

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерної інженерії та управління _____
Кафедра _____ Автоматизації проєктування обчислювальної техніки _____
Рівень вищої освіти _____ перший (бакалаврський) _____
Спеціальність _____ 123 Комп'ютерна інженерія _____
Тип програми _____ Освітньо-професійна _____
Освітня програма _____ Комп'ютерна інженерія _____

ЗАТВЕРДЖУЮ:
Зав. _____ кафедри _____

_____ (підпис)
«____» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Антипенку Іллі Андрійовичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Система клімат контролю у приміщенні на базі Arduino Uno _____

затверджена наказом університету від _21_ _____ 05 _____ 2025 р. № 403Ст

2. Термін подання студентом роботи до екзаменаційної комісії _10_ ____ 06 ____ 2025 р.

3. Вихідні дані до роботи _____

Arduino

Програмно-апаратний модуль

4. Перелік питань, що потрібно опрацювати в роботі _____

Аналіз та огляд існуючих систем.

Постановка задачі.

Розробка структурної схеми пристрою.

Розробка функціональної схеми програми.

Розробка алгоритму роботи пристрою.


Тестування


5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) _____
17 слайдів

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Видача теми проєкту, узгодження і затвердження теми	06.05.2025 – 09.05.2025	
2	Аналіз проблемної галузі, постановка задачі, вибір інструментальних засобів	09.05.2025 – 14.05.2025	
3	Розробка структурної схеми пристрою, вибір апаратної платформи	14.05.2025 – 16.05.2025	
4	Розробка функціональної схеми програми	16.05.2025 – 17.05.2025	
5	Розробка програмних модулів. Проведення тестування	17.05.2025 – 23.05.2025	
6	Оформлення пояснювальної записки	23.05.2025 – 25.05.2025	
7	Перевірка виконаного проєкту керівником, допуск до захисту	25.05.2025 – 10.06.2025	
8	Захист проєкту	12.06.2025 – 24.06.2025	

Дата видачі завдання 06.05.2025 р.

Студент 
(підпис)

Керівник роботи  проф. Хаханова Г.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи містить 47 сторінок, 14 рисунків, 11 джерел посилання.

ARDUINO, LCD, I2C, ДАТЧИК, ТЕМПЕРАТУРА, ВОЛОГІСТЬ, ТИСК, ГАЗ, РЕЛЕ, EEPROM.

Метою кваліфікаційної роботи є розробка доступної та надійної системи клімат-контролю для житлових і робочих приміщень, яка на базі мікроконтролера Arduino UNO поєднує датчики температури, вологості, атмосферного тиску та горючих газів, відображає дані на LCD-дисплеї, дозволяє користувачеві налаштовувати поріг увімкнення вентилятора за допомогою кнопок та зберігає вибрані параметри в EEPROM і своєчасно попереджає про небезпечні відхилення за допомогою світлодіодів і звукового сигналу. У межах роботи проаналізовано сучасні системи клімат-контролю, сформульовано технічне завдання, підібрано апаратну базу, розроблено програмний код мовою C++ у середовищі Arduino IDE, зібрано та протестовано макет, перевірено точність вимірювань, працездатність індикації й стабільність керування вентилятором, а також визначено напрямки подальшого вдосконалення системи у майбутньому.

ABSTRACT

The explanatory note of the qualification work contains 47 pages, 14 figures, 11 references.

ARDUINO, LCD, I2C, SENSOR, TEMPERATURE, HUMIDITY, PRESSURE, GAS, RELAY, EEPROM.

The purpose of the qualification work is to develop an affordable and reliable climate control system for residential and commercial premises, which, based on the Arduino UNO microcontroller, combines temperature, humidity, atmospheric pressure and combustible gas sensors, displays data on the LCD display, allows the user to adjust the fan threshold using the buttons and saves the selected parameters in EEPROM and timely warns of dangerous deviations using LEDs and an audible signal. As part of the work, we analyzed modern climate control systems, formulated the technical task, selected the hardware base, developed the C++ program code in the Arduino IDE, assembled and tested the layout, checked the measurement accuracy, display performance, and stability of the fan control, and identified areas for further improvement of the system in the future.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ.....	8
ВСТУП	9
1 СИСТЕМИ КЕРУВАННЯ КЛІМАТОМ В ПРИМІЩЕННІ.....	10
1.1 Застосування систем керування кліматом	10
1.2 Історичний огляд та еволюція систем клімат-контролю.....	12
1.3 Класифікація систем керування кліматом	13
1.4 Постановка завдання.....	16
2 СТРУКТУРНО-ФУНКЦІОНАЛЬНА МОДЕЛЬ СИСТЕМИ КЛІМАТ КОНТРОЛЮ.....	17
2.1 Загальний опис та призначення системи	17
2.2 Компоненти системи клімат-контролю	17
2.2.1 Керуючий мікроконтролерний пристрій Arduino	18
2.2.2 Сенсори кліматичних параметрів	20
2.2.3 LCD дисплей	22
2.2.4 Управління системою охолодження.....	23
2.2.5 Датчик газу.....	24
3 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ СИСТЕМИ КЛІМАТ КОНТРОЛЮ	26
3.1 Загальна структура програмного забезпечення та конфігурація пінів мікроконтролера	26
3.2 Опис використаних бібліотек.....	28
3.3 Розробка алгоритмів роботи основних модулів програми.....	30
3.3.1 Алгоритм зчитування даних з сенсорів.....	31
3.3.2 Алгоритм обробки даних та прийняття рішень	32
3.3.3 Обробка натискань кнопок.....	34
3.3.4 Алгоритм відображення інформації на LCD-дисплеї.....	36

4 ТЕСТУВАННЯ СИСТЕМИ	38
ВИСНОВКИ.....	45
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	46
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	48
ДОДАТОК Б Лістинг коду	57

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

BME280 – комбінований сенсор температури, вологості та тиску виробництва Bosch Sensortec.

DHT11 – датчик температури та вологості.

MQ-2 – датчик газу для вимірювання концентрації різних горючих газів, таких як метан, пропан.

LCD – рідкокристалічний дисплей, що використовується для відображення інформації.

I2C – інтерфейс для зв'язку між пристроями через дві лінії передачі даних (SDA та SCL).

VCC – напруга живлення пристрою.

GND – заземлення.

VDD – основне живлення датчика.

ppm – частин на мільйон, одиниця вимірювання концентрації газу.

Arduino – мікроконтролерна плата для прототипування електронних пристроїв.

SDA – лінія передачі даних в інтерфейсі I2C.

SCL – лінія годинникового сигналу в інтерфейсі I2C.

ШИМ – широтно-імпульсна модуляція, метод управління потужністю в електронних пристроях.

гПа – гектопаскалі, одиниця вимірювання атмосферного тиску.

EEPROM – енергонезалежна пам'ять Arduino, у якій зберігаються дані навіть після вимкнення живлення.

ВСТУП

Розвиток систем клімат контролю є одним із ключових напрямків у сучасній автоматизації, який дозволяє створювати комфортні умови для життя, роботи та виробничих процесів. Забезпечення оптимальних показників температури, вологості та освітленості є важливим аспектом для підвищення якості життя, збереження здоров'я та енергетичної ефективності.

Завдяки появі мікроконтролерів, таких як Arduino, створення інтелектуальних систем клімат контролю стало доступним навіть для початківців у сфері електроніки та програмування. Ці платформи дозволяють інтегрувати широкий спектр сенсорів та пристроїв, які здатні аналізувати умови навколишнього середовища та автоматично адаптувати параметри клімату в реальному часі.

Даний проєкт має на меті створення інтерактивної системи клімат-контролю, яка забезпечує автоматизоване вимірювання, моніторинг та регулювання основних параметрів мікроклімату: температури, вологості, атмосферного тиску та рівня забрудненості повітря. Особливістю системи є можливість відображення даних на OLED-дисплеї, інтеграція RGB-світлодіодів для візуалізації стану середовища, а також використання вентилятора як аналог системи кондиціонування повітря.

Розробка системи передбачає використання мови програмування C++ та апаратної платформи Arduino. Реалізація даного проєкту сприятиме вирішенню практичної задачі та поглибленню знань у галузі автоматизації.

1 СИСТЕМИ КЕРУВАННЯ КЛІМАТОМ В ПРИМІЩЕННІ

1.1 Застосування систем керування кліматом

У сучасному світі, де більшість нашого життя проходить у приміщеннях – чи то затишна оселя, динамічний офіс, навчальна аудиторія або виробничий цех – якість внутрішнього середовища відіграє ключову роль у формуванні нашого здоров'я, комфорту та загальної продуктивності. Саме тому увага до створення та підтримки оптимального мікроклімату невпинно зростає, перетворюючись із простого бажання на важливий елемент забезпечення високого рівня життя. Адже саме збалансовані умови сприяють не лише фізичному, але й психологічному благополуччю, дозволяючи нам почуватися енергійними та здатними до концентрації.

Проте, створення та підтримка ідеального мікроклімату – завдання не з легких. Сучасні будівлі часто стикаються з проблемами дисбалансу температури та вологості, що безпосередньо впливає на якість повітря. Взимку опалювальні системи нерідко призводять до надмірної сухості повітря, провокуючи дискомфорт, сухість шкіри, проблеми з дихальними шляхами. Влітку ж надмірна спека та вологість можуть створити задушливу атмосферу, суттєво знижуючи працездатність та загальне самопочуття. До цих факторів додається невидима, але не менш підступна загроза – забруднення повітря шкідливими або потенційно небезпечними газами. Такі речовини, як чадний газ від несправних опалювальних приладів, підвищені концентрації вуглекислого газу в погано провітрюваних приміщеннях, або витоки побутового метану, можуть спричинити погіршення самопочуття, головний біль, алергічні реакції, загострення хронічних респіраторних захворювань, а у високих концентраціях – призвести до серйозних отруєнь чи навіть вибухонебезпечних ситуацій. Варто зазначити, що багато з цих негативних

чинників можуть діяти приховано, поступово накопичуючи свій згубний вплив, перш ніж проблема стане очевидною.

Саме тому системи моніторингу мікроклімату стають незамінними інструментами в боротьбі за здорове та безпечне середовище. Ці технологічні рішення призначені для безперервного або періодичного збору точних даних про ключові параметри повітря – температуру, відносну вологість, атмосферний тиск, а також концентрації різноманітних газів. Зібрана інформація аналізується, дозволяючи не лише фіксувати поточний стан, але й прогнозувати потреби в регулюванні мікроклімату та своєчасно інформувати користувачів про можливі загрози.

Такий ретельний моніторинг є особливо критичним для приміщень з підвищеними вимогами до умов перебування: сучасних офісів, де важлива максимальна концентрація працівників; медичних установ та лікарень, де чистота та стабільність повітря є запорукою одужання пацієнтів; бібліотек та архівів, що потребують специфічних умов для збереження цінностей; і, звичайно, для домівок, де проживають люди з хронічними захворюваннями дихальної системи, алергіями чи астмою.

Впровадження інтелектуальних систем моніторингу відкриває шлях до оперативного реагування на будь-які зміни. Вони дозволяють не просто пасивно спостерігати, а й активно втручатися: наприклад, автоматично керувати системами вентиляції при підвищенні рівня вуглекислого газу, оптимізувати роботу опалення чи кондиціонування. Наприклад, інтелектуальні термостати та системи вентиляції, керовані датчиками присутності або концентрації CO₂, активуються лише за реальної потреби, що усуває марні витрати енергії на обігрів, охолодження чи вентиляцію порожніх або вже оптимально провітрених приміщень. Це не тільки підвищує рівень здоров'я, безпеки та загального благополуччя людей, створюючи умови для ефективної роботи та повноцінного відпочинку, але й сприяє раціональному використанню енергоресурсів. Таким чином, сучасні системи клімат-контролю та моніторингу є не просто технологічними пристроями, а

надійними партнерами у створенні справді здорового, комфортного та безпечного простору для життя і діяльності.

1.2 Історичний огляд та еволюція систем клімат-контролю

Розуміння історії розвитку систем клімат-контролю допомагає оцінити сучасні технології та спрогнозувати напрями їх удосконалення. Ідея створення комфортних умов існування сягає глибокої давнини, задовго до появи електрики чи складних механічних пристроїв.

Ранні цивілізації застосовували природні принципи для регулювання клімату. У Стародавньому Єгипті, наприклад, для охолодження використовували змочений очерет у віконних отворах – це сприяло природному випаровуванню вологи й охолодженню повітря. У Римі поширена була система опалення гіпокауст, що циркулювала гаряче повітря під підлогою та стінами. У Китаї винайшли вентилятори для циркуляції повітря, а на Близькому Сході застосовували ловці вітру – вежі, що спрямовували прохолодне повітря у приміщення. Ці методи були енергоефективними та стали основою сучасних принципів зеленого будівництва[10].

Зародження сучасних систем опалення, вентиляції і кондиціонування (HVAC) припало на Промислову революцію. У XVIII-XIX століттях почали з'являтися перші механічні пристрої охолодження та опалення. У 1902 році Вілліс Керріер розробив першу систему кондиціонування, що контролювала температуру й вологість, спочатку для друкарень. Паралельно розвивалось централізоване опалення із застосуванням радіаторів і повітряних систем.

Після Другої світової війни HVAC-системи стали масовими у побуті. Енергетичні кризи 1970-х стимулювали розробку енергоефективних рішень, зокрема програмованих термостатів. Різні регіони світу адаптували технології під свої кліматичні умови: в Японії популярні інверторні спліт-системи, у Китаї – масове виробництво кондиціонерів, в Європі – пріоритет на ефективне опалення, а в Африці – енергонезалежні і пасивні рішення[10].

У кінці XX – на початку XXI століття системи HVAC перейшли до цифрової ери: з'явилися "розумні" термостати, дистанційне керування та інтелектуальні алгоритми. Ключову роль у цій трансформації відіграв Інтернет речей (IoT), який дозволив об'єднати різноманітні датчики та виконавчі пристрої в єдину мережу для комплексної взаємодії. Це дозволяє підвищувати комфорт, знижувати енергоспоживання і реагувати на зміни навколишнього середовища максимально ефективно. Розвиток цифрових технологій і поширення мікроконтролерів зробили можливим створення доступних, гнучких та інтелектуальних систем клімат-контролю, які широко застосовуються як у промисловості, так і в побуті. Ці системи дозволяють не тільки автоматизувати процеси, але й збирати великі обсяги даних для подальшого аналізу та оптимізації [10]. Сучасні рішення часто інтегруються в єдину екосистему "розумного будинку", взаємодіючи із системами освітлення та безпеки. Таким чином, еволюція перетворила клімат-контроль із простих механічних приладів на складні кіберфізичні системи, що використовують самонавчання для автономної оптимізації мікроклімату.

1.3 Класифікація систем керування кліматом

Системи керування кліматом представляють собою комплексні інженерні рішення, які забезпечують підтримання комфортних та безпечних умов мікроклімату у приміщеннях різного призначення. Враховуючи різноманітність таких систем, виникає потреба у їх класифікації за низкою критеріїв, що сприяє більш ефективному проектуванню, вибору, монтажу і експлуатації. Наявність уніфікованої системи класифікації полегшує взаємодію між фахівцями різних профілів і дозволяє проводити порівняльний аналіз ефективності систем у різних умовах застосування.

Першим важливим критерієм класифікації є функціональне призначення систем. Серед них виділяють системи опалення, які відповідають за компенсацію тепловтрат і підтримання заданої температури в холодний період

року. Вони розрізняються за видом теплоносія, наприклад, водяні, парові, повітряні чи електричні, а також за типом нагрівальних пристроїв. Наступною категорією є системи вентиляції, що забезпечують необхідний повітрообмін у приміщеннях, подаючи свіже повітря і видаляючи забруднене. Вони можуть бути припливними, витяжними або припливно-витяжними, часто обладнаними рекуператорами для підвищення енергоефективності. Системи кондиціонування повітря виконують завдання автоматичного підтримання заданих параметрів температури, вологості та чистоти повітря незалежно від зовнішніх умов. Сучасні кондиціонери, особливо інверторні, часто поєднують функції охолодження й опалення, стираючи межі між традиційними системами. Крім того, існують інтегровані системи ОВКП (опалення, вентиляція, кондиціонування повітря), які забезпечують комплексний контроль параметрів мікроклімату, а також системи, що спеціалізуються на підтриманні окремих характеристик, таких як рівень вологості, концентрація вуглекислого газу чи очищення повітря.

Другим важливим критерієм є масштаб застосування та зона обслуговування системи. Локальні системи призначені для одного або декількох невеликих приміщень і включають кімнатні кондиціонери, локальні нагрівачі та вентилятори. Централізовані системи розраховані на обслуговування великої кількості приміщень або цілої будівлі, з одним або кількома центральними агрегатами. Вони можуть одночасно підтримувати однакові параметри у всьому обсязі (однозональні системи) або забезпечувати індивідуальне регулювання клімату в різних зонах (багатозональні системи). Багатозональні системи є більш гнучкими і дозволяють підвищувати комфорт і енергоефективність, адже враховують різні умови й потреби окремих приміщень.

Класифікація за способом керування визначає рівень участі користувача у регулюванні кліматичних параметрів. Ручні системи потребують постійного втручання для зміни налаштувань і не підтримують автоматичного регулювання. Автоматизовані системи оснащені датчиками та виконавчими

пристроями, що дозволяє підтримувати задані параметри без постійної участі користувача. Програмовані системи мають можливість задавати часові режими роботи, що забезпечує оптимізацію споживання енергії і підвищення комфорту. Найсучасніші інтелектуальні системи володіють здатністю до самонавчання, адаптуються до зовнішніх і внутрішніх умов, можуть взаємодіяти з іншими інженерними системами будівлі в рамках BMS та керуються дистанційно через мобільні пристрої.

Ще одним важливим аспектом класифікації є конструктивне виконання систем. Моноблочні системи містять усі основні компоненти в одному корпусі і зазвичай застосовуються в невеликих приміщеннях, наприклад, віконні або мобільні кондиціонери. Роздільні системи (спліт, мультиспліт) мають окремі зовнішні та внутрішні блоки, що підключаються між собою трубопроводами, і можуть обслуговувати кілька приміщень одночасно. Канальні системи відрізняються прихованим монтажем внутрішнього блоку і розподілом повітря через мережу повітроводів, що забезпечує рівномірне кондиціонування без видимих елементів обладнання. Безканальні системи не використовують повітроводи, поширені серед спліт-систем і мобільних кондиціонерів.

Класифікація за типом використовуваного енергоносія або принципом дії відображає основне джерело енергії системи та фізичний принцип роботи. Електричні системи найчастіше застосовуються в сучасних кондиціонерах і обігрівачах, парокомпресійні холодильні машини є основою більшості з них. Опалювальні системи можуть працювати на викопному паливі (газ, дизель, тверде паливо) або використовувати відновлювані джерела енергії, такі як теплові насоси і сонячні колектори. Останні набувають все більшого поширення завдяки екологічності та енергоефективності. Принцип роботи холодильної машини може бути парокомпресійним або абсорбційним, залежно від джерела енергії та конструктивних особливостей.

Важливою складовою класифікації є відповідність систем нормативним вимогам і стандартам. Відповідно до будівельних норм, міжнародних

стандартів ISO та ASHRAE, системи оцінюються за класом забезпечення метеорологічних умов, енергоефективністю, рівнем фільтрації повітря, безпекою і якістю повітря в приміщеннях [11]. Дотримання цих стандартів є ключовим фактором для забезпечення надійності, комфорту та безпеки користувачів, а також для сприяння сталому розвитку і впровадженню новітніх технологій у сфері клімат-контролю.

1.4 Постановка завдання

Для виконання поставлених завдань розроблювана система моніторингу мікроклімату та детекції газів повинна забезпечувати наступні основні функції:

- збір даних з датчиків температури, вологості, атмосферного тиску та концентрації газу;
- виведення поточних параметрів мікроклімату, рівня газу та системних налаштувань на рідкокристалічний дисплей;
- налаштування користувачем робочих параметрів системи, зокрема температурного порогу для керування реле, зі збереженням цих налаштувань в енергонезалежній пам'яті (EEPROM);
- автоматичне керування виконавчим пристроєм (через реле) на основі аналізу температурних показників;
- сигналізування про різні рівні концентрації газу за допомогою світлодіодної індикації та звукового сповіщення при критичних значеннях.

2 СТРУКТУРНО-ФУНКЦІОНАЛЬНА МОДЕЛЬ СИСТЕМИ КЛІМАТ КОНТРОЛЮ

2.1 Загальний опис та призначення системи

Система кліматичного контролю є комплексним рішенням для автоматичного моніторингу та регулювання мікроклімату в закритих приміщеннях. Її основне призначення – забезпечення комфортних умов для людей та оптимальних параметрів середовища для різних видів діяльності чи технологічних процесів. Система здатна автоматично вимірювати та контролювати такі важливі показники, як температура, відносна вологість і атмосферний тиск, а також, за потреби, вживати заходів для коригування цих параметрів через інтегровані виконавчі пристрої.

Завдяки можливості дистанційного моніторингу та управління, система може бути використана як у побутових, так і в промислових умовах. Вона забезпечує збори даних у реальному часі, їх обробку та подальшу передачу на зовнішні пристрої для аналізу або для використання в автоматизованих процесах. Це дозволяє ефективно реагувати на зміни мікроклімату і підтримувати задані умови без постійної участі користувача, що значно підвищує зручність і ефективність експлуатації приміщень.

2.2 Компоненти системи клімат-контролю

Для розробки системи кліматичного контролю було прийнято рішення спочатку створити початкову схему пристрою, яка включає інтеграцію всіх основних компонентів: датчиків, дисплея та виконавчих елементів. Було розроблено базову схему системи, яка відповідатиме за етапи зчитування даних, їх обробки та відображення результатів (рисунок 2.1).

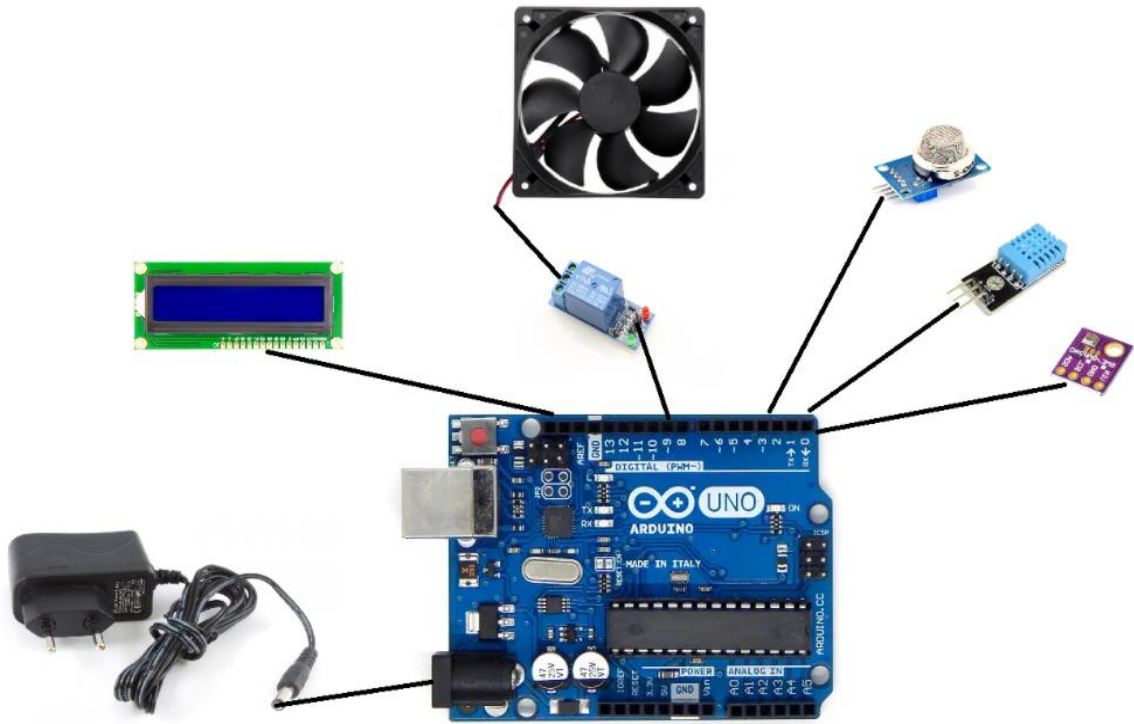


Рисунок 2.1 – Початкова схема пристрою

2.2.1 Керуючий мікроконтролерний пристрій Arduino

Основним елементом системи клімат-контролю є керуючий пристрій на базі мікроконтролера Arduino. Для таких завдань часто вибирають плати Arduino Nano або Arduino Uno завдяки їх доступності, простоті програмування та достатній потужності. Використання цих платформ для студентських та дослідницьких проектів у галузі клімат-контролю є поширеним вибором. Це пояснюється низьким порогом входження, великою спільнотою користувачів та наявністю численних бібліотек для роботи з різними датчиками та периферійними пристроями, що значно пришвидшує процес розробки та створення прототипів[1].

У процесі проектування пристрою було обрано плату Arduino Uno R3 (рисунок 2.2), що працює на мікроконтролері ATmega328P та включає у собі всі необхідні компоненти для ефективного використання у вигляді процесорного ядра для малобюджетної системи керування кліматом. Плата

містить у собі 14 пінів вводу/виводу, з яких 6 можуть використовуватися як ШІМ виходи. Плата також має USB-з'єднання, кварцовий резонатор на 16 МГц, роз'єм живлення, ICSP-хедер і кнопку Reset.

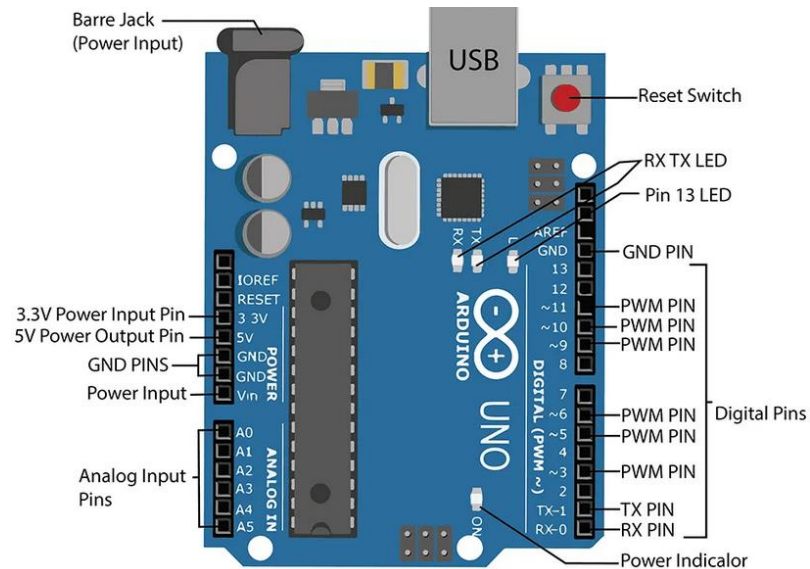


Рисунок 2.2 – Призначення контактів плати Arduino Uno

ATmega328P – це компактний, потужний та популярний мікроконтролер серії AVR від Microchip Technology (раніше Atmel), який активно застосовується у вбудованих системах, побутовій електроніці, робототехніці та інших сферах. Він має архітектуру RISC і вбудований восьмибітний ALU, працюючи при тактовій частоті до 20 МГц[2].

Пам'ять мікроконтролера складається з 32 Кб флеш-пам'яті для програм, 2 Кб оперативної пам'яті (ОЗУ) та 1 Кб EEPROM для зберігання даних, що дозволяє зберігати інформацію навіть після вимкнення живлення. Мікроконтролер працює в діапазоні робочої напруги від 1,8 В до 5,5 В [2].

Мікроконтролер підтримує завантаження Arduino bootloader, що значно полегшує процес розробки та тестування проектів. Мікроконтролер здатний виконувати інструкції за один тактовий цикл, що забезпечує пропускну здатність до 1 MIPS на МГц і сприяє ефективному використанню енергії[2].

Докладну інформацію про розпіновку та технічні характеристики можна знайти в офіційному datasheet мікроконтролера (рисунок 2.3) [3].

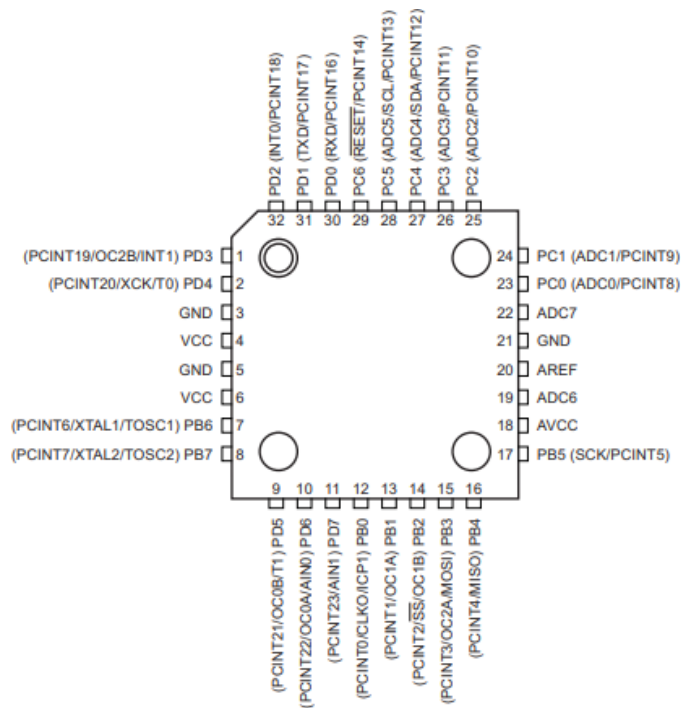


Рисунок 2.3 – ATmega328P pinout

Мікроконтролер у системі виконує основні функції, такі як ініціалізація та періодичне опитування датчиків кліматичних параметрів, обробка отриманих даних, перетворення одиниць вимірювання та порівняння з пороговими значеннями, а також виведення інформації на дисплей. Він також керує світлодіодною індикацією для відображення стану забруднення повітря, забезпечує обмін даними з персональним комп'ютером через USB-інтерфейс для передачі вимірювань і отримання команд, а також керує виконавчими пристроями. Підключення до інших компонентів здійснюється через цифрові та аналогові порти, а також інтерфейси I2C та UART.

2.2.2 Сенсори кліматичних параметрів

Одним із ключових елементів системи є комбінований сенсор BME280 виробництва Bosch Sensortec (рисунок 2.4). Цей сенсор дозволяє одночасно

вимірювати температуру повітря, відносну вологість та атмосферний тиск. Діапазон вимірювання температури становить від $-40\text{ }^{\circ}\text{C}$ до $+85\text{ }^{\circ}\text{C}$ з типовою точністю від $\pm 0.5\text{ }^{\circ}\text{C}$ до $\pm 1.5\text{ }^{\circ}\text{C}$ в залежності від діапазону. Діапазон вимірювання вологості варіюється від 0% до 100% з точністю $\pm 3\%$. Атмосферний тиск вимірюється в діапазоні від 300 гПа до 1100 гПа з абсолютною точністю $\pm 1.0\text{ гПа}$. Сенсор підключається до мікроконтролера Arduino через інтерфейс I2C, що забезпечує швидкий обмін даними. Напряга живлення сенсора варіюється від 1.71 В до 3.6 В , а споживання струму є дуже низьким, наприклад, 3.6 мкА при частоті оновлення даних 1 Гц для всіх трьох параметрів[4]. Використання комбінованого сенсора дозволяє зменшити кількість окремих компонентів на платі, спростити розведення та програмування, а також забезпечити високу точність вимірювань, що важливо для систем клімат контролю.

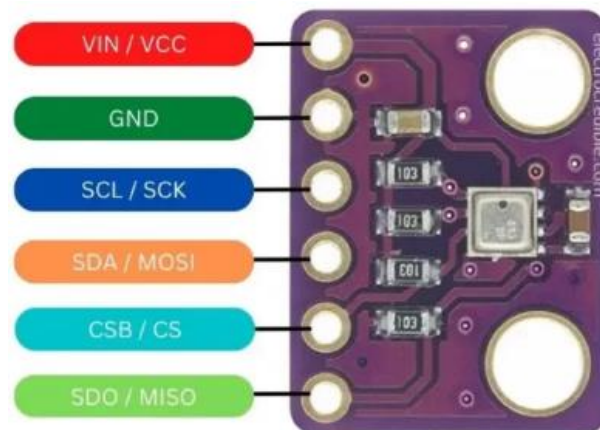


Рисунок 2.4 – BME280 3.3v pinout

Для зчитування параметрів вологості повітря у пристрої використовується додатковий датчик DHT11 (рисунок 3.5). Він працює на термісторі, з допомогою якого може вимірювати температуру та на резистивному елементі для вимірювання вологості. Хоч він і надає менш точні дані ніж BME280, у процесі тестування було виявлено, що останній є дефективним та надає не зовсім правильну інформацію щодо вологості

навколишнього середовища. Діапазон вимірювання температури DHT11 становить від 0 °С до +50 °С з точністю ± 2 °С. Відносна вологість вимірюється в діапазоні від 20 % до 90 % з точністю ± 5 %. Напруга живлення сенсора може коливатися від 3.3 В до 5.5 В. Сенсор має цифровий однопровідний інтерфейс зв'язку і мінімальний час відгуку між запитами складає 1 секунду[5].

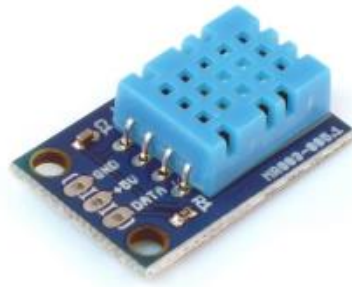


Рисунок 2.5 – Датчик DHT11

2.2.3 LCD дисплей

Рідкокристалічний дисплей використовується для відображення детальних значень кліматичних параметрів, системних повідомлень або меню налаштувань. Для таких цілей часто застосовують дисплеї типу 16x2, що мають 16 символів у двох рядках. Щоб спростити підключення та заощадити порти мікроконтролера такі дисплеї можна підключати за допомогою I2C-модуля на базі мікросхеми PCF8574 (рисунок 2.6)[7]. Такий підхід є особливо зручним для платформ з обмеженою кількістю пінів, оскільки стандартне підключення дисплея за допомогою паралельного інтерфейсу потребує великої кількості цифрових портів. Використання I2C-адаптера дозволяє зменшити кількість необхідних ліній зв'язку до двох (SDA та SCL), які можуть також використовуватися іншими I2C-пристроями, наприклад, сенсором BME280.

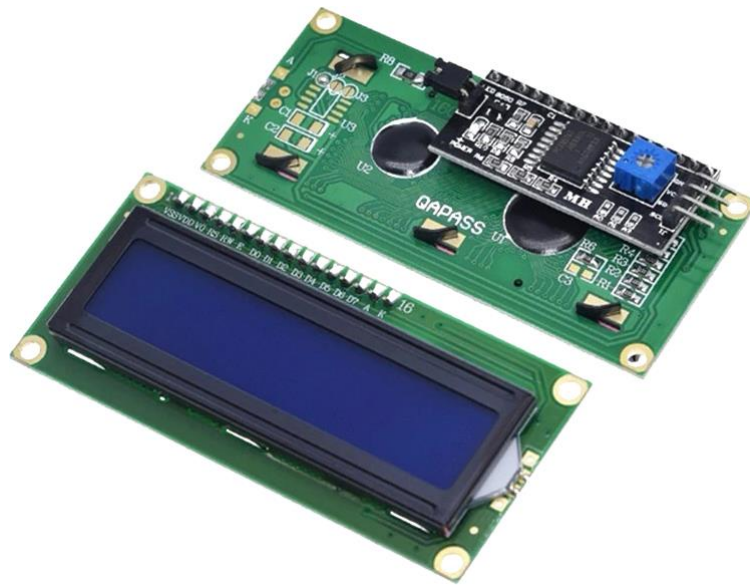


Рисунок 2.5 – LCD1602 з I2C модулем

LCD-дисплей з I2C-модулем підключається до пінів A4 та A5 мікроконтролера Arduino, які зазвичай використовуються для інших I2C-пристроїв. Для роботи з дисплеєм використовується бібліотека `LiquidCrystal_I2C.h`, яка містить функції для ініціалізації дисплея, керування підсвічуванням, встановлення позиції курсора, виведення тексту та очищення екрану. Це дозволило легко інтегрувати дисплей у систему клімат контролю для відображення актуальних даних у реальному часі.

2.2.4 Управління системою охолодження

Розроблений пристрій здатен активно впливати на деякі параметри мікроклімату, а саме на температуру навколишнього середовища. У проекті використовується вентилятор GAMEMAX GMX-WFBK що керується за допомогою релейного модуля. Реле дозволяє слабкострумовому сигналу від мікроконтролера керувати потужним електричним колом. Котушка реле, що керується сигналом 5 В від цифрового виводу Arduino, створює магнітне поле, яке замикає або розмикає силові контакти, через які проходить струм до навантаження.

Важливою особливістю якісних релейних модулів є гальванічна розв'язка між мікроконтролером і силовим колом, зазвичай реалізована через оптопару. Це захищає мікроконтролер від високовольтних сплесків або перешкод, що особливо важливо при роботі з індуктивними навантаженнями, такими як двигуни або котушки. Релейний модуль підключається до мікроконтролера через керуючий вхід IN, який з'єднується з цифровим виводом Arduino, а також через піни живлення (рисунок 2.6).

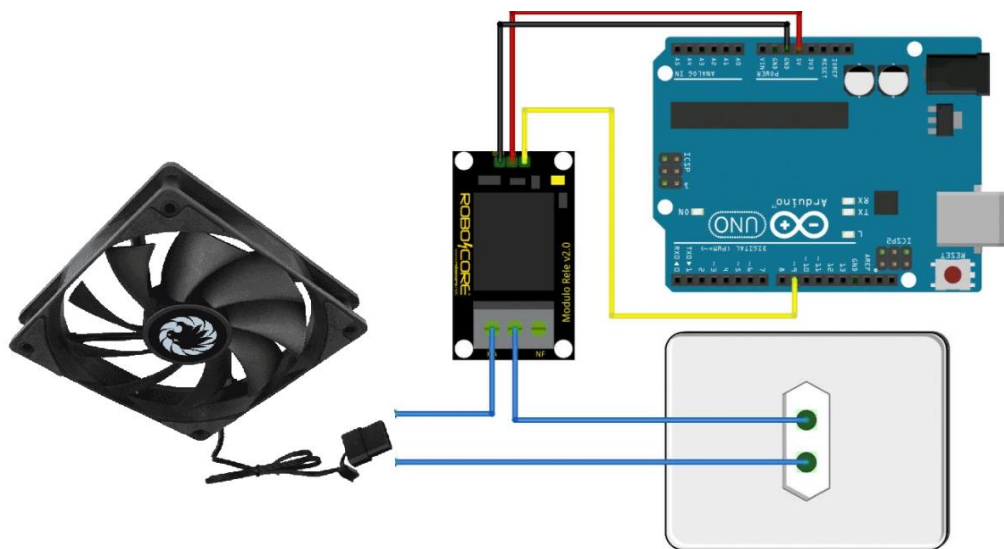


Рисунок 2.6 – Електрична схема підключення релейного модуля та вентилятора системи охолодження

2.2.5 Датчик газу

Для забезпечення безпеки та контролю за якістю повітря в приміщеннях у системі клімат контролю використовується датчик газу MQ-2 (рисунок 2.7). Цей сенсор дозволяє вимірювати рівень різних газів, таких як метан, пропан, дим та інші горючі гази, що особливо важливо для моніторингу повітря в закритих приміщеннях.

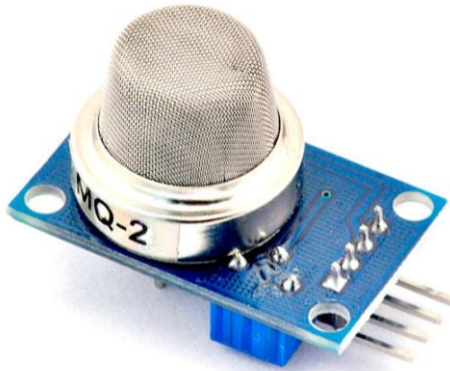


Рисунок 2.6 – Датчик газу MQ-2

MQ-2 працює завдяки резистивному елементу, чий опір змінюється при контакті з газами. Це дозволяє точно визначати їх концентрацію в повітрі. Датчик може вимірювати рівень газів у діапазоні від 300 до 10000 ppm, зокрема для метану та інших горючих газів[8]. Завдяки живленню від 5 В і низькому споживанню енергії, він ідеально підходить для використання в енергоефективних системах на основі Arduino.

Порівняно з іншими датчиками, такими як MQ-7, що спеціалізується на виявленні монооксиду вуглецю CO, MQ-2 є більш універсальним, оскільки він здатний виявляти кілька видів газів одночасно. Однак, якщо спосіб використання потребує точного вимірювання конкретних газів, наприклад як монооксид вуглецю, краще використовувати датчик MQ-7, який спеціалізується на цьому газі[9].

Вибір між MQ-2 та іншими датчиками залежить від конкретних вимог проекту. Якщо потрібно вимірювати кілька типів газів і точність не є критичною, MQ-2 буде відмінним варіантом. Для завдань, де потрібно точно вимірювати рівень монооксиду вуглецю або інших специфічних газів, можна звернутися до спеціалізованих датчиків, таких як MQ-7[9].

Інтеграція MQ-2 в систему дозволяє швидко реагувати на підвищені концентрації небезпечних газів, що є важливим параметром навколишнього середовища для кінцевого споживача.

3 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ СИСТЕМИ КЛІМАТ КОНТРОЛЮ

3.1 Загальна структура програмного забезпечення та конфігурація пінів мікроконтролера

Програмний модуль є ядром системи клімат-контролю, відповідальним за збір даних з сенсорів, їх обробку, прийняття рішень щодо керування виконавчими пристроями та взаємодію з користувачем. Розробка програмного забезпечення здійснювалася у середовищі Arduino IDE мовою програмування C++ з використанням специфічних для платформи Arduino бібліотек.

Програмне забезпечення розробленої системи клімат-контролю побудоване на базі стандартної для платформи Arduino структури, що включає дві основні функції: `setup()` та `loop()`. Функція `setup()` виконується один раз при старті мікроконтролера і призначена для ініціалізації апаратних компонентів та початкових налаштувань системи. Функція `loop()` виконується циклічно протягом усього часу роботи пристрою, реалізуючи основну логіку системи: опитування датчиків, обробку даних, керування виконавчими механізмами та оновлення інформації на дисплеї.

Важливим аспектом конфігурації програмного забезпечення є визначення відповідності між фізичними підключеннями компонентів до пінів мікроконтролера Arduino Uno та їх символічними іменами у програмному коді. Для цього активно використовуються препроцесорні директиви `#define`. Такий підхід дозволяє призначити змістовні імена номерам пінів, до яких підключені датчики, виконавчі пристрої, елементи індикації та кнопки керування. Окрім пінів, директива `#define` використовується також для встановлення значень констант за замовчуванням, наприклад, початкового порогу температури `TEMPERATURE_THRESHOLD_DEFAULT` 26.

Застосування директив `#define` для конфігурації апаратних підключень має дві ключові переваги. По-перше, це значно покращує читабельність та

зрозумілість програмного коду, оскільки замість абстрактних числових значень пінів використовуються їх логічні імена, що відображають функціональне призначення. По-друге, такий підхід суттєво спрощує процес реконфігурації системи. У випадку, якщо виникне необхідність змінити підключення будь-якого компонента до іншого піна мікроконтролера, достатньо буде внести зміни лише в одному місці – у відповідній директиві `#define` на початку програмного файлу, без необхідності пошуку та модифікації всіх місць використання цього піна в коді.

Функціонування системи базується на послідовному виконанні операцій у циклі `loop()` без застосування формальної операційної системи реального часу (RTOS) або складного диспетчера подій. Це забезпечує м'які гарантії реального часу, що є цілком прийнятним для системи клімат-контролю, де миттєва реакція з точністю до мілісекунд зазвичай не є критичною. Такий підхід виправданий своєю простотою та ефективністю. Однак у разі потреби у жорстких часових гарантіях, наприклад, для швидкої обробки аварійних сигналів або управління високодинамічними процесами, обрана архітектура може не забезпечувати своєчасність реакцій. Особливо це проявляється при зростанні обчислювального навантаження у циклі `loop()`, коли тривалість виконання певних операцій (наприклад, взаємодії з повільними датчиками або оновлення складного графічного інтерфейсу) може спричинити затримки в обробці інших важливих завдань.

Для наочного представлення використання апаратних ресурсів мікроконтролера Arduino Uno в розробленій системі клімат-контролю наведено опис основних підключень ключових компонентів. У програмному коді кожен пін має своє символічне ім'я, яке відповідає конкретному фізичному піну на платі Arduino Uno, а також визначає тип сигналу та функціональне призначення.

Датчик вологості DHT11 підключений до цифрового піна D2 (символьне ім'я `DHTPIN`). Цей пін використовується як цифровий вхід/вихід з однопровідним інтерфейсом для зчитування даних температури та вологості,

зокрема, вологості повітря. Релейний модуль, який керує живленням вентилятора системи охолодження, підключений до цифрового виходу D9 (RELAY_PIN). Для моніторингу концентрації газів використовується датчик MQ-2, який під'єднаний до аналогового входу A0 (MQ2_PIN) і забезпечує зчитування аналогового сигналу, пропорційного рівню газу.

Світлодіодні індикатори стану газу розподілені між цифровими виходами D10 (LED_GREEN) для сигналізації безпечного рівня, D11 (LED_YELLOW) для попереджувального середнього рівня та D12 (LED_RED) для небезпечного високого рівня концентрації. Звукове сповіщення при критичних рівнях газу забезпечує зумер, підключений до цифрового виходу D8 (BUZZER_PIN).

Керування пороговими значеннями температури реалізується за допомогою двох кнопок, підключених до цифрових входів D4 (BUTTON_INC_PIN) та D5 (BUTTON_DEC_PIN) з внутрішньою підтяжкою. Вони призначені для збільшення та зменшення значення температурного порогу відповідно.

Для виведення інформації на користувача використовується LCD-дисплей, який підключений через шину I2C, лінії якої відповідають аналоговим пінам A4 (SDA) та A5 (SCL). Аналогічним чином, через цю ж I2C-шину підключений датчик BME280, який зчитує параметри температури та атмосферного тиску.

3.2 Опис використаних бібліотек

Для забезпечення ефективної взаємодії з різними апаратними компонентами системи клімат-контролю, такими як датчики, дисплей та енергонезалежна пам'ять, було використано низку спеціалізованих бібліотек Arduino. Ці бібліотеки надають високорівневий інтерфейс програмування, який інкапсулює складність низькорівневих протоколів обміну даними та специфіку роботи з відповідними пристроями. Підключення необхідних

бібліотек здійснюється за допомогою препроцесорної директиви `#include`. Нижче наведено опис основних бібліотек, які застосовуються у програмному модулі.

Бібліотека `DHT.h` призначена для роботи з датчиками сімейства DHT, зокрема DHT11, що використовується для вимірювання відносної вологості повітря. У розробленому проекті основними функціями цієї бібліотеки є ініціалізація датчика за допомогою `dht.begin()` та зчитування вологості за допомогою `dht.readHumidity()`. Хоча датчик DHT11 також може вимірювати температуру, у системі ця функція замінена більш точними вимірюваннями від датчика BME280. Вибір DHT11 обґрунтований необхідністю отримувати достовірні дані щодо вологості, оскільки в ході тестування BME280 показав некоректні результати за цим параметром.

Бібліотека `Wire.h` є стандартною для платформи Arduino та забезпечує підтримку інтерфейсу I2C (Inter-Integrated Circuit). Цей двопровідний протокол обміну даними дозволяє підключати до мікроконтролера кілька периферійних пристроїв, використовуючи всього два пінових контакти – SDA та SCL. У проекті бібліотека `Wire.h` використовується опосередковано через інші бібліотеки, зокрема `LiquidCrystal_I2C.h` для роботи з LCD-дисплеєм та `GyverBME280.h` для роботи з датчиком BME280. Використання I2C-інтерфейсу значно оптимізує розподіл апаратних ресурсів, звільняючи пінові виходи мікроконтролера для підключення інших компонентів.

Бібліотека `LiquidCrystal_I2C.h` застосовується для керування рідкокристалічним LCD-дисплеєм, підключеним через I2C-модуль на основі контролера PCF8574 або аналогічного. Вона надає зручні функції для ініціалізації дисплея, керування підсвічуванням, очищення екрану, позиціонування курсора та виведення тексту. Використання цієї бібліотеки з I2C-модулем значно спрощує електричне підключення дисплея, оскільки замість паралельного інтерфейсу, що потребує численних ліній, застосовується лише дві шини, що особливо важливо для плат з обмеженою кількістю пінів, таких як Arduino Uno.

Бібліотека GyverBME280.h призначена для роботи з комбінованими датчиками BME280 виробництва Bosch Sensortec, які вимірюють температуру, вологість та атмосферний тиск. У програмному модулі використовуються функції ініціалізації датчика (`bme.begin()`), а також зчитування температури (`bme.readTemperature()`) та тиску (`bme.readPressure()`). Відзначається, що у наданому коді функція зчитування вологості не застосовується, що співпадає з інформацією про некоректну роботу каналу вологості в конкретній реалізації датчика, через що й застосовується окремий датчик DHT11.

Бібліотека EEPROM.h забезпечує роботу з внутрішньою енергонезалежною пам'яттю мікроконтролера ATmega328P, на якому базується плата Arduino Uno. Вона надає можливість зберігати та читати дані, які зберігаються навіть після відключення живлення. У системі клімат-контролю бібліотека використовується для збереження налаштувань користувача, зокрема значення температурного порогу, що гарантує збереження цих параметрів при перезавантаженні або вимкненні пристрою.

Взаємодія між бібліотеками організована за принципом абстракції: Wire.h забезпечує низькорівневий обмін даними по шині I2C, на базі якого працюють бібліотеки LiquidCrystal_I2C.h та GyverBME280.h для керування відповідними периферійними пристроями. Такий підхід значно спрощує розробку, дозволяючи сконцентруватися на реалізації основної функціональності системи без необхідності детально опрацьовувати протоколи та особливості апаратної взаємодії.

3.3 Розробка алгоритмів роботи основних модулів програми

Функціонування системи клімат-контролю визначається набором алгоритмів, реалізованих у програмному модулі. Ці алгоритми охоплюють процеси ініціалізації системи, зчитування даних з сенсорів, обробки цих даних, прийняття рішень щодо керування виконавчими пристроями, а також взаємодії з користувачем через кнопки та LCD-дисплей.

3.3.1 Алгоритм зчитування даних з сенсорів

Зчитування актуальних параметрів навколишнього середовища є однією з ключових функцій, що виконуються в циклі `loop()` програмного модуля. В рамках кожної ітерації відбувається послідовний опит датчиків, що підключені до системи. Зокрема, для отримання відносної вологості виконується виклик функції `dht.readHumidity()` з бібліотеки `DHT.h`. Саме цей датчик використовується для вимірювання вологості через виявлені апаратні особливості датчика `BME280`, описані раніше. Температура навколишнього середовища зчитується з `BME280` за допомогою функції `bme.readTemperature()`, що повертає точне значення для подальшої логіки керування кліматом. Атмосферний тиск з цього ж сенсора зчитується функцією `bme.readPressure()/100.0F`, де поділ на 100 виконує конвертацію із Паскалів у гектопаскалі, що відповідає стандартним метеорологічним одиницям. Для контролю рівня небезпечних газів використовується аналоговий вхід, до якого підключено датчик `MQ-2`. Значення рівня газу зчитується та повертає числовий показник у діапазоні 0-1023, пропорційний концентрації газів у повітрі.

Для забезпечення коректності отриманих даних застосовується базова валідація. Функція `isnan()` використовується для перевірки, чи не повернули датчики `DHT11` та `BME280` нечислові значення, які можуть свідчити про збої у зв'язку або помилки читання (лістинг 3.1):

Лістинг 3.1 – Функція валідації `isnan()`

```
if (isnan(humidity) || isnan(bmpTemperature)) {  
    return;  
}
```

У разі виявлення таких помилок ітерація циклу `loop()` негайно завершується, що дозволяє уникнути подальшої обробки некоректних даних. Такий підхід суттєво підвищує надійність системи, запобігаючи прийняттю рішень на основі хибних сигналів.

Крім того, на початковому етапі ініціалізації системи передбачена перевірка доступності датчика BME280 (лістинг 3.2). Функція ініціалізації `bme.begin()` повертає логічне значення, що сигналізує про успішність встановлення зв'язку з датчиком. Якщо датчик не відповідає, програма виводить відповідне діагностичне повідомлення в серіальний порт і переходить у нескінченний цикл, зупиняючи подальше виконання коду. Це забезпечує запобігання роботі системи з відсутніми або некоректними даними.

Лістинг 3.2 – Обробка ініціалізації BME280

```
if (!bme.begin()) {  
    Serial.println("Не вдається ініціалізувати BME280!");  
    while (1); // Зупинка виконання  
}
```

Зазначена послідовність опитування сенсорів формує відносно синхронний знімок стану навколишнього середовища для кожної ітерації циклу. Враховуючи сумарний час обробки кожного датчика, повний цикл виконується з частотою, що зазвичай становить декілька разів на секунду. Фактична швидкість оновлення параметрів залежить від обсягу додаткових операцій, таких як оновлення інтерфейсу користувача або обробка команд з кнопок.

3.3.2 Алгоритм обробки даних та прийняття рішень

Після зчитування даних із сенсорів програмний модуль переходить до їх обробки та прийняття рішень у межах циклу `loop()`. Для зручності користувача

значення атмосферного тиску, що отримується з датчика BME280 у гектопаскалях, конвертується у міліметри ртутного стовпця за формулою:

$$\text{тиск (мм рт. ст.)} = \frac{\text{тиск (гПа)} * 100}{133,322} \quad (3.1)$$

Це значення виводиться на LCD-дисплей для інформування користувача. Основна логіка керування релейним модулем полягає у порівнянні поточної температури, зчитаної з датчика BME280, з встановленим порогом `temperatureThreshold`. Якщо температура перевищує цей поріг, відбувається активація реле (лістинг 3.3).

Лістинг 3.3 – Активація реле та передача живлення на вентилятор

```
if (bmpTemperature > temperatureThreshold) {
    digitalWrite(RELAY_PIN, HIGH); // Увімкнення вентилятора
} else {
    digitalWrite(RELAY_PIN, LOW); // Вимкнення вентилятора
}
```

Ще одним важливим елементом системи є моніторинг концентрації газів за допомогою датчика MQ-2, значення якого зчитується аналоговим входом. В залежності від рівня сигналу реалізується керування світлодіодами та зумером. При значенні менше 150 вмикається зелений світлодіод, при значеннях від 150 до 350 – жовтий, а при перевищенні 350 – червоний світлодіод та звуковий сигнал (лістинг 3.4):

Лістинг 3.4 – Керування світлодіодами та зумером залежно від значень датчика газу MQ-2

```
if (mq2Value < 150) {
    digitalWrite(LED_GREEN, HIGH);
    digitalWrite(LED_YELLOW, LOW);
    digitalWrite(LED_RED, LOW);
}
```

```

    digitalWrite(BUZZER_PIN, LOW);
} else if (mq2Value >= 150 && mq2Value <= 350) {
    digitalWrite(LED_GREEN, LOW);
    digitalWrite(LED_YELLOW, HIGH);
    digitalWrite(LED_RED, LOW);
    digitalWrite(BUZZER_PIN, LOW);
} else {
    digitalWrite(LED_GREEN, LOW);
    digitalWrite(LED_YELLOW, LOW);
    digitalWrite(LED_RED, HIGH);
    digitalWrite(BUZZER_PIN, HIGH);
}

```

Логіка обробки та прийняття рішень у системі є детермінованою, тобто при однакових вхідних параметрах вона генерує стабільні та передбачувані вихідні сигнали. Це важлива властивість для систем контролю, що спрощує тестування, налагодження та підвищує надійність роботи.

3.3.3 Обробка натискань кнопок

Система клімат-контролю надає користувачеві можливість налаштувати один із ключових параметрів – поріг температури, при якому активується система охолодження. Для цього використовуються дві фізичні кнопки, підключені до цифрових входів мікроконтролера. Алгоритм обробки натискань кнопок реалізований в основному циклі `loop()`.

Під час кожної ітерації циклу програма зчитує логічні рівні на пінах, до яких підключені кнопки збільшення (`BUTTON_INC_PIN`) та зменшення (`BUTTON_DEC_PIN`) порогу температури. Кнопки підключені з використанням внутрішніх підтягуючих резисторів (режим `INPUT_PULLUP`), через що ненависнутий стан відповідає високому логічному рівню (`HIGH`), а натиснутий – низькому (`LOW`). Поточні стани зберігаються у змінних `buttonIncState` та `buttonDecState`.

Для уникнення помилкових спрацьовувань, спричинених явищем брязкоту контактів – короточасних багаторазових замикань і розмикань під час натискання чи відпускання кнопки – реалізовано програмний антибрязкіт. Логіка а антибрязкіту перевіряє кілька умов: по-перше, чи поточний стан кнопки відповідає натиснутому (`buttonIncState == LOW` або `buttonDecState == LOW`); по-друге, чи попередній стан був ненатиснутим (`lastButtonIncState == HIGH` або `lastButtonDecState == HIGH`), що дозволяє зафіксувати лише один перехід у стан натиснутої кнопки; і по-третє, чи минув достатній інтервал часу з останнього зареєстрованого натискання (`millis() - lastButtonPressTime > debounceDelay`). Змінна `lastButtonPressTime` зберігає час останнього спрацювання, а `debounceDelay` встановлено у 50 мілісекунд для фільтрації брязкоту контактів.

Якщо усі ці умови виконані, вважається, що відбулося коректне одиничне натискання. Відповідно, якщо спрацьовує кнопка збільшення (`BUTTON_INC_PIN`), значення `temperatureThreshold` інкрементується, а якщо кнопка зменшення (`BUTTON_DEC_PIN`) – декрементується. Після кожної зміни нового значення воно одразу записується у енергонезалежну пам'ять EEPROM за адресою 0 за допомогою функції `EEPROM.write(0, temperatureThreshold)`; що забезпечує збереження налаштувань навіть при вимкненні живлення.

Час останнього натискання оновлюється командою `lastButtonPressTime = millis()`; що використовується як у логіці антибрязкіту, так і в керуванні відображенням інформації на LCD-дисплеї. Попередні стани кнопок зберігаються у змінних `lastButtonIncState` та `lastButtonDecState` для подальшого коректного відслідковування переходів.

Використання програмного антибрязкіту є важливою практикою вбудованих систем з фізичними кнопками, оскільки це значно підвищує надійність вводу та запобігає помилковим багаторазовим спрацюванням одного натискання. Негайний запис порогу температури в EEPROM після

кожного коректного натискання забезпечує персистентність налаштувань, що підвищує зручність експлуатації.

3.3.4 Алгоритм відображення інформації на LCD-дисплеї

Рідкокристалічний дисплей (LCD) служить основним інтерфейсом візуального інформування користувача про поточний стан мікроклімату та налаштування системи. Алгоритм керування відображенням інформації реалізовано в основному циклі `loop()`, де передбачено декілька механізмів, що забезпечують своєчасність оновлення та зручність взаємодії з користувачем.

Щоб уникнути надмірно частого оновлення екрану, що може спричинити мерехтіння та додаткове навантаження на мікроконтролер, оновлення дисплея обмежене інтервалом у 2 секунди. Це контролюється перевіркою умови `if (millis() - lastUpdate > 2000)`, де змінна `lastUpdate` зберігає час останнього оновлення. Після виконання оновлення цей таймер оновлюється викликом `lastUpdate = millis();`

Відображення інформації адаптується залежно від взаємодії користувача з кнопками налаштування порогу температури. Якщо протягом останніх 3 секунд кнопки не були натиснуті (умова `if (millis() - lastButtonPressTime > 3000)`), система демонструє основні кліматичні параметри. Ця логіка реалізована через змінну-прапорець `showFirstScreen`, що забезпечує циклічне перемикання двох інформаційних екранів.

На першому екрані (коли `showFirstScreen == true`) відображаються значення температури та вологості: після очищення екрану (`lcd.clear()`) курсор встановлюється в початок першого рядка (`lcd.setCursor(0, 0)`) для виведення тексту "Temp: ", після чого показується температура з датчика BME280 (`bmpTemperature`) з позначкою "C". Другий рядок (`lcd.setCursor(0, 1)`) містить текст "Humidity: " і відносну вологість з датчика DHT11 (`((int)humidity)`) із символом "%".

На другому екрані (`showFirstScreen == false`) виводяться поточні значення атмосферного тиску та сире аналогове значення газового датчика. Екран очищується, у першому рядку встановлюється курсор на позицію (0, 0) та виводиться "Psr: " із значенням тиску в мм рт. ст. (`pressureInMmHg`) та одиницями вимірювання "mmHg". Другий рядок (`lcd.setCursor(0, 1)`) містить "Gas: " із аналоговим числом з датчика MQ-2 (`mq2Value`). Після відображення екрану прапорець `showFirstScreen` інвертується (`showFirstScreen = !showFirstScreen`), що забезпечує циклічне переключення між двома екранами кожні 2 секунди, доки користувач не почне змінювати налаштування.

У випадку, коли кнопки були натиснуті менш ніж 3 секунди тому (`else-гілка умови millis() - lastButtonPressTime > 3000`), система переходить у режим налаштування порогу температури. Екран очищується, у першому рядку виводиться текст "Temp threshold:", а у другому (`lcd.setCursor(6, 1)`) центрується та відображається актуальне значення температурного порогу (`temperatureThreshold`) з позначкою "C". Цей екран залишається активним доти, доки користувач здійснює налаштування або не припиняє взаємодію протягом 3 секунд, після чого автоматично повертається до циклу основних екранів.

4 ТЕСТУВАННЯ СИСТЕМИ

Тестування є невід’ємним етапом розробки будь-якого програмно-апаратного комплексу, оскільки дозволяє перевірити його працездатність, надійність та відповідність поставленим вимогам.

Працездатність всіх компонентів системи було перевірено окремо перед збиранням приладу. Живлення здійснювалось через зовнішній блок живлення на 12W для того щоб не перегрівати плату мікроконтролера. На рисунку 4.1 зображено зібрану систему клімат контролю.

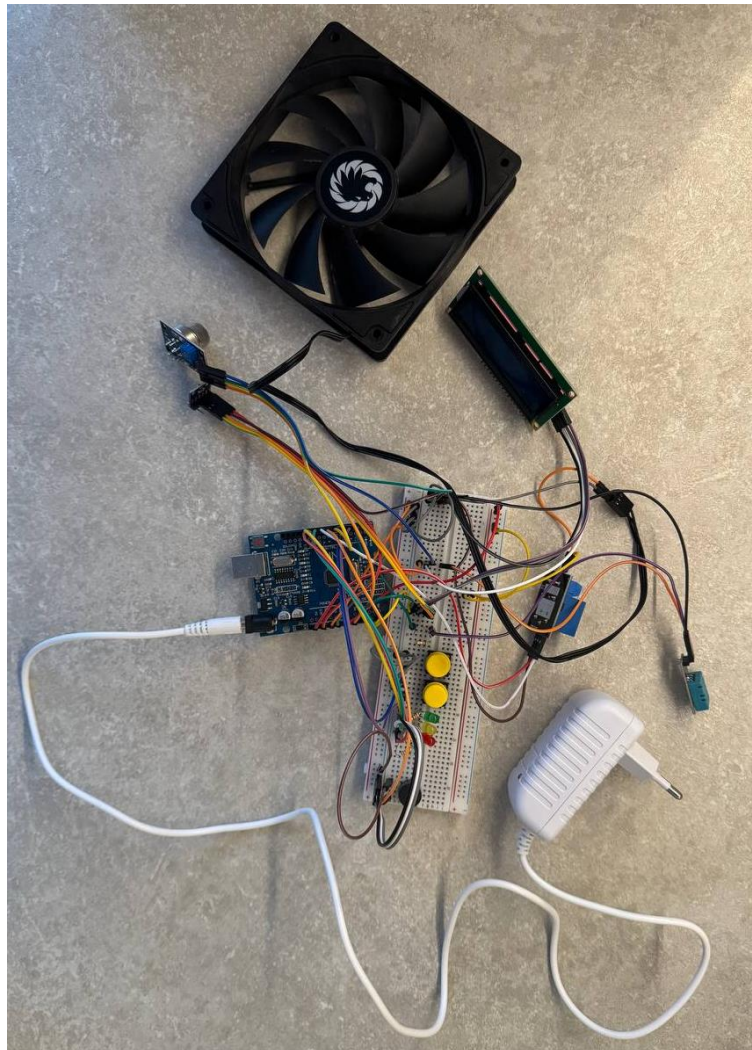


Рисунок 4.1 – Підключення системи клімат контролю у приміщенні

Метою тестування розробленої системи клімат-контролю є верифікація коректності функціонування кожного окремого компонента та їх взаємодії в інтегрованому середовищі, а також підтвердження того, що система в цілому виконує всі заявлені функції згідно з технічним завданням. Щоб оцінити працездатність приладу підключимо живлення та задамо параметри роботи вентилятора. Під час ініціалізації компонентів на LCD екран виводиться значення температури при якому буде працювати система охолодження повітря у вигляді зовнішнього вентилятора (рисунок 4.2).



Рисунок 4.2 – Максимально допустима комфортна температура задана у системі

Через декілька секунд, коли всі компоненти системи ініціалізуються, на екрані можна побачити вивід даних щодо температури та вологості, що знімаються з датчиків BME280 та DHT11 відповідно (рисунок 4.3).



Рисунок 4.3 – Зчитування та вивід температури і вологості на дисплей

Через декілька секунд екран системи оновлюється і виводить дані про атмосферний тиск та відносний рівень забрудненості повітря небезпечними газами (рисунок 4.4). Також про рівень газів у повітрі свідчать три світлодіоди, що світяться різними кольорами в залежності від стану навколишнього середовища.

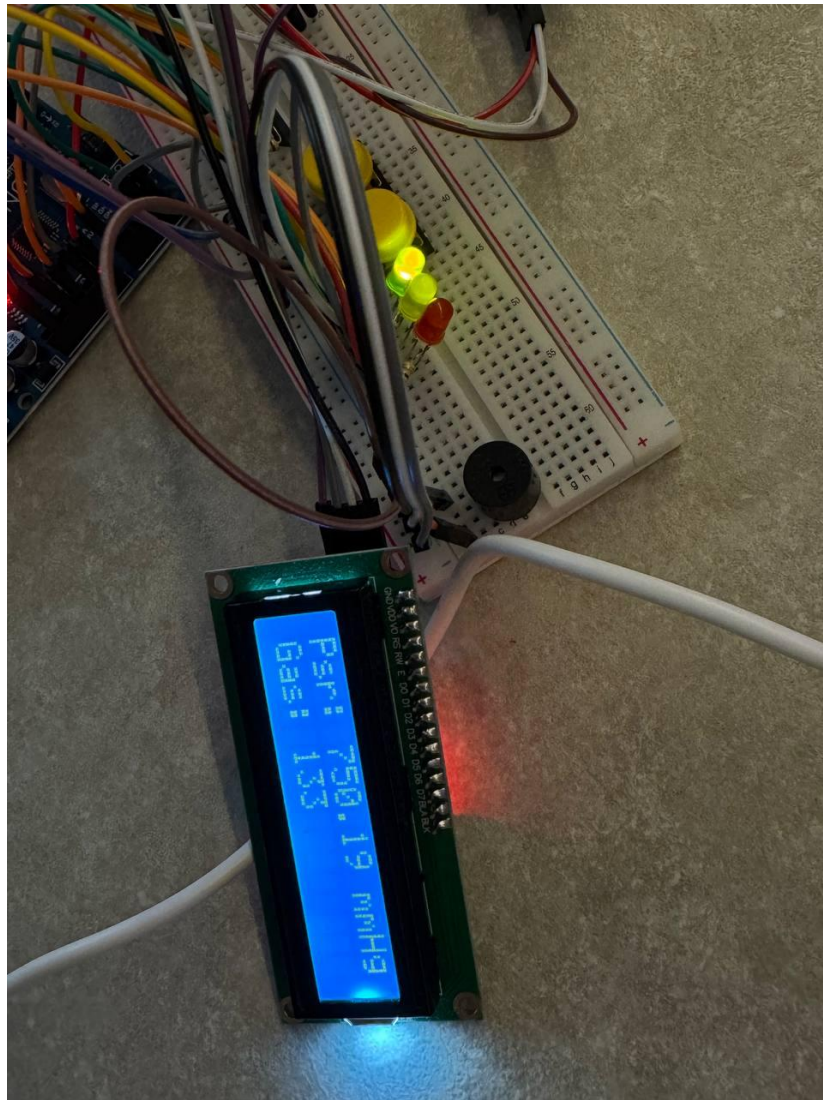


Рисунок 4.4 – Вивід даних про атмосферний тиск і рівень забрудненості повітря та світлова індикація

Якщо температура стане вищою ніж встановлений користувачем поріг, то спрацює реле, яке запустить систему охолодження (рисунок 4.5). Таким чином система може регулювати один з найважливіших параметрів навколишнього середовища – температуру.

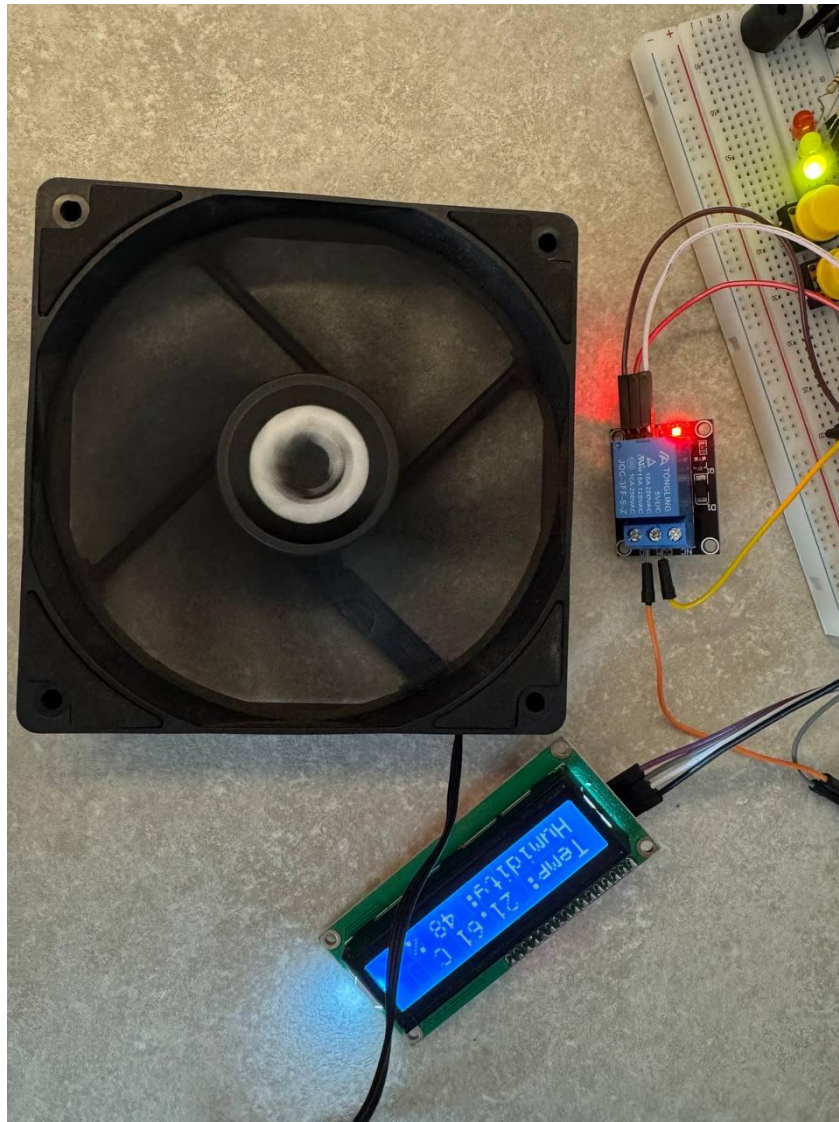


Рисунок 4.5 – Реакція на перевищення ліміту температури та ввімкнення охолодження

Також було перевірено реакцію пристрою на підвищений рівень забрудненості повітря газами за допомогою імітації витоку метану. При середньому рівні газу загоряється жовтий світлодіод. При високому рівні газу загоряється червоний світлодіод та активується зумер. LCD-дисплей продовжує відображати актуальні дані. Після усунення джерела газу та зниження його концентрації індикація повертається до нормального стану (зелений світлодіод).

Користувач за допомогою кнопок має можливість змінити поріг нормальної температури використовуючи кнопки, встановлені у системі. Під час нормальної роботи системи користувач натискає кнопки для зміни температурного порогу. LCD-дисплей переходить в режим відображення та налаштування поточного температурного порогу. Кожне натискання кнопки коректно змінює значення порогу на дисплеї. Після завершення налаштування (через 3 секунди бездіяльності кнопок) нове значення порогу зберігається в EEPROM (перевіряється шляхом перезавантаження системи або спостереженням за поведінкою реле відносно нового порогу). Система коректно реагує на новий поріг, наприклад, реле може вимкнутись, якщо новий поріг став вищим за поточну температуру, або увімкнутись, якщо нижчим (рисунок 4.6).

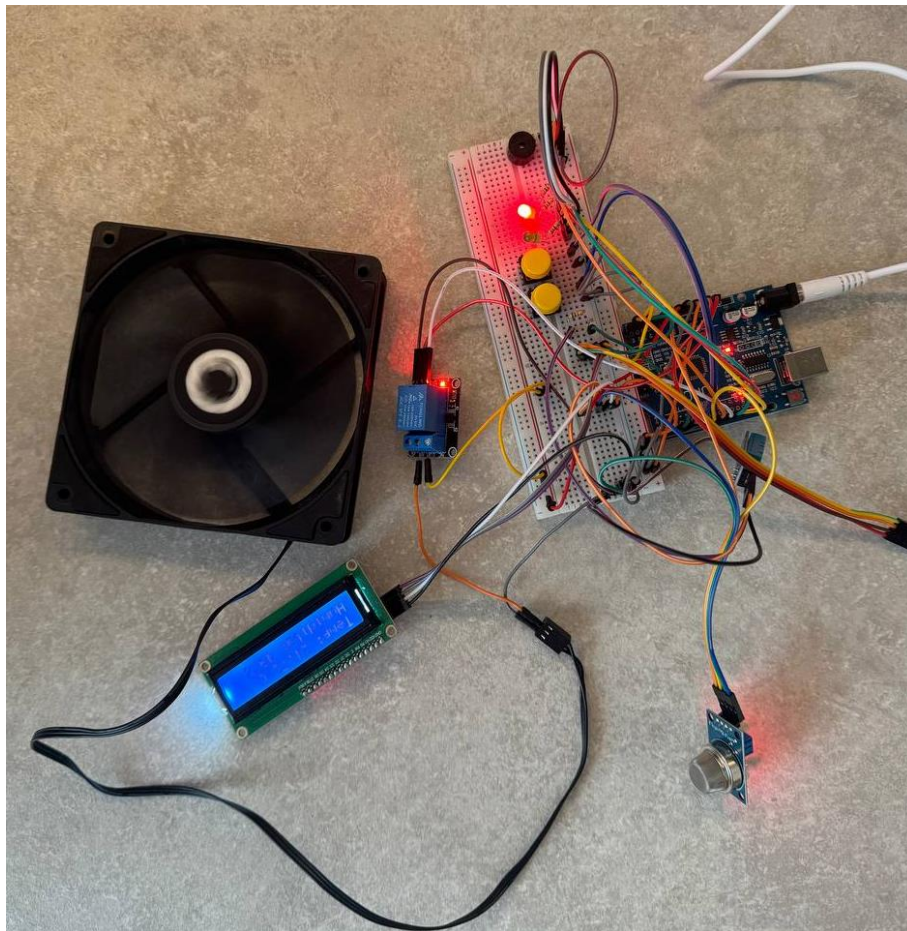


Рисунок 4.6 – Загальний вигляд системи під час тестування

Комплексне тестування системи за різними сценаріями використання показало коректну взаємодію всіх компонентів. Система адекватно реагувала на зміни параметрів навколишнього середовища, такі як температура, вологість, концентрація газу та на команди користувача. Логіка керування системою охолодження та система сповіщення про рівень газу функціонували відповідно до закладених алгоритмів.

Протягом тестових сесій система демонструвала стабільну роботу. Збоїв у роботі мікроконтролера чи окремих модулів не спостерігалось. Дані з датчиків зчитувалися коректно та з прийнятною періодичністю. Логіка прийняття рішень (керування реле, активація індикації) працювала без помилок. Користувацький інтерфейс, реалізований за допомогою LCD-дисплея та кнопок, функціонував належним чином, забезпечуючи можливість моніторингу та налаштування системи.

Таким чином, результати тестування підтверджують, що розроблена система клімат-контролю є працездатною та виконує покладені на неї функції. Виявлені особливості роботи окремих компонентів були враховані при проектуванні та успішно інтегровані в загальну архітектуру системи.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було реалізовано доступну мікроконтролерну систему клімат-контролю для приміщень на базі Arduino UNO. Проведено огляд сучасних рішень і сформульовано технічне завдання, на підставі якого підібрано апаратну частину: комбінований сенсор BME280 для температури й тиску, DHT11 для вологості, газовий датчик MQ-2, LCD екран з I2C-адаптером, кнопковий інтерфейс та релейний модуль для керування вентилятором. Розроблено програмне забезпечення мовою C++ у середовищі Arduino IDE з використанням бібліотек Wire, LiquidCrystal_I2C, GyverBME280, DHT та EEPROM; передбачено обробку можливих збоїв ініціалізації датчиків і фільтрацію некоректних показників під час роботи.

Зібраний макет пройшов тестування, яке підтвердило точність зчитування даних, стабільність роботи алгоритмів керування вентилятором та коректність світлової й звукової індикації різних рівнів забрудненості повітря. Система успішно зберігає вибраний користувачем температурний поріг у EEPROM і після перезапуску відновлює налаштування без втручання оператора.

Створено недороге, енергоефективне та легко масштабоване рішення, що може бути інтегроване у розумний дім або використане як навчальний стенд для вивчення основ вбудованих систем.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Arduino UNO // Офіційний сайт Arduino. URL: <https://arduino.ru/Hardware/ArduinoBoardUno> (дата звернення: 18.05.2025).
2. Arduino UNO R3 // Офіційний магазин Arduino. URL: <https://store.arduino.cc/products/arduino-uno-rev3> (дата звернення: 18.05.2025).
3. ATmega328P Automotive: Datasheet / Atmel. 2016. URL: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf (дата звернення: 18.05.2025).
4. Башир Х. М. BME280 Pinout, Specs, & Applications // Electrocredible. 2024. URL: <https://electrocredible.com/bme280-pinout-specs-applications/> (дата звернення: 18.05.2025).
5. Technical Data MQ-2 Gas Sensor V2.5 / Zhengzhou Winsen Electronics Technology Co., Ltd. 2022. URL: <https://www.tme.eu/Document/7a4fd48d400b8c4c8309ef1e2b13cdd4/MR003-005-1.pdf> (дата звернення: 18.05.2025).
6. Arduino_uno_guy. I2C Liquid Crystal Displays // Arduino Project Hub. 2020. URL: https://projecthub.arduino.cc/arduino_uno_guy/i2c-liquid-crystal-displays-5eb615 (дата звернення: 18.05.2025).
7. Міра Л. How to Use 16x2 Lcd and I2C Module With Arduino // Instructables. 2024. URL: <https://www.instructables.com/How-to-Use-16x2-Lcd-and-I2C-Module-With-Arduino/> (дата звернення: 18.05.2025).
8. MQ-2 Semiconductor Sensor for Combustible Gas: Technical Data / Hanwei Electronics. 2009. URL: <https://www.pololu.com/file/0j309/mq2.pdf> (дата звернення: 18.05.2025).
9. MQ-7 Semiconductor Sensor for Carbon Monoxide: Technical Data / Hanwei Electronics. 2009. URL: <https://cdn.sparkfun.com/assets/b/b/b/3/4/MQ-7.pdf> (дата звернення: 18.05.2025).

10. The History of HVAC Systems: How They Evolved and Gained Popularity Worldwide // Dows ClimateCare. 2023. URL: <https://www.dowclimatecare.com/blog/the-history-of-hvac-systems-how-they-evolved-and-gained-popularity-worldwide/> (дата звернення: 20.05.2025).

11. ДБН В.2.5-67:2013. Опалення, вентиляція та кондиціювання / Мінрегіон України. URL: https://e-construction.gov.ua/laws_detail/3074971619479783152 (дата звернення: 20.05.2025).

12. Векторні моделі логіки і структури для тестування та моделювання цифрових схем / Г.В. Хаханова, В.І. Хаханов, С.В. Чумаченко та ін. // Радіоелектроніка, інформатика, управління. – Запоріжжя: ЗНТУ. – 2021. – №3 . – С.69–85.

13. Квантові цифро-аналогові обчислення / Г.В. Хаханова, С.В. Чумаченко, Д.Ю. Рахліс та ін. // Радіоелектроніка, інформатика, управління. – Запоріжжя: ЗНТУ. – 2022. – №4 (63). – С.40–60.

14. Vector Models for Modeling Logic Based on XOR-Relations / [V. Hahanov, E. Litvinova, O. Shevchenko et al.] // 16th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, 24-27 February 2022: proceedings. – IEEE, 2022. – P. 823–828.