорове зображення, розроблене у Microsoft Visio. Такий підхід дозволив досягти чіткої відповідності між графічним оформленням і реальним розміщенням компонентів. Розмітка містить також позиційні мітки для вирізання отворів під перемикачі, потенціометр, кнопки, світлодіодні індикатори й дисплей.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

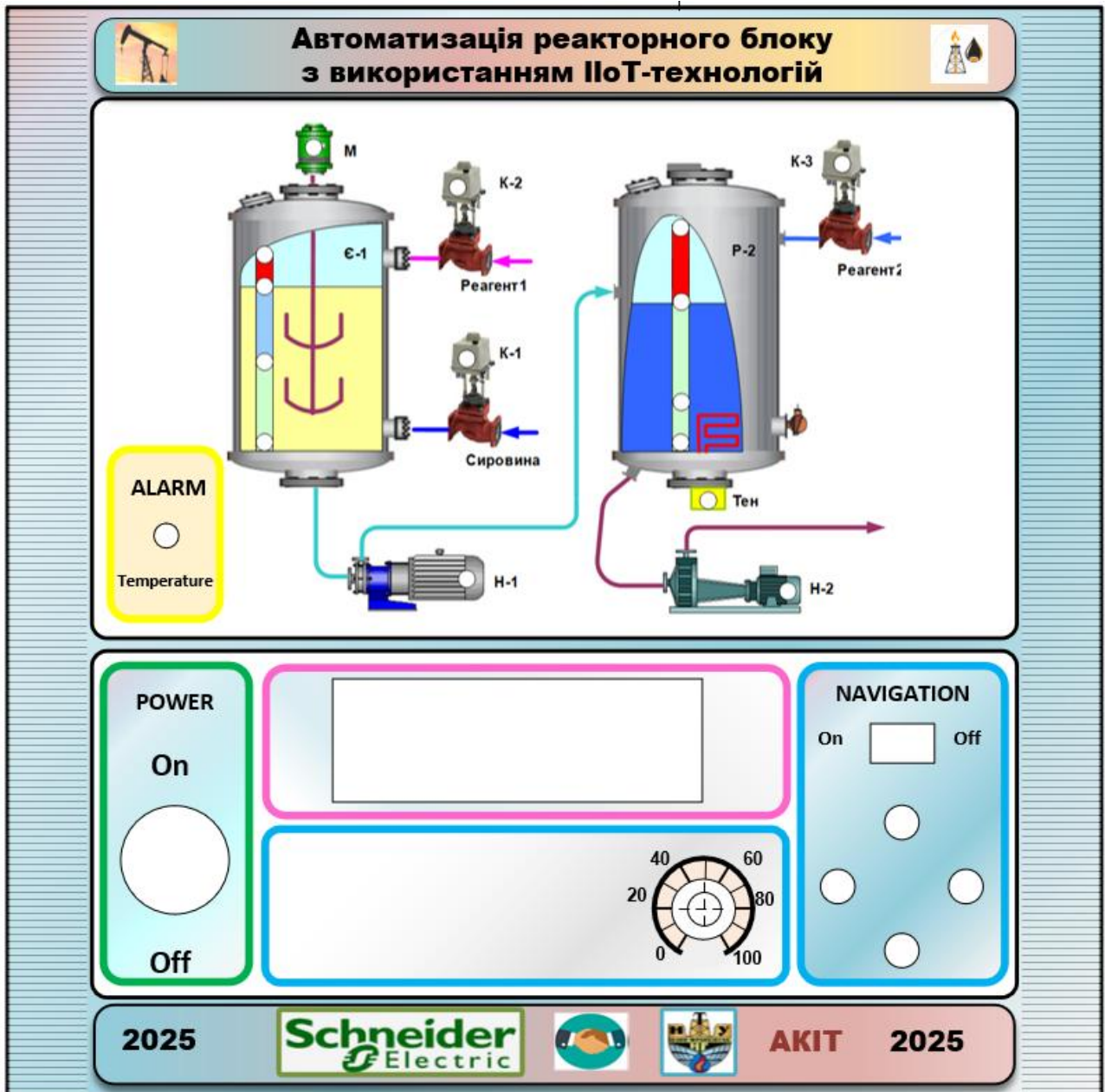


Рисунок 4.11 - Дизайн передньої панелі

Передня панель має чітку зональну організацію, яка дозволяє оператору швидко орієнтуватися під час взаємодії з системою.

- Зона живлення (POWER). Розташована зліва внизу. Містить великий перемикач «On/Off», за допомогою якого здійснюється подача напруги 220 В на систему. Цей вузол є базовим для активації або вимкнення всіх інших модулів, включаючи мікроконтролери, виконавчі механізми й сенсорну апаратуру. Його виділено зеленим кольором для візуального акценту.
- Зона дисплея (DISPLAY). Розміщена по центру панелі й оформлена

										Арк.
										45
Змн.	Арк.	№ докум.	Підпис	Дата	БР.АКП-62.00.00.000 ПЗ					

рожево-червоним контуром. У цій зоні передбачено вікно під рідкокристалічний дисплей, через який оператор може спостерігати за параметрами процесу (температура, стан клапанів, режим роботи тощо). Дисплей взаємодіє з мікроконтролером NodeMCU, а інформація оновлюється в реальному часі завдяки зв'язку з хмарною платформою Adafruit IO.

- Зона потенціометра (MANUAL CONTROL). Безпосередньо під зоною дисплея розташовано область із градуванням від 0 до 100. Тут встановлюється поворотний потенціометр, який слугує для ручного налаштування окремих параметрів у локальному режимі (наприклад, температури ТЕНу або швидкості обертання мішалки). Такий підхід дозволяє швидко адаптувати параметри без необхідності переходу в онлайн-інтерфейс.
- Зона навігації (NAVIGATION). Праворуч на панелі розташована група з чотирьох кнопок (вгору, вниз, «Enter», «Назад»), які слугують для переміщення по меню дисплея. Також передбачено перемикач «Start/Stop» для ініціації або зупинки процесу пастеризації. Ця зона оформлена в синіх тонах і забезпечує повний ручний контроль над функціями пристрою.
- Зона аварійної індикації (ALARM). У верхньому лівому куті передбачено панель світлової сигналізації. Світлодіод автоматично активується при перевищенні критичного значення температури або за відсутності зв'язку з одним із сенсорів. Це підвищує рівень оперативного контролю та дозволяє своєчасно вжити заходів для зупинки або переходу в аварійний режим.
- Візуальна схема процесу. У верхній частині панелі розташована ілюстрація структури реакторного блоку, що дозволяє оператору візуально співвідносити показання дисплея зі станом об'єкта керування. На схемі вказані компоненти, такі як резервуари, насосні агрегати, клапани, змішувачі й ТЕН.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

Передня панель не є лише декоративним або допоміжним елементом, а фактично виконує функцію операторської станції в компактному виконанні. Завдяки логічній організації зони керування, реалізації аварійної індикації та поєднанню локального та віддаленого доступу до параметрів, панель є повноцінною частиною ПоТ-системи. Вона дозволяє реалізувати гібридну модель керування — як вручну через фізичні елементи, так і віддалено через мережу інтернет.

Панель забезпечує інтуїтивну зручність у роботі, простоту обслуговування та можливість подальшої модернізації (наприклад, інтеграція додаткових сенсорів, сенсорного дисплея чи модулів зворотного зв'язку). Її розробка та реалізація дозволяє ефективно демонструвати роботу автоматизованої системи.

4.6. Код програми для NodeMCU8266

У даному підрозділі розглянуто програмне забезпечення автоматизованої системи пастеризації фруктового пюре з використанням технологій Інтернету речей (ПоТ). Програма реалізована на двох мікроконтролерах: NodeMCU ESP8266 та Arduino Pro Mini, які взаємодіють між собою через UART і забезпечують як хмарне керування, так і локальну логіку технологічного процесу. На рис. 4.12 приставлена загальна блок-схема роботи програми.

Цей модуль виконує функцію інтерфейсу між хмарним сервісом Adafruit IO та Arduino Pro Mini. Його головне завдання — обробляти MQTT-команди, отримані з хмари, передавати їх у вигляді текстових інструкцій через послідовний порт до другого мікроконтролера, а також приймати зворотні повідомлення та оприлюднювати їх в Adafruit IO.

Робота програми включає підключення до Wi-Fi, налаштування MQTT-з'єднання, підписку на відповідні топіки, обробку отриманих команд, формування інструкцій типу K1_ON, HEATER_OFF, TEMP:65 та передачу їх у UART-канал. Також вона виконує публікацію відповідей Arduino, зокрема поточної температури або повідомлень про виконані фази.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

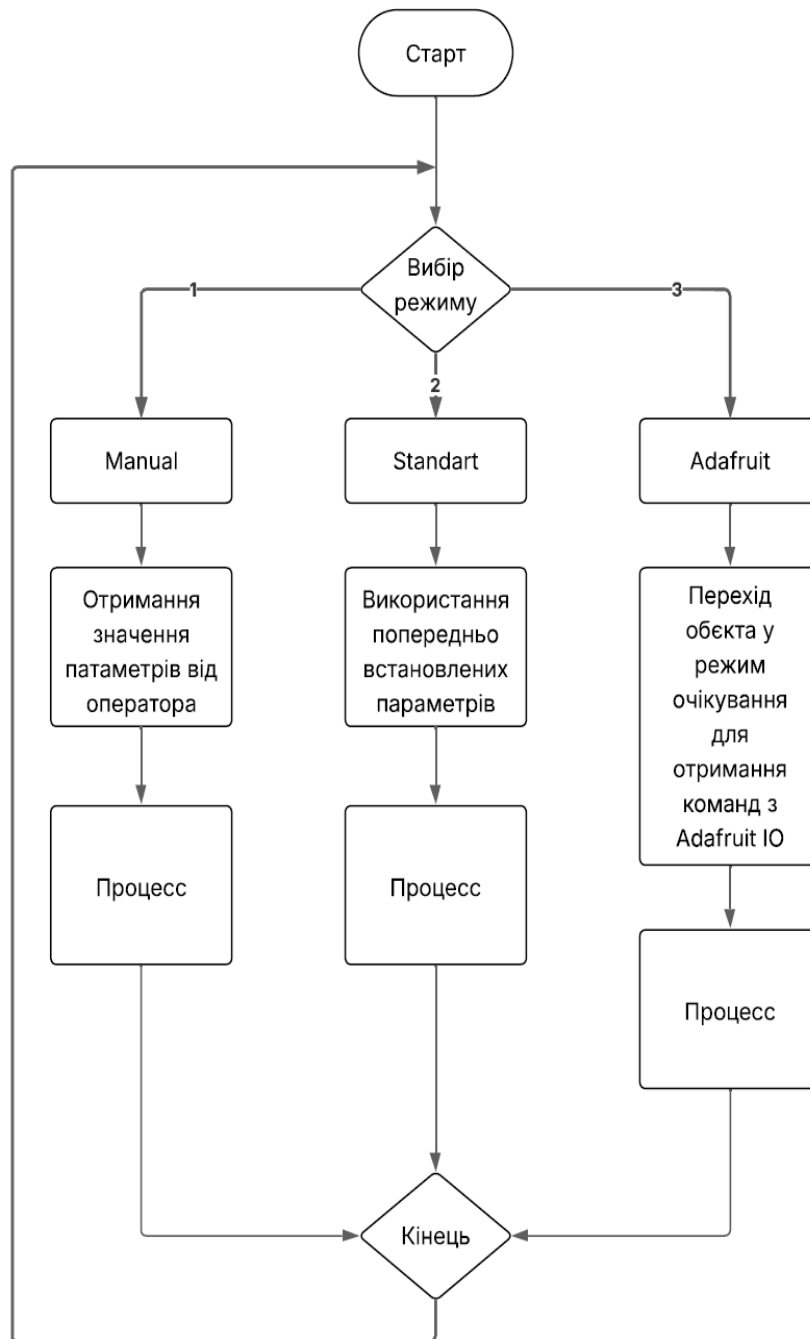


Рисунок 4.12 - Блок-схема роботи програми

Код програми NodeMCU8266

Особливу роль відіграє функція автоперепідключення до MQTT-брокера, що забезпечує стабільність хмарного з'єднання при втраті сигналу. Обробка вхідних повідомлень реалізується у циклі loop(), де кожна підписка порівнюється з відповідним об'єктом і перетворюється у відповідну команду.

Наприклад, обробка топиків виконується в межах конструкції зображеній на рисунку нижче.

елементів із затримками, нагрів до заданої температури та контроль досягнення цільових значень:

```
227     lcd.setCursor(0, 0); lcd.print("HEATING...");
228     targetTemp = tTarget;
229     while (true) {
230         sensors.requestTemperatures();
231         currentTemp = sensors.getTempCByIndex(0);
232         tempPID.Compute();
233         analogWrite(pinHeater, (int)heaterOutput);
234
235         lcd.setCursor(0, 1);
236         lcd.print(currentTemp); lcd.print((char)223); lcd.print("C/");
237         lcd.print(targetTemp);
238
239         if (currentTemp >= targetTemp || checkForStop()) break;
240         delay(1000);
241     }
```

Рисунок 4.14 - Регулятор потужності нагрівача

Цей цикл регулює подачу потужності на нагрівач відповідно до помилки температури за допомогою PID-контролю.

Функція `handleAdafruitManualMode()` відповідає за реакцію на команди з хмари. Вона розпізнає інструкції на зразок `K1_ON`, `HEATER_OFF`, `TEMP:70`, виконує дії та відправляє зворотні повідомлення типу `PHASE:K1_ON` або `TEMP:68.4` через UART.

Таким чином, логіка програми дозволяє як автономне виконання технології, так і її керування в режимі реального часу.

Висновки до розділу

У четвертому розділі розроблено проектну частину системи автоматичного керування. Проведено вибір технічних засобів, таких як мікроконтролери, датчики температури та рН, модулі керування навантаженням, інтерфейси зв'язку та живлення. Побудовано структурну, функціональну й електричну схеми системи. Створено макет передньої панелі для ручного керування.

Розроблено програмне забезпечення для NodeMCU та Arduino Pro Mini з інтеграцією в хмарну платформу Adafruit IO. Це дозволяє здійснювати як локальне, так і дистанційне керування та моніторинг усіх ключових параметрів.

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докum.	Підпис	Дата		50

ЗАГАЛЬНІ ВИСНОВКИ

У межах даної бакалаврської роботи було розроблено автоматизовану систему керування для реакторного блоку пастеризації фруктового пюре з використанням сучасних технологій Інтернету речей (ІоТ). Робота охоплює всі етапи створення системи: від аналізу об'єкта керування та математичного моделювання до розробки алгоритмів регулювання і реалізації прототипу з передачею даних до хмарної платформи.

На першому етапі було обґрунтовано вибір об'єкта автоматизації — реакторного блоку, що складається з двох резервуарів, насосів, нагрівального елемента, змішувача та системи подачі сировини. Проведено структурний аналіз процесу пастеризації фруктового пюре з визначенням основних параметрів, що підлягають контролю: температури, часу витримки при пастеризації, рівня рідини та кислотності (рН). Визначені межі номінальних значень цих параметрів і обґрунтовано необхідність їх автоматичного контролю для забезпечення стабільної якості готової продукції.

У другому розділі було побудовано математичну модель реакторного блоку на основі теплофізичних залежностей і передавальних функцій. Ідентифікація моделі дозволила отримати адекватне уявлення про динаміку системи при впливі збурень і зміні вхідних параметрів. Така модель стала основою для синтезу регулятора.

У третьому розділі здійснено синтез системи керування із застосуванням ПІ-регулятора. Моделювання в середовищі MATLAB/Simulink підтвердило ефективність обраного регулятора — система забезпечує швидке зростання сигналу, обмежене перерегулювання (~15 %) і відсутність коливань у сталому режимі. Це свідчить про достатню точність і надійність запропонованої системи.

Ключовим досягненням роботи є реалізація демонстраційного макету на базі мікроконтролерів NodeMCU та Arduino Pro Mini. Використано модульну архітектуру для керування виконавчими механізмами та збору даних з датчиків. Інтеграція з хмарною платформою Adafruit ІО забезпечує можливість

						БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата			51

віддаленого моніторингу та контролю за технологічним процесом у режимі реального часу, що відповідає сучасним вимогам ІоТ у промисловій автоматизації.

Загалом, виконана робота доводить доцільність і практичну ефективність використання технологій Інтернету речей для автоматизації харчових технологічних процесів. Розроблена система є гнучкою, масштабованою та економічно доступною, що дозволяє адаптувати її до інших подібних об'єктів автоматизації. Отримані результати можуть бути використані як у виробничій сфері, так і в навчальному процесі для підготовки фахівців у галузі автоматизації та ІоТ-рішень.

					БР.АКП-62.00.00.000 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Глухов В. В., Мельник А. Г. Автоматизація технологічних процесів харчової промисловості. — Київ: НУХТ, 2019. — 248 с.
2. Кириченко О. Ю., Мудрак Ю. М. Інтернет речей та його застосування в системах керування. — Харків: ХНУРЕ, 2020. — 212 с. (Джерело, що описує IoT, хмарні сервіси, Arduino та ESP8266/NodeMCU у промисловості.)
3. Сайт UAmpere – технічні характеристики обладнання [Електронний ресурс] – Режим доступу до ресурсу <https://uampere.com/index.php?route=common/home>.
4. Сайт BERTOCCHI – технічні характеристики обладнання [Електронний ресурс] – Режим доступу до ресурсу <https://www.bertocchi.com/ucx-series.php>
5. Стаття про господарство «Гадз» – інформація про імпорт обладнання [Електронний ресурс] – Режим доступу до ресурсу https://agrotimes.ua/ovochi-sad/fg-gadz-zavezlo-obladnannya-dlya-zavodu-z-vyrobnyctva-pyure-ta-sokiv/?utm_source=chatgpt.com
6. Сайт «GADZ» – стаття про співпрацю із іноземними фірмами [Електронний ресурс] – Режим доступу до ресурсу <https://www.fg-gadz.com.ua/novyny/41-fg-gadz-pidpisalo-dogovir-z-italijskoju-kompanieju-rossi>
7. Сайт Adafruit – технічна документація [Електронний ресурс] – Режим доступу до ресурсу <https://io.adafruit.com/api/docs/#rate-limiting>
8. Сайт ARDUINO.UA – технічні характеристики обладнання [Електронний ресурс] – Режим доступу до ресурсу https://arduino.ua/prod1492-wi-fi-modyl-nodemcu-esp8266?srsId=AfmBOopelIEfVS9656DrD1b7I-YJ8Uk5Ymi4O0U_hUT9S_1_3Y36bOjR
9. Сайт BeeGreen – технічні характеристики обладнання [Електронний ресурс] – Режим доступу до ресурсу <https://beegreen.com.ua/>
10. Онлайн бібліотека – технічні характеристики обладнання та інформація про технологічний процес [Електронний ресурс] – Режим доступу до ресурсу <https://www.wikipedia.org/>

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докum.	Підпис	Дата		53

11. Сайт PmdWay – технічні характеристики обладнання [Електронний ресурс] – Режим доступу до ресурсу <https://pmdway.com>

					БР.АКП-62.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		54

ДОДАТКИ

ДОДАТОК А

Програмний код для Arduino Pro Mini

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <OneWire.h>
#include <DallasTemperature.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

const int buttonPin = A0;
const int potPin = A1;

// Реле / виконавчі механізми
const int pinK1 = 10;
const int pinK2 = 9;
const int pinK3 = 8;
const int pinP1 = 11;
const int pinP2 = 12;
const int pinM = 7; // мішалка (інверторне реле)
const int pinHeater = 6; // нагрівач (ШИМ)
const int pinAlarm = 5; // світлодіод сигналізації (інвертований)

const int tempSensorPin = 2;
OneWire oneWire(tempSensorPin);
DallasTemperature sensors(&oneWire);

enum AdafruitState { IDLE, RUNNING, STOPPED };
AdafruitState adafruitState = IDLE;

int mode = 0;
unsigned long lastPress = 0;
```

```
void setup() {  
    lcd.init();  
    lcd.backlight();  
    showMenu();  
  
    pinMode(pinK1, OUTPUT);  
    pinMode(pinK2, OUTPUT);  
    pinMode(pinK3, OUTPUT);  
    pinMode(pinP1, OUTPUT);  
    pinMode(pinP2, OUTPUT);  
    pinMode(pinM, OUTPUT);  
    pinMode(pinHeater, OUTPUT);  
    pinMode(pinAlarm, OUTPUT);  
  
    // Вимикаємо всі виконавчі механізми  
    digitalWrite(pinK1, HIGH);  
    digitalWrite(pinK2, HIGH);  
    digitalWrite(pinK3, HIGH);  
    digitalWrite(pinP1, HIGH);  
    digitalWrite(pinP2, HIGH);  
    digitalWrite(pinM, HIGH);  
    analogWrite(pinHeater, 0);  
    digitalWrite(pinAlarm, HIGH);  
  
    sensors.begin();  
}  
  
void loop() {  
    int btn = analogRead(buttonPin);  
    unsigned long now = millis();
```

```

if (now - lastPress > 300) {
  if (btn > 1000) {
    mode--;
    if (mode < 0) mode = 2;
    showMenu();
    lastPress = now;
  } else if (btn > 415 && btn < 440) {
    mode++;
    if (mode > 2) mode = 0;
    showMenu();
    lastPress = now;
  } else if (btn > 500 && btn < 530) {
    lcd.clear();
    if (mode == 0) runManual();
    if (mode == 1) runStandard();
    if (mode == 2) runAdafruit();
    delay(1500);
    showMenu();
    lastPress = now;
  }
}
}

void showMenu() {
  lcd.clear();
  lcd.setCursor(0, 0);
  if (mode == 0) lcd.print("> MANUAL");
  if (mode == 1) lcd.print("> STANDARD");
  if (mode == 2) lcd.print("> ADAFRUIT");
}

```

```
lcd.setCursor(0, 1);  
if (mode == 0) lcd.print(" STANDARD");  
if (mode == 1) lcd.print(" ADAFRUIT");  
if (mode == 2) lcd.print(" MANUAL");  
}
```

```
void runManual() {  
    const int paramCount = 7;  
    int paramValues[paramCount] = {0, 0, 0, 0, 0, 0, 20}; // В секундах (або °C для  
TEMP)  
    String paramLabels[paramCount] = {"K1", "K2", "K3", "P1", "P2", "M",  
"TEMP"};  
    int paramIndex = 0;  
    int btn;  
    unsigned long now;  
    unsigned long lastInteraction = millis();  
  
    lcd.clear();  
    lcd.setCursor(0, 0);  
    lcd.print("Edit Params:");  
  
    while (true) {  
        btn = analogRead(buttonPin);  
        now = millis();  
  
        if (now - lastInteraction > 300) {  
            if (btn > 1000) { // UP  
                paramIndex--;  
                if (paramIndex < 0) paramIndex = paramCount - 1;  
                lastInteraction = now;  
            } else if (btn > 415 && btn < 440) { // SELECT
```

```

    paramIndex++;
    if (paramIndex >= paramCount) paramIndex = 0;
    lastInteraction = now;
} else if (btn > 500 && btn < 530) { // RIGHT = Confirm/start
    break;
} else if (btn > 390 && btn < 415) { // LEFT = Back
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Cancelled");
    delay(1000);
    return;
}
}

int potVal = analogRead(potPin);
if (paramIndex == 6)
    paramValues[paramIndex] = map(potVal, 0, 1023, 20, 101); // TEMP (°C)
else
    paramValues[paramIndex] = map(potVal, 0, 1023, 0, 201); // секунди

lcd.setCursor(0, 0);
lcd.print(">");
lcd.print(paramLabels[paramIndex]);
lcd.print(": ");
lcd.print(paramValues[paramIndex]);
lcd.print((paramIndex == 6) ? "C  " : "s  ");

int next = (paramIndex + 1) % paramCount;
lcd.setCursor(0, 1);
lcd.print(" ");
lcd.print(paramLabels[next]);

```

```
    lcd.print(": ");
    lcd.print(paramValues[next]);
    lcd.print((next == 6) ? "C  " : "s  ");

    delay(100);
}

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Starting...");
delay(1000);

runProcess(
    paramValues[0], paramValues[1], paramValues[5], paramValues[3],
    paramValues[2], paramValues[6], paramValues[4]
);
}

void runStandard() {
    lcd.print("Standard Mode");
    runProcess(5, 5, 5, 2, 5, 60, 2);
}

void runAdafruit() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Adafruit Mode");
    lcd.setCursor(0, 1);
    lcd.print("Waiting Serial");
```

```
adafruitState = IDLE;
bool shouldExit = false;

while (!shouldExit) {
  if (Serial.available()) {
    String input = Serial.readStringUntil('\n');
    input.trim();

    if (input == "STATUS?") {
      if (adafruitState == IDLE) Serial.println("STATUS:IDLE");
      else if (adafruitState == RUNNING) Serial.println("STATUS:RUNNING");
      else if (adafruitState == STOPPED) Serial.println("STATUS:STOPPED");
    }

    else if (input == "TEMP?") {
      sensors.requestTemperatures();
      float t = sensors.getTempCByIndex(0);
      Serial.print("TEMP:"); Serial.println(t);
    }

    else if (input == "STOP") {
      emergencyStop();
      Serial.println("PHASE:STOPPED");
      shouldExit = true;
    }

    else if (input.startsWith("RUN:")) {
      if (adafruitState == RUNNING) {
        Serial.println("ERROR:ALREADY_RUNNING");
        continue;
      }
    }
  }
}
```

```
}

input.remove(0, 4);
int k1, k2, k3, p1, p2, mix, temp;
sscanf(input.c_str(), "%d,%d,%d,%d,%d,%d,%d", &k1, &k2, &k3, &p1, &p2,
&mix, &temp);

Serial.println("PHASE:STARTING");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Adafruit Running");

adafruitState = RUNNING;
runProcess(k1, k2, mix, p1, k3, temp, p2);

if (adafruitState != STOPPED) {
    adafruitState = IDLE;
    Serial.println("PHASE:DONE");
}
}

else {
    Serial.println("ERROR:UNKNOWN_COMMAND");
}
}

delay(50);
}

lcd.clear();
lcd.setCursor(0, 0);
```

```
lcd.print("Back to Menu");  
delay(1000);  
}
```

```
void runProcess(int k1_s, int k2_s, int m_s, int p1_s, int k3_s, int tempTarget, int  
p2_s) {
```

```
    // Перетворюємо в мілісекунди
```

```
    unsigned long k1 = k1_s * 1000UL;
```

```
    unsigned long k2 = k2_s * 1000UL;
```

```
    unsigned long m = m_s * 1000UL;
```

```
    unsigned long p1 = p1_s * 1000UL;
```

```
    unsigned long k3 = k3_s * 1000UL;
```

```
    unsigned long p2 = p2_s * 1000UL;
```

```
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("K1 ON");
```

```
    digitalWrite(pinK1, LOW); delay(k1); digitalWrite(pinK1, HIGH);
```

```
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("K2 ON");
```

```
    digitalWrite(pinK2, LOW); delay(k2); digitalWrite(pinK2, HIGH);
```

```
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("MIXING...");
```

```
    digitalWrite(pinM, LOW); delay(m); digitalWrite(pinM, HIGH);
```

```
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("P1 ON");
```

```
    digitalWrite(pinP1, LOW); delay(p1); digitalWrite(pinP1, HIGH);
```

```
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("K3 ON");
```

```
    digitalWrite(pinK3, LOW); delay(k3); digitalWrite(pinK3, HIGH);
```

```
    // ==== НАГРІВ, якщо tempTarget > 20 ====
```

```
    if (tempTarget > 20) {
```

```
lcd.clear(); lcd.setCursor(0, 0); lcd.print("Heating...");
```

```
for (int pwm = 0; pwm <= 255; pwm += 5) {  
  analogWrite(pinHeater, pwm);  
  delay(50);  
}
```

```
unsigned long startTime = millis();  
while (millis() - startTime < 900000UL) { // 15 XB  
  analogWrite(pinHeater, 255);  
  delay(500);  
  if (adafruitState == STOPPED) return;  
}
```

```
for (int pwm = 255; pwm >= 0; pwm -= 5) {  
  analogWrite(pinHeater, pwm);  
  delay(50);  
}
```

```
lcd.clear(); lcd.setCursor(0, 0); lcd.print("Heating Done!");  
digitalWrite(pinAlarm, LOW);  
delay(3000);  
digitalWrite(pinAlarm, HIGH);  
} else {  
  lcd.clear(); lcd.setCursor(0, 0); lcd.print("Skip Heating");  
  delay(1000);  
}
```

```
lcd.clear(); lcd.setCursor(0, 0); lcd.print("P2 ON");  
digitalWrite(pinP2, LOW); delay(p2); digitalWrite(pinP2, HIGH);
```

```
lcd.clear(); lcd.setCursor(0, 0); lcd.print("Done!");  
delay(2000);  
}
```

```
void emergencyStop() {  
    digitalWrite(pinK1, HIGH);  
    digitalWrite(pinK2, HIGH);  
    digitalWrite(pinK3, HIGH);  
    digitalWrite(pinP1, HIGH);  
    digitalWrite(pinP2, HIGH);  
    digitalWrite(pinM, HIGH);  
    analogWrite(pinHeater, 0);  
    digitalWrite(pinAlarm, HIGH);  
    lcd.clear(); lcd.setCursor(0, 0); lcd.print("EMERGENCY STOP");  
    adafruitState = STOPPED;  
}
```

ДОДАТОК Б

Програмний код для ESP8266

```
#include <ESP8266WiFi.h> // або <WiFi.h> для ESP32
```

```
#include "Adafruit_MQTT.h"
```

```
#include "Adafruit_MQTT_Client.h"
```

```
// Wi-Fi
```

```
#define WIFI_SSID "TP-LINK_4D38"
```

```
#define WIFI_PASSWORD "26244648"
```

```
// Adafruit IO
```

```
#define AIO_SERVER "io.adafruit.com"
```

```
#define AIO_SERVERPORT 1883
```

```
#define AIO_USERNAME "Vadym241"
```

```
#define AIO_KEY "aio_hUJi76AvPpiiIpVS0AMZRFnJvz1P"
```

```
WiFiClient client;
```

```
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT,  
AIO_USERNAME, AIO_KEY);
```

```
// --- Feeds ---
```

```
Adafruit_MQTT_Subscribe feed_k1 = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/k1");
```

```
Adafruit_MQTT_Subscribe feed_k2 = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/k2");
```

```
Adafruit_MQTT_Subscribe feed_k3 = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/k3");
```

```
Adafruit_MQTT_Subscribe feed_p1 = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/p1");
```

```
Adafruit_MQTT_Subscribe feed_p2 = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/p2");
```

```
Adafruit_MQTT_Subscribe feed_mix = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/mix");
```

```
Adafruit_MQTT_Subscribe feed_temp = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/temp");
```

```
Adafruit_MQTT_Subscribe feed_start = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/start");
```

```
Adafruit_MQTT_Subscribe feed_stop = Adafruit_MQTT_Subscribe(&mqtt,  
AIO_USERNAME "/feeds/stop");
```

```
// Значения
```

```
int val_k1 = 0, val_k2 = 0, val_k3 = 0;
```

```
int val_p1 = 0, val_p2 = 0, val_mix = 0, val_temp = 50;
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
```

```

while (WiFi.status() != WL_CONNECTED) delay(500);

mqtt.subscribe(&feed_k1); mqtt.subscribe(&feed_k2); mqtt.subscribe(&feed_k3);

mqtt.subscribe(&feed_p1); mqtt.subscribe(&feed_p2);
mqtt.subscribe(&feed_mix);

mqtt.subscribe(&feed_temp);

mqtt.subscribe(&feed_start); mqtt.subscribe(&feed_stop);
}

void loop() {

  MQTT_connect();

  Adafruit_MQTT_Subscribe *sub;

  while ((sub = mqtt.readSubscription(100))) {

    if (sub == &feed_k1) val_k1 = atoi((char*)feed_k1.lastread);
    if (sub == &feed_k2) val_k2 = atoi((char*)feed_k2.lastread);
    if (sub == &feed_k3) val_k3 = atoi((char*)feed_k3.lastread);
    if (sub == &feed_p1) val_p1 = atoi((char*)feed_p1.lastread);
    if (sub == &feed_p2) val_p2 = atoi((char*)feed_p2.lastread);
    if (sub == &feed_mix) val_mix = atoi((char*)feed_mix.lastread);
    if (sub == &feed_temp) val_temp = atoi((char*)feed_temp.lastread);

    if (sub == &feed_start) {

```

```
String cmd = "RUN:" + String(val_k1) + "," + String(val_k2) + "," +  
String(val_k3) + "," +  
String(val_p1) + "," + String(val_p2) + "," + String(val_mix) + ","  
+  
String(val_temp);  
Serial.println(cmd);  
}
```

```
if (sub == &feed_stop) {  
String val = String((char*)feed_stop.lastread);  
val.trim();  
if (val == "ON") {  
Serial.println("STOP");  
}  
}  
}
```

```
delay(200);  
}
```

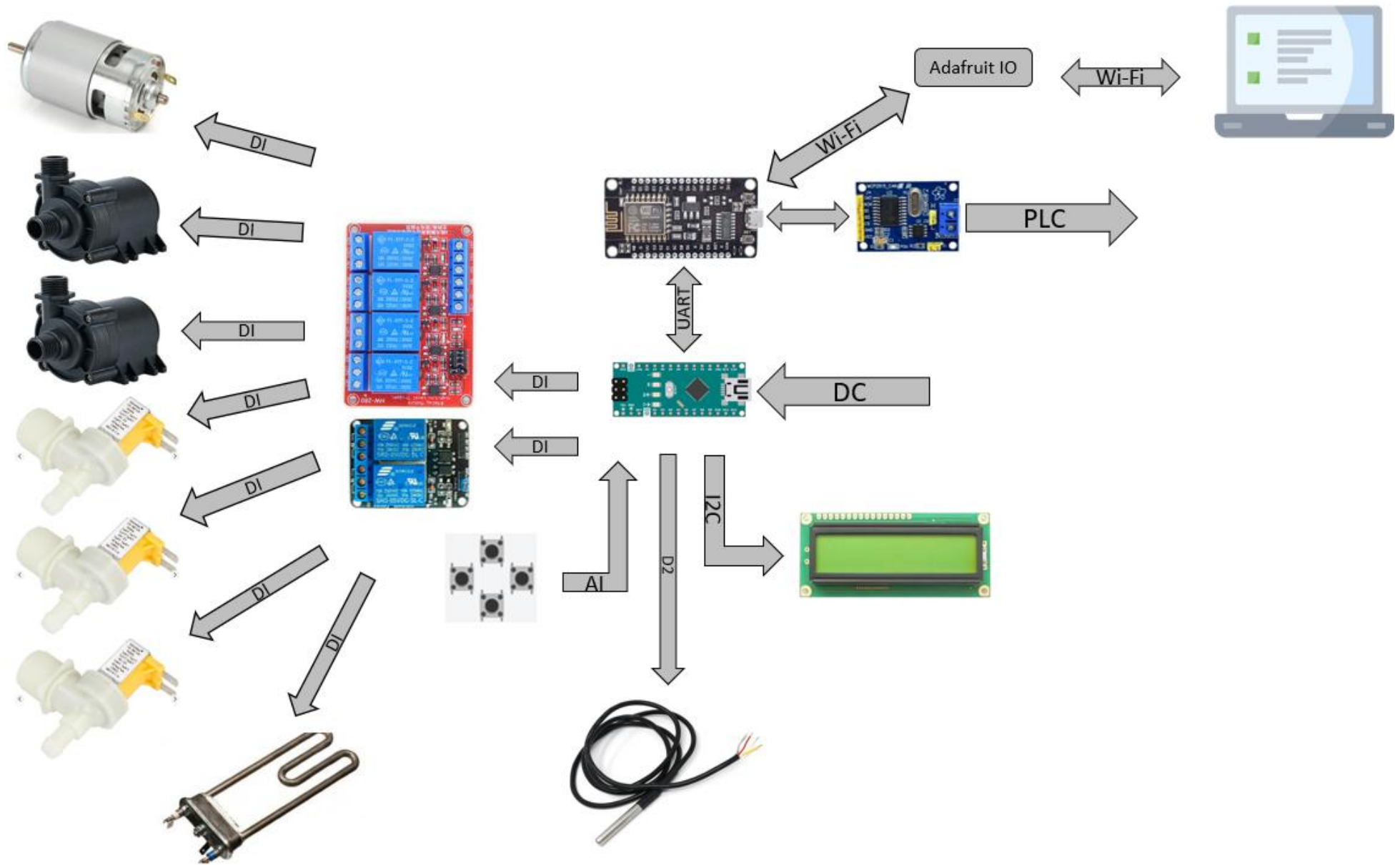
```
void MQTT_connect() {  
if (mqtt.connected()) return;  
  
while (mqtt.connect() != 0) {
```

```
delay(5000); // Повторне підключення
```

```
}
```

```
}
```

					БР.АКП-62.00.00.000. 01				
Змн.	Лист	№ докум.	Підпис	Дата	Автоматизація реакторного блоку з використанням ІоТ-технологій	Літера		Маса	Масштаб
Розробив		Стефурак В.Л.							
Перевірів		Когутяк М.І.							
Т. Контроль									
Рецензент		Борин В.С.				Арк.		Аркушів	
					Структурна схема об'єкта	АКП-23-2К			
Н. Контроль		Кучмистенко О.В.				ІФНТУНГ			
Затвердив		Лагойда А.І.							



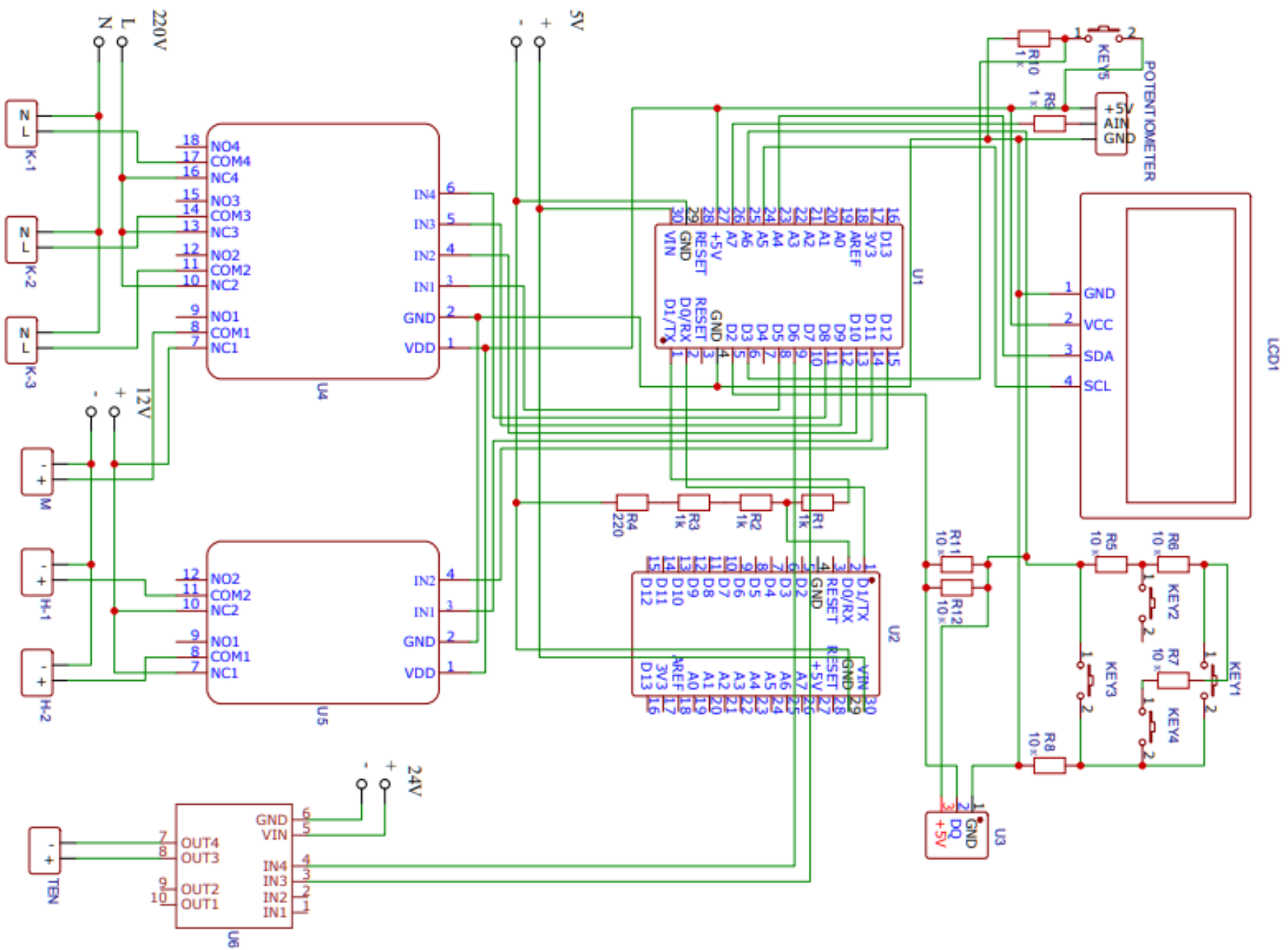
БР.АКП-62.00.00.000. 02

Змн.	Лист	№ докум.	Підпис	Дата			
					Автоматизація реакторного блоку з використанням ІоТ-технологій		
Розробив		Стефурак В.Л.					
Перевірів		Когутяк М.І.					
Т. Контроль							
Рецензент		Борин В.С.			Літера	Маса	Масштаб
					Арк.	Аркушів	
Н. Контроль		Кучмистенко О.В.			АКП-23-2К ІФНТУНГ		
Затвердив		Лагойда А.І.					
					Функціональна схема об'єкта		

БР.АКП-62.00.00.000. 03

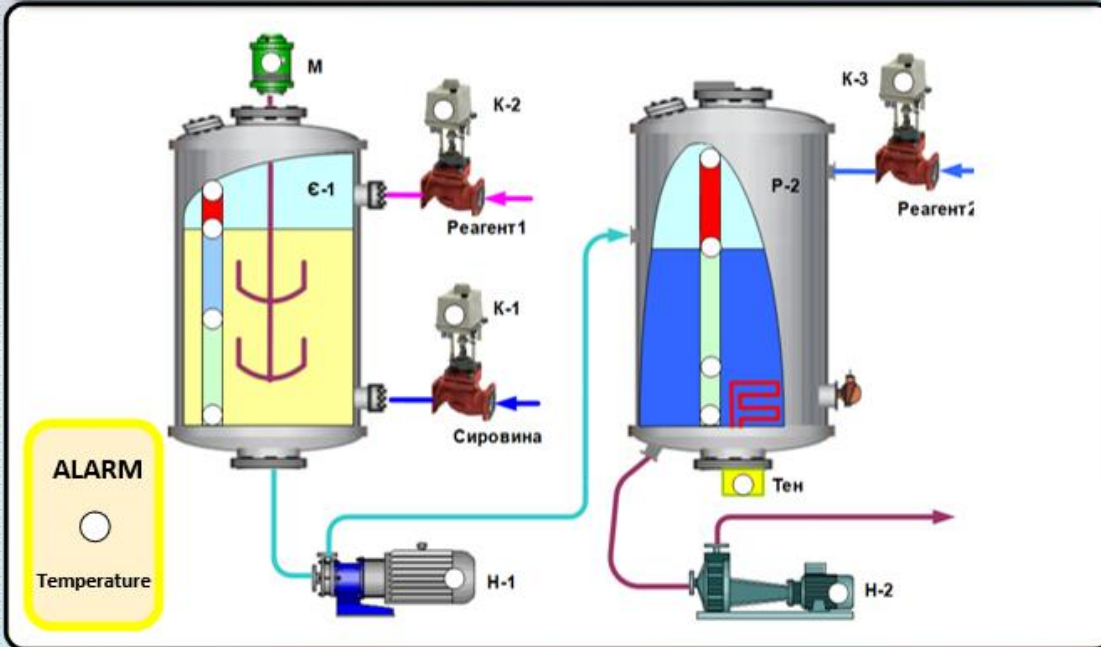
Змн.	Лист	№ докум.	Підпис	Дата							
					Автоматизація реакторного блоку з використанням ІоТ-технологій						
Розробив		Стефурак В.Л.									
Перевірив		Когутяк М.І.									
Т. Контроль											
Рецензент		Борин В.С.							Арк.	Аркушів	
					АКП-23-2К						
Н. Контроль		Кучмистенко О.В.			ІФНТУНГ						
Затвердив		Лагойда А.І.									

Електрична-принципова схема підключень





Автоматизація реакторного блоку з використанням ІоТ-технологій



ALARM

○

Temperature

POWER

On

○

Off

NAVIGATION

On Off

○

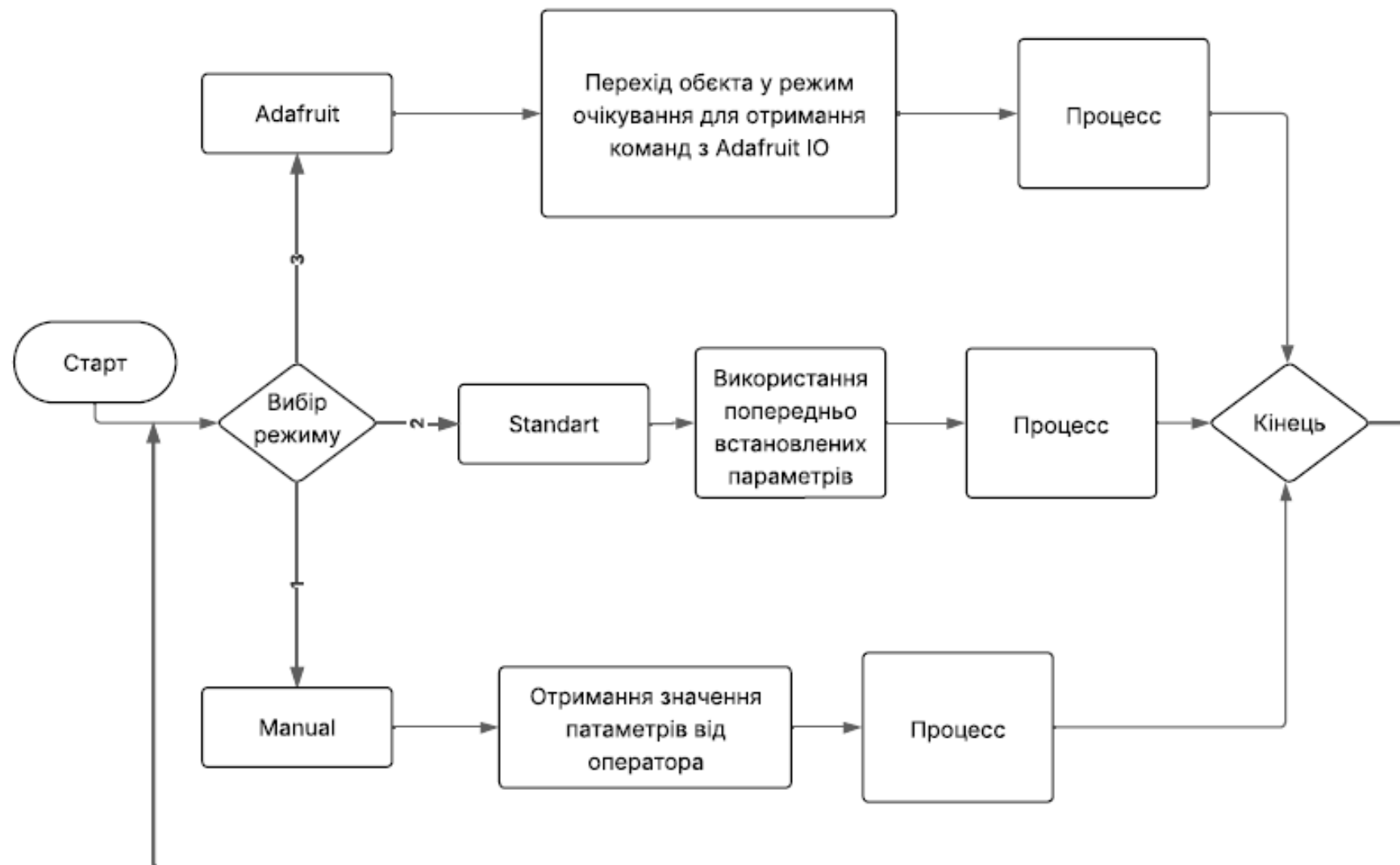
○

○

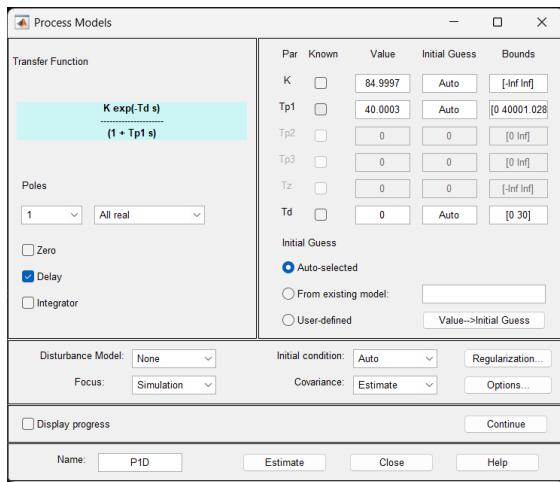
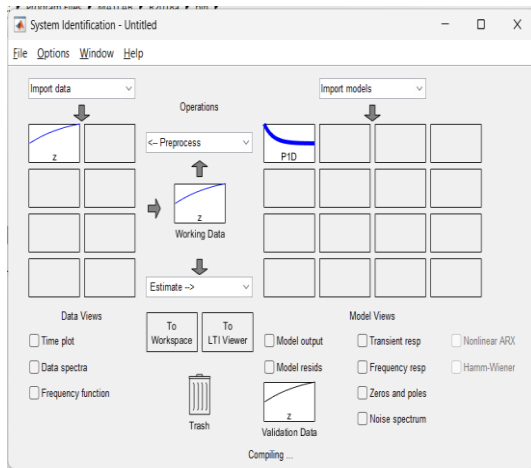
○

40 60 80 100

20 0



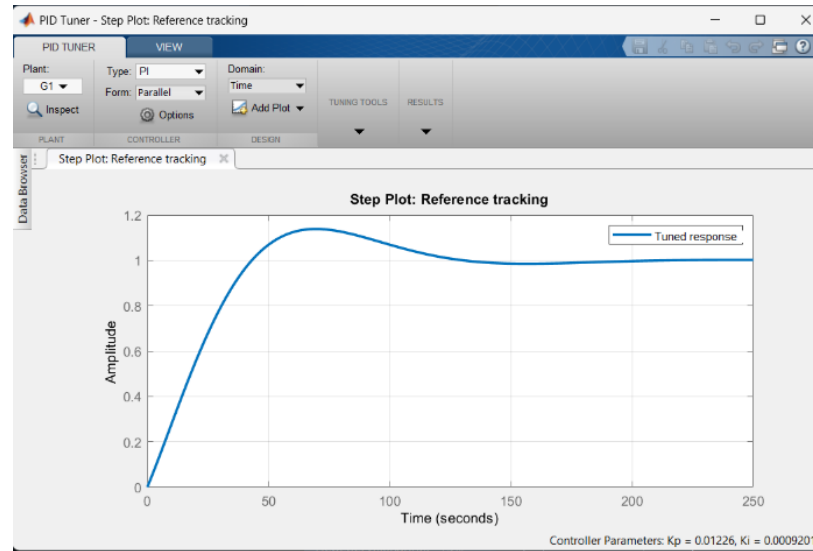
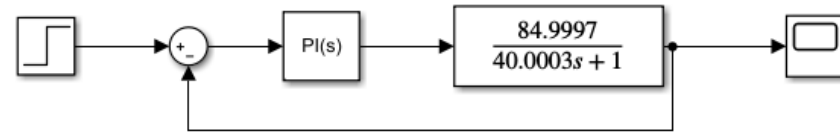
Ідентифікація об'єкта



$$G(s) = \frac{84.9997}{40.0003s + 1}$$

Точність апроксимації — понад 95% відповідності

Дослідження АСР



Параметр	Значення	Примітка
Статична похибка	≈ 0	Завдяки інтегральній складовій ПІ-регулятора
Час встановлення	~ 200 сек	Досягнення значення 1 з допустимим відхиленням
Перерегулювання	$\approx 15\%$	Визначено з графіку перехідної характеристики
Коливання в сталому режимі	Відсутні	Система стабільна після виходу на заданий рівень
Точність апроксимації моделі	$> 95\%$	За результатами System Identification Toolbox