

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра комп'ютерних систем та мереж

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Комп'ютеризована система керування камінним опаленням та розподілом теплого повітря в приватному будинку

Виконав: студент 4 курсу, групи СІ-42

спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

Мехин О. Ю.

(підпис)

(прізвище та ініціали)

Керівник

Жаровський Р.

(підпис)

(прізвище та ініціали)

Нормоконтроль

Tum € . B.

(підпис)

(прізвище та ініціали)

Завідувач кафедри

Осухівська Г.М.

(підпис)

(прізвище та ініціали)

Рецензент

Мудрик И.Я.

(підпис)

(прізвище та ініціали)

Тернопіль
2025

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Осухівська Г.М.
(підпис) (прізвище та ініціали)
«__» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавр
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

студента Мехіна Олександра Юрійовича
(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютеризована система керування камінним опаленням та розподілом
теплого повітря в приватному будинку

Керівник роботи кандидат технічних наук, доцент кафедри КС Жаровський Руслан Олегович
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 27 » січня 2025 року № 4/7–53

2. Термін подання студентом завершеної роботи 25.06.2025 р.

3. Вихідні дані до роботи Технічне завдання

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Аналіз технічного завдання

2. Проєктна частина

3. Практична частина

4. Безпека життєдіяльності, основи охорони праці.

Висновки

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

1. Структурна схема

2. Схема електрична принципова

3. Блок-схема роботи

4. Результати роботи

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Безпека життєдіяльності, основи охорони праці</i>	<i>Пилипець М.І., д.т.н., професор кафедри МТ</i>		

7. Дата видачі завдання 27.01.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	<i>Розробка технічного завдання</i>	<i>27.01 – 02.02</i>	<i>Викон.</i>
2	<i>Аналіз технічного завдання</i>	<i>03.02-01.03</i>	<i>Викон.</i>
3	<i>Аналіз вимог до системи</i>	<i>01.03-27.03</i>	<i>Викон.</i>
4	<i>Проектування структури системи</i>	<i>27.03-01.05</i>	<i>Викон.</i>
5	<i>Проектування, реалізація програмної і апаратної частини</i>	<i>01.05-15.05</i>	<i>Викон.</i>
6	<i>Налаштування і тестування системи</i>	<i>15.05-01.06</i>	<i>Викон.</i>
7	<i>Безпека життєдіяльності, основи охорони праці</i>	<i>01.06 – 05.06</i>	<i>Викон.</i>
8	<i>Оформлення пояснювальної записки і графічного матеріалу</i>	<i>06.06 – 08.06</i>	<i>Викон.</i>
9	<i>Перевірка на академічний плагіат, перевірка керівником та консультантами</i>	<i>9.06 – 15.06</i>	<i>Викон</i>
10	<i>Попередній захист кваліфікаційної роботи бакалавра</i>	<i>16.06 – 22.06</i>	<i>Викон.</i>
11	<i>Захист кваліфікаційної роботи бакалавра</i>	<i>25.06</i>	<i>Викон.</i>

Студент

(підпис)

Мехін Олександр Юрійович

(прізвище та ініціали)

Керівник роботи

(підпис)

Жаровський Руслан Олегович

(прізвище та ініціали)

АНОТАЦІЯ

Мехін О. Ю. Комп'ютеризована система керування камінним опаленням та розподілом теплого повітря в приватному будинку: робота на здобуття кваліфікаційного ступеня бакалавра: спец. 123 – комп'ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2025.

Ключові слова: комп'ютеризована система, кіберфізична система, камінне опалення, Arduino Nano, датчик температури, серво-привід, автоматичне керування, інтерфейс користувача.

У кваліфікаційній роботі розроблено комп'ютеризовану систему моніторингу та керування камінним опаленням із розподілом теплого повітря в приватному будинку. Проведено аналіз існуючих систем автоматизації повітряних камінів, що дозволило обґрунтувати вибір мікроконтролерної архітектури на базі Arduino Nano, температурних сенсорів DS18B20 та MAX6675, газових датчиків MQ-2 і MQ-7, а також виконавчих механізмів на основі серво-приводів, електромагнітних заслінок і вентиляторів.

Проектна частина містить розробку алгоритмів багаторежимного керування процесами горіння, тління, розподілу теплого повітря між приміщеннями, а також реалізацію аварійного захисту при перевищенні концентрації чадного газу та диму. Забезпечено функції моніторингу параметрів у реальному часі, зручний інтерфейс користувача на базі LCD-дисплея з меню налаштувань та сигналізацію аварійних ситуацій. Результати експериментальної перевірки підтвердили стабільну роботу системи в усіх режимах, високу точність підтримання заданих температурних режимів та ефективність роботи захисних функцій.

ANNOTATION

Mekhin O. U. Computerized system for managing fireplace heating and distributing warm air in a private house: Bachelor's Graduation Thesis: speciality 123 – computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2025.

Keywords: computerized system, cyber-physical system, fireplace heating, Arduino Nano, temperature sensor, servo drive, automatic control, user interface.

The bachelor thesis presents the development of a computerized monitoring and control system for fireplace heating with warm air distribution in a private house. An analysis of existing automation systems for air-based fireplaces was conducted, which substantiated the choice of a microcontroller architecture based on Arduino Nano, temperature sensors DS18B20 and MAX6675, gas sensors MQ-2 and MQ-7, as well as actuators based on servo drives, electromagnetic dampers, and air circulation fans.

The project part includes the development of multi-mode control algorithms for combustion, smoldering, and warm air distribution between rooms, as well as the implementation of emergency protection in case of exceeding the permissible concentrations of carbon monoxide and smoke. Real-time monitoring functions are provided, along with a user-friendly interface based on an LCD display with a configuration menu and audible emergency alerts. The results of experimental verification confirmed the stable operation of the system in all modes, high accuracy in maintaining the specified temperature parameters, and the effectiveness of the safety functions.

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ	10
1.1 Огляд предметної області	10
1.2 Аналіз можливих рішень поставленого завдання	13
1.2.1 Регулювання процесу горіння	13
1.2.2 Керування циркуляцією теплого повітря	15
1.3 Технічний огляд існуючих систем автоматизації камінного опалення ...	17
1.4 Вимоги до системи моніторингу і керування камінним опаленням і вентиляцією	20
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА.....	23
2.1 Розробка узагальненої структури.....	23
2.2 Обґрунтування вибору апаратного забезпечення	24
2.3 Розробка електричної схеми	32
2.4 Підбір програмного забезпечення.....	37
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА.....	38
3.1 Моделювання проектних рішень	38
3.2 Розробка алгоритму роботи і програмного забезпечення	41
3.3 Практична реалізація проекту	43
РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ..	51
4.1 Долікарська допомога при опіках	51

					КС КРБ 123.227.00.00 ПЗ						
Змн.	Арк.	№ докум.	Підпис	Дата	Комп'ютеризована система керування камінним опаленням та розподілом теплого повітря в приватному будинку			Літ.	Арк.	Аркуші	
Розроб.		Мехін О. Ю.									
Перевір.		Жаровський								6	
Реценз.								ТНТУ, каф. КС, гр. CI-42			
Н. Контр.		Тиш Є. В.									
Затверд.		Осухівська Г.М.									

4.2 Вимоги безпеки праці під час експлуатації систем вентиляції, опалення чи кондиціонування повітря.....53

ВИСНОВКИ.....56

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....58

Додаток А. Технічне завдання

Додаток Б. Код порграми

Додаток В. Перелік елементів

					<i>КС КРБ 123.227.00.00 ПЗ</i>	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК СКОРОЧЕНЬ

PWM — широтно-імпульсна модуляція (Pulse-Width Modulation)

PID — пропорційно-інтегрально-диференціальний регулятор

LCD — рідкокристалічний дисплей (Liquid Crystal Display)

I2C — послідовний інтерфейс двопровідної шини (Inter-Integrated Circuit)

SPI — послідовний периферійний інтерфейс (Serial Peripheral Interface)

EEPROM — енергонезалежна пам'ять з електричним стиранням та програмуванням (Electrically Erasable Programmable Read-Only Memory)

CO — чадний газ (Carbon Monoxide)

USB — універсальна послідовна шина (Universal Serial Bus)

					<i>КС КРБ 123.227.00.00 ПЗ</i>	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

У сучасних умовах підвищеної уваги до енергоефективності, безпеки та комфорту житлових будинків значно зростає актуальність автоматизованих систем керування кліматичними установками, зокрема системами опалення. Одним із популярних та ефективних джерел тепла у приватних будинках є камінні системи з повітряною подачею тепла. Проте традиційні каміни мають низький коефіцієнт корисної дії та потребують постійного втручання користувача для регулювання процесу горіння та розподілу тепла між приміщеннями. Для покращення ефективності функціонування камінного опалення виникає необхідність розробки комп'ютеризованих систем керування, які здатні в реальному часі здійснювати моніторинг та автоматичне регулювання процесів горіння та розподілу тепла.

Розроблена система базується на використанні мікроконтролера Arduino Nano, який у взаємодії з температурними, газовими та іншими сенсорами, сервоприводами та виконавчими механізмами забезпечує автоматичне регулювання процесу горіння каміна, а також керування потоками гарячого повітря між окремими кімнатами. У системі реалізовано багаторівневий алгоритм управління з поділом роботи на режими розпалу, стабілізації горіння, тління та згасання, що дозволяє значно підвищити ефективність використання енергії дров, забезпечити комфортну температуру у приміщеннях і мінімізувати втрати тепла. Крім того, важливою складовою є система аварійного захисту, що реагує на перевищення порогів концентрації чадного газу та диму для забезпечення безпечної експлуатації каміна.

У процесі виконання роботи було проведено аналіз предметної області, визначено технічні вимоги, розроблено структурну і функціональну схеми системи, обґрунтовано вибір апаратного забезпечення, створено алгоритм функціонування системи, розроблено програмне забезпечення та здійснено практичну реалізацію макету з подальшим тестуванням роботи системи [1].

					КС КРБ 123.227.00.00 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Огляд предметної області

Камінне опалення – це як традиційний елемент інтер'єру, так і альтернативне джерело тепла для приватного будинку. Проте ефективність обігріву будинку залежить від конструкції каміна та способу розповсюдження тепла. Звичайний відкритий камін використовує енергію палива неефективно – більшість тепла виходить з димовими газами, і приміщення обігрівається переважно за рахунок інфрачервоного випромінювання полум'я, поки воно горить. ККД традиційного відкритого каміна становить лише близько 10–15%, тому його доцільно застосовувати лише як додаткове джерело затишку, а не основного обігріву (за наявності основної системи опалення, наприклад, газового котла). Натомість сучасні каміни із закритою топкою та системою розподілу теплого повітря значно підвищують ефективність: оснащення каміна повітропроводами для конвекційного обігріву може збільшити ККД до 70–80%. Іншими словами, правильно обладнаний камін може слугувати повноцінною частиною системи опалення будинку.

Принцип дії повітряної системи камінного опалення (рис 1.1) полягає в нагріванні повітря від гарячої поверхні топки та його транспортуванні по мережі каналів до інших приміщень. Холодне кімнатне повітря забирається (з приміщення або ззовні), нагрівається в конвекційній камері між корпусом топки і облицюванням каміна, після чого піднімається вгору і розподіляється по сусідніх кімнатах через повітропроводи. При цьому продукти згоряння відводяться окремо через димохід і не змішуються з теплим повітрям, що подається в приміщення. Розповсюдження теплого повітря може

					КС КРБ 123.227.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Мехін О. Ю.			Аналіз технічного завдання	Літ.	Арк.
Перевір.		Жаровський					10
Реценз.		Мудрик І.Я.				ТНТУ, каф. КС, гр. СІ-42	
Н. Контр.		Тиш Є. В.					
Затверд.		Осухівська Г.М.					

здійснюватися двома способами: природною (гравітаційною) конвекцією та примусовою подачею повітря.

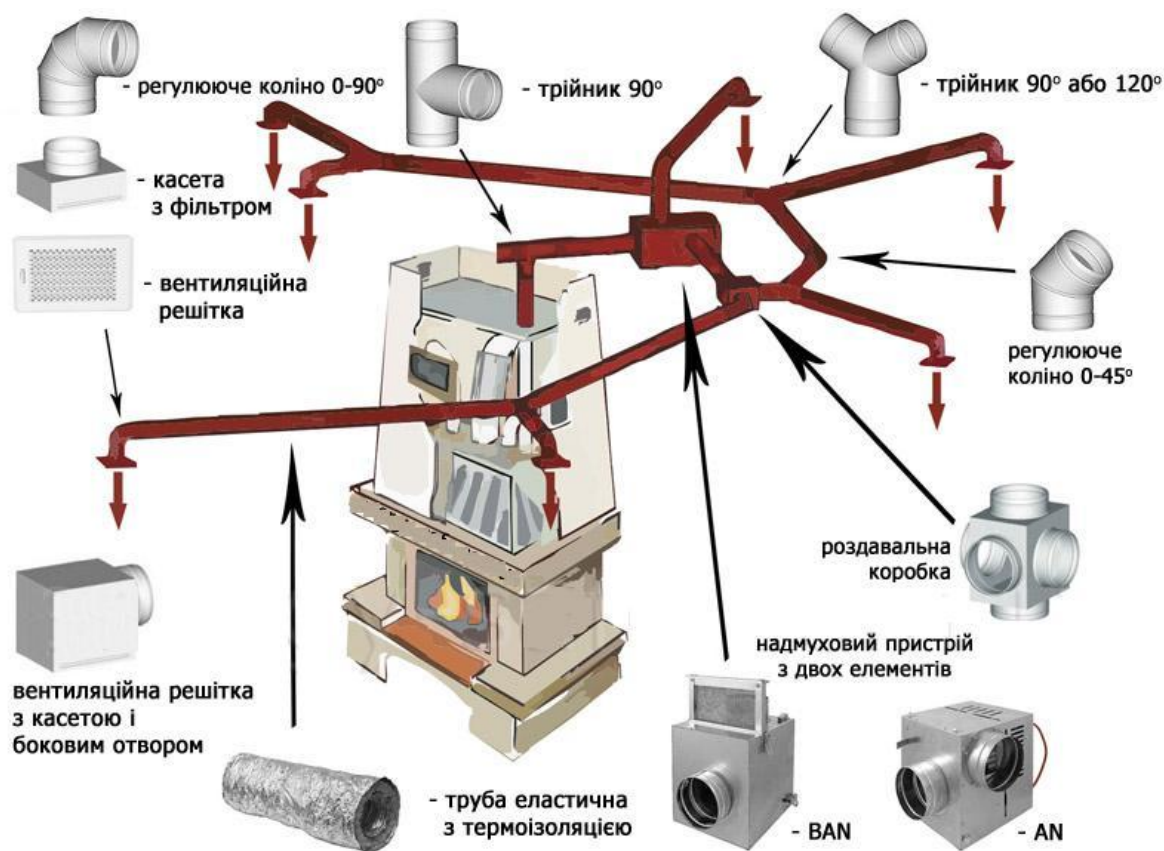


Рисунок 1.1 – Принцип дії повітряної системи камінного опалення

Гравітаційна циркуляція відбувається завдяки тому, що тепле повітря легше за холодне – воно самопливом підіймається у вертикальних каналах та може рухатись горизонтально на відстань до ~4 метрів без використання вентиляторів. Такий спосіб достатньо ефективний для обігріву невеликих одноповерхових чи двоповерхових будинків, особливо якщо камін розташувати по центру будівлі, аби повітряні канали були короткими та рівномірними. У гравітаційних системах кожна кімната зазвичай має окремий канал від каміна, без розгалужень, і довжина горизонтальних ділянок обмежена для запобігання втрат тяги. Примусова система передбачає встановлення електричного вентилятора (турбіни), який під'єднується до

камінна і активно прокачує нагріте повітря через мережу каналів. Камінний вентилятор долає опір довгих або складних повітропроводів, тому теплі повітряні маси можуть доставлятися на значно більші відстані від каміна і навіть вниз – на нижчі рівні будинку. Примусова система дозволяє обігріти віддалені кімнати та поверхи, недоступні для гравітаційної подачі. Отже, у більших за розміром будинках або при вимушеному розташуванні каміна далеко від центру приміщення встановлення вентилятора є практично єдиним рішенням для забезпечення температурного комфорту в усіх кімнатах.

Популярність повітряного опалення від каміна серед власників приватних будинків зумовлена можливістю забезпечити тепло в усьому будинку та зменшити витрати на інші енергоносії. Така система є альтернативним шляхом опалення як невеликих котеджів, так і просторих осель. Важливо врахувати особливості використання камінного опалення в приватному будинку.

По-перше, для ефективної роботи повітряних каналів бажано ще на стадії проектування будівлі передбачити місця для прокладання повітропроводів і розташувати камін якомога ближче до геометричного центру опалюваної зони. Централізоване розміщення каміна та проста, симетрична мережа каналів мінімізують теплові втрати і опір повітря, що підвищує ефективність системи.

По-друге, слід забезпечити приплив свіжого повітря для горіння в топці – найкраще по окремому каналу ззовні безпосередньо до каміна, з розрахунку не менше $\sim 10 \text{ м}^3/\text{год}$ на 1 кВт потужності каміна. Надходження зовнішнього повітря поліпшує гігієнічність опалення: якщо брати повітря для нагріву прямо з кімнати, пил у ньому вигорає на гарячих поверхнях каміна і повертається в приміщення, погіршуючи якість повітря, тоді як при використанні припливного повітря ззовні вдасться цього уникнути. Хоч подача зовнішнього повітря дещо знижує ККД (частина тепла витрачається на нагрівання холодного повітря з вулиці), зате забезпечується потрібний баланс

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

тяги та свіжості повітря у приміщеннях.

По-третє, камінне опалення має інерційність і циклічність: камін дає інтенсивне тепло лише під час активного горіння дров, тому для стабільного обігріву потрібен регулярний контроль та підкладання палива. У типових умовах рекомендується комбінувати камін з іншим джерелом тепла або акумулюючими системами, щоб підтримувати температуру в будинку, коли вогонь поступово згасає.

Саме необхідність контролю процесу горіння та розподілу тепла породжує потребу в автоматизації камінного опалення, особливо в сучасних «розумних будинках», де комфорт користувача і енергоефективність виходять на перший план.

1.2 Аналіз можливих рішень поставленого завдання

Автоматизація процесу обігріву каміном охоплює два ключові аспекти:

- регулювання горіння (отримання тепла)
- керування циркуляцією теплого повітря (передача тепла приміщенням).

Розглянемо технічні підходи до кожного з цих аспектів [2].

1.2.1 Регулювання процесу горіння

У традиційних дров'яних камінах інтенсивність горіння контролюється вручну – за допомогою заслінок подачі повітря, шиберів димоходу або зміною кількості дров. Проте ручне регулювання не забезпечує стабільної температури і не реагує оперативно на зміни умов.

Технічне рішення – система автоматичного контролю горіння, яка підтримує задану температуру згоряння палива. Сучасна електронна автоматика для камінів складається з декількох компонентів: датчиків температури (в топці, димових газів, теплоносія тощо), блоку керування з

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

мікропроцесором та приводу повітряної заслінки (або вентилятора наддуву) для регулювання подачі повітря в камеру згоряння. Принцип дії такої системи: датчики вимірюють поточну температуру в каміні, а контролер порівнює її із заданою і відкриває або прикриває повітряний канал відповідно до потреб – збільшуючи подачу кисню при недостатньому горінні або зменшуючи при перегріві.



Рисунок 1.2 - Автоматичний оптимізатор горіння SAVEN AutoBurn GR-100 з датчиком відкриття дверцят

Наприклад, польські та чеські виробники пропонують автоматику, що орієнтується на температуру димових газів: при падінні температури контролер відкриває заслінку для посилення горіння, а при надмірно високій температурі прикриває, щоб уникнути перегріву і марного виходу тепла в димохід.

Такий підхід оптимізує використання енергії деревини і робить процес горіння більш керованим і безпечним. Результатом є продовження часу горіння однієї закладки дров та підвищення теплосефективності каміна.

Автоматика може подовжити цикл горіння до 30%, тим самим зменшити витрату палива, а також запобігти перегріванню топки і димоходу.

Додатковою функцією розумних контролерів є сповіщення користувача – наприклад, сигналізація про необхідність підкласти дрова, коли температура падає нижче оптимальної. Таким чином, автоматизоване регулювання горіння дозволяє підтримувати стабільну тепловіддачу каміна без постійного втручання людини і з підвищеною безпекою експлуатації [3].

1.2.2 Керування циркуляцією теплого повітря

Другим завданням є автоматизація розподілу тепла від каміна по будинку, тобто керування вентиляторами та повітряними потоками. У найпростішому випадку камінні вентилятори працюють за принципом термостатичного реле: вентилятор автоматично вмикається, коли температура повітря або корпусу каміна досягає певного порогу (наприклад, 50–60 °С), і вимикається після остигання нижче порогу.

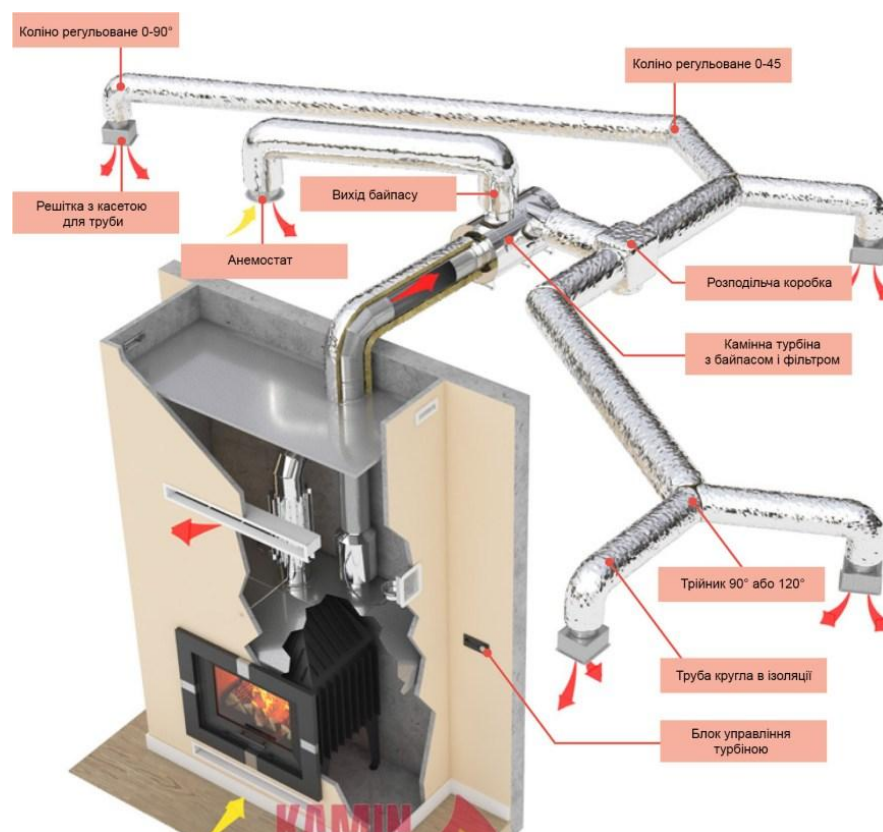


Рисунок 1.3 – Елементи камінної системи РГП

Змн.	Арк.	№ докум.	Підпис	Дата

КС КРБ 123.227.00.00 ПЗ

Такі рішення реалізуються за допомогою біметалевих датчиків або термостатів, встановлених у камері нагріву – коли камін розпалений, контакт замикається і подає живлення на мотор вентилятора. Подібна автоматика є нескладною і недорогою, вона запобігає ситуаціям, коли користувач забуває увімкнути або вимкнути вентилятор вручну. Однак для повноцінної автоматизації клімат-контролю цього недостатньо, адже простого вмикання/вимикання може не вистачати для підтримання комфортної температури у різних кімнатах. Більш просунуті підходи передбачають плавне регулювання швидкості вентиляторів та зональне керування потоками повітря. Наприклад, встановлення вентилятора з регульованими обертами і використання датчиків температури в кожній ключовій зоні будинку дозволяє автоматично змінювати інтенсивність подачі теплого повітря, підтримуючи задану температуру в цих зонах. В літературі зазначено, що застосування примусової подачі повітря з можливістю плавного регулювання дозволяє контролювати розподіл теплого повітря по приміщеннях і тим самим підтримувати в них бажаний температурний режим. Тобто камін з вентилятором та відповідним контролером може керовано обігрівати кілька кімнат: коли температура в віддаленій кімнаті нижча за встановлену, система підвищує швидкість вентилятора або відкриває відповідну заслінку, спрямовуючи більше теплого повітря в цю кімнату. Навпаки, досягнувши комфорту, подача тепла може зменшуватися, аби уникнути перегріву. Зональне керування можливе, якщо мережа повітропроводів оснащена регульованими заслінками або кількома вентиляторами – контролер тоді вирішує, куди саме і скільки теплого повітря подати в кожен момент часу.

Технічна реалізація автоматизації повітряного опалення камином може базуватись на різних платформах.

Поширеним є використання мікроконтролерів (Arduino, STM32 тощо) або вбудованих систем (наприклад, Raspberry Pi) для збору даних з датчиків і керування виконавчими пристроями. Такі контролери можуть бути

					КС КРБ 123.227.00.00 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

спеціалізованими (промислові контролери для HVAC) чи створеними індивідуально під проект. В якості датчиків застосовуються терморезистори, термопари або цифрові сенсори температури, здатні працювати при високих температурах біля каміна та передавати точні показники. Виконавчі механізми включають електродвигуни вентиляторів, сервоприводи повітряних заслінок і реле або тиристорні схеми для керування насосами (у випадку комбінованих систем з водяним контуром).

Алгоритми керування можуть варіюватися від простого гістерезисного контролю (включення/вимкнення за порогоми) до PID-регуляторів, що плавно підтримують потрібну температуру. Важливим аспектом є безпечна та надійна робота: система повинна мати захист від аварійних ситуацій – наприклад, відключати подачу повітря при перегріві каміна або, навпаки, відкривати заслінки на повний обсяг для охолодження, вмикати аварійну сигналізацію при небезпечних температурах. Додатково може передбачатися резервне живлення (UPS) для вентилятора, щоб при зникненні електрики уникнути перегріву каміна без охолоджувальної циркуляції повітря.

1.3 Технічний огляд існуючих систем автоматизації камінного опалення

На сучасному ринку представлено кілька основних технічних підходів до побудови автоматизованих систем керування камінним опаленням. Системи варіюються від простих локальних контролерів до багатозонних комп'ютеризованих систем з мережевим керуванням та віддаленим моніторингом.

Системи контролю та обчислювальні платформи є базовим елементом більшості рішень. Електронні блоки керування представляють собою контролери, які реалізують логіку управління системою. За рівнем складності їх можна умовно поділити на кілька груп:

					<i>КС КРБ 123.227.00.00 ПЗ</i>	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

– Локальні спеціалізовані контролери: наприклад, моделі Tatarek (Польща), Timrex (Чехія), Plum (Польща), SAVEN (Україна). Ці контролери оснащуються вбудованими алгоритмами регулювання горіння, мають інтегровані дисплеї для локального керування та дозволяють задавати параметри без зовнішнього програмування. Вони орієнтовані на конкретні моделі камінів і переважно працюють автономно, без інтеграції з розумними мережами.

– Універсальні контролери на базі мікрокомп'ютерів: рішення з використанням мікроконтролерних платформ (STM32, ESP32, Arduino, Raspberry Pi Compute Module). Такі системи мають значно більші можливості для програмування складних сценаріїв керування, інтеграції з мережами розумного дому, збору статистики, дистанційного моніторингу та аналітики. Перевагою є відкритість протоколів, масштабованість і гнучкість.

– Промислові PLC-контролери (Siemens LOGO!, WAGO, Schneider Electric): застосовуються в складних об'єктах з великою кількістю зон та високими вимогами до надійності. Підтримують вбудовану логіку, захищені комунікації та аварійні сценарії.

Датчики для моніторингу використовуються для забезпечення повноцінного контролю температурних, безпекових та динамічних характеристик камінної системи використовуються різні типи сенсорів. До таких датчиків ставляться спеціальні вимоги.

Температурні датчики високих температур топки і димових газів:

- термопари типу К (до +1000 °C);
- платинові датчики PT100/PT1000 (до +500 °C).

Температурні датчики повітря в повітропроводах і приміщеннях:

- цифрові сенсори DS18B20 (до +125 °C);
- датчики SHT31, SHT85 (температура + вологість, точність $\pm 0,1$ °C).

Газові датчики для виявлення CO₂ (чадний газ):

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

- MQ-7 датчик CO₂;
- MiCS-5524 - електрохімічні сенсори середнього діапазону.

Датчики диму і полум'я:

- фоточутливі та оптичні сенсори з видимим/ІЧ каналом;
- MQ-2 датчик диму.

Датчики положення для сервоприводів заслінок:

- потенціометричні енкодери;
- магнітні індуктивні сенсори положення.

Забезпечення стійкої передачі даних між датчиками, контролерами і виконавчими пристроями досягається шляхом використання надійних протоколів обміну.

Внутрішня локальна шина:

- I²C — для цифрових сенсорів низької швидкості (DS18B20, SHT);
- 1-Wire — для температурних датчиків у пасивних мережах;
- RS485 (Modbus RTU) — для промислових систем з великою кількістю вузлів;
- CAN — у багатозонних системах для зонального керування.

Зовнішні мережеві протоколи:

- Ethernet (TCP/IP, MQTT, HTTP REST);
- Wi-Fi (802.11n/ac);
- Zigbee, Z-Wave — для інтеграції у бездротові мережі розумного будинку;
- GSM/3G/4G — для мобільного віддаленого моніторингу [4].

Виконавчі механізми використовуються для реалізації фізичного регулювання процесів в камінній системі такі групи актуаторів:

- Регулювання подачі повітря до топки: сервоприводи заслінок (наприклад, Siemens GDB161, Belimo), електромеханічні приводи з позиційним управлінням (90°, 180°).

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

– Вентилятори подачі теплого повітря: осьові або центробіжні вентилятори з фазовим або ШІМ-регулюванням (230 В АС або 24 В DC), продуктивність 300–800 м³/год.

– Заслінки зонального контролю повітряних магістралей: моторизовані повітряні клапани Ø100–160 мм з 2-позиційним або аналоговим керуванням.

– Насоси (для комбінованих водно-повітряних систем): циркуляційні насоси з PWM або релейним управлінням.

Засоби виведення даних та інтерфейси користувача є важливою складовою сучасних систем контролю. Інформаційне забезпечення користувача здійснюється на кількох рівнях:

– Локальні дисплеї контролерів: графічні LCD/OLED (128×64, 320×240), сенсорні TFT-панелі, кнопкові панелі або енкодери для налаштування параметрів.

– Веб-інтерфейси: доступ через браузер з будь-якого пристрою в локальній мережі або через захищене VPN-з'єднання.

– Мобільні додатки: нативні застосунки (Android, iOS) з віддаленим контролем і отриманням повідомлень (Push-нотифікації).

– Інтеграція з системами «розумного дому»: сумісність з Home Assistant, Google Home, Amazon Alexa через стандартні API (MQTT, Webhooks) [5].

1.4 Вимоги до системи моніторингу і керування камінним опаленням і вентиляцією

Система моніторингу й керування камінним повітряним опаленням повинна задовольняти наступні вимоги:

– Точність і діапазон вимірювань: датчики температури мають охоплювати приблизно від -20...40 °С (зовнішнє повітря/кімнатний

					КС КРБ 123.227.00.00 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

термостат) до +100...+125 °С (нагріте повітря від вентилятора). Допустима похибка вимірювань – не гірше $\pm 0,5$ °С (цифрові датчики DS18B20) або $\pm 0,1$ °С (PT1000). Відхилення напруги живлення пристроїв – $\pm 10\%$.

– Особливості вентиляції: система повинна підтримувати заданий об'ємний потік повітря (наприклад, 300–600 м³/год) та плавне регулювання його швидкості. Вентилятори спроектовані на переміщення повітря до ~ 150 °С, тому система керування має враховувати температуру повітря, аби не перевантажити двигун.

– Надійність і безпека: електронні компоненти та проводка повинні відповідати захисту IP (щонайменше IP20 для внутрішніх пристроїв). Передбачаються апаратні засоби захисту від перегріву (термоконтакти, автомати) і аварійне відключення. Система керування повинна забезпечувати сигналізацію аварійних ситуацій (відсутність тяги, перевищення температури) і можливість дистанційного відключення.

– Функціональність: підтримка програмованих режимів (комфортний/економний), можливість інтеграції з кімнатними і виносними термостатами, відображення стану системи на дисплеї чи в мобільному додатку. Забезпечити модульність прошивки із гнучкою підтримкою підключення нових датчиків або зон без переписування основного коду. Передбачити підтримку віртуальних зон управління для оптимізації режимів при змінній присутності людей.

– Апаратна платформа контролера. Контролер має бути побудований на основі вбудованої обчислювальної системи з мікроконтролером (наприклад, STM32, ESP32, Raspberry Pi). Це дозволить забезпечити високу швидкодію для обробки кількох датчиків одночасно, складних алгоритмів регулювання та мережевих протоколів обміну. Забезпечити наявність апаратних watchdog-механізмів для апаратного перезавпуску при зависанні або збоях програмного забезпечення. Плати повинні мати гальванічну розв'язку входів/виходів для забезпечення безпеки

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		21

в системах з високими температурами та при індустріальних перешкодах (ЕМС).

– Архітектура передачі даних. Внутрішню комунікацію датчиків - реалізувати як шину з багатоточковим підключенням (наприклад, I²C, 1-Wire для температурних датчиків). Зовнішня комунікація (інтерфейси інтеграції з системою розумного дому, SCADA-системами, мобільними додатками) через мережеві інтерфейси Ethernet / Wi-Fi / GSM / Zigbee з підтримкою стандартних мережевих протоколів (MQTT, REST API, WebSocket). Передбачити локальну буферизацію даних у контролері (на flash чи SD-карті) з журналюванням параметрів для автономної роботи при втраті зв'язку [6].

Аналіз предметної області та існуючих технічних рішень засвідчує актуальність проблеми створення комп'ютеризованої системи моніторингу та керування камінним опаленням у приватних будинках.

Традиційні системи обігріву мають обмежену ефективність через циклічність процесу горіння та нерівномірність розподілу тепла. Запровадження сучасних вбудованих обчислювальних систем, інтелектуальних датчиків і виконавчих механізмів дозволяє реалізувати адаптивне регулювання, багатозонний контроль мікроклімату та інтеграцію у концепцію «розумного дому».

Особливу складність розробки становить необхідність обробки даних у реальному часі, забезпечення високої надійності, безпеки та гнучкості системи при її масштабуванні. Застосування комп'ютерної інженерії в даній сфері відкриває широкі можливості для створення ефективних, енергоощадних і комфортних систем побутового теплопостачання.

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

2.1 Розробка узагальненої структури

Структурна схема наведена на рисунку 2.1. Ця структурна схема показує взаємозв'язок між різними компонентами системи.

В основі схеми — центральний блок керування каміном і розподілом теплого повітря. До нього підключено три основні групи підсистем:

- датчики;
- виконавчі механізми;
- інтерфейс користувача та індикація.

Вся система є замкнутим циклом зворотного зв'язку, що відповідає класичній системі автоматичного регулювання:

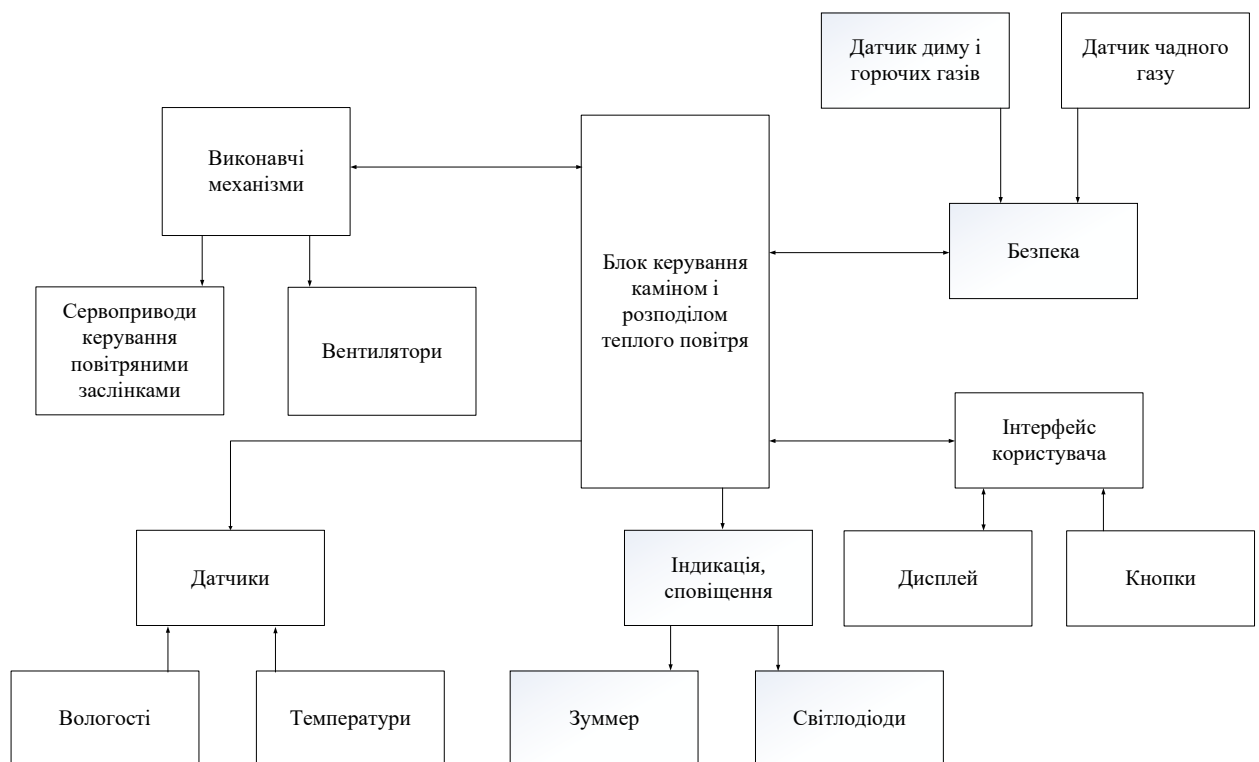


Рисунок 2.1 – Структурна схема системи

					КС КРБ 123.227.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата	Проектна частина		
Розроб.		Мехін О. Ю.					
Перевір.		Жаровський					
Реценз.		Мудрик І.Я.					
Н. Контр.		Тиш Є. В.					
Затверд.		Осухівська Г.М.					
					Літ.	Арк.	Аркушів
						23	
					ТНТУ, каф. КС, гр. CI-42		

Датчики включають модулі температури (для контролю температури повітря у кімнатах і на виході з каміну з термопарою для контролю температури димових газів), датчики диму і горючих газів, датчик чадного газу. Всі датчики формують зворотний зв'язок системи як для терморегуляції, так і для забезпечення безпеки [7].

Блок безпеки обробляє сигнали від газових сенсорів і формує аварійні сигнали при перевищенні порогових значень чадного газу, диму чи CO₂. У випадку загрози система переходить в аварійний стан, закриває заслінки, вимикає вентилятори та активує світлову та звукову індикацію тривоги.

Виконавчі механізми включають сервоприводи керування повітряними заслінками, які регулюють обсяг свіжого повітря, що подається до топки каміна, а також заслінки зонального розподілу повітря по кімнатах. Вентилятори забезпечують активну циркуляцію теплого повітря у вентиляційних каналах.

Інтерфейс користувача реалізується через LCD-дисплей для відображення температурних та аварійних параметрів системи, а також через кнопки і ротаційний енкодер для налаштування режимів роботи. Індикатори та зумери забезпечують оперативне інформування користувача про стан системи та аварійні ситуації.

Дана структура забезпечує гнучку модульну побудову системи, можливість масштабування на більшу кількість зон опалення, підвищену безпеку та комфорт управління процесом опалення приміщення каміном. Для реалізації проекту необхідно у відповідності до вищенаведеної схеми вибрати відповідне апаратне і програмне забезпечення [8].

2.2 Обґрунтування вибору апаратного забезпечення

Вибір апаратного забезпечення для комп'ютеризованої системи керування камінним опаленням базується на технічних вимогах,

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

функціональних задачах системи та умовах експлуатації.

В якості центрального обчислювального модуля вибрано контролер Arduino Nano на базі мікроконтролера ATmega328P. Цей контролер має достатню кількість цифрових та аналогових входів/виходів для підключення всіх необхідних сенсорів, актуаторів та інтерфейсів користувача.

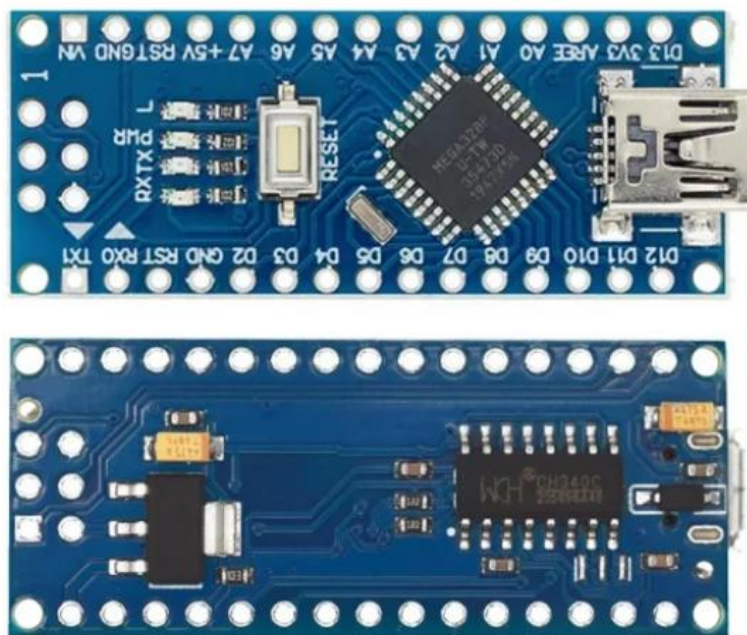


Рисунок 2.2 – Контролер Arduino Nano (ATmega328P)

Його перевагами є компактність, енергоефективність, доступність великої кількості бібліотек для роботи з різними периферійними модулями, а також простота інтеграції додаткових модулів через стандартні інтерфейси UART, SPI та I2C (табл. 2.1).

Для вимірювання температури у контрольних точках обрано різні типи датчиків, що відповідають умовам вимірювання. Для контролю температури димових газів (Т1) з топки каміна використовується термопара типу К (рис. 2.3) з модулем MAX6675 (рис. 2.4). Ця комбінація дозволяє знімати температуру до +1024 °С з високою точністю у високотемпературних умовах топки [10].

Таблиця 2.1 – Характеристики Arduino Nano (ATmega328P)

Параметр	Значення
Мікроконтролер	ATmega328P
Напруга живлення (через USB)	5 В
Робоча напруга логіки	5 В
Напруга на вході VIN (зовнішнє)	7–12 В (макс. 20 В через стабілізатор)
Тактова частота	16 МГц
Об'єм Flash-пам'яті	32 КБ (з них 2 КБ для бутлоадера)
SRAM (оперативна пам'ять)	2 КБ
EEPROM	1 КБ
Кількість цифрових входів/виходів	22 (з них 6 PWM)
Кількість аналогових входів	8 (A0–A7)
UART (послідовних портів)	1 апаратний UART (TX/RX)
SPI	1 апаратний SPI (MISO, MOSI, SCK, SS)
I2C	1 апаратний I2C (SDA, SCL)
USB-конвертер на платі	CH340G
Конектор USB	Mini-USB
Рівень логіки UART	5 В
Габарити плати	45×18 мм
Струм споживання контролера	до 50 мА (без зовнішньої периферії)



Рисунок 2.3 – Високотемпературний датчик температури

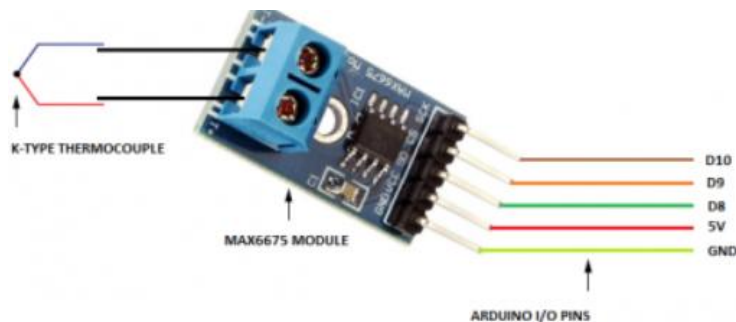


Рисунок 2.4 – Модуль MAX6675

Для вимірювання температури в тепловій камері каміна (T2) та в кімнатах (T3) застосовано цифрові датчики DS18B20 (рис.2.5), які мають високий клас точності та зручний протокол OneWire, що дозволяє з'єднувати декілька датчиків в одну шину з мінімальною кількістю з'єднувальних проводів [11].



Рисунок 2.5 - Датчик температури DS18B20

Таблиця 2.2 – Характеристики датчика температури

Напруга живлення	3.0..5.5 В
Діапазон температур	-55°C..+125°C
Точність показань температури	0.5 °C
Крок показів	0.0625 °C
Інтерфейс	1-Wire
Споживаний струм	1 мА

З метою забезпечення безпеки обрано газові сенсори MQ-2 та MQ-7 (характеристики наведені в таблиці 2.3). Сенсор MQ-2 дозволяє виявити дим

та горючі гази в діапазоні концентрацій 200–10000 ppm, а MQ-7 спеціалізований для контролю чадного газу (CO) у діапазоні 20–2000 ppm. Обидва датчики підключаються до аналогових входів Arduino, що дозволяє контролювати рівень небезпечних речовин у реальному часі. Для контролю рівня CO₂ у повітрі додано високоточний інфрачервоний сенсор MH-Z19, який працює по UART-інтерфейсу та дозволяє проводити вимірювання концентрації CO₂ у межах до 5000 ppm.

Таблиця 2.3 – Характеристики газових сенсорів MQ-2 та MQ-7

Параметр	MQ-2	MQ-7
Тип сенсора	Газовий сенсор диму і горючих газів	Газовий сенсор чадного газу (CO)
Контрольовані гази	Метан (CH ₄), LPG, CO, H ₂ , спирти, дим	Чадний газ (CO), водень (H ₂)
Діапазон концентрацій	200 – 10000 ppm	20 – 2000 ppm
Напруга живлення	5 В	5 В
Робоча температура	-20°C ... +50°C	-10°C ... +50°C
Вихідний сигнал	Аналоговий	Аналоговий
Точність	~ ±10% після калібрування	~ ±10% після калібрування
Час прогріву (перший запуск)	до 24 годин	до 24 годин
Час стабілізації (робочий прогрів)	кілька хвилин	кілька хвилин
Споживання струму	~150 мА (нагрівальний елемент)	~150 мА (нагрівальний елемент)
Час відгуку	10 – 30 секунд	10 – 60 секунд
Особливості роботи	Постійний нагрів	Перемінний підігрів
Розміри модуля	~32×20 мм	~32×20 мм

У якості виконавчих механізмів застосовано серво-привід для управління аналоговою заслінкою K1 (рис.2.6), що регулює подачу повітря в топку каміна, дискретні релейні виконавчі механізми KZ1-KZn для керування відкриттям зональних заслінок подачі теплого повітря в кімнати. Так як реалізація заслінок в каналних системах вентиляції є різною, відповідно необхідно використати більш потужний привід. А підключення буде здійснено через драйвер на силовому MOSFET транзисторі IRF520 (рис. 2.7)

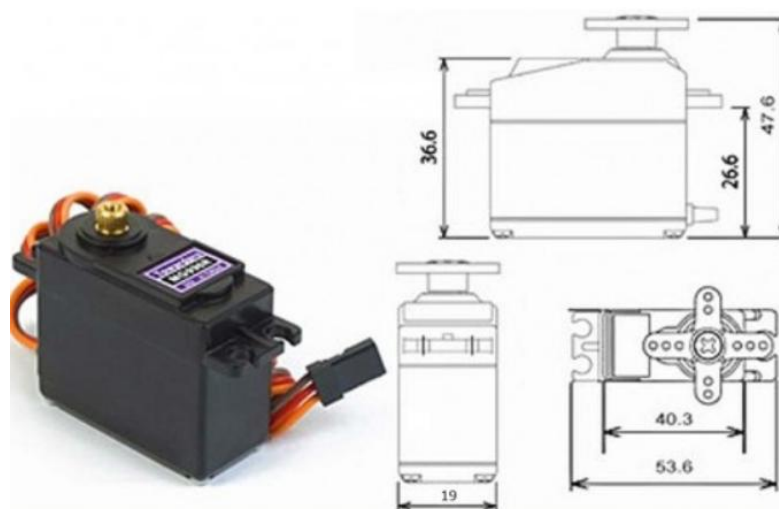


Рисунок 2.6 - Сервопривід керування аналоговою заслінкою K1



Рисунок 2.7 – Драйвер на силовому MOSFET транзисторі IRF520

Характеристики:

– розміри: 33 x 24 мм

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		29

- маса: 10 г
- керуюча напруга: від 3.5В
- управління: цифрове
- вихідна напруга: 0-24В
- Струм: до 5А, при струмі понад 1А необхідно використовувати радіатор.

Також вентилятор V1, який покращує циркуляцію повітря в системі. Підключається через реле SSR-25DA.

Серво-привід підключається до ШІМ-виходу Arduino, що дозволяє точно регулювати кут відкриття заслінки. Реле комутуються цифровими виходами та дозволяють керувати живленням зональних магістралей.

Для зручної взаємодії з користувачем система обладнана LCD-дисплеєм 16x2 символів з інтерфейсом I2C, ротаційним енкодером з кнопкою для налаштування параметрів, світлодіодами для індикації станів та звуковим бузером для сигналізації аварійних ситуацій. LCD-дисплей дозволяє відображати актуальну інформацію про температуру, режими роботи та аварійні стани системи.

Для організації безперервного живлення будемо жити всю систему від одного акумулятора TR 18650 – його ємності має вистачити приблизно на півгодини роботи блоку керування.

Блок живлення виберу зовнішній з вилкою на 12 В, 2 А. Знижувальні перетворювачі LM2596 зменшать напругу з 12 В до 5 В та 9 В для живлення складових системи управління каміном. Вони недорогі й досить потужні - витримують струм до 2 А.

Для живлення контролера від акумулятора застосуємо підвищувальний перетворювач 5V DC-DC Booster, а для живлення сервоприводу - підвищувальний перетворювач напруги MT3608, що забезпечує напругу 6 В.

Для заряджання та захисту акумулятора буде використано два модулі, причому один з них - у нестандартній схемі підключення. Перший модуль

					КС КРБ 123.227.00.00 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

забезпечує повноцінне заряджання акумулятора і одночасно подає напругу на контролер. Другий модуль лише подає напругу на сервопривід, виконуючи при цьому функцію захисту акумулятора від глибокого розряду. Це зроблено тому, що струму 1 А, який обмежує перший зарядний модуль, недостатньо для живлення всіх споживачів. Таким чином, другий модуль додатково виконує функцію захисту без підключення споживачів безпосередньо до акумулятора.

Таблиця 2.4 – Перелік обладнання для реалізації системи

Компонент	Призначення	Кількість
Arduino Nano	Центральний контролер	1
MAX6675 з термопарою К	Вимірювання температури димових газів (T1) каміна	1
DS18B20	Вимірювання температури в нагрівачі та кімнатах (T2, T3)	2+
MQ-2	Контроль диму та горючих газів	1
MQ-7	Контроль чадного газу (CO)	1
Серво-привід	Регулювання повітряної заслінки (K1) каміна	1
Реле KZ	Керування заслінками зональних каналів	Залежить від кі-ті кімнат
Вентилятор V1	Циркуляція повітря	1
LCD 16x2 (I2C)	Відображення інформації	1
Ротаційний енкодер	Навігація по меню	1
Світлодіоди та буюер	Індикація та аварійна сигналізація	2/1
Джерело живлення 12 В, 2 А	Забезпечення стабільного живлення	1

2.3 Розробка електричної схеми

Для роботи системи реалізовано з'єднання всіх модулів з Arduino Nano. Живлення від джерела 12 В подається на вентилятори та на входи регулятора напруги, який формує стабільні 5 В для Arduino, логіки датчиків та сервоприводів. Arduino Nano живиться від 5 В (як по USB, так і через Vin/5V). Для кожного датчика передбачено окремий провід живлення +5 В і GND. Додаткові резистори 4.7 кОм під'єднані між лінією даних і живленням для DS18B20 (підтягування) [12].

На рисунках 2.8 – 2.13 наведені схеми елементів і особливості їх підключення.

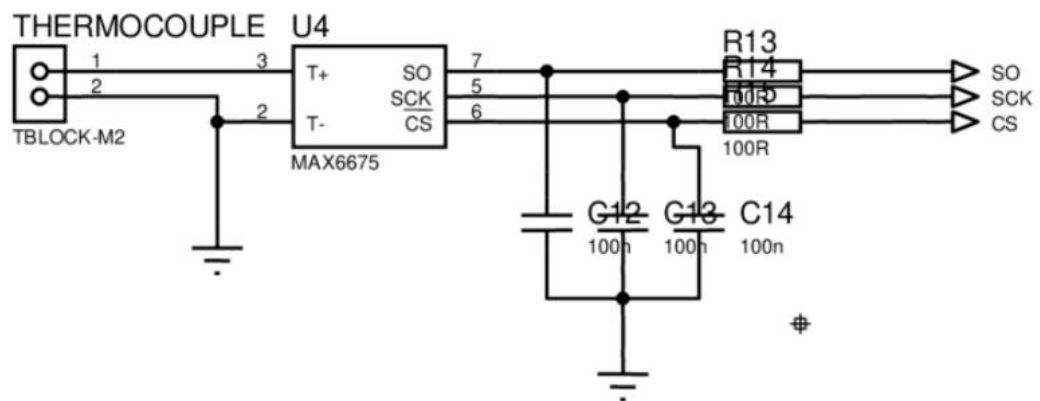


Рисунок 2.8 – Схема модуля MAX6675 для підключення термопар

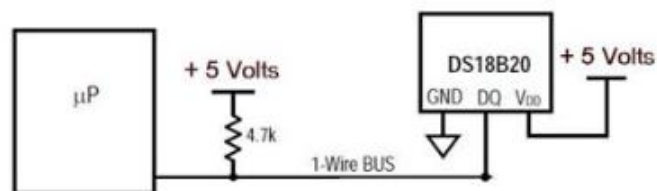


Рисунок 2.9 – Схема підключення датчика температури

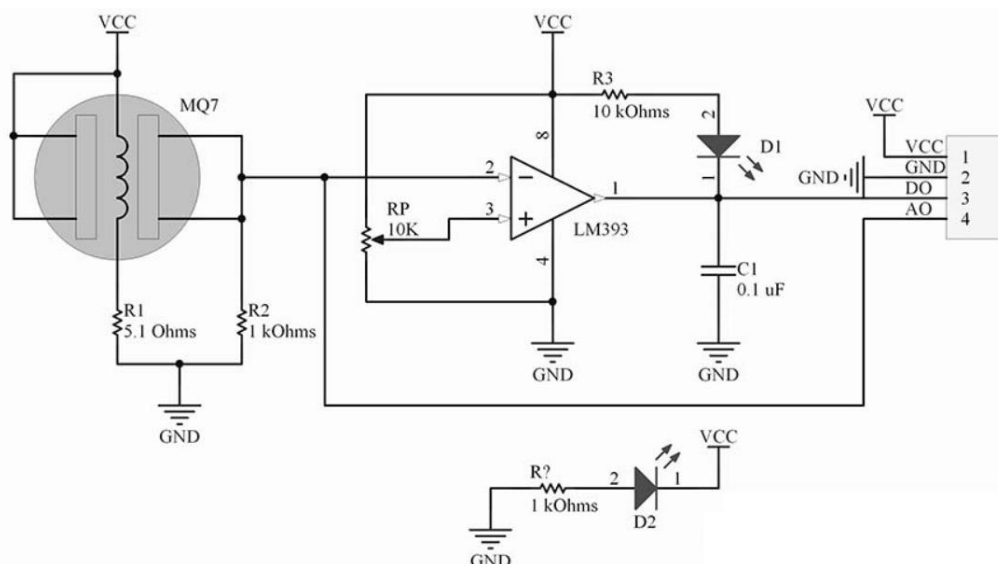


Рисунок 2.13 – Схема під'єднання датчика MQ-7

Підключення портів контролера виконано за такими правилами:

- SPI-інтерфейс (MAX6675): вивід SO (Data) датчика MAX6675 підключено до порту D12 (MISO), SCK – до D13, а CS – до D10 Arduino (вхід CS). Ця схема відповідає апаратному SPI-навантаженню Arduinoanalog.com.
- Шина 1-Wire (DS18B20): «датчик даних» DS18B20 під'єднано до порту D2 через лінію з резистором підтягування; живлення +5 В подається окремо. Бібліотека OneWire дозволяє працювати з цим лінійним протоколом.
- Аналогові входи (MQ-2, MQ-7): аналогові виходи газових сенсорів з'єднані відповідно з A0 (MQ-2) та A1 (MQ-7) Arduino. Датчики потребують постійної напруги +5 В та GND, сигнал подається на АЦП контролера.
- UART (MH-Z19): модуль MH-Z19 підключено по послідовному інтерфейсу. TX-модулю наводиться на RX (D3), а RX модулю – на TX (D4) або через SoftwareSerial, забезпечуючи обмін ASCII-командами.
- I²C (LCD-дисплей): використовуються контакти A4 (SDA) та A5 (SCL) для підключення I²C-бекпленду LCD 16×2. Лінії I²C потребують підтягувальних резисторів (часто уже вбудовані в модуль).

- Серводвигуни: керуються імпульсами з цифрових ШІМ-виходів (наприклад, D5, D6). Джерелом живлення 5 В для серво виступає той же регулятор, що і для плати.
- Вентилятори та реле: вентилятори можуть підключатися через драйвер (напр. L298N, як на макеті) або безпосередньо через транзистори до ШІМ-виходу (D9) та маси. Реле (для навантажень 220 В чи 12 В) комутуються цифровими сигналами з D7, D8.
- Індикація та керування: буюер підключено до D11 (з можливістю ШІМ для мелодій), світлодіоди аварії – до D12/D13 з резисторами обмеження струму. Ротаційний енкодер має два сигнальні виходи (наприклад, D9, D10) та кнопку (підключається до D8 через внутрішнє підтягування) (рис.2.14).

Таблиця 2.5 – Відповідності портів Arduino та призначення кожного з'єднання

Назва пристрою	Призначення	Порт Arduino Nano	Тип з'єднання / Примітка
MAX6675 - SO (DO)	Температура димових газів	D12 (MISO)	SPI
MAX6675 - CS	Чіп-селект	D10	SPI
MAX6675 - SCK	Тактова лінія	D13 (SCK)	SPI
DS18B20 (спільна шина 1-Wire)	Температура повітря в нагрівачі і кімнаті	D2	1-Wire (обидва на одній шині через підтягуючий резистор 4.7 кОм)
MQ-2	Дим, горючі гази	A0	Аналоговий вхід
MQ-7	Чадний газ CO	A1	Аналоговий вхід
Серво-привід (K1)	Заслінка подачі повітря в топку	D5	PWM

Назва пристрою	Призначення	Порт Arduino Nano	Тип з'єднання / Примітка
Реле KZ1	Заслінка подачі повітря в кімнату	D6	Цифровий вихід
Вентилятор V1 (через драйвер)	Примусова циркуляція повітря	D9	Цифровий вихід
Кнопка RUN	Старт системи	D4	Цифровий вхід (з підтягуванням до +5В)
Кнопка STOP	Зупинка системи	D7	Цифровий вхід
Енкодер CLK	Меню, налаштування	A2	Цифровий вхід
Енкодер DT	Меню, налаштування	A3	Цифровий вхід
Кнопка енкодера (SW)	Вибір в меню	A4	Цифровий вхід
LCD I2C SDA	Дисплей	A5	I ² C SDA
LCD I2C SCL	Дисплей	A6	I ² C SCL
Світлодіоди аварійної індикації	Стан системи	D11 / D8	Цифровий вихід
Бuzzer	Звукова сигналізація	D3	PWM
Датчик живлення	Контроль живлення	A7	Аналоговий вхід
Джерело живлення	Загальне живлення системи 5В / 12В	VIN / 5V	Подається через стабілізатори та DC-DC

Загальна схема наведена на кресленні КС КРБ 123.226.00.00 ЕЗ.

2.4 Підбір програмного забезпечення

Програмна частина реалізована у середовищі Arduino IDE мовою C++. У коді програми будуть використані стандартні бібліотеки: OneWire (1-Wire), SPI (SPI-шина), Wire (I²C), Servo (керування сервоприводами), LiquidCrystal_I2C (LCD-дисплей. Функція setup() конфігурує виходи та входи, ініціалізує серійну передачу, LCD та сенсорні бібліотеки. У безперервному циклі loop() відбувається послідовне зчитування всіх сенсорів (методами analogRead, функціями бібліотек для DS18B20, MAX6675, МН-Z19 тощо), після чого здійснюється аналіз даних і відповідне керування виконавчими елементами.

Логічна структура скетчу побудована у вигляді станів: наприклад, ініціалізація, зчитування сенсорів, перевірка безпеки, обчислення керуючих сигналів, оновлення індикаторів. Після кожного циклу контролер затримується на заданий інтервал або очікує події. Зчитані дані та стан системи відображаються на дисплеї. Бібліотека LiquidCrystal_I2C забезпечує прості функції print() для відображення інформації на екрані, а Servo.h – метод write() для повороту заслінок.

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		37

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Моделювання проектних рішень

Макет пристрою спроектовано за модульною структурою, що передбачає три групи компонентів: датчики, виконавчі механізми та інтерфейс користувача. Усі підсистеми об'єднані в замкнений контур зворотного зв'язку згідно з класичними принципами автоматичного регулювання. Основні компоненти макета показані на рисунку 3.1.

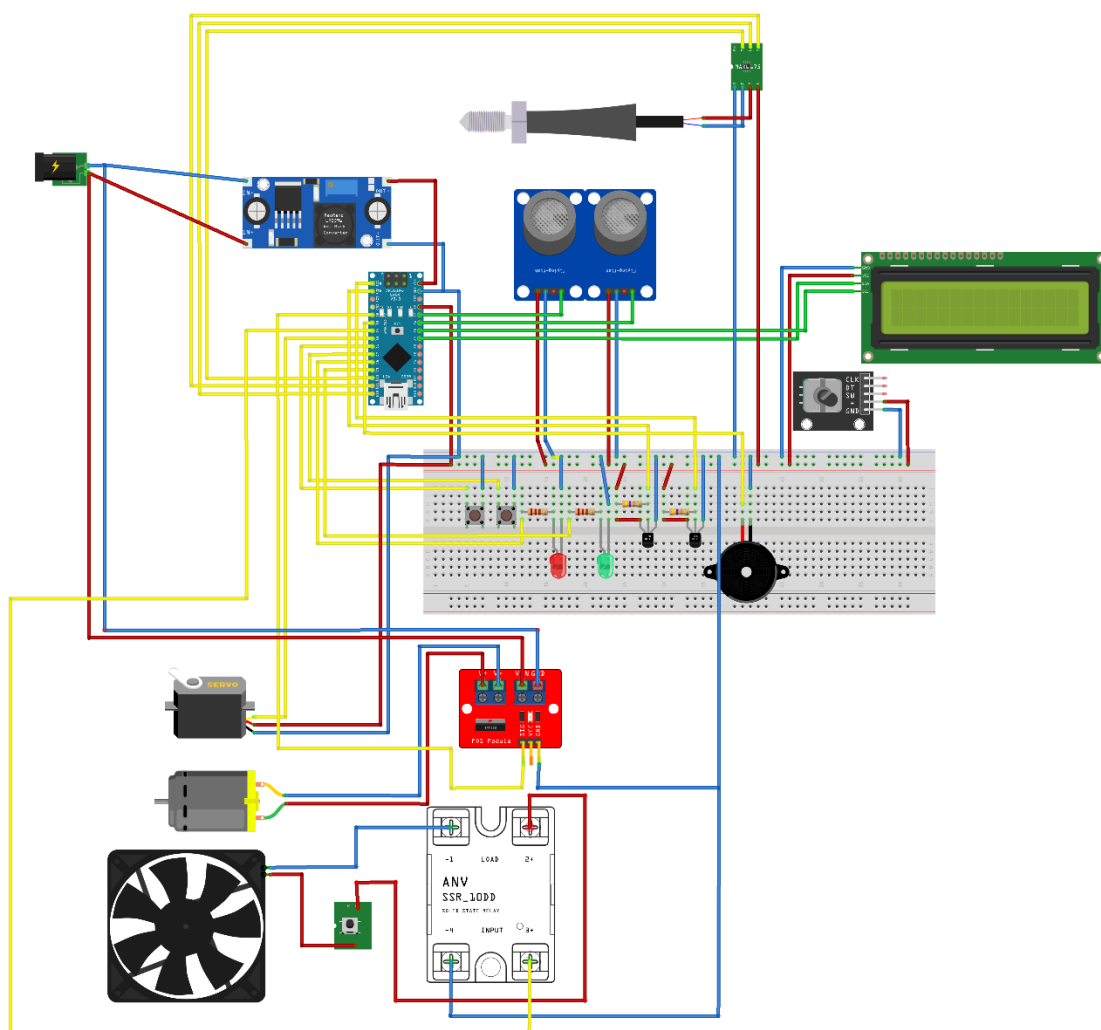


Рисунок 3.1 – Макет системи керування каміном і опаленням

					КС КРБ 123.227.00.00 ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата								
Розроб.		Мехін О. Ю.			Практична частина			Літ.	Арк.	Аркушів		
Перевір.		Жаровський								38		
Реценз.		Мудрик І.Я.						ТНТУ, каф. КС, гр. CI-42				
Н. Контр.		Тиш Є. В.										
Затверд.		Осухівська Г.М.										

Датчики: модулі температури (термопара К-типа з модулем MAX6675 для контролю температури димових газів, цифрові датчики DS18B20 для контролю температури в камері та в кімнаті); сенсори диму і горючих газів (MQ-2) та чадного газу (MQ-7) для безпеки.

Виконавчі механізми: сервоприводи керують повітряними заслінками (подача первинного повітря в топку та зональне розподілення теплого повітря), а вентилятори забезпечують примусову циркуляцію теплого повітря по каналах.

Інтерфейс: LCD-дисплей відображає поточні температури та стан системи, кнопки і ротаційний енкодер служать для вибору режимів та налаштувань; світлові й звукові індикатори (LED, зумер) попереджають про аварійні ситуації.

Режим розпалювання: після натискання кнопки RUN контролер входить у режим інтенсивного розпалювання. У цьому режимі сервопривод максимально відкриває заслінку подачі повітря до камери, а вентилятор включається для подачі додаткового повітря. Це сприяє швидкому запаленню палива. Як тільки температура в камері або димових газах досягне заданого порогу, система автоматично переходить у основний робочий режим з адаптивним регулюванням подачі повітря.

Обробка температур: макет обробляє температурні показники трьох зон. Дані з термопари (T_1 – димові гази) та з DS18B20 (T_2 – топкова камера, T_3 – кімната) надходять до контролера.

ПІД-регулятор порівнює фактичну температуру приміщення з цільовою та формує керуючий сигнал для сервопривода заслінки і вентилятора. Якщо температура нижче бажаної, заслінка відкривається ширше і потужність вентилятора зростає, щоб пришвидшити нагрів. При досягненні або перевищенні цільового значення регулювання зменшує подачу повітря для підтримки стабільної температури.

Системи керування вентиляцією за допомогою Arduino та PWM-вентилятора. Вентилятор підключається через широтно-імпульсну модуляцію

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		39

та реле.

Системи вентиляції і заслінок макету працюють узгоджено. Сервопривод керує аналоговим положенням повітряної заслінки, пропорційно змінюючи об'єм подачі повітря (аналогічно прикладу HVAC-системи). Вентилятор вмикається для підтримки циркуляції повітря через канали. Швидкість вентилятора може регулюватися PWM; для повного відключення подачі повітря передбачене застосування реле. Аварійна сигналізація: блок безпеки постійно контролює роботу газових сенсорів і температур. MQ-2 фіксує наявність горючих газів та диму (H_2 , CH_4 , LPG, CO тощо), MQ-7 – концентрацію чадного газу (CO). При перевищенні заданих порогових значень на будь-якому з цих сенсорів, або при критичному підвищенні температури чи виході напруги живлення за межі норми, система негайно переходить в аварійний режим. У разі небезпеки контролер закриває всі заслінки, відключає вентилятори та активує світлову/звукову тривогу, щоб запобігти накопиченню небезпечних газів і перегріву.

Ротаційний енкодер з кнопкою використовується для навігації по меню та вводу параметрів системи. Обертання енкодера змінює виділений пункт меню, кнопка підтверджує вибір.

Меню і налаштування: користувач взаємодіє з системою через LCD та ротаційний енкодер. Обертання енкодера дозволяє прокручувати пункти меню та налаштовувати значення параметрів системи (наприклад, граничних температур чи PID-коефіцієнтів), кнопка на енкодері підтверджує вибір. Змінені налаштування записуються у вбудовану EEPROM-пам'ять мікроконтролера, що гарантує їх збереження при вимкненні живлення. Моделювання на макетній платі та тестування: зібраний на макетній платі пристрій дозволив перевірити всі ключові функції системи. На початковому етапі було перевірено з'єднання всіх компонентів згідно з схемою. Далі протестовано коректність зчитування температур: покази MAX6675 і DS18B20 відображались на дисплеї при зміні умов (підміна терморпарі змінним резистором або прогрів і охолодження). Сервоприводи заслінок

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		40

успішно реагували на команди контролера (змінювали кут згідно з PID-розрахунками), а вентилятор стабільно включався і вимикався за кодом керування. Меню системи протестовано: за допомогою енкодера виконувалися зміна та збереження параметрів – значення успішно записувались у EEPROM та відновлювались після перезавантаження. Імітування аварійних ситуацій (наприклад, підвищення вихідної напруги сенсорів газу) приводило до спрацювання сигналізації і переходу в безпечний режим. Також перевірено роботу ПД-регулятора: при зміні цільової температури в кодї макета заслінка переміщувалась до відповідного кута, стабілізуючи реальну температуру на новому рівні. Усі зазначені функції вдалося успішно протестувати на макеті, що підтвердило обґрунтованість обраних рішень.

3.2 Розробка алгоритму роботи і програмного забезпечення

Після ввімкнення відбувається ініціалізація контролера, портів та змінних, після чого програма послідовно переходить у режими: розпал, контроль горіння, тління. Паралельно система постійно відстежує газові датчики і при перевищенні їх порогових значень негайно активує аварійний режим.

Ініціалізація. Налаштування контролера Arduino Nano та підключених датчиків/виконавчих пристроїв (ініціалізація змінних, встановлення початкових параметрів). Після завершення ініціалізації відбувається перехід у режим розпал.

Розпал. Дія: серво-привід K1 повністю відкритий (максимальна подача повітря у камеру згоряння); заслінки KZ1 закриті (немає потоку повітря у кімнати).

Умова завершення: режим триває доти, доки температура димових газів T1 не перевищить 250 °C та температура в повітряній камері T2 стане вище 50 °C. При виконанні цих умов відбувається перехід у режим контроль

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		41

горіння.

Інакше: залишатися в режимі розпал, продовжуючи подавати повітря.

Контроль горіння. Дія: PID-регулювання сервоприводу K1 для підтримання температури T1 у межах 300–500 °С; заслінки KZ1 відкриті; вентилятор увімкнено.

Умови підтримання: система підтримує T1 у діапазоні 300–500 °С і T2 у межах 80–150 °С.

Додаткові умови: якщо температура у кімнаті T3 перевищує 20 °С – вентилятор вимикається, якщо T3 > 22 °С – заслінки KZ1 закриваються.

Перехід до тління: якщо температури виходять за межі (наприклад, T1 падає нижче 300 °С або T2 виходить за 80–150 °С), система переходить у режим Тління.

Режим тління. Перевірка дров: якщо навіть при відкритих заслінках T1 опускається нижче 150 °С – виводиться повідомлення «Завантажити дрова» і активується бузер.

Якщо дрова завантажено (користувач підтвердив завантаження) – перехід у режим розпал.

Якщо дрова не завантажено – сервопривід K1 закривається, вентилятор вимикається.

Подальші дії: якщо після цього температура T2 падає нижче 25 °С – заслінки KZ1 закриваються.

Завершення роботи: коли T1 опускається нижче 50 °С – показується повідомлення «Завершення роботи» і бузер подає сигнал (згасання каміна); раз на 10 хвилин звучить короткий сигнал.

Вимкнення: через 30 хвилин після початку цього стану система повністю вимикається (OFF).

Аварійний режим (паралельно). Умова: якщо MQ-2 > 400 або MQ-7 > 300 (значне задимлення чи висока концентрація CO).

Дія: усі заслінки KZ1 закриваються, вентилятор вимикається, бузер активується безперервно.

					КС КРБ 123.227.00.00 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

Повідомлення: відображається «АВАРІЯ: дим або СО» (сигнал тривоги).

Наслідок: система переходить у безпечний стан (необхідне втручання користувача), основні режими тимчасово зупинені.

Описаний алгоритм роботи наведений на кресленні КС КРБ 123.226.00.01

3.3 Практична реалізація проекту

Для розміщення всіх елементів системи було обрано корпус від електрощита (рис.3.2).



Рисунок 3.2 – Зовнішній вигляд зібраної системи

Внутрішня частина системи контролю наведено на рисунку 3.3.

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43



Рисунок 3.3 – Реалізація системи контролю

На рисунках 3.4- 3.8 показані змонтовані елементи і їх призначення.

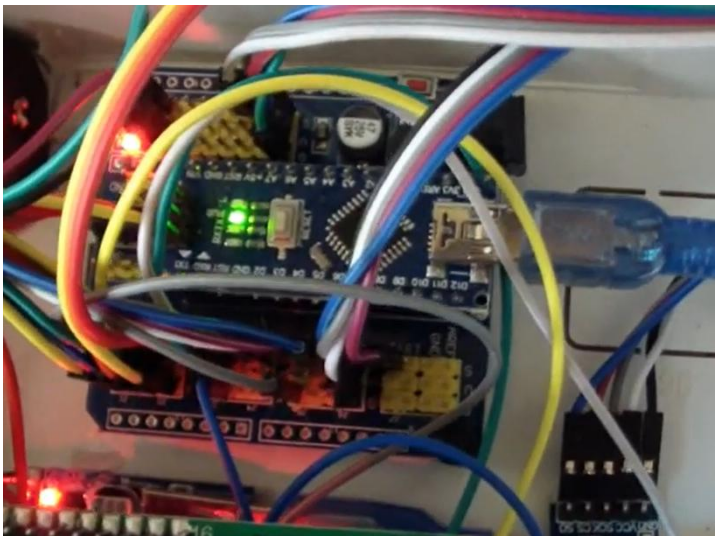


Рисунок 3.4 – Змонтована на шилді плата Arduino Nano

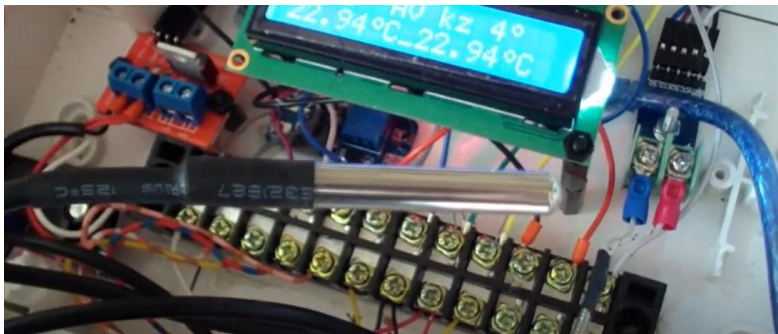


Рисунок 3.5 – Під’єднаний датчик температури

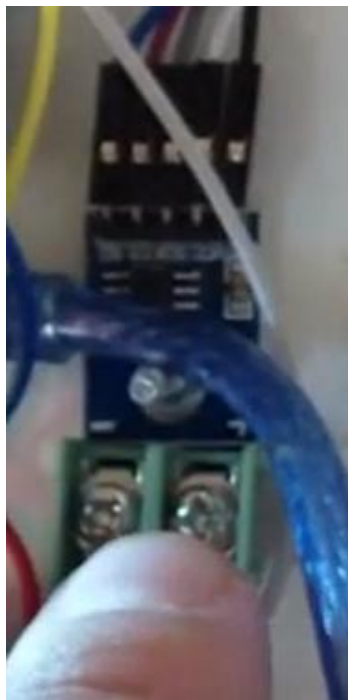


Рисунок 3.6 – Модуль для підключення термопари



Рисунок 3.7 – Модуль комутації заслонки

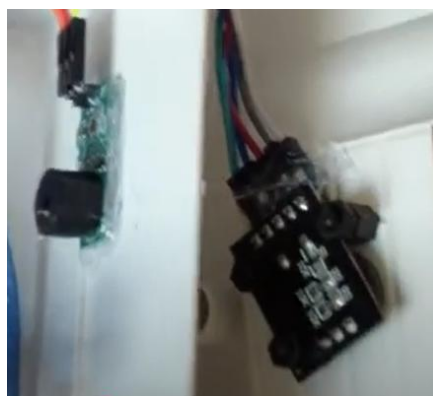


Рисунок 3.8 – Змонтований п'єзодинамік і кнопка керування

Розглянемо роботу пристрою і основних меню. Після старту системи бачимо температуру на виході з камінної топки; температуру, температуру в камері камінаї температуру в кімнаті, а також значення відкриття заслінки каміна. AV – відображає наявність зовнішнього живлення, якщо воно відсутнє то маленькі букви



Рисунок 3.9 – Виведення інформації про температуру підключення і стан заслінки

Всі регулювання і переходи і налаштування здійснюються з допомогою однієї кнопки (рис.3.10)



Рисунок 3.10 – Перехід по меню налаштувань



Рисунок 3.11 – Перехід в налаштування режиму розпалювання і налаштування режиму температури на виході з каміна

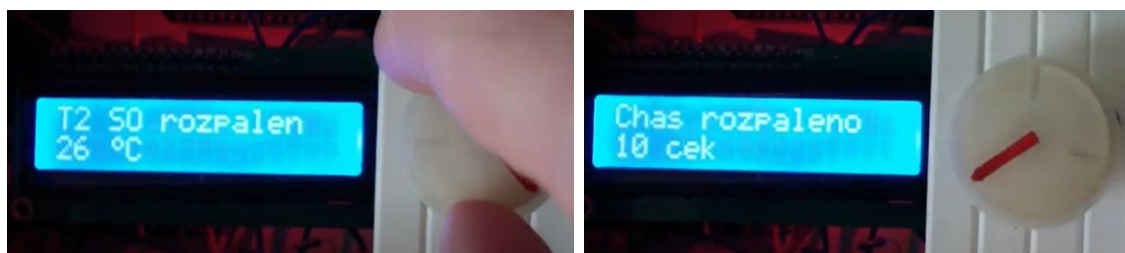


Рисунок 3.12 – Налаштування температури і часу затримки при якій система переходить в режим контроль горіння

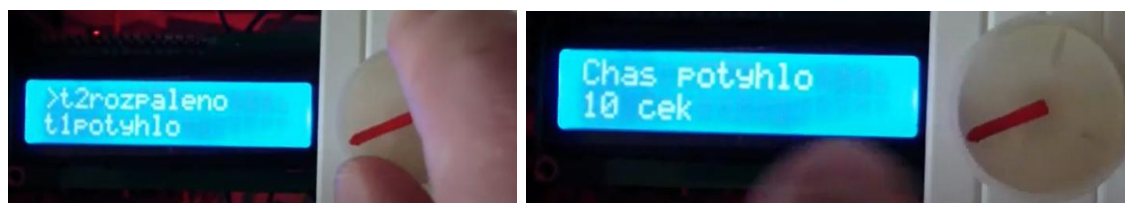


Рисунок 3.13 – Налаштування режиму тління



Рисунок 3.14 – Перехід в режим очікування коли відсутнє 220В

Тепер наведу декілька фото індикації режимів роботи рис. 3.15 – 3.17.



Рисунок 3.15 – Режим розпалювання



Рисунок 3.16 – Режим контроль горіння



Рисунок 3.17 – Перехід в режим тління (очікування)

На рисунку 3.18 наведено залежність температури в каміні від часу протягом повного циклу горіння. Виділено чотири ключові фази роботи системи:

- Розпал (0–30 хв): швидке зростання температури з кімнатної (~20°C) до 250°C. Лінія графіка тут йде на стрімке підвищення (червоний фон), і саме ця фаза має відрізок часу 30 хвилин.
- Контроль горіння (30–150 хв): підтримка високої температури приблизно в діапазоні 300–500°C. На графіку це зелена ділянка часу (2

години). Температура у цьому режимі залишається на високому рівні (майже плоска червона лінія з невеликими коливаннями), демонструючи повільне зростання до середньої точки діапазону та стабілізацію.

– Тління (150–330 хв): поступове падіння температури до 150°C. Ця фаза триває 3 години і позначена бежевим фоном на графіку. Червона лінія плавно спадає вниз, показуючи поступове охолодження палива.

– Згасання (330–360 хв): фінальна 30-хвилинна фаза, у якій температура падає до 50°C. Видно синю ділянку часу, де температура різко, але плавно, знижується до кінцевого значення.

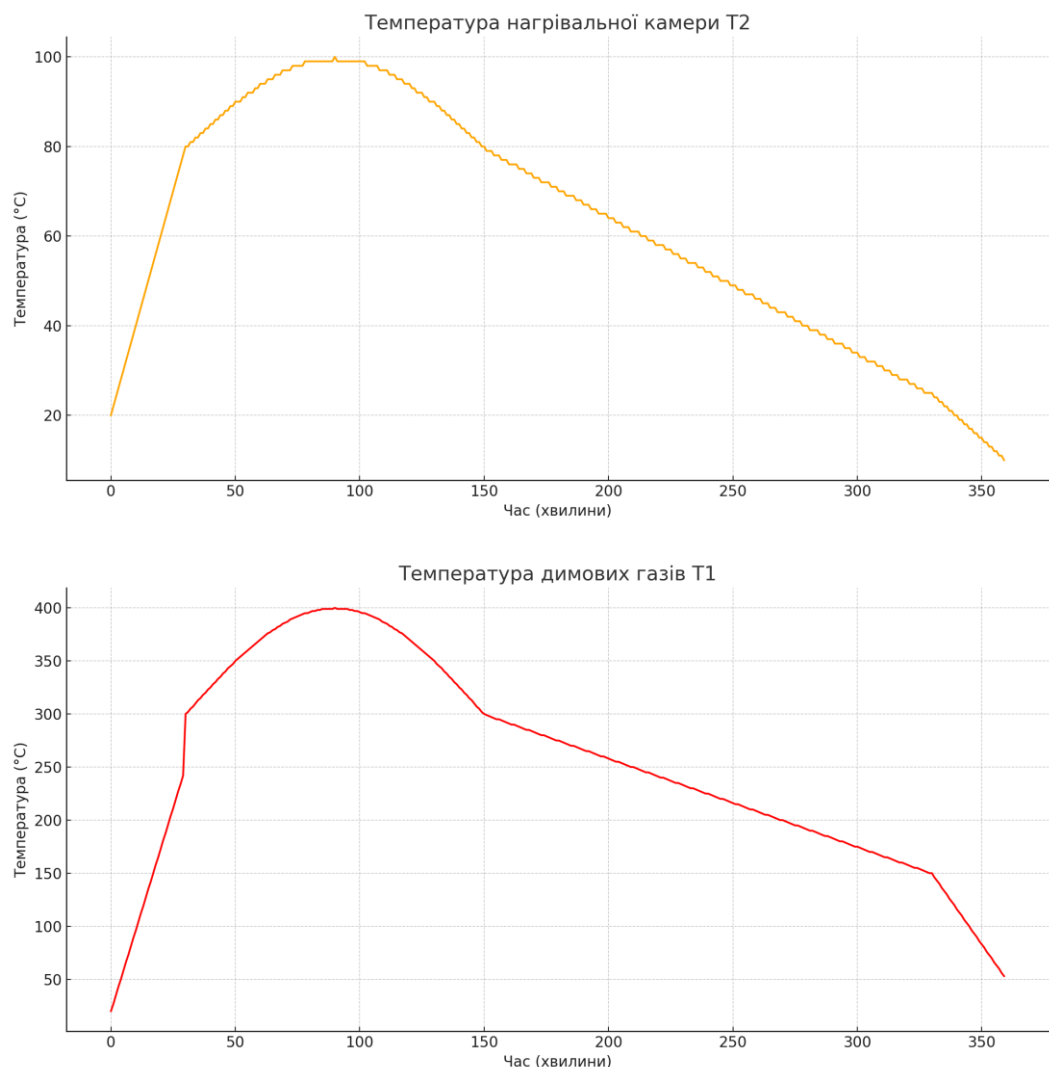


Рисунок 3. 18 – Графік отриманий з температурних датчиків каміну

Такий графік демонструє основні режими циклу роботи каміна і

дозволяє наочно побачити, як змінюється температура в кожній фазі – від розпалювання дров до повного згасання вогню. Візуалізація створена за заданими температурними порогами та тривалістю кожного режиму, щоб вихідна інформація була максимально зрозумілою та наочною.

Таким чином було реалізовано і протестовано систему контролю за каміном і системою опалення. Система була розроблена з врахуванням модульного підходу. Тобто можна додавати нові датчики температури і керуючі механізми для регулювання подачі повітря в різні кімнати.

					<i>КС КРБ 123.227.00.00 ПЗ</i>	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Долікарська допомога при опіках

Долікарська допомога при опіках в контексті комп'ютеризованої системи керування камінним опаленням є важливим аспектом для забезпечення безпеки користувачів.

У побуті найчастішими є термічні опіки від відкритого вогню. Вторинні опіки можуть виникнути при займанні одягу або контакті з нагрітими поверхнями. Такі ураження викликають сильний біль, порушення функцій шкіри, інтоксикацію організму та можуть призводити до опікового шоку [14].

Перше, що потрібно зробити при побутовому опіку – припинити дію ушкоджувального чинника. Якщо на потерпілому горить одяг, полум'я слід збити, обгорнувши людину щільною тканиною (ковдрою, пальтом, пледом) або облити водою. Після припинення горіння важливо обережно звільнити уражені ділянки тіла, уникаючи пошкодження пухирів. За необхідності одяг розрізають [15].

При обмежених термічних опіках першу допомогу надають шляхом охолодження ураженої поверхні холодною водою протягом 10–15 хвилин. Це дозволяє зменшити глибину ураження та зняти біль. Потім на уражену ділянку накладають асептичну пов'язку з чистої тканини або стерильного бинта. Заборонено проколювати пухирі, наносити мазі чи олії, а також відривати прилиплий одяг [16].

У разі великих опікових поверхонь потерпілого загортають у чисте простирадло або покривають ковдрою, забезпечуючи тепло, та дають пити теплий солодкий чай чи воду з додаванням солі (1 ч. ложка солі та 0,5 ч. ложки соди на 1 л води). Для полегшення стану можна дати заспокійливі засоби,

					КС КРБ 123.227.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Мехін О. Ю.			Безпека життєдіяльності, основи охорони праці	Літ.	Арк.
Перевірів		Жаровський Р.					51
Консульт.						ТНТУ, каф. КС, гр. СІ-42	
Н. Контр.							
Затверд.		Осухівська Г.М.					

наприклад, настойку валер'янки (20 крапель на воду). При наявності ознак шоку – блідість, слабкий пульс, зниження температури тіла – слід покласти потерпілого горизонтально та злегка підняти ноги [14, 15].

При хімічних опіках 10-20 хвилин виконують промивання обпеченого місця струменем проточної води. У випадку отримання опіку лугом накладають марлеву пов'язку, яка змочена розчином борної кислоти (1 грам на склянку води).

Заборонено відривати пухирі, а також відривати від них одяг, каніфоль, сургуч, бо це може спричинити інфекцію та затягнути тривалість загоєння ран. Заборонено засипати рани порошками, змащувати мазями та маслом. При обширних і важких опіках (більше 15-30% всієї поверхні тіла) виникає загальне ураження організму, яке супроводжується важким шоком (опікова хвороба), викликає інтоксикацію організму, зміни складу крові, зміни в роботі центральної нервової системи (біль). Чим більша опікова поверхня, тим більше нервових закінчень уражено і тим сильніше проявляються явища травматичного шоку.

При опіках з'являється велика кількість виділень крізь опікову поверхню плазми крові, виникає отруєння організму від продуктів розпаду змертвілої тканини, які із зони ушкодження всмоктуються організмом. З'являються такі симптоми як блювання, загальна слабкість, головний біль. Потерпілому потрошки і часто дають пити воду з питною сіллю (одна чайна ложка солі + пів чайної ложки соди на один літр води).

Для пов'язок застосовують індивідуальний пакет, стерильний бинт. Опечену поверхню можна накрити чистою тканиною з бавовни, пропрасованою гарячою праскою чи змоченою перманганатом калію, горілкою або етиловим спиртом, які зменшують біль.

До приїзду швидкої допомоги важливо спостерігати за диханням і свідомістю потерпілого. Якщо потерпілий у свідомості – заборонено залишати його без нагляду. У випадку втрати свідомості, але збереженого дихання, потрібно покласти людину у стабільне бокове положення [15].

					КС КРБ 123.227.00.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

Під час фахової медичної допомоги застосовують серцеві, знеболювальні препарати, протишокові заходи, інфузійну терапію. Медичні працівники також проводять профілактику інфікування рани та оцінюють необхідність подальшого хірургічного втручання.

Правильна та своєчасна долікарська допомога при опіках відіграє вирішальну роль у зменшенні тяжкості наслідків та збереженні життя потерпілого. Основні принципи – припинення дії шкідливого чинника, охолодження, накладення стерильної пов'язки, уникнення додаткових пошкоджень та якнайшвидше звернення за медичною допомогою.

4.2 Вимоги безпеки праці під час експлуатації систем вентиляції, опалення чи кондиціонування повітря

Системи вентиляції, опалення та кондиціонування повітря є невід'ємною частиною багатьох будівель і сприяють забезпеченню комфортних умов для роботи і проживання. Проте, під час експлуатації цих систем виникають певні ризики, пов'язані з безпекою праці. З метою забезпечення безпечного та здорового середовища існують вимоги, які необхідно враховувати під час проектування, монтажу та експлуатації цих систем.

Один з основних аспектів безпеки праці пов'язаний з відповідним проектуванням систем вентиляції, опалення та кондиціонування повітря. Проектування повинно враховувати тип приміщення, його розмір, кількість мешканців, а також їх потреби у повітрі та комфортних умовах.

Забезпечення належного обсягу повітря та правильного розподілу температури є важливими аспектами, що впливають на безпеку та здоров'я мешканців будинку.

Під час монтажу систем вентиляції, опалення та кондиціонування повітря важливо дотримуватись вимог безпеки. Роботи з монтажу повинні проводитись кваліфікованими спеціалістами з використанням відповідного обладнання та інструментів. Крім того, необхідно дотримуватись правил

електробезпеки, особливо при підключенні електричних компонентів систем. При встановленні систем вентиляції слід також забезпечувати безпечний доступ до всіх елементів системи для обслуговування та ремонту.

Експлуатація систем вентиляції, опалення та кондиціонування повітря повинна здійснюватися відповідно до рекомендацій виробника та нормативних документів. Регулярна технічна перевірка та обслуговування систем допомагає попередити можливі аварійні ситуації та забезпечує безперебійну роботу.

Очищення фільтрів, перевірка роботи вентиляторів та контроль параметрів повітря (таких як вологість, температура, швидкість потоку повітря) є важливими аспектами регулярного обслуговування.

Додатковою мірою безпеки є навчання користувачів щодо правильного використання систем вентиляції, опалення та кондиціонування повітря.

Природна та штучна вентиляції повинні відповідати наступним санітарно-гігієнічним вимогам:

- створювати в робочій зоні приміщень нормовані метеорологічні умови праці (температуру, вологість і швидкість руху повітря);
- повністю усувати з приміщень шкідливі гази, пари, пил та аерозолі або розчиняти їх до гранично допустимих концентрацій;
- не вносити в приміщення забруднене повітря ззовні або шляхом засмоктування забрудненого повітря з суміжних приміщень;
- не створювати протягів чи різкого охолодження;
- бути доступними для управління та ремонту під час експлуатації;
- не створювати під час експлуатації додаткових незручностей (наприклад, шуму, вібрацій, попадання дощу, снігу).

Користувачі повинні бути ознайомлені з принципами роботи систем, процедурами заміни фільтрів, а також правилами експлуатації в разі аварійних ситуацій, наприклад, виникнення задимлення або пожежі. Знання про запобіжні заходи та процедури евакуації можуть врятувати життя і здоров'я працівників у випадку надзвичайних ситуацій.

					КС КРБ 123.227.00.00 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

Одним з основних аспектів безпеки праці є контроль якості повітря, яке постачається системами вентиляції. Регулярний аналіз якості повітря на наявність шкідливих речовин, таких як газів, пил, бактерії чи алергени, допомагає виявити можливі проблеми з системою та вжити відповідних заходів для їх вирішення.

Крім того, важливо враховувати нормативні вимоги щодо розміщення систем вентиляції, опалення та кондиціонування повітря. Наприклад, деякі компоненти систем можуть вимагати певної відстані від робочих місць або повинні бути відокремлені від інших приміщень з метою запобігання поширенню шкідливих речовин.

Усі користувачі, які мають прямий контакт з системами вентиляції, опалення та кондиціонування повітря, повинні мати необхідні засоби індивідуального захисту. Це можуть бути респіратори, захисні окуляри, відповідний одяг тощо. Застосування відповідних засобів індивідуального захисту допомагає зменшити ризик впливу шкідливих речовин на здоров'я працівників [17].

Забезпечення належного проектування, монтажу, обслуговування і користування цими системами допомагає забезпечити безпеку та здоров'я мешканців будівель, знизити ризик виникнення аварій та забезпечити комфортні умови проживання

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		55

ВИСНОВКИ

У процесі виконання роботи було створено комп'ютеризовану систему автоматизованого керування камінним опаленням та розподілом теплого повітря в приватному будинку, яка забезпечує ефективну, стабільну та безпечну роботу каміна без постійного втручання користувача. На основі аналізу предметної області виявлено основні недоліки традиційних систем повітряного опалення, пов'язані з нестабільністю температурних режимів, нераціональним використанням палива, нерівномірністю обігріву приміщень і потенційною небезпекою для користувачів через можливе накопичення чадного газу та диму. Визначено вимоги до функціональних можливостей, технічних характеристик, апаратної конфігурації та програмної логіки майбутньої системи.

Розроблена система побудована на базі мікроконтролера Arduino Nano, що став центральною платформою для збору даних з датчиків, виконання алгоритмів керування та управління виконавчими пристроями. Для реалізації функцій моніторингу стану системи використано термопару MAX6675 для вимірювання температури димових газів, цифрові датчики температури DS18B20 для контролю температури у нагрівальній камері та в приміщеннях, а також газові сенсори MQ-2 та MQ-7 для контролю вмісту диму і чадного газу. Виконавча частина системи включає серво-привід для регулювання подачі повітря в камеру згоряння, блок керування заслінками для розподілу гарячого повітря між кімнатами, вентилятор циркуляції повітря, а також інформаційно-сигнальні елементи для зворотного зв'язку з користувачем.

У розробленому програмному забезпеченні реалізовано поетапну логіку роботи системи із переходами між режимами розпалу, контролю горіння, тління та завершення роботи. Основою роботи системи в режимі горіння є PID-регулятор, який дозволяє плавно регулювати положення повітряної заслінки відповідно до відхилення температури димових газів від заданої величини, забезпечуючи стабільність процесу згоряння. Режим розпалу

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

здійснюється при повністю відкритій повітряній заслінці до досягнення необхідної температури димових газів та нагрівальної камери. У режимі стабільного горіння система підтримує температуру димових газів у межах 300–500°C, регулює розподіл гарячого повітря у кімнати в залежності від поточної температури в приміщеннях. При переході в режим тління відбувається поступове зниження температури з відповідним коригуванням роботи заслінок і вентилятора. Завершення роботи передбачає відключення усіх виконавчих елементів після досягнення безпечних температурних порогів. Одночасно постійно працює аварійна система захисту, яка забезпечує негайне блокування роботи каміна при перевищенні критичних порогів концентрації чадного газу або диму.

Практична реалізація макету системи підтвердила правильність прийнятих технічних рішень, стабільність роботи алгоритмів, точність дотримання температурних режимів та коректність переходів між режимами. Система продемонструвала здатність ефективно адаптуватися до зміни умов роботи, своєчасно реагувати на позаштатні ситуації та забезпечувати стабільний комфортний мікроклімат у приміщеннях. Запропонована система має високий потенціал для подальшої модернізації, зокрема для розширення кількості зон опалення, впровадження віддаленого керування через мобільні застосунки, інтеграції в системи «розумного дому» та розширення функціональних можливостей для повної автоматизації процесу експлуатації камінних систем опалення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Жаровський Р.О., Луцик Н.С., Осухівська Г.М., Паламар А.М., Тиш Є.В. Методичні вказівки до виконання кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 39 с.
2. Буров Є.В., Митник М.М. Комп'ютерні мережі. Підручник. Том другий. Львів: «Магнолія 2006», 2024. 204 с.
3. Буров Є.В., Митник М.М. Комп'ютерні мережі. Підручник. Том перший. Львів: «Магнолія 2006», 2024. 333 с.
4. Лупенко С.А., Пасічник В.В., Тиш Є.В. Комп'ютерна логіка. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 354 с.
5. Луцків А., Лупенко С., Пасічник В. Паралельні та розподільнені обчислення. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 566 с.
6. Микитишин А.Г., Митник М.М., Стухляк П.Д., Пасічник В.В. Комп'ютерні мережі. Книга 1. Львів: «Магнолія 2006», 2024. 256 с.
7. Микитишин А.Г., Митник М.М., Стухляк П.Д., Пасічник В.В. Комп'ютерні мережі. Книга 2. Львів: «Магнолія 2006», 2024. 328 с.
8. Паламар М.І., Стрембіцький М.О., Паламар А.М. Проектування комп'ютеризованих вимірювальних систем і комплексів. Навчальний посібник. Тернопіль: ТНТУ, 2019. 150 с.
9. Kharchenko A., Bodnarchuk I., Yatcysyn V. The Method for Comparative Evaluation of Software Architecture with Accounting of Trade-offs. American Journal of Information Systems. 2014. Vol. 2, No. 1. P. 20-25.
10. Lupenko S., Lutsyk N., Yasniy O., Sobaszek Ł. Statistical analysis of human heart rhythm with increased informativeness. Acta mechanica et automatic. 2018. Vol. 12. P. 311-315.

					КС КРБ 123.227.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

11. Lupenko S., Orobchuk O., Stadnik N., Zozulya A. Modeling and signals processing using cyclic random functions. In Proceedings of the 13th IEEE International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT). Lviv, Ukraine, 11–14 September, 2018. P. 360–366.

12. Yatsyshyn V., Pastukh O., Palamar A., Zharovskyy R. Technology of relational database management systems performance evaluation during computer systems design. Scientific Journal of TNTU, Ternopil, Ukraine, 2023. Vol. 109, No 1. P. 54–65.

13. Yatsyshyn V., Pastukh O., Zharovskiy R., Shabliy N. Software tool for productivity metrics measure of relational database management system. Mathematical Modeling. No 1 (48). 2023. P. 7-17.

14. Shabliy N., Lupenko S., Lutsyk N., Yasniy O., Malyshevskaya O. Keystroke dynamics analysis using machine learning methods. Applied Computer Science. 2021. Vol. 17, No. 4. P. 75-83.

15. Савчин М. В. Безпека життєдіяльності: навч. посіб. Львів: Афіша, 2020. 176 с.

16. Мисливий В. І. Домедична допомога: навч. посіб. Тернопіль: ТДМУ, 2019. 144 с.

17. Жидецький В.Ц. Охорона праці користувачів комп'ютерів: підручник. Львів: Афіша, 2020. 176 с. management system. Mathematical Modeling. No 1 (48). 2025. P. 7-17.

Додаток А.
Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

“Затверджую”

Завідувач кафедри КС

_____ Осухівська Г.М.

“ ____ ” _____ 2025 р

КОМП'ЮТЕРИЗОВАНА СИСТЕМА КЕРУВАННЯ КАМІННИМ ОПАЛЕННЯМ
ТА РОЗПОДІЛОМ ТЕПЛОГО ПОВІТРЯ В ПРИВАТНОМУ БУДИНКУ

ТЕХНІЧНЕ ЗАВДАННЯ

на 8 листках

Вид робіт:

Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»

Керівник кваліфікаційної роботи

_____ к.т.н., доц. Жаровський Р.

« ____ » _____ 2025 р.

«ВИКОНАВЕЦЬ»

Студент групи СІ-42

_____ Мехін О. Ю.

« ____ » _____ 2025 р.

Тернопіль 2025

1 Загальні відомості

1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп'ютеризована система керування камінним опаленням та розподілом теплого повітря в приватному будинку».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.227.00.00

1.2 Виконавець

Студент групи СІ-42, факультету комп'ютерно-інформаційних систем і програмної інженерії, кафедри комп'ютерних систем та мереж, Тернопільського національного технічного університету імені Івана Пулюя, Мехін Олександр Юрійович .

1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету « 27 » січня 2025 року № 4/7–53

1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 27.01.2025 р.

Плановий термін завершення виконання кваліфікаційної роботи – 19.06.2025 р.

1.5 Порядок оформлення та пред'явлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу здійснюється у відповідності до чинних норм та правил ISO, ЕСКД, ЕСПД та ДСТУ.

Пред'явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи. Попередній захист кваліфікаційної роботи відбувається при готовності роботи на 90% , наявності пояснювальної записки та графічного матеріалу.

Пред'явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

2 Призначення і цілі створення системи

2.1 Призначення системи

Розроблена комп'ютеризована система призначена для:

- автоматизованого контролю процесу горіння в каміні;
- автоматичного регулювання розподілу теплого повітря між приміщеннями;
- забезпечення енергоефективної роботи камінної системи опалення;
- підвищення безпеки експлуатації шляхом контролю концентрації диму та чадного газу.

2.2 Мета створення системи

Метою кваліфікаційної роботи є розробка та практична реалізація комп'ютеризованої системи керування камінним опаленням та розподілом

теплого повітря в приватному будинку на базі мікроконтролера Arduino Nano.

2.3 Характеристика об'єкту

Розроблювана система призначена для автоматичного контролю температури процесу горіння, розподілу теплого повітря по кімнатах та забезпечення безпечної роботи каміна з використанням:

- датчиків температури (термопара К через MAX6675, цифрові DS18B20);
- газових сенсорів диму MQ-2 та чадного газу MQ-7;
- серво-приводів та реле для керування заслінками подачі повітря;
- вентилятора для покращення циркуляції повітря;
- LCD дисплея, енкодера та кнопок для інтерфейсу користувача;
- аварійної індикації (світлової та звукової).

3 Вимоги до системи

3.1 Вимоги до системи в цілому

Система повинна забезпечувати:

- стабільне автоматичне керування процесами розпалу, горіння, тління та завершення роботи каміна;
- PID-регулювання температури димових газів у межах 300–500°C;
- контроль температур у нагрівальній камері каміна та в приміщеннях;
- зональне керування заслінками подачі теплого повітря в кімнати;
- моніторинг газових сенсорів диму та CO для аварійного відключення системи у разі небезпеки.

3.1.1 Вимоги до структури та функціонування системи

Структура системи включає:

- Arduino Nano (ATmega328P);

- термопару K + MAX6675 (T1);
- датчики DS18B20 (T2, T3);
- MQ-2 (дим), MQ-7 (чадний газ);
- серво-привід K1 (аналогова заслінка);
- реле для KZ1 (зональні заслінки);
- вентилятор V1;
- інтерфейс користувача (LCD 16x2 I2C, енкодер, кнопки);
- система аварійної індикації (бузер, світлодіоди).

3.1.2 Вимоги до способів та засобів зв'язку між компонентами системи
Комунікації реалізуються через:

- SPI (MAX6675);
- OneWire (DS18B20);
- Аналогові входи (MQ-2, MQ-7);
- I2C (LCD дисплей);
- Цифрові GPIO для сервоприводів, реле, енкодера та індикації.

3.1.3 Вимоги до режимів функціонування системи

Передбачити режими роботи:

- Розпал (інтенсивна подача повітря, закриті заслінки в кімнати);
- Контроль горіння (PID-регулювання, відкриття заслінок при досягненні температури);
- Тління (падіння температури, сигнал про необхідність підкласти дрова);
- Завершення роботи (поступове вимкнення);
- Аварійний режим (перевищення диму або CO, повне блокування роботи системи).

3.1.4 Вимоги по діагностуванню системи

Система повинна відображати поточні параметри роботи та повідомляти користувача про наявність аварійних ситуацій на дисплеї та звуковим

сигналом.

3.1.5 Перспективи розвитку, проектування системи

Система може бути розширена шляхом:

- додавання нових зон опалення (додаткові кімнати);
- підключення до системи «розумного дому»;
- реалізації віддаленого моніторингу.

3.2 Показники призначення

Система повинна мати наступні характеристики:

- точність вимірювання температури не гірше $\pm 0,5^{\circ}\text{C}$ (DS18B20);
- швидкість реакції на зміну параметрів — не більше 1 с;
- надійність роботи протягом усього опалювального циклу;
- стійкість до перебоїв електроживлення (підтримка автономного живлення).

3.2.1 Вимоги до надійності

Система повинна працювати стабільно при:

- збоях електроживлення (відновлення роботи);
- помилках апаратної або програмної частини (автоматичне перезавантаження);
- забезпеченні гальванічної розв'язки силових і логічних ланцюгів.

3.3 Вимоги до безпеки

- Захист від ураження електричним струмом (заземлення);
- Захист від пожежі у випадку аварійної ситуації;
- Наявність аварійного ручного вимкнення.

3.3.1 Вимоги до експлуатації і технічного обслуговування

- Температура експлуатації: від +10°C до +35°C;
- Вологість: 30%–80%;
- Атмосферний тиск: 760±25 мм рт. ст.;
- Регулярне технічне обслуговування системи — не рідше 1 разу на рік.

3.4 Вимоги до захисту інформації від несанкціонованого доступу

Забезпечення захищеного доступу до налаштувань системи;

Блокування апаратного втручання в роботу контролера.

3.4.1 Вимоги по збереженню інформації при аваріях

Збереження основних налаштувань в EEPROM мікроконтролера при зникненні живлення.

3.4.2 Вимоги по стандартизації і уніфікації

Використання стандартних сертифікованих електронних компонентів;

Відповідність нормам ергономіки та безпеки побутового електрообладнання.

4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

- пояснювальної записки;
- графічного матеріалу:
- 1. Структурна схема
- 2. Схема електрична принципова
- 3. Блок-схема роботи
- 4. Результати роботи

*Примітка: У комплект документації можуть вноситися міни та доповнення в процесі розробки.

5 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи бакалавра

№ етапу	Назва етапу виконання кваліфікаційної роботи	Термін виконання
1	<i>Розробка технічного завдання</i>	<i>27.01 – 02.02</i>
2	<i>Аналіз технічного завдання</i>	<i>03.02-01.03</i>
3	<i>Аналіз вимог до системи</i>	<i>01.03-27.03</i>
4	<i>Проектування структури системи</i>	<i>27.03-01.05</i>
5	<i>Проектування, реалізація програмної і апаратної частини</i>	<i>01.05-15.05</i>
6	<i>Налаштування і тестування системи</i>	<i>15.05-01.06</i>
7	<i>Безпека життєдіяльності, основи охорони праці</i>	<i>01.06 – 05.06</i>
8	<i>Оформлення пояснювальної записки і графічного матеріалу</i>	<i>06.06 – 08.06</i>
9	<i>Перевірка на академічний плагіат, перевірка керівником та консультантами</i>	<i>9.06 – 15.06</i>
10	<i>Попередній захист кваліфікаційної роботи бакалавра</i>	<i>16.06 – 22.06</i>
11	<i>Захист кваліфікаційної роботи бакалавра</i>	<i>25.06</i>

6 Додаткові умови виконання кваліфікаційної роботи

Під час виконання кваліфікаційної роботи у дане технічне завдання можуть вноситися зміни та доповнення.

Додаток Б.

Код порграми

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h> // Бібліотека для роботи з LCD
дисплеєм по I2C
#include <EEPROM.h>           // Для збереження параметрів у
EEPROM
#include <OneWire.h>           // Для роботи з датчиками
температури DS18B20
#include <max6675.h>           // Для роботи з модулем термопари
MAX6675
#include <Servo.h>             // Для роботи із сервоприводом

// --- ЗМІННИ ---
const int MQ2_pin = A0;       // Вхід для датчика диму MQ-2
const int MQ7_pin = A1;       // Вхід для датчика СО MQ-7
const int V1pin = 9;          // Вихід для вентилятора
const int ButRUN = 6;          // Кнопка запуску системи
const int ButSTOP = 7;         // Кнопка зупинки
const int LampRUN = 8;         // Світлодіод "система активна"
const int LampSTOP = 9;        // Світлодіод "система зупинена"
const int KZ1pin = 2;          // Реле заслінки зони подачі
повітря
const int powerPin = A7;       // Контроль живлення
const int button = 14;         // Кнопка меню
const int pinA = 16;           // Енкодер А
const int pinB = 15;           // Енкодер В
const int thermoDO = 10;       // MAX6675 DO
const int thermoCS = 12;       // MAX6675 CS
const int thermoCLK = 11;      // MAX6675 CLK

// --- ЗМІННИ ДАТЧИКІВ ГАЗУ ---
int mq2_value = 0;
int mq7_value = 0;
const int MQ2_threshold = 400; // Попіг аварії по MQ-2
const int MQ7_threshold = 300; // Попіг аварії по MQ-7

// --- ЗМІННИ ТЕМПЕРАТУРНИХ ДАТЧИКІВ ---
OneWire temp1(0);
OneWire temp2(1);
byte addr1[8];
byte addr2[8];
float celsius[3]; // Массив температур: celsius[0]=T2,
celsius[1]=T3, celsius[2]=T1
int dsreset[2] = {0, 0};
unsigned long prevMillis_ds[3] = {0, 0, 0};

// --- PID-регулятор та параметри ---
Servo myservo;
unsigned int K1 = 0; // Позиція заслінки
```

```

float PIDOUT = 0;
float error, errSum, output, derror;
float Kp, Ki, T1sp, T3sp, dT3sp, T2rozpaleno, T1potyhlo, T2max,
dT2max;
unsigned int t2rozpaleno, t1potyhlo, powerout;
unsigned long rozpalenoMillis = 0;
unsigned long potyhloMillis = 0;
unsigned long poweroutMillis = 0;
int OUTMIN, OUTMAX, MANOUT;
bool KZ1MAN, KZ1AUTO;

// --- ІНТЕРФЕЙС LCD ---
LiquidCrystal_I2C lcd(0x27, 16, 2);
MAX6675 thermocouple(thermoCLK, thermoCS, thermoDO);

// --- СТАН СИСТЕМИ ---
int astep;
int avto = 0; // 0 - ручний режим, 1 - автомат

// --- МЕНЮ ТА ЕНКОДЕР ---
int encoderPosCount = 0;
int pinALast;
int aVal;
boolean bCW;
int menu = 0;
int menu_ = 0;
int button_old = 1;
unsigned long previousMillis = 0;
const long interval = 500;

// === ІНІЦІАЛІЗАЦІЯ СИСТЕМИ ===
void setup() {
    // Зчитуємо параметри з EEPROM
    EEPROM.get(50, Kp); EEPROM.get(55, T1sp); EEPROM.get(70,
T3sp);
    EEPROM.get(75, dT3sp); EEPROM.get(40, T2rozpaleno);
    EEPROM.get(45, T1potyhlo);
    EEPROM.get(21, Ki); EEPROM.get(31, OUTMIN); EEPROM.get(25,
OUTMAX);
    EEPROM.get(27, t2rozpaleno); EEPROM.get(29, t1potyhlo);
    EEPROM.get(33, powerout);
    EEPROM.get(60, T2max); EEPROM.get(65, dT2max); EEPROM.get(70,
derror);

    myservo.attach(5);
    MANOUT = OUTMIN; myservo.write(MANOUT);

    // Налаштування пінів
    pinMode(ButRUN, INPUT_PULLUP); pinMode(ButSTOP, INPUT_PULLUP);
    pinMode(KZ1pin, OUTPUT); pinMode(LampRUN, OUTPUT);
    pinMode(LampSTOP, OUTPUT);
    pinMode(V1pin, OUTPUT); pinMode(MQ2_pin, INPUT);
    pinMode(MQ7_pin, INPUT);

```

```

    pinMode(button, INPUT_PULLUP); pinMode(pinA, INPUT);
    pinMode(pinB, INPUT);

    lcd.begin(); lcd.backlight();
    lcd.setCursor(0, 0); lcd.print("Privit,"); lcd.setCursor(7,
1); lcd.print("gospodar!");

    temp1.search(addr1); temp2.search(addr2); delay(2000);
    pinALast = digitalRead(pinA);
}

// === ОСНОВНИЙ ЦИКЛ ===
void loop() {
    // Зчитуємо газові датчики
    mq2_value = analogRead(MQ2_pin);
    mq7_value = analogRead(MQ7_pin);

    // Контроль аварії по газах
    if ((mq2_value > MQ2_threshold) || (mq7_value >
MQ7_threshold)) {
        avto = 0; myservo.write(OUTMIN);
        digitalWrite(V1pin, LOW); digitalWrite(KZ1pin, LOW);
        lcd.setCursor(0, 0); lcd.print(" GAS ALARM!!! ");
        delay(1000);
    }

    // Зчитування температурних датчиків
    func_dallas(temp1, addr1, 0);
    func_dallas(temp2, addr2, 1);
    celsius[2] = thermocouple.readCelsius();

    // Обробка кнопок запуску та зупинки
    if (!digitalRead(ButRUN)) { avto = 1; astep = 100;
rozpalenoMillis = millis(); potyhloMillis = millis(); }
    if (!digitalRead(ButSTOP)) { avto = 0; astep = 0; MANOUT =
OUTMIN; }

    if (avto == 1) func_auto();
    func_outs();
    func_encoder();
    func_button();
    func_menu();
}

// === ВИХОДИ СИСТЕМИ ===
void func_outs() {
    digitalWrite(LampRUN, avto);
    digitalWrite(LampSTOP, !avto);
    digitalWrite(KZ1pin, KZ1MAN);
    if ((avto == 1) && (celsius[1] >= T3sp)) digitalWrite(V1pin,
HIGH);
    else digitalWrite(V1pin, LOW);
}

```

```

// === ЗЧИТУВАННЯ ДАТЧИКІВ DS18B20 ===
void func_dallas(OneWire ds, byte addr[8], int num) {
    byte present = 0; int temp;
    if (dsreset[num] == 0) {
        ds.reset(); ds.select(addr); ds.write(0x44); dsreset[num] =
1;
    }
    if (millis() - prevMillis_ds[num] >= 1000) {
        prevMillis_ds[num] = millis(); dsreset[num] = 0;
        present = ds.reset(); ds.select(addr); ds.write(0xBE);
        temp = (ds.read() | ds.read() << 8);
        celsius[num] = (present == 1) ? ((float)temp / 16.0) :
1000.00;
    }
}

// === ЛОГІКА АВТОМАТИКИ ===
void func_auto() {
    if (celsius[1] < 1000.0) {
        if (celsius[1] > T3sp) KZ1AUTO = 0;
        if (celsius[1] < T3sp - dT3sp) KZ1AUTO = 1;
    } else { KZ1AUTO = 1; }

    if (astep == 100) {
        PIDOUT = OUTMIN;
        if (celsius[0] > T2rozpaleno) {
            if (millis() - rozpalenoMillis > t2rozpaleno * 1000) {
astep = 200; potyhloMillis = millis(); }
            } else { rozpalenoMillis = millis(); }
        }

        if (astep == 200) {
            error = T1sp - celsius[2]; errSum += error;
            output = abs(Kp) * error + Ki * errSum / 100.0;
            if ((output > OUTMAX) || (output < OUTMIN)) errSum = ((K1 -
(abs(Kp) * error)) / Ki) * 100.0;
            PIDOUT = constrain(output, OUTMIN, OUTMAX);
            K1 = PIDOUT; myservo.write(K1);

            if (celsius[2] < T1potyhlo) {
                if (millis() - potyhloMillis > t1potyhlo * 1000) astep =
300;
            } else { potyhloMillis = millis(); }
            if (celsius[0] > T2max) astep = 400;
        }

        if (astep == 300) { avto = 0; MANOUT = OUTMIN; KZ1MAN = 0; }
        if (astep == 400) {
            PIDOUT = OUTMIN;
            if (celsius[0] < T2max - dT2max) { astep = 100;
rozpalenoMillis = millis(); }
        }
}

```

```

}

// === ЕНКОДЕР ===
void func_encoder() {
    aVal = digitalRead(pinA);
    if (aVal != pinALast) {
        if (digitalRead(pinB) != aVal) encoderPosCount++;
        else encoderPosCount--;
    }
    pinALast = aVal;
}

// === КНОПКА МЕНЮ ===
void func_button()
{
    if ((digitalRead(button)==LOW) && (button_old==1))
    {
        if ((menu == 1)and(button_old==1)) {
            switch (menu_) {
                case 0: menu = 100; encoderPosCount=MANOUT;
                    break;
                case 1: menu = 200; encoderPosCount=0;
                    break;
                case 2: menu = 300; encoderPosCount=0;
                    break;
                case 3: menu = 400; encoderPosCount=0;
                    break;
                case 4: menu = 0; encoderPosCount=0;
                    break;
            }
            button_old=0;
        }

        if ((menu == 100)and(button_old==1)) {
            menu = 1;

            encoderPosCount = 0;
            button_old=0;
        }

        if ((menu == 200)and(button_old==1)) {
            switch (menu_) {
                case 0: KZ1MAN = !KZ1MAN;
                    break;
                case 1: menu = 1; encoderPosCount=1;
                    break;
            }
            button_old=0;
        }

        if ((menu == 300)and(button_old==1)) {
            switch (menu_) {
                case 0: menu = 301; encoderPosCount = T1sp;

```

```

        break;
        case 1: menu = 302; encoderPosCount = T2rozpaleno;
        break;
        case 2: menu = 303; encoderPosCount = T1potyhlo;
        break;
        case 3: menu = 304; encoderPosCount = t2rozpaleno;
        break;
        case 4: menu = 305; encoderPosCount = t1potyhlo;
        break;
        case 5: menu = 306; encoderPosCount = T3sp;
        break;
        case 6: menu = 307; encoderPosCount = dT3sp;
        break;
        case 7: menu = 308; encoderPosCount = powerout;
        break;
        case 8: menu = 309; encoderPosCount = T2max;
        break;
        case 9: menu = 310; encoderPosCount = dT2max;
        break;
        case 10: menu = 1; encoderPosCount=2;
        break;
    }
    button_old=0;
}

if ((menu == 301)and(button_old==1)) {
    menu = 300;
    T1sp = encoderPosCount;
    EEPROM.put(55, T1sp);
    encoderPosCount = 0;
    button_old=0;
}

if ((menu == 302)and(button_old==1)) {
    menu = 300;
    T2rozpaleno = encoderPosCount;
    EEPROM.put(40, T2rozpaleno);
    encoderPosCount = 1;
    button_old=0;
}

if ((menu == 303)and(button_old==1)) {
    menu = 300;
    T1potyhlo = encoderPosCount;
    EEPROM.put(45, T1potyhlo);
    encoderPosCount = 2;
    button_old=0;
}

if ((menu == 304)and(button_old==1)) {
    menu = 300;
    t2rozpaleno = encoderPosCount;
    EEPROM.put(27, t2rozpaleno);

```

```

        encoderPosCount = 3;
        button_old=0;
    }

    if ((menu == 305)and(button_old==1)) {
        menu = 300;
        t1potyhlo = encoderPosCount;
        EEPROM.put(29, t1potyhlo);
        encoderPosCount = 4;
        button_old=0;
    }

    if ((menu == 306)and(button_old==1)) {
        menu = 300;
        T3sp = encoderPosCount;
        EEPROM.put(70, T3sp);
        encoderPosCount = 5;
        button_old=0;
    }

    if ((menu == 307)and(button_old==1)) {
        menu = 300;
        dT3sp = encoderPosCount;
        EEPROM.put(75, dT3sp);
        encoderPosCount = 6;
        button_old=0;
    }

    if ((menu == 308)and(button_old==1)) {
        menu = 300;
        powerout = encoderPosCount;
        EEPROM.put(33, powerout);
        encoderPosCount = 7;
        button_old=0;
    }

    if ((menu == 309)and(button_old==1)) {
        menu = 300;
        T2max = encoderPosCount;
        EEPROM.put(60, T2max);
        encoderPosCount = 8;
        button_old=0;
    }

    if ((menu == 310)and(button_old==1)) {
        menu = 300;
        dT2max = encoderPosCount;
        EEPROM.put(65, dT2max);
        encoderPosCount = 9;
        button_old=0;
    }

    if ((menu == 400)and(button_old==1)) {

```

```

switch (menu_) {
    case 0: menu = 401; encoderPosCount = Kp*10;
        break;
    case 1: menu = 402; encoderPosCount = Ki;
        break;
    case 2: menu = 403; encoderPosCount = OUTMIN;
        break;
    case 3: menu = 404; encoderPosCount = OUTMAX;
        break;
    case 4: menu = 405; encoderPosCount = derror;
        break;
    case 5: menu = 1; encoderPosCount=3;
        break;
}
button_old=0;
}

if ((menu == 401)and(button_old==1)) {
    menu = 400;
    Kp = encoderPosCount*0.1;
    EEPROM.put(50, Kp);
    encoderPosCount = 0;
    button_old=0;
}

if ((menu == 402)and(button_old==1)) {
    menu = 400;
    Ki = encoderPosCount;
    EEPROM.put(21, Ki);
    encoderPosCount = 1;
    button_old=0;
}

if ((menu == 403)and(button_old==1)) {
    menu = 400;
    OUTMIN = encoderPosCount;
    EEPROM.put(31, OUTMIN);
    encoderPosCount = 2;
    button_old=0;
}

if ((menu == 404)and(button_old==1)) {
    menu = 400;
    OUTMAX = encoderPosCount;
    EEPROM.put(25, OUTMAX);
    encoderPosCount = 3;
    button_old=0;
}

if ((menu == 405)and(button_old==1)) {
    menu = 400;
    derror = encoderPosCount;
    EEPROM.put(70, derror);
}

```

```

        encoderPosCount = 4;
        button_old=0;
    }

    if ((menu == 0)and(button_old==1)) {
        menu = 1;
        menu_ = 0;
        button_old=0;
        encoderPosCount=0;
    }

    menu_ = 0;
}
button_old = digitalRead(button);
}

// === МЕНЮ ВІДОБРАЖЕННЯ ===
void func_menu()
{
    int celsius2;
    if (currentMillis - previousMillis >= interval) {
        previousMillis = currentMillis;
        if (menu == 1){

            if (encoderPosCount>4) encoderPosCount = 0;
            if (encoderPosCount<0) encoderPosCount = 4;
            menu_ = encoderPosCount;
            lcd.setCursor(0, 0);
            lcd.print(">");
            switch (menu_) {
                case 0: lcd.print("K1                ");
                break;
                case 1: lcd.print("KZ1                ");
                break;
                case 2: lcd.print("Parametri          ");
                break;
                case 3: lcd.print("PID regulator      ");
                break;
                case 4: lcd.print("EXIT                ");
                break;
            }

            lcd.setCursor(0, 1);
            switch (menu_) {

                case 4: lcd.print("K1                ");
                break;
                case 0: lcd.print("KZ1                ");
                break;
                case 1: lcd.print("Parametri          ");
                break;
                case 2: lcd.print("PID regulator      ");

```

```

        break;
        case 3: lcd.print("EXIT          ");
        break;
    }
}

if (menu == 0){
    lcd.setCursor(0, 0);
    celsius2 = celsius[2];

    if (String(celsius[2],0)=="NAN"){lcd.print(" ERR ");} else
{lcd.print(celsius2);}

    lcd.print("\1C ");
    lcd.setCursor(6, 0);
    if (!power) lcd.print("AV "); else lcd.print("av ");
    if (KZ1) lcd.print("KZ"); else lcd.print("kz");

    lcd.print(" ");
    lcd.print(K1); lcd.print("\1 ");
    lcd.setCursor(0, 1);

    //lcd.print("er"); lcd.print(error); lcd.print("ers");
    lcd.print(errSum);lcd.print(" ");

    celsius2 = celsius[0];
    if (celsius[0]<1000.0) {lcd.print(celsius2);lcd.print("\1C
");} else {lcd.print("ERROR ");}
    switch (astep) {
        case 100: lcd.print("#");
        break;
        case 200: lcd.write(byte(178));
        break;
        case 400: lcd.write(byte(64));
        break;
        default: lcd.print("_");
        break;
    }
    lcd.print(" ");
    celsius2 = celsius[1];
    if (celsius[1]<1000.0)
{lcd.print(celsius2);lcd.print("\1C");} else lcd.print("ERROR");
    lcd.print(" ");
}

if (menu == 100){
    if (encoderPosCount>180) encoderPosCount = 180;
    if (encoderPosCount<0) encoderPosCount = 0;
    MANOUT = encoderPosCount;
    lcd.setCursor(0, 0);
    lcd.print("Otkritie K1 ");
    lcd.setCursor(0, 1);

```

```

        lcd.print(encoderPosCount); lcd.print(" grad
");
    }

    if (menu == 200){
        if (encoderPosCount>1) encoderPosCount = 0;
        if (encoderPosCount<0) encoderPosCount = 1;
        menu_ = encoderPosCount;
        lcd.setCursor(0, 0);
        lcd.print(">");
        switch (menu_) {
            case 0: if (KZ1) lcd.print("KZ1 OTKRITO      "); else
lcd.print("KZ1 ZAKRITO      ");
                break;
            case 1: lcd.print("EXIT                ");
                break;
        }
        lcd.setCursor(0, 1);
        switch (menu_) {
            case 1: if (KZ1) lcd.print("KZ1 OTKRITO      "); else
lcd.print("KZ1 ZAKRITO      ");
                break;
            case 0: lcd.print("EXIT                ");
                break;
        }
    }

    if (menu == 300){
        if (encoderPosCount>10) encoderPosCount = 0;
        if (encoderPosCount<0) encoderPosCount = 10;
        menu_ = encoderPosCount;
        lcd.setCursor(0, 0);
        lcd.print(">");
        switch (menu_) {
            case 0: lcd.print("T1sp          DG  ");
                break;
            case 1: lcd.print("T2rozpaleno SO  ");
                break;
            case 2: lcd.print("T1potyhlo   DG  ");
                break;
            case 3: lcd.print("t2rozpaleno      ");
                break;
            case 4: lcd.print("t1potyhlo      ");
                break;
            case 5: lcd.print("T3sp          GVS ");
                break;
            case 6: lcd.print("dT3sp         GVS ");
                break;
            case 7: lcd.print("t3 nema 220V    ");
                break;
            case 8: lcd.print("T2max          SO  ");
                break;
            case 9: lcd.print("dT2max         SO  ");

```

```

        break;
        case 10:lcd.print("EXIT          ");
        break;
    }

    lcd.setCursor(0, 1);
    switch (menu_) {
        case 10:lcd.print("T1sp          DG  ");
        break;
        case 0: lcd.print("T2rozpaleno SO  ");
        break;
        case 1: lcd.print("T1potyhlo   DG  ");
        break;
        case 2: lcd.print("t2rozpaleno        ");
        break;
        case 3: lcd.print("t1potyhlo        ");
        break;
        case 4: lcd.print("T3sp          GVS ");
        break;
        case 5: lcd.print("dT3sp          GVS ");
        break;
        case 6: lcd.print("t3 nema 220V      ");
        break;
        case 7: lcd.print("T2max          SO  ");
        break;
        case 8: lcd.print("dT2max          SO  ");
        break;
        case 9: lcd.print("EXIT          ");
        break;
    }
}

if (menu == 301){
    if (encoderPosCount>600) encoderPosCount = 600;
    if (encoderPosCount<0) encoderPosCount = 0;
    lcd.setCursor(0, 0);
    lcd.print("Zadana T1sp DG  ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount); lcd.print(" \1C          ");
}

if (menu == 302){
    if (encoderPosCount>100) encoderPosCount = 100;
    if (encoderPosCount<0) encoderPosCount = 0;
    lcd.setCursor(0, 0);
    lcd.print("T2 SO rozpalen  ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount); lcd.print(" \1C          ");
}

if (menu == 303){
    if (encoderPosCount>600) encoderPosCount = 600;
    if (encoderPosCount<0) encoderPosCount = 0;

```

```

        lcd.setCursor(0, 0);
        lcd.print("T1 DG potyhlo  ");
        lcd.setCursor(0, 1);
        lcd.print(encoderPosCount); lcd.print(" \1C          ");
    }

    if (menu == 304){
        if (encoderPosCount>60000) encoderPosCount = 60000;
        if (encoderPosCount<1) encoderPosCount = 1;
        lcd.setCursor(0, 0);
        lcd.print("Chas rozpaleno  ");
        lcd.setCursor(0, 1);
        lcd.print(encoderPosCount); lcd.print(" cek
    ");
    }

    if (menu == 305){
        if (encoderPosCount>60000) encoderPosCount = 60000;
        if (encoderPosCount<1) encoderPosCount = 1;
        lcd.setCursor(0, 0);
        lcd.print("Chas potyhlo  ");
        lcd.setCursor(0, 1);
        lcd.print(encoderPosCount); lcd.print(" cek
    ");
    }

    if (menu == 306){
        if (encoderPosCount>100) encoderPosCount = 100;
        if (encoderPosCount<0) encoderPosCount = 0;
        lcd.setCursor(0, 0);
        lcd.print("Zadana T3sp GVS ");
        lcd.setCursor(0, 1);
        lcd.print(encoderPosCount); lcd.print(" \1C          ");
    }

    if (menu == 307){
        if (encoderPosCount>30) encoderPosCount = 30;
        if (encoderPosCount<0) encoderPosCount = 0;
        lcd.setCursor(0, 0);
        lcd.print("Delta dT3sp GVS ");
        lcd.setCursor(0, 1);
        lcd.print(encoderPosCount); lcd.print(" \1C          ");
    }

    if (menu == 308){
        if (encoderPosCount>60000) encoderPosCount = 60000;
        if (encoderPosCount<1) encoderPosCount = 1;
        lcd.setCursor(0, 0);
        lcd.print("Chas koly 'ac'");
        lcd.setCursor(0, 1);
        lcd.print(encoderPosCount); lcd.print(" cek
    ");
    }

```

```

if (menu == 309){
    if (encoderPosCount>100) encoderPosCount = 100;
    if (encoderPosCount<0) encoderPosCount = 0;
    lcd.setCursor(0, 0);
    lcd.print("Zadana T2max SO ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount); lcd.print(" \1C          ");
}

if (menu == 310){
    if (encoderPosCount>100) encoderPosCount = 100;
    if (encoderPosCount<0) encoderPosCount = 0;
    lcd.setCursor(0, 0);
    lcd.print("Delta dT2max SO ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount); lcd.print(" \1C          ");
}

if (menu == 400){
    if (encoderPosCount>5) encoderPosCount = 0;
    if (encoderPosCount<0) encoderPosCount = 5;
    menu_ = encoderPosCount;
    lcd.setCursor(0, 0);
    lcd.print(">");
    switch (menu_) {
        case 0: lcd.print("Kp          ");
        break;
        case 1: lcd.print("Ki          ");
        break;
        case 2: lcd.print("OUTMIN      ");
        break;
        case 3: lcd.print("OUTMAX      ");
        break;
        case 4: lcd.print("dBAND       ");
        break;
        case 5: lcd.print("EXIT        ");
        break;
    }

    lcd.setCursor(0, 1);
    switch (menu_) {
        case 5: lcd.print("Kp          ");
        break;
        case 0: lcd.print("Ki          ");
        break;
        case 1: lcd.print("OUTMIN      ");
        break;
        case 2: lcd.print("OUTMAX      ");
        break;
        case 3: lcd.print("dBAND       ");
        break;
        case 4: lcd.print("EXIT        ");

```

```

        break;
    }
}

if (menu == 401){
    if (encoderPosCount>1000) encoderPosCount = 1000;
    if (encoderPosCount<-1000) encoderPosCount = -1000;
    lcd.setCursor(0, 0);
    lcd.print("Kp -100...100  ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount*0.1); lcd.print("
");
}

if (menu == 402){
    if (encoderPosCount>10000) encoderPosCount = 10000;
    if (encoderPosCount<0) encoderPosCount = 0;
    lcd.setCursor(0, 0);
    lcd.print("Ki 0...10000  ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount); lcd.print("
");
}

if (menu == 403){
    if (encoderPosCount>180) encoderPosCount = 180;
    if (encoderPosCount<0) encoderPosCount = 0;
    lcd.setCursor(0, 0);
    lcd.print("OUTMIN 0...180  ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount); lcd.print(" grad
");
}

if (menu == 404){
    if (encoderPosCount>180) encoderPosCount = 180;
    if (encoderPosCount<0) encoderPosCount = 0;
    lcd.setCursor(0, 0);
    lcd.print("OUTMAX 0...180 ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount); lcd.print(" grad
");
}

if (menu == 405){
    if (encoderPosCount>200) encoderPosCount = 200;
    if (encoderPosCount<0) encoderPosCount = 0;
    lcd.setCursor(0, 0);
    lcd.print("dBAND 0...200 ");
    lcd.setCursor(0, 1);
    lcd.print(encoderPosCount); lcd.print(" \1C
");
}
}
}

```

```

void func_pisk()
{
    if ((pisk_once) || (astep==300)){
        currentMillis = millis();
        if (currentMillis - piskwaitMillis > piskwait) {
            piskwaitMillis = currentMillis;
            if (!pisk1){
                pisk1 = 1;
                digitalWrite(piskpower, HIGH);
                tone (p, 1000); //пищим на частоте 1000 Гц}
                piskwait = 1000;
            }else{
                pisk1 = 0;
                noTone(p); //выключаем звук
                digitalWrite(piskpower, LOW);
                piskwait = piskset;
            }
        }
        currentMillis = millis();
        if (currentMillis - piskMillis > 3600000) {
            pisk_once = 0;
            if (astep==300) astep = 0;
            noTone(p); //выключаем звук
            digitalWrite(piskpower, LOW);}
        }else {piskMillis = currentMillis; piskwait = 1000;
        piskwaitMillis = currentMillis; noTone(p);}
    }
}

```

Додаток В.
Перелік елементів

