

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра електроніки, загальної та прикладної фізики

«До захисту допущено»
Завідувачка кафедри

_____ Лариса ОДНОДВОРЕЦЬ
_____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня «магістр»

за спеціальністю 171 «Електроніка»
освітньо-професійної програми «Електронні інформаційні системи»
на тему **«Автоматизована система керування датчиками на основі мікроконтролерів за технологією «Розумний будинок»»**

Здобувача групи ЕП.м-32 Шевченка Миколи Юрійовича

Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

_____ Микола ШЕВЧЕНКО

Керівник д-р фіз.-мат. наук, професор Лариса ОДНОДВОРЕЦЬ _____

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Факультет електроніки та інформаційних технологій
Кафедра електроніки, загальної та прикладної фізики
Спеціальність 171 – Електроніка, освітньо-професійна програма
«Електронні інформаційні системи»

ЗАТВЕРДЖУЮ

Завідувачка кафедри

_____ Лариса ОДНОДВОРЕЦЬ

«04» листопада 2024 р.

**ІНДИВІДУАЛЬНЕ ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА**

Шевченка Миколи Юрійовича

1. Тема роботи **«Автоматизована система керування датчиками на основі мікроконтролерів за технологією «Розумний будинок»**

затверджена наказом СумДУ від «01» листопада 2024 р., № 1130-VI

2. Термін здачі здобувачем закінченої роботи: 11 грудня 2024 р.

3. Вихідні дані до роботи (актуальність, мета):

Актуальність тематики полягає у важливості впровадження інтелектуальних технологій для підвищення комфорту, безпеки та енергоефективності житлових і комерційних приміщень, що є критично важливим у сучасних умовах розвитку інформаційних технологій та автоматизації. Мета кваліфікаційної роботи полягала в розробці та аналізі автоматизованої системи керування датчиками, що базується на використанні мікроконтролерів для реалізації технології «Розумний будинок», з урахуванням її функціональних, енергетичних і практичних характеристик.

4. Зміст текстової частини роботи (перелік питань, які необхідно розробити):

- 1) Розглянути сучасні мікроконтролерні системи за технологією «Розумний будинок».
- 2) Розробити елементну базу для створюваної системи управління датчиками.
- 3) Провести експериментальні дослідження системи.

5. Перелік графічного матеріалу для презентації:

Слайд №1, 2 – Актуальність і мета роботи.

Слайди №3, 4 – Промислові системи «Розумний будинок».

Слайди №5-7 – Принцип роботи мікроконтролерів і датчиків.

Слайд №8 – Алгоритм роботи головного модуля системи.

Слайди №9-13 – Електричні принципіальні схеми.

Слайд №14 – Висновки.

КАЛЕНДАРНИЙ ПЛАН

№ пор.	Назва етапів виконання кваліфікаційної роботи	Термін виконання етапів роботи	Примітка про стан виконання роботи
1	Теоретичний огляд систем віддаленого моніторингу навколишнього середовища	04.11.2024 р.	Виконано
2	Проведення експерименту, моделювання, розрахунків, обробка результатів	11.11.2024 р.	Виконано
3	Розробка схем приладу	18.11.2024 р.	Виконано
4	Симуляція роботи системи	25.11.2024 р.	Виконано
5	Підготовка тексту магістерської роботи	01.12.2024 р.	Виконано
6	Підготовка презентації до захисту кваліфікаційної роботи	07.12.2024 р.	Виконано
7	Попередній захист роботи	12.12.2024 р., 10 ⁰⁵ (онлайн)	Виконано
8	Захист кваліфікаційної роботи	17.12.2024 р., 11 ⁴⁰	

6. Дата видачі індивідуального завдання: 01.11.2024 р.

Здобувач

Микола ШЕВЧЕНКО

Керівник

Лариса ОДНОДВОРЕЦЬ

АНОТАЦІЯ

Кваліфікаційна робота викладена на 34 сторінках, зокрема, містить 13 рисунків, 8 таблиць, список використаних джерел складається з 11 найменувань.

Актуальність теми обумовлена важливістю впровадження інтелектуальних технологій для підвищення комфорту, безпеки та енергоефективності житлових і комерційних приміщень, що є критично важливим у сучасних умовах розвитку інформаційних технологій та автоматизації. Дослідження в цій галузі сприяють оптимізації використання ресурсів та створенню інноваційних рішень для удосконалення систем керування в розумних будинках.

Мета кваліфікаційної роботи магістра полягала у розробці та аналізі автоматизованої системи керування датчиками, що базується на використанні мікроконтролерів для реалізації технології «Розумний будинок», з урахуванням її функціональних, енергетичних і практичних характеристик.

Методи: системний аналіз, моделювання, оптимізація автоматизованих систем керування, програмування мікроконтролерів для інтеграції датчиків у технологію «Розумного будинку».

Отримані результати:

1. Показано, що автоматизована система керування датчиками на основі мікроконтролерів ефективно підтримує функціональність «Розумного будинку». Вона дозволяє інтегрувати різні датчики (температури, вологості, освітленості, руху тощо) у єдину мережу для забезпечення комфорту, безпеки та енергоефективності.

2. Одержано нові дані щодо впровадження мікроконтролерів у системи керування датчиками. Зокрема, використання мікроконтролерів ATmega2560 та ATmega328 забезпечило високу стабільність роботи та спрощення інтеграції сенсорів завдяки застосуванню бездротових модулів.

3. Установлено, що використання алгоритму адаптивного вибору частотного діапазону підвищує надійність бездротового зв'язку в системах «Розумний

будинок». Це сприяє мінімізації перешкод і підвищує безперебійність передачі даних.

4. Розроблено та протестовано апаратну і програмну частини автоматизованої системи. Проведені експериментальні дослідження підтвердили її здатність реагувати на критичні ситуації (задимлення, витік води) та оперативно передавати відповідні сповіщення.

5. Практичне значення одержаних результатів полягає у можливості впровадження розробленої системи в житлових і комерційних приміщеннях. Система сприяє оптимізації енергоспоживання, покращенню безпеки та забезпеченню зручних умов проживання.

Ключові слова: розумний будинок, датчики, сенсори, мікроконтролер, система.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 ТЕХНОЛОГІЇ «РОЗУМНИЙ БУДИНОК»	8
1.1 Промислові системи «Розумний будинок»	8
1.1.1 Ajax	8
1.1.2 Fibaro	10
1.1.3 Xiaomi.....	11
1.2 Аналіз вимог до автоматизації в житлових приміщеннях.....	13
1.3 Принцип роботи мікроконтролерів.....	14
1.4 Принцип роботи датчиків.....	15
РОЗДІЛ 2 АПАРАТНА РЕАЛІЗАЦІЯ СИСТЕМИ «РОЗУМНИЙ БУДИНОК»	17
2.1 Вибір апаратної платформи.....	17
2.2 Алгоритм роботи головного модуля системи розумний будинок	21
2.3 Алгоритм роботи інформаційних модулів системи «Розумний будинок».	22
2.4 Функціональні модулі системи	23
2.5 Схема електрична принципова керуючого блоку розумного будинку	25
2.6 Схема електрична принципова модулів системи розумний будинок	26
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ «РОЗУМНИЙ БУДИНОК»..	28
ВИСНОВКИ	31
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	32
ДОДАТОК А	33
ДОДАТОК Б.....	34
ДОДАТОК В.....	35
ДОДАТОК Г	38

ВСТУП

Тема магістерської роботи «Автоматизована система керування датчиками на основі мікроконтролерів за технологією «Розумний будинок»» є актуальною в умовах розвитку сучасних інформаційних технологій та автоматизації. Концепція «Розумного будинку» дозволяє значно підвищити комфорт, безпеку та енергоефективність житлових і комерційних приміщень, використовуючи автоматизовані системи управління на основі датчиків і мікроконтролерів. Завдяки цьому дослідження у цій сфері сприяють розв'язанню важливих завдань енергоефективності, оптимізації ресурсів та створення зручних умов для проживання та роботи.

Мета роботи полягала в аналізі автоматизованої системи керування датчиками, що базується на використанні мікроконтролерів для реалізації технології «Розумний будинок», з урахуванням її функціональних, енергетичних і практичних характеристик.

Для досягнення поставленої мети визначено такі завдання: провести аналіз систем автоматизованого керування датчиками для «Розумного будинку»; вибрати оптимальну архітектуру системи на основі мікроконтролерів; розробити апаратну та програмну частини системи; виконати тестування розробленої системи та оцінити її ефективність.

Практична цінність роботи полягає у створенні ефективної автоматизованої системи, яка забезпечує інтеграцію декількох датчиків різного призначення (температури, вологості, освітленості тощо) у єдину мережу під керуванням мікроконтролера, з використанням сучасних технологій IoT (Інтернет речей) та можливості впровадження розробленої системи для автоматизації управління в житлових і офісних приміщеннях. Крім того, напрацювання роботи можуть бути використані для вдосконалення існуючих рішень у сфері «Розумних будинків», а також у навчальних цілях для підготовки фахівців у галузі автоматизації та IoT.

РОЗДІЛ 1 ТЕХНОЛОГІЇ «РОЗУМНИЙ БУДИНОК»

1.1 Промислові системи «Розумний будинок»

1.1.1 Ajax

Система автоматизації розумного будинку від українського виробника Ajax (рис. 1.1) забезпечує високий рівень комфорту та зручності в управлінні життєдіяльністю будівель і гарантує безпеку житла. Ajax активно контролює периметр будівель, запобігає крадіжкам і виявляє можливі загрози електрики, пожежі, газу, може виявляти інші потенційні загрози для будинку. Важливою особливістю системи є підтримка українського та російського інтерфейсів, що робить її доступною для широкого кола користувачів.



Рисунок 1.1 – Система розумного будинку Ajax

Ajax працює на базі Jeweller - захищеного, зашифрованого двостороннього бездротового зв'язку, розробленого власниками компанії. Ця технологія гарантує стабільний зв'язок між елементами системи. Система також повністю незалежна від електромережі завдяки резервному джерелу живлення (хабу), що підвищує надійність у разі відключення електроенергії. Усі пристрої Ajax мають стильний дизайн і гармонійно вписуються в будь-який інтер'єр.

Переваги та недоліки цієї системи наведені в таблиці 1.1.

Таблиця 1.1 – Переваги та недоліки системи Ajax

Переваги	Недоліки
Простий монтаж без спеціальних навичок	Залежність від центрального контролера (Hub)
Бездротовий канал зв'язку між елементами	Відсутність власної камери відеоспостереження
Велика зона дії сигналу (до 2000 м)	Управління тільки через мобільний додаток
Захист від зняття датчиків	
Можливість надання доступу іншим користувачам	
Автономна робота хаба до 16 годин	
Різноманітність способів інформування	
Розумна розетка, що показує витрату електроенергії	
Підключення до 100 пристроїв	
Наявність тривожної кнопки на пульті	
Демократична ціна комплекту (від 200\$)	

Системи розумної домашньої автоматизації Ajax вважаються одними з найкращих на ринку. Вони багатofункціональні, надійні та прості у використанні. Високоякісні протиугінні функції, стильний дизайн і зручний інтерфейс роблять цю систему привабливою для широкої аудиторії. [6]

1.1.2 Fibaro

Fibaro (рис. 1.3) – це система автоматизації та безпеки, розроблена в Польщі та зареєстрована як торгова марка в США. Зверніть увагу, що підтримка російської мови в інтерфейсі обмежена і може бути складною для користувачів, для яких англійська мова не є рідною.



Рисунок 1.3 – Система розумного будинку Fibaro

Fibaro пропонує професійне рішення для домашньої автоматизації та безпеки з широким спектром функцій. Однак, у порівнянні з іншими подібними системами, воно вимагає спеціальної установки і налаштування, що може бути складно для пересічного користувача.

Переваги та недоліки системи розумного будинку від Fibaro наведено у таблиці 1.3.

Таблиця 1.3 – Переваги та недоліки системи Fibaro

Переваги				Недоліки		
Великий асортимент пристроїв	датчиків	і		Висока вартість обладнання (починається від 600\$)		

Наявність відеокамери для спостереження	Необхідність професійного встановлення та налаштування
Широкі можливості для створення автоматизованих сценаріїв	Потрібне підключення центрального контролера Fibaro Home Center до інтернету через LAN-кабель
Сповідення можуть надходити на кілька телефонів	Неможливість роботи без центрального хаба
Використання протоколу Z-Wave для взаємодії з іншими пристроями	Відсутність резервного живлення для хаба
Датчик протікання обладнаний сиреною	Обмежена дальність сигналу (до 50 м без перешкод)
Розумна розетка, що контролює витрати електроенергії	Затримка в отриманні Push-повідомлень
Можливість голосового управління через Google, але тільки англійською	Необхідність встановлення програмного забезпечення на ПК, а також обмежена функціональність мобільного додатку

Системи розумного будинку Fibaro пропонують широкий спектр функцій завдяки численним датчикам, які контролюють приміщення і автоматизують управління пристроями. Однак дороге обладнання, необхідність спеціалізованого монтажу та обмеження інтерфейсу можуть стати бар'єром для деяких користувачів. Найкраще використовувати систему за допомогою експерта, який забезпечить правильне встановлення всіх компонентів. [6]

1.1.3 Xiaomi

Xiaomi (рис. 1.5) – китайський виробник, посідає п'яте місце в рейтингу систем розумного будинку. Система має англійський та китайський інтерфейси, але не має

української чи російської версій. Це може викликати труднощі під час встановлення та налаштування для користувачів, які не знають цих мов.

Система Xiaomi пропонує доступне рішення для домашньої автоматизації, яке дозволяє користувачам легко керувати різними пристроями та приладами.



Рисунок 1.5 – Система розумного будинку Xiaomi

Переваги та недоліки системи розумного будинку від Xiaomi наведені в таблиці 1.5.

Таблиця 1.5 – Переваги та недоліки системи Xiaomi

Переваги	Недоліки
Повна автономність роботи пристроїв	Обмежена зона дії сигналу (до 10 м)
Можливість розширення системи з додатковими пристроями	Невеликий набір сенсорів і виконавчих пристроїв у базовому комплекті
Наявність вбудованої камери для відеоспостереження	Кожен датчик вимагає окремого монтажу
Використання бездротового протоколу ZigBee для зв'язку	Відсутність резервного живлення для хаба

Переваги	Недоліки
Зручне управління через смартфон за допомогою Wi-Fi	
Можливість налаштування сценаріїв	
Стильний і компактний дизайн	
Доступна ціна базового комплекту (приблизно 90\$)	

Система розумного будинку Xiaomi – ідеальний вибір для тих, хто хоче почати автоматизувати свій житловий простір. Вона пропонує можливість інтеграції з іншими датчиками та пристроями, в тому числі сторонніх виробників. Кожен компонент може працювати незалежно або як частина єдиної системи, створюючи функціональну платформу для моніторингу безпеки і комфорту вашого будинку. Невисока ціна Xiaomi робить її вдалим вибором для тих, хто хоче познайомитися з концепцією розумного будинку. [6]

1.2 Аналіз вимог до автоматизації в житлових приміщеннях

Автоматизація будівель спрямована на підвищення комфорту, безпеки та енергоефективності. Сучасні системи «розумного будинку» дозволяють об'єднувати різні пристрої в єдину мережу, забезпечуючи моніторинг і автоматизацію за допомогою датчиків, мікроконтролерів і систем обробки даних.

Одним із ключових аспектів автоматизації є безпека. Камери спостереження забезпечують контроль території та запис подій у реальному часі, дозволяючи мешканцям бути в курсі ситуації. Датчики руху та відкривання дверей і вікон, такі як HC-SR501 і AM312, використовуються в системах сигналізації для виявлення можливих вторгнень. Крім того, детектори диму і газу, наприклад MQ-2, MQ-7 і MQ-135, дають змогу своєчасно реагувати на загрози, пов'язані з пожежею або витоком газу.

Ще одним важливим напрямком автоматизації є енергоефективність. Датчики температури та вологості, такі як DHT11, DHT22 і BME280, допомагають підтримувати комфортне середовище в приміщенні та оптимізувати роботу опалювальних і вентиляційних систем. Це дозволяє уникнути надмірного використання енергії, зменшуючи витрати на її споживання.

Для реалізації таких систем широко використовуються мікроконтролери. Arduino є простим і доступним варіантом для базових проєктів, тоді як ESP8266 і ESP32, оснащені підтримкою Wi-Fi та Bluetooth, забезпечують інтеграцію в середовище Інтернету речей (IoT), що дозволяє створювати складніші системи з централізованим управлінням.

Зручний інтерфейс користувача також відіграє важливу роль в автоматизації. Керування пристроями через смартфони або голосові команди стає стандартом завдяки підтримці бездротових протоколів, таких як Wi-Fi і Bluetooth. Це дозволяє користувачам легко налаштовувати систему, отримувати сповіщення та реагувати на події, що робить автоматизацію доступною та комфортною у використанні.

1.3 Принцип роботи мікроконтролерів

Мікроконтролери — це компактні комп'ютери на одному кристалі, призначені для виконання конкретних завдань у вбудованих системах, зокрема в технологіях «розумного дому». Їхня структура включає центральний процесор (для арифметичних і логічних операцій), пам'ять (оперативну, кеш і ПЗП), інтерфейсний блок (для взаємодії з периферійними пристроями), а також допоміжні схеми (тактові генератори, сторожові таймери тощо) (рис. 2.1). Взаємодія між компонентами забезпечується внутрішньою шиною управління, що складається з шини даних, адресної шини та шини управління. Тактовий генератор синхронізує роботу всіх блоків, забезпечуючи стабільну й узгоджену роботу системи. Завдяки інтеграції з периферійними пристроями, такими як датчики та таймери, мікроконтролери можуть виконувати широкий спектр функцій, включаючи збір даних, обробку сигналів і керування пристроями.

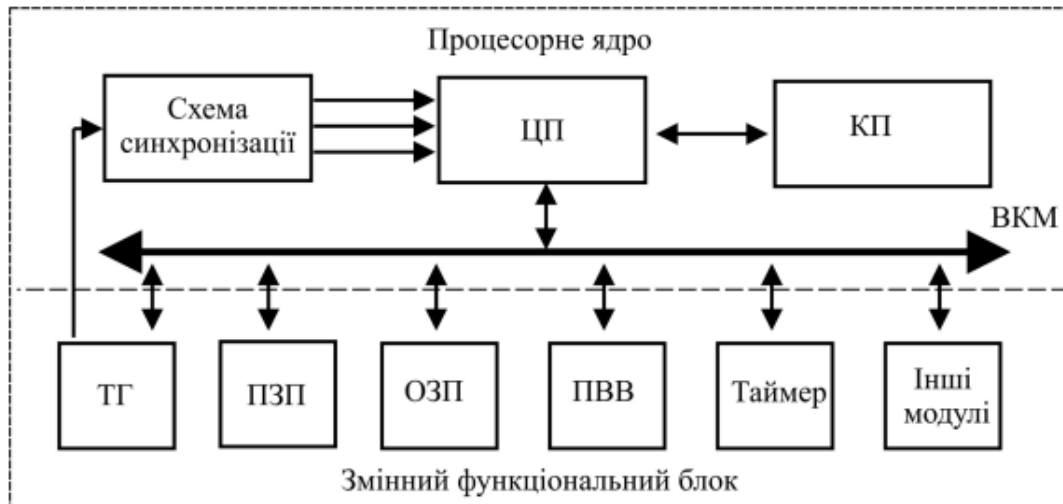


Рисунок 2.1 – Структурна схема сучасного мікроконтролера [10]

Гнучкість і функціональність мікроконтролерів роблять їх важливими компонентами сучасних систем керування та автоматизації, забезпечуючи ефективність і надійність роботи в різних галузях.

1.4 Принцип роботи датчиків

Датчики відіграють ключову роль у забезпеченні автоматизації та контролю в системах розумного будинку. Вони дозволяють моніторити стан середовища, реагувати на зміни та забезпечувати комфорт, безпеку й енергоефективність.

Датчики температури і вологості широко застосовуються для створення оптимального мікроклімату. Їх чутливі елементи змінюють електричні характеристики залежно від параметрів навколишнього середовища. Сигнал, згенерований при зміні температури або вологості, надходить до мікроконтролера, який обробляє інформацію та регулює системи опалення, вентиляції й кондиціонування повітря.

Детектори диму і газу є важливими елементами для забезпечення безпеки. Вони реагують на наявність частинок диму або газів, таких як чадний газ (CO) чи метан (CH₄). Принцип їх дії базується на зміні електричного опору чутливого

елемента. Коли концентрація диму чи газу перевищує безпечний рівень, датчик активує тривогу або надсилає сигнал системі управління.

Датчики руху, як-от пасивні інфрачервоні (PIR), виявляють інфрачервоне випромінювання, яке випромінюють люди або тварини. Вони реагують на зміну інфрачервоного фону, генеруючи сигнал для активації освітлення, охоронних систем або інших автоматичних пристроїв. Їх часто використовують для економії енергії та підвищення безпеки, наприклад, у коридорах чи на входах.

Датчики освітленості вимірюють рівень природного чи штучного світла в приміщенні. Їх чутливі елементи змінюють електричний опір залежно від інтенсивності світла. Це дозволяє регулювати яскравість освітлення або автоматично вмикати світло при недостатньому рівні освітленості, забезпечуючи комфорт і економію енергії.

Датчики відкриття дверей і вікон використовують магнітні контакти для визначення їх стану. Коли двері чи вікно зачинені, контакти замкнені, а сигнал не генерується. При відкритті контакти розмикаються, і сигнал надходить до системи, яка може активувати сигналізацію або надіслати повідомлення власнику.

Детектори витоку води працюють на основі зміни електричних властивостей при контакті води з чутливими електродами. При виявленні води генерується сигнал тривоги або надсилається сповіщення, що дозволяє швидко реагувати на протікання, зменшуючи ризик пошкоджень.

Усі ці датчики взаємодіють із мікроконтролерами та автоматизованими системами управління, забезпечуючи зручність, безпеку та раціональне використання ресурсів у сучасних розумних будинках.

РОЗДІЛ 2 АПАРАТНА РЕАЛІЗАЦІЯ СИСТЕМИ «РОЗУМНИЙ БУДИНОК»

2.1 Вибір апаратної платформи

Основними критеріями вибору елементної бази для реалізації бездротових систем є такі характеристики, як енергоспоживання, параметри радіочастот (чутливість приймача, потужність передавача), обсяг пам'яті, доступність додатків і рівень захисту переданих даних.

Сьогодні виробники бездротових компонентів пропонують три основних підходи до проектування бездротових вузлів.

Перший підхід передбачає використання приймача разом із керуючим мікроконтролером. У цьому варіанті мікроконтролер забезпечує функціонування мережевого стека і вирішення прикладних задач. Приймач з'єднується з мікроконтролером через інтерфейси SPI або UART. Такий метод дозволяє комбінувати бездротові модулі і мікроконтролери для досягнення оптимальних характеристик за енергоспоживанням, габаритами, функціоналом, доступністю програмних бібліотек і вартістю. Крім того, цей підхід допускає використання компонентів від різних виробників, хоча зазвичай вистачає елементної бази, запропонованої одним виробником.

Другий підхід базується на застосуванні систем-на-кристалі (СНК), які об'єднують у одному корпусі як приймач, так і мікроконтролер. У цьому випадку ядро процесора виконує функції стека протоколів і прикладних задач. Такий варіант є більш компактним і вимагає меншої кількості компонентів, що спрощує проектування та зменшує розміри друкованих плат.

Обидва підходи вимагають ретельного тестування як стека протоколів, так і прикладних задач під час налаштування програмного забезпечення. Неправильна робота програми може спричинити збій у виконанні процедур стека протоколів, що призведе до виходу вузла з мережі. Це, у свою чергу, може негативно вплинути на функціонування всієї розподіленої системи. Помилки в роботі стека протоколів

також можуть стати причиною зависання мікроконтролера або припинення виконання програми, внаслідок чого вузол перестає виконувати свої функції.

Сучасні багатофункціональні однокристальні бездротові рішення дають розробникам можливість створювати пристрої, які відповідають найвищим вимогам споживачів, забезпечуючи гнучкість у застосуванні, зменшуючи кількість необхідних компонентів і зменшуючи розміри пристроїв. Основними виробниками таких компонентів є компанії Atmel, Microchip, Nordic, STMicroelectronics, Texas Instruments та інші. Розробники постійно вдосконалюють архітектуру СНК, оптимізуючи параметри приймачів, впроваджуючи режими енергозбереження і покращуючи програмне забезпечення через пропонування бібліотек, протокольних стеків і інструментів для розробки.

Мікроконтролери в таких системах можуть відрізнитися за розрядністю (від 8 до 32 біт) і максимальною робочою частотою. Часто використовуються процесорні ядра, такі як 8-бітові архітектури x51 або 32-бітові ядра ARM. Деякі компанії пропонують готові рішення у вигляді бездротових систем збору даних або інтегральних сенсорів фізичних величин.

У цьому проєкті для функцій головного модуля системи обрано плату Arduino Mega2560 R3, яка побудована на базі мікроконтролера ATmega2560 від компанії ATMEL. Цей контролер належить до сімейства AVR і оснащений 256 КБ флеш-пам'яті для зберігання прошивки, 8 КБ оперативної пам'яті та 4 КБ енергонезалежної пам'яті EEPROM для даних. Основні характеристики мікроконтролера ATmega2560 наведені в таблиці 2.1.

Таблиця 2.1 – Технічні характеристики мікроконтролера ATmega2560

Параметр	Значення
Ядро мікропроцесора	AVR 8-біт
Flash-пам'ять	256 кБайт
EEPROM-пам'ять	4 кБайт
RAM-пам'ять	8 кБайт

Параметр	Значення
Кількість каналів АЦП	16
Розширення АЦП	10
Кількість каналів ШІМ	12
Розширення ШІМ	16
Сторожовий таймер	1
Годинник реального часу	1
Кількість інтерфейсів UART	4
Напруга живлення ядра	4,5...5,5 В
Напруга живлення периферії	4,5...5,5 В
Робоча температура	-40...85 °С

Для реалізації інформаційних модулів системи бездротового зв'язку в рамках розумного будинку пропонується використати плату Arduino Nano R3. Ця плата базується на мікропроцесорі виробництва компанії ATMEL та оснащена 8-бітним мікроконтролером серії AVR, а саме ATmega328. Arduino Nano R3 обрано як платформу для прототипування та тестування, завдяки її функціональності та сумісності з необхідними компонентами. Основні технічні характеристики мікроконтролера ATmega328 наведено у таблиці 2.2.

Таблиця 2.2 – Характеристики мікроконтролера ATmega328

Параметр	Значення
Робоча напруга	5 В
Вхідна напруга (рекомендована)	7-12 В
Вхідна напруга (гранична)	6-20 В
Цифрові входи/виходи	14 (6 з яких можуть використовуватися як виходи ШІМ)
Аналогові входи	6
Постійний струм через вхід/вихід	40 мА

Параметр	Значення
Постійний струм для виведення 3.3 В	50 мА
Флеш-пам'ять	32 Кб (АТmega328) з яких 0.5 Кб використовуються для завантажувача
ОЗУ	2 Кб (АТmega328)
EEPROM	1 Кб (АТmega328)
Тактова частота	16 Гц

Важливим компонентом апаратного забезпечення є модуль бездротового зв'язку nRF24L01, який використовується для створення радіомережі на частоті 2,4 ГГц. Цей модуль, побудований на основі мікросхеми nRF24L01+, підтримує передачу даних зі швидкістю 250 Кбіт/с, 1 Мбіт/с або 2 Мбіт/с. Крім того, модуль може функціонувати на 126 окремих каналах, що дозволяє одночасно взаємодіяти кільком пристроям без взаємних перешкод.

Для роботи з мікроконтролером модуль використовує інтерфейс SPI, завдяки чому підключення до мікроконтролера АТmega328 виконується через SPI-порти, доступні на платі Arduino.

Призначення виводів радіомодуля nRF24L01+ наведені в таблиці 2.3.

Таблиця 2.3 – Призначення виводів на nRF24L01

Найменування виводу	Призначення
SCK	Трагований вивід (синхронізація)
MOSI/MI	Вхід даних
MOSI/MO	Вихід даних
CE/SS	Вибір на шині SPI з декількох пристроїв
SCN	Вибір режиму прийому/передача, фактично – CE
IRQ	Вихід переривання, найчастіше не використовується проте необхідний для негайної реакції мікроконтролера при прийомі нового пакета

Найменування виводу	Призначення
GND	Земля, маса
Vcc	Живлення модуля 3.3 В

2.2 Алгоритм роботи головного модуля системи розумний будинок

Запропонований алгоритм розроблений для тестової програми, що реалізує метод адаптивної селекції каналів зв'язку (КЗ) для системи «Розумний будинок». Основною його задачею є встановлення з'єднання з інформаційними модулями, організація ефективного обміну інформацією між ними та виведення отриманих даних або аварійних повідомлень на LCD-дисплей головного модуля «Блок управління».

Робота алгоритму починається з підключення та ініціалізації необхідних бібліотек, оголошення змінних, а також налаштування портів вводу/виводу і зовнішніх модулів системи. Після цього виконується сканування частотного діапазону для визначення доступних частот і створення їхнього переліку. На основі отриманих даних встановлюється базова частота для подальшого прослуховування лінії.

Якщо виявляється спроба з'єднання з новим модулем системи, виконується підключення до нього, отримання даних конфігурації, таких як ідентифікатор модуля та його адреса, а також передача переліку вільних частот і базової частоти. У разі відсутності спроб з'єднання алгоритм продовжує моніторинг лінії на предмет появи подій.

Стан мережі відображається на LCD-дисплеї разом із меню вибору дій для користувача. Алгоритм реагує на команди, введені через клавіатуру, і виконує їх обробку. Якщо команди не надходять, система переходить у режим очікування на встановлений час, після чого відновлюється моніторинг лінії та перевірка з'єднання з модулями системи.

При перевірці з'єднання у разі невдачі система виводить відповідне повідомлення на LCD-дисплей і повторює спробу встановлення радіозв'язку до

трьох разів. У разі успіху надсилається запит на отримання інформації від підключених модулів. Отримані дані відображаються на дисплеї, а у випадку отримання попереджувального повідомлення система негайно виводить відповідні показники на екран.

Після завершення кожного циклу система знову очікує встановлений час і повертається до стану моніторингу мережі, забезпечуючи постійний контроль за станом системи та своєчасне реагування на події чи запити користувача.

2.3 Алгоритм роботи інформаційних модулів системи «Розумний будинок»

Алгоритм роботи інформаційних модулів, таких як «Датчик руху» (Motion sensor), «Датчик газу» (Smoke sensor) та «Датчик затоплення» (Flood sensor), передбачає послідовне виконання низки дій, спрямованих на забезпечення стабільного збору, аналізу та передачі даних до головного модуля системи.

На початковому етапі виконання програми здійснюється підключення та ініціалізація необхідних бібліотек, що забезпечують функціонування датчиків та їх взаємодію із системою. Після цього оголошуються змінні, які зберігатимуть конфігураційні параметри модулів та оперативні дані.

Далі виконується функція сканування частотного діапазону, що дозволяє визначити оптимальні параметри для радіозв'язку. Паралельно ініціалізуються зовнішні модулі, необхідні для роботи датчиків. Наступним важливим етапом є встановлення радіозв'язку із головним модулем. У разі невдалої спроби модуль прослуховує лінію, очікує певний проміжок часу та повторює спробу підключення.

Після успішного встановлення зв'язку здійснюється передача конфігураційних даних, таких як ідентифікатор модуля та його адреса. Головний модуль передає інформацію про доступні частоти, включно з базовою частотою. На основі цих даних виконується функція стрибкоподібної зміни частоти передачі, що підвищує стабільність та безпеку зв'язку.

На наступному етапі здійснюється зчитування даних з інформаційних датчиків. Зібрані дані перевіряються на відповідність встановленим критичним рівням показників. Якщо значення перевищують допустиму норму, модуль відправляє попереджувальне повідомлення до головного модуля.

Програма продовжує прослуховувати лінію радіозв'язку, очікуючи команди від головного модуля. У разі надходження запиту на передачу даних модуль передає актуальну інформацію. Якщо запит відсутній, програма переходить до наступного циклу. Після завершення всіх поточних операцій модуль очікує певний встановлений час і повторює зчитування даних, забезпечуючи безперервний моніторинг ситуації.

Ця послідовність дій гарантує стабільну роботу модулів, а також своєчасне виявлення критичних ситуацій та їх опрацювання у системі «Розумного будинку».

Блок-схема алгоритму роботи програми головного модуля системи РБ «Блок управління» представлена в додатку А.

2.4 Функціональні модулі системи

Розроблена система складається з чотирьох бездротових функціональних модулів, які взаємодіють між собою (рис. 2.1). До її складу входить основний модуль – «Блок управління» та три модулі збору інформації: «Датчик руху», «Датчик газу» і «Датчик затоплення».

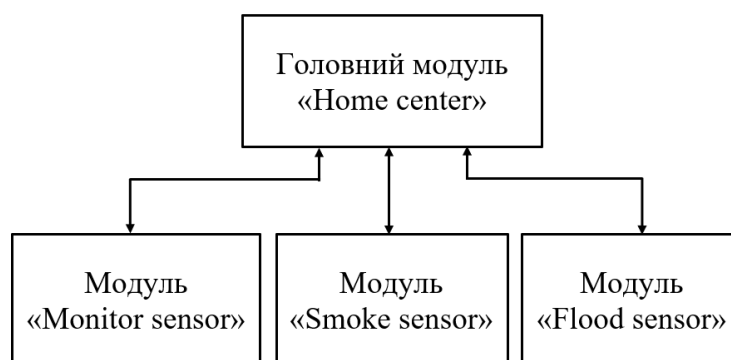


Рисунок 2.1 – Схема радіозв'язку між модулями системи «Розумний будинок»

Концепція «Розумного будинку» ґрунтується на використанні радіозв'язку між усіма компонентами системи. Такий зв'язок забезпечує передачу як команд керування, так і даних від інформаційних модулів до центрального блоку. Кожен модуль у системі має унікальний ідентифікаційний номер і функціонує як окремий об'єкт. У даній реалізації передбачається наявність єдиного керуючого елемента – «Блоку управління» (або «Home Center»), який координує роботу інших модулів та отримує від них дані.

Система включає три інформаційні модулі: «Датчик руху» («Motion Sensor»), «Датчик газу» («Smoke Sensor») і «Датчик затоплення» («Flood Sensor»). Основне завдання цих модулів полягає у зборі інформації з відповідних датчиків, передачі цих даних по радіозв'язку, а також у миттєвому реагуванні на перевищення допустимих рівнів параметрів, таких як рух, концентрація газу чи рівень води. Додатково можуть бути інтегровані датчики температури, вологості чи освітленості для контролю відповідних параметрів згідно із встановленими нормами (додаток Б).

Головний модуль «Блок управління» виконує такі функції: встановлює зв'язок із периферійними модулями для отримання даних або передачі керуючих команд; відображає стан системи та її робочий режим на дисплеї; керує побутовими пристроями через релейний модуль; забезпечує можливість ручного управління через клавіатуру, розташовану на самому пристрої.

Периферійні модулі мають бути максимально простими у конструкції, компактними за розмірами, енергоєфективними та виконувати лише ті функції, які визначені головним модулем відповідно до обраного режиму роботи.

Функції інформаційних модулів передбачають:

- встановлення радіозв'язку з головним модулем;
- зчитування показників із підключених датчиків;
- перевірку даних на відповідність встановленим нормам;
- обробку команд, отриманих від головного модуля;
- передачу інформації про стан датчиків у відповідь на запити головного модуля;

– автоматичне відправлення аварійних сигналів у разі перевищення допустимих значень параметрів.

Уся система орієнтована на забезпечення високої надійності та оперативності в обміні даними, що є ключовою вимогою для технології «Розумний будинок».

2.5 Схема електрична принципова керуючого блоку розумного будинку

Розробка електричної принципової схеми є одним із ключових етапів проектування, який визначає функціональну структуру, масштаби системи та її окремих компонентів. У рамках цієї роботи створено основний модуль системи — «Блок управління». Основними складовими електричної схеми модуля є елементи, подані в таблиці 2.4.

Таблиця 2.4 – Елементи електричної принципової схеми керуючого блоку

Позначення	Назва компонента
DM1	Arduino Mega2560
DM2	HCM1205X
DM3	nRF24L01+
DM4	Bluetooth модуль
DM5	DS1302
DM6	Модуль реле 1
DM7	Модуль реле 2
KM1	Матрична клавіатура 4*4
DD1	Модуль I2C на базі мікросхеми PCF8574
HG1	Модуль LCD-дисплею

Схема електрична принципова головного модуля «Блок управління» зображена на додатку В (рис. В.1) .

2.6 Схема електрична принципова модулів системи розумний будинок

Розроблена автоматизована система РБ складається з декількох периферійних модулів: «Датчик руху», «Датчик газу» та «Датчик затоплення», кожен з яких має модульну структуру з можливістю взаємозаміни. Основною відмінністю між модулями є використання різних основних датчиків, які і визначають їхні назви. Периферійні компоненти, такі як мікроконтролер і радіомодуль, є спільними для всіх схем.

Принципові електричні схеми цих модулів детально відображають структуру їхніх пристроїв, основні блоки та вузли з вказанням зв'язків між ними. Відповідно до таких схем можна зрозуміти принцип роботи кожного блоку, його підключення та взаємодію частин системи. Повні електричні принципові схеми модулів системи РБ наведені в додатку В (рис. В.2-В.4) цієї дипломної роботи. Основні блоки електричних принципових схем модулів системи РБ зазначено в таблиці 2.5.

Таблиця 2.5 – Основні блоки електричних принципових схем модулів системи

Блоки	«Датчик руху» (Motion Sensor)	«Датчик газу» (Smoke Sensor)	«Датчик затоплення» (Flood Sensor)
DM1	Arduino Nano, призначений для управління роботою модуля	Arduino Nano, призначений для управління роботою модуля	
DM2	П'єзоелектричний випромінювач моделі HCM1205X, розроблений для забезпечення звукових сигналів		
DM3	Модуль бездротового зв'язку nRF24L01+ використовується для створення радіомережі, що працює на частоті 2,4 ГГц		
DAT1	Датчик освітленості (фоторезистор) призначений для визначення рівня освітлення в приміщеннях		

Блоки	«Датчик руху» (Motion Sensor)	«Датчик газу» (Smoke Sensor)	«Датчик затоплення» (Flood Sensor)
DAT2	Датчик вологості та температури (DHT-11) забезпечує вимірювання показників вологості та температури у внутрішньому середовищі		
DAT3	Датчик руху (PIR) використовується для виявлення руху людей у приміщенні	Датчик газу (MQ-5) призначений для вимірювання концентрації природного газу, пропану, ізобутану, коксового газу, алкоголю та диму	Датчик затоплення (Water sensor) допомагає визначити наявність витоку води в приміщенні

РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ «РОЗУМНИЙ БУДИНОК»

Опис роботи програмного забезпечення основного модуля системи «розумний будинок» представлено на рисунках 3.1–3.8.

На рисунку 3.1 зображено код, який відповідає за підключення необхідних бібліотек для коректного функціонування системи.

```
#include <Wire.h>
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>
#include <LiquidCrystal_I2C.h> //LCD-дисплей
#include <Keypad.h>           //Клавіатура
#include <RTC.h>               //Модуль реального часу
```

Рисунок 3.1 – Код підключення бібліотек

Код для оголошення виводів основного модуля наведено на рисунку 3.2.

```
pinMode (led1Pin, OUTPUT);
pinMode (led2Pin, OUTPUT);
pinMode (BuzzPin, OUTPUT);
pinMode (RALAY1, OUTPUT);
pinMode (RALAY2, OUTPUT);
```

Рисунок 3.2 – Код оголошення виводів

Ініціалізація модуля реального часу описана в коді, представленому на рисунку 3.3.

```
time.begin (RTC_DS1302, 8, 10, 9);
```

Рисунок 3.3 – Код ініціалізації модуля реального часу

Код, що забезпечує точний вивід поточної дати та часу на LCD-дисплей, зображено на рисунку 3.4.

```
Serial.println(time.gettime("d-m-Y, H:i:s, D")); //виводимо час
    lcd.print(time.gettime("d-m-Y, H:i:s, D"));
    delay(2000);
    lcd.clear();
    Serial.println();
```

Рисунок 3.4 – Код для відображення даних на дисплеї

На рисунку 3.5 наведено програмний код для налаштування роботи радіомодуля nRF24L01+.

```
#define CE_PIN    48
#define CSN_PIN   48

const uint64_t pipe01 = 0xF0F1F2F3F4LL;
const uint64_t pipe02 = 0xF0F1F2F3F1LL;
const uint64_t pipe03 = 0xE8E8F0F0E1LL;

RF24 radio (CE_PIN, CSN_PIN); //Створюємо радіо - ініціалізуємо CE, CSN
    radio.begin(); //Ініціалізуємо включення радіомодуля
    delay(10);
    radio.setChannel(9);
    radio.setDataRate(RF24_1MBPS);
    radio.setPALevel(RF24_PA_HIGH);
radio.openReadingPipe(1, pipe01);
radio.openReadingPipe(2, pipe02);
radio.openReadingPipe(2, pipe02);
// radio.openReadingPipe(0, pipe01); //або відкриваємо всі "труби разом"
radio.startListening(); //вмикаємо радіо-модуль на прослуховування ефіру
// radio.stopListening(); //зупиняємо прослуховування (потрібно перед початком передачі)
```

Рисунок 3.5 – Код для налаштування радіомодуля nRF24L01+

Код для налаштування LCD-дисплея та виведення привітального повідомлення представлено на рисунку 3.6.

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
for a 16 chars and 2 line display
    lcd.init();
    lcd.init();
        lcd.backlight();
        lcd.print("# Smart Home #");
        delay(1500);
        lcd.clear();
```

Рисунок 3.6 – Код налаштування дисплея та виведення тексту

Передавання даних у системі здійснюється за допомогою коду, зображеного на рисунку 3.7.

```
radio.stopListening();  
radio.openWritingPipe(1, pipe);  
radio.wrire(&stan1, sizeof(stan1));  
radio.startListening();
```

Рисунок 3.7 – Код для передавання даних

На рисунку 3.8 показано код для відображення аварійного повідомлення на екрані.

```
if (data[1] == 1) {  
    Serial.println("Warning = moving");  
    Serial.println("Sensor = ");  
    lcd.setCursor(0, 0);  
    lcd.print("Warning = moving");  
    lcd.setCursor(0, 1);  
    lcd.print("Sensor = Moving ");  
}
```

Рисунок 3.8 – Код для виводу аварійного повідомлення

Повний програмний код, який охоплює функціонал всієї системи «розумний будинок» (включно з головним і інформаційними модулями), а також всі змінні, константи, функції, методи та розрахункові формули, що використовуються для реалізації адаптивного вибору каналу зв'язку, подано в додатку Г.

ВИСНОВКИ

1. Показано, що автоматизована система керування датчиками на основі мікроконтролерів ефективно підтримує функціональність «Розумного будинку». Вона дозволяє інтегрувати різні датчики (температури, вологості, освітленості, руху тощо) у єдину мережу для забезпечення комфорту, безпеки та енергоефективності.

2. Одержано нові дані щодо впровадження мікроконтролерів у системи керування датчиками. Зокрема, використання мікроконтролерів ATmega2560 та ATmega328 забезпечило високу стабільність роботи та спрощення інтеграції сенсорів завдяки застосуванню бездротових модулів.

3. Установлено, що використання алгоритму адаптивного вибору частотного діапазону підвищує надійність бездротового зв'язку в системах «Розумний будинок». Це сприяє мінімізації перешкод і підвищує безперебійність передачі даних.

4. Розроблено та протестовано апаратну і програмну частини автоматизованої системи. Проведені експериментальні дослідження підтвердили її здатність реагувати на критичні ситуації (задимлення, витік води) та оперативно передавати відповідні сповіщення.

5. Практичне значення одержаних результатів полягає у можливості впровадження розробленої системи в житлових і комерційних приміщеннях. Система сприяє оптимізації енергоспоживання, покращенню безпеки та забезпеченню зручних умов проживання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Голубнов А. Н. - "Розумний будинок" своїми руками – 2019.
2. Ніколайчук О. Б. Автоматизована система керування функціями будівлі за технологією «Розумний будинок». - Київ, Україна, 2018. - 2 с. <https://dspace.nau.edu.ua/bitstream/NAU/35583/1/Ніколайчук.pdf> .
3. Типи, принципи роботи «Розумного будинку» [Електронний ресурс] - <https://www.smarthouse.ua/umnyj-dom-plyusy-i-minusy.html> .
4. Supriya S., Charanya R., Madhumitha S. J. A review on smart home automation using IoT. International Conference on Emerging Trends in Information Technology and Engineering. 2020. - P. 1–5.
5. Buchanan M. The smart home manual. 3rd ed. Part of: Home Technology Manuals, 2020. - 74 p.
6. Рейтинг систем «Розумний будинок» по виробникам 2023-2024 року по якості, надійності. Інтернет-магазин ВЕНКОН - магазин техніки та інженерних рішень в Україні / Київ, Львів. URL: <https://vencon.ua/ua/articles/rejting-sistem-umnyy-dom-po-proizvoditelyam> .
7. What is Arduino?. Arduino. URL: <https://www.arduino.cc/>.
8. Знайомимося з модулем ESP8266. HobbyTech. <https://hobbytech.com.ua/>.
9. Мікроконтролер ESP32. IT Master - електроніка та програмування. <https://itmaster.biz.ua/directory/microcontrollers/esp32.html#:~:text=ESP32%20має%20Wi-Fi,%20Bluetooth,криптографічні%20модулі%20AES%20і%20SHA> .
10. Татарчук Д. Д., Діденко Ю. В. Мікропроцесори та мікроконтролери : курс лекцій. Київ : Київ. політехн. ін-т ім. Ігоря Сікорського, 2020. - 238 с.
11. Тимченко О. В., Демченко В. О. Архітектура та застосування технології розумного дому. - Львів, 2023. - С.99–108. <https://kk.uad.lviv.ua/wp-content/uploads/2024/04/1-23-99-108.pdf> .

ДОДАТОК А

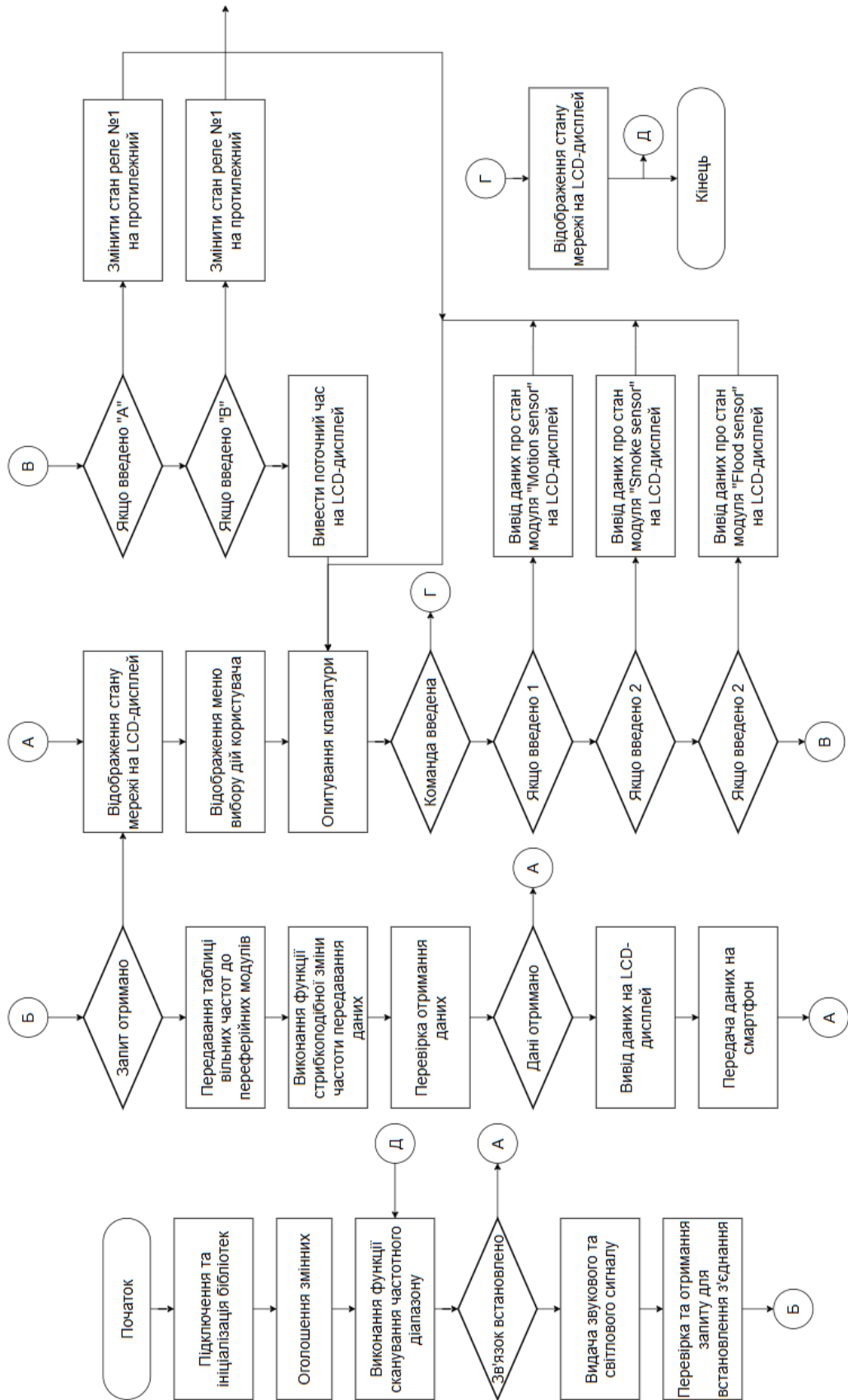


Рисунок А.1 – Блок-схема алгоритму роботи головного модуля

ДОДАТОК Б



Рисунок Б.1 – Структурна схема системи

ДОДАТОК В

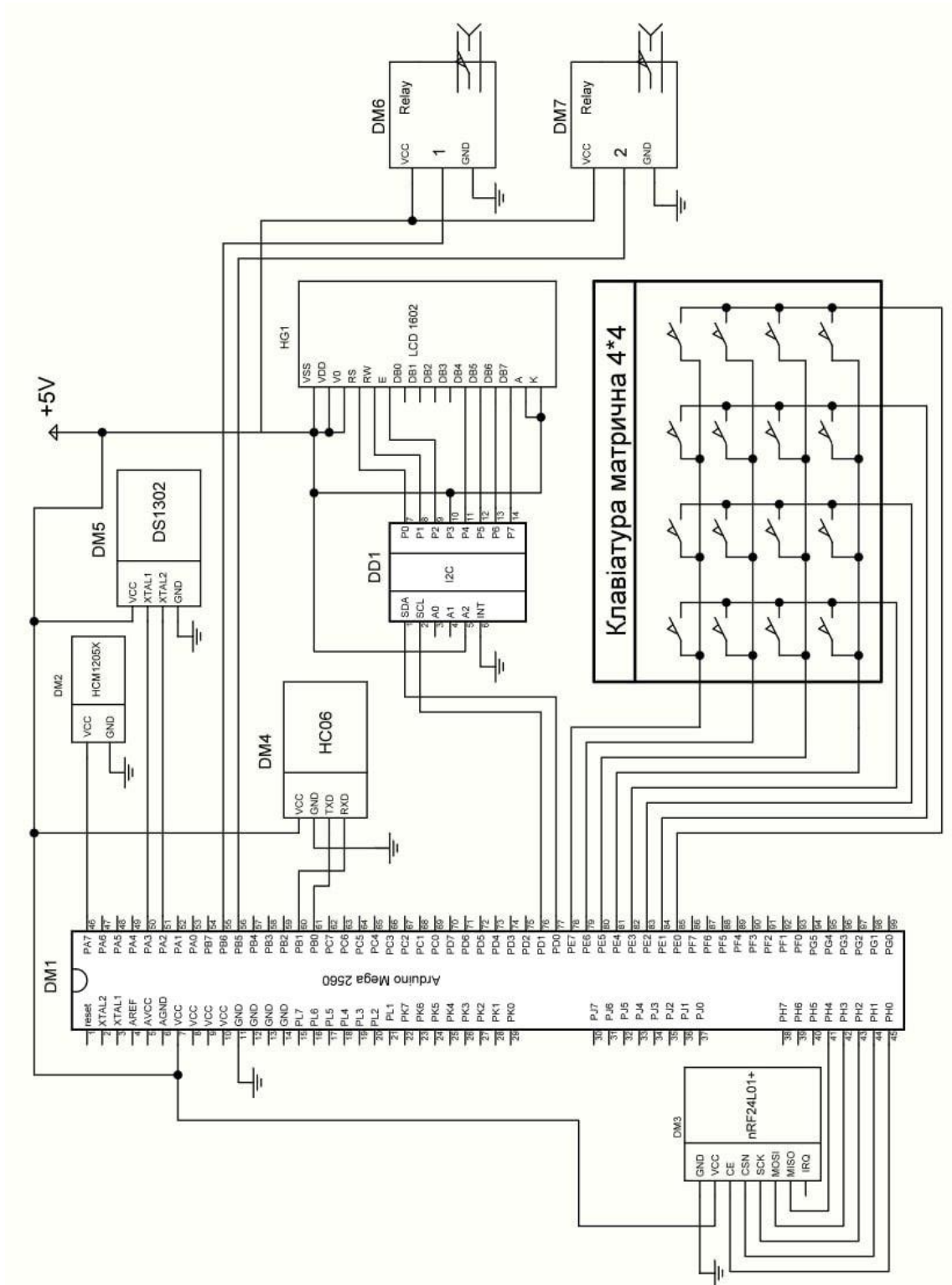


Рисунок В.1 – Схема електрична принципова керуючого модуля

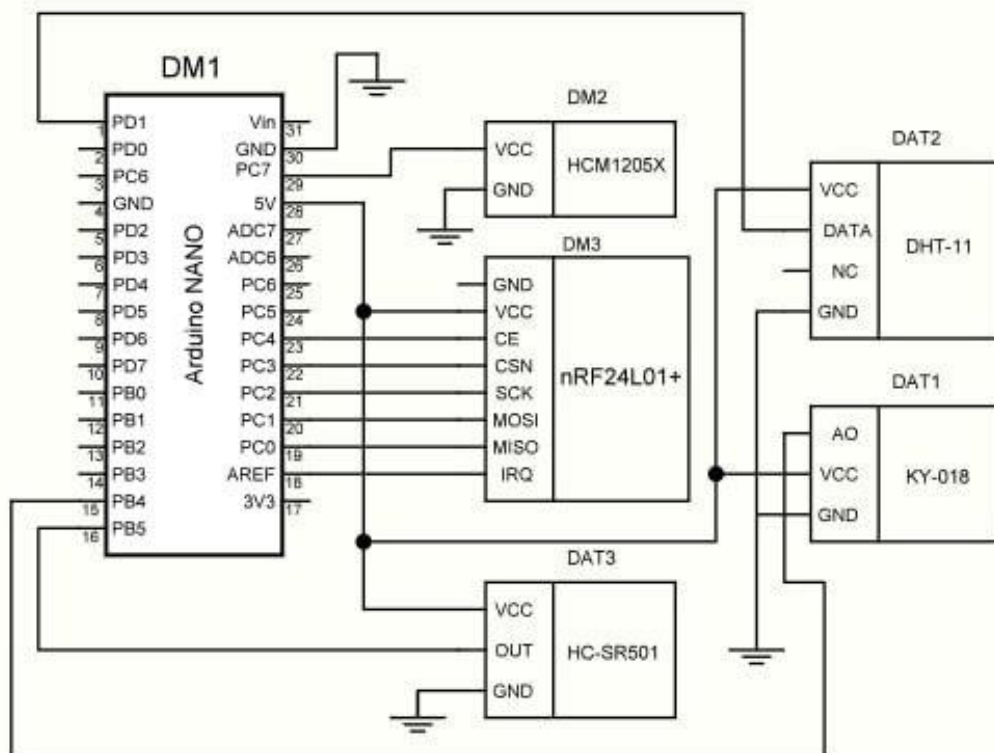


Рисунок В.2 – Схема електрична принципова модуля датчика руху

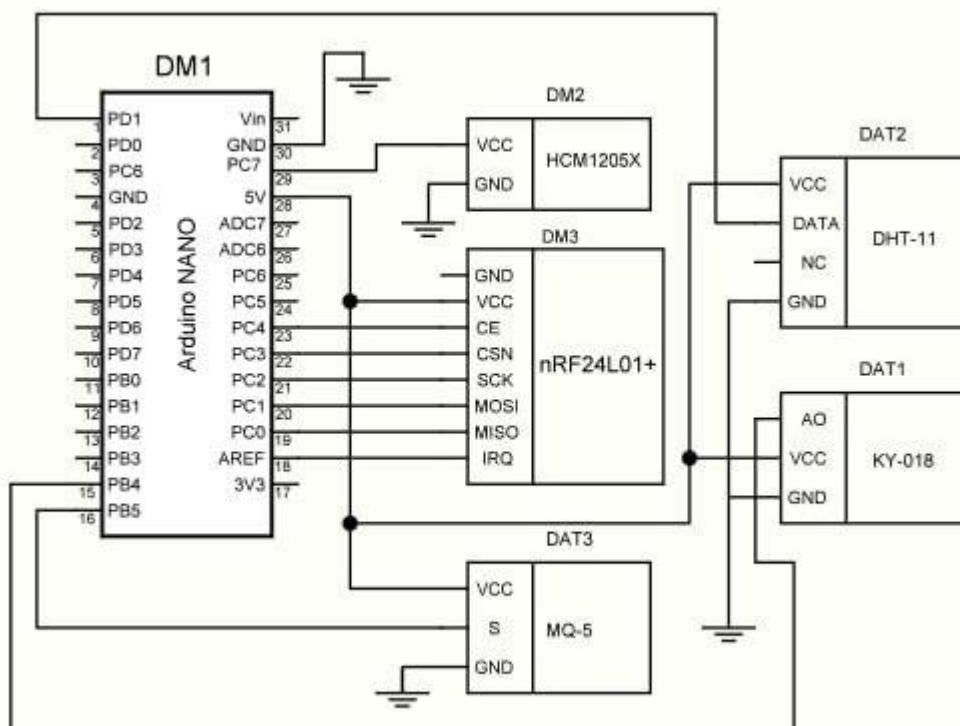


Рисунок В.3 – Схема електрична принципова модуля датчика газу

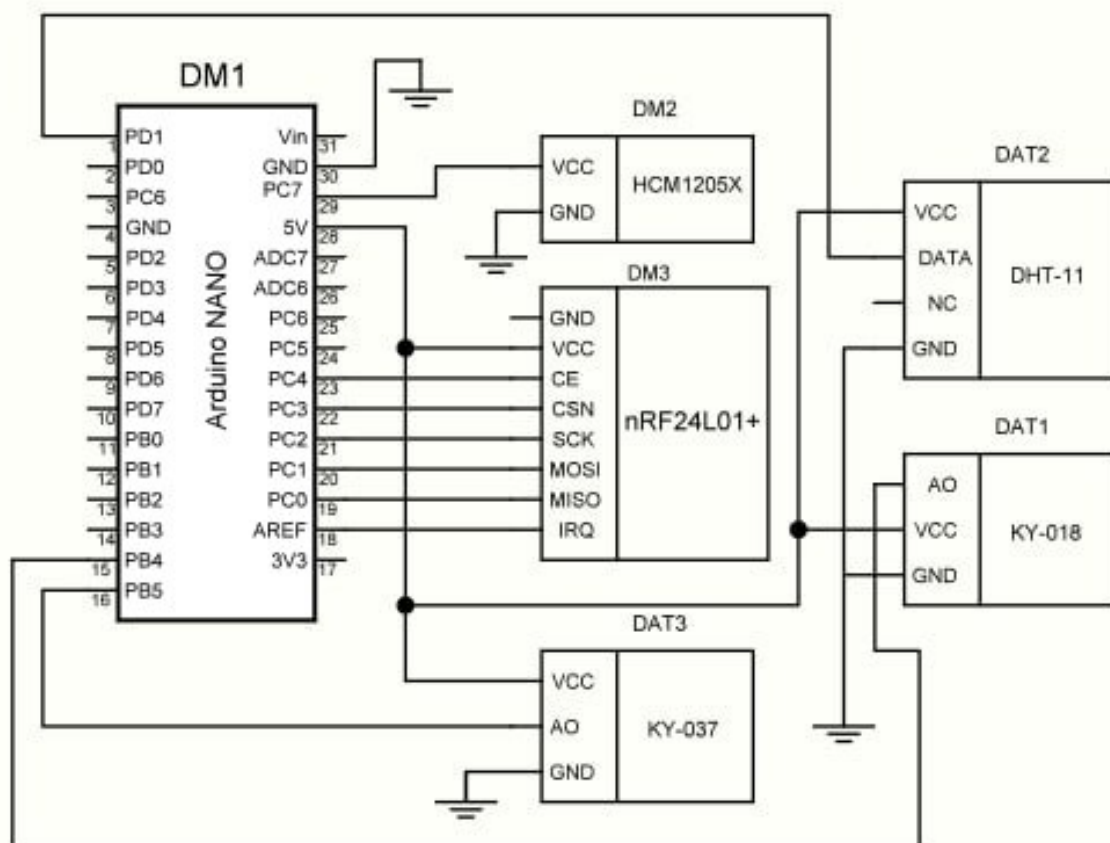


Рисунок В.4 – Схема електрична принципова модуля датчика затоплення

ДОДАТОК Г

Код функції:

```
#include <SPI.h>
#include "nRF24L01.h"
#include "RF24.h"
#include "printf.h"

RF24 radio(7, 8);

const uint8_t totalChannels = 126;
uint8_t channelData[totalChannels];

void setup() {
    Serial.begin(115200);
    printf_begin();
    Serial.println(F("\n\rRF24 Scanner Example"));

    radio.begin();
    radio.setAutoAck(false);
    radio.startListening();
    radio.stopListening();
    radio.printDetails();

    for (uint8_t i = 0; i < totalChannels; i++) {
        Serial.print((i >> 4), HEX);
    }
    Serial.println();

    for (uint8_t i = 0; i < totalChannels; i++) {
        Serial.print((i & 0xF), HEX);
    }
    Serial.println();
}

const int repetitions = 100;

void loop() {
```

```

memset(channelData, 0, sizeof(channelData));

for (int r = 0; r < repetitions; r++) {
    for (uint8_t ch = 0; ch < totalChannels; ch++) {
        radio.setChannel(ch);
        radio.startListening();
        delayMicroseconds(128);
        radio.stopListening();

        if (radio.testCarrier()) {
            channelData[ch]++;
        }
    }
}

for (uint8_t i = 0; i < totalChannels; i++) {
    Serial.print(min(0xF, channelData[i]), HEX);
}
Serial.println();
}

```

Код для RX метод FHHS:

```

#include <SPI.h>
#include "nRF24L01.h"
#include "RF24.h"
#include "printf.h"

RF24 transceiver(6, 7);
const uint64_t channels[2] = { 0xABCDABCD71LL, 0x544d52687CLL };
int timer_interval = 10;
byte timer_count = 0;
byte max_timer_count = 5;
volatile bool timer_flag = false;
unsigned long last_received_time = 0;
bool hopping_enabled = false;
byte hopping_sequence[] = {11, 46, 32, 49, 2, 19, 3, 33, 30, 14, 9, 13,
                           6, 1, 34, 39, 44, 43, 54, 24, 42, 37, 31};
byte sequence_index = 0;

```

```
struct RXData {
    int value1;
    int value2;
    int value3;
    int value4;
    byte value5;
} received_data;

struct TXData {
    int value1;
    int value2;
    int value3;
    int value4;
    int value5;
    int value6;
} transmitted_data;

void setup() {
    Serial.begin(115200);
    printf_begin();

    transceiver.begin();
    transceiver.setPALevel(RF24_PA_MAX);
    transceiver.setRetries(4, 9);
    transceiver.setAutoAck(true);
    transceiver.enableAckPayload();
    transceiver.enableDynamicPayloads();
    transceiver.setDataRate(RF24_250KBPS);
    transceiver.setChannel(hopping_sequence[sequence_index]);
    transceiver.openWritingPipe(channels[0]);
    transceiver.openReadingPipe(1, channels[1]);

    memset(&received_data, 0, sizeof(received_data));
    memset(&transmitted_data, 0, sizeof(transmitted_data));

    transceiver.startListening();
    transceiver.printDetails();

    TCCR1A = 0;
    TCCR1B = (1 << WGM12) | (1 << CS11);
    TCNT1 = 0;
```



```

OCR1A = (timer_interval * 2000) - 1;
TIFR1 |= _BV(OCF1A);
TIMSK1 = _BV(OCIE1A);

last_received_time = millis();
}

void loop() {
  if (timer_flag) {
    timer_flag = false;
    timer_count++;

    if (timer_count > max_timer_count) {
      timer_count = 0;
    }

    if (timer_count == (max_timer_count - 1) && hopping_enabled) {
      sequence_index++;
      if (sequence_index >= sizeof(hopping_sequence)) {
        sequence_index = 0;
      }
      transceiver.setChannel(hopping_sequence[sequence_index]);
      send_ack_payload();
    }
  }

  if ((millis() - last_received_time) > (((sizeof(hopping_sequence) + 5) * 10 *
max_timer_count))) {
    last_received_time = millis();
    hopping_enabled = false;
    sequence_index++;
    if (sequence_index >= sizeof(hopping_sequence)) {
      sequence_index = 0;
    }
    transceiver.setChannel(hopping_sequence[sequence_index]);
  }

  if (transceiver.available()) {
    TCNT1 = 0;
    hopping_enabled = true;
    last_received_time = millis();
  }
}

```

```

    transceiver.read(&received_data, sizeof(received_data));

    Serial.print("Received data: ");
    Serial.print(received_data.value1);
    Serial.print(", ");
    Serial.print(received_data.value2);
    Serial.print(", ");
    Serial.print(received_data.value3);
    Serial.print(", ");
    Serial.print(received_data.value4);
    Serial.print(", ");
    Serial.println(received_data.value5);

    timer_count = 0;
} else {
    transmitted_data.value1++;
    transmitted_data.value2++;
    transmitted_data.value3++;
    transmitted_data.value4++;
    transmitted_data.value5++;
    transmitted_data.value6++;
}
}

void send_ack_payload() {
    transceiver.flush_tx();
    transceiver.writeAckPayload(1, &transmitted_data, sizeof(transmitted_data));
}

ISR(TIMER1_COMPA_vect) {
    timer_flag = true;
}

#include <SPI.h>
#include "nRF24L01.h"
#include "RF24.h"
#include "printf.h"

RF24 radio(6, 7);

const uint64_t pipes[2] = { 0xABCDABCD71LL, 0x544d52687CLL };

```

```
int timerInterval = 10;
byte interruptCounter = 0;
byte maxInterrupts = 5;
volatile boolean timerTriggered = false;

byte frequencyHopping[] = {11, 46, 32, 49, 2, 19, 3, 33, 30, 14, 9, 13, 6, 1, 34, 39, 44,
43, 54, 24, 42, 37, 31};
byte frequencyIndex = 0;

typedef struct {
    int value1;
    int value2;
    int value3;
    int value4;
    byte value5;
} TXData;

typedef struct {
    int value1;
    int value2;
    int value3;
    int value4;
    int value5;
    int value6;
} RXData;

TXData dataToSend;
RXData dataReceived;

void setup() {
    Serial.begin(115200);
    printf_begin();

    radio.begin();
    radio.setPALevel(RF24_PA_LOW);
    radio.setRetries(4, 9);
    radio.setAutoAck(true);
    radio.enableAckPayload();
    radio.enableDynamicPayloads();
    radio.setDataRate(RF24_250KBPS);
```

```

radio.setChannel(frequencyHopping[frequencyIndex]);
radio.openWritingPipe(pipes[1]);
radio.openReadingPipe(1, pipes[0]);

dataToSend.value1 = 1;
dataToSend.value2 = 2;
dataToSend.value3 = 3;
dataToSend.value4 = 4;
dataToSend.value5 = 0x05;

radio.startListening();
radio.printDetails();

// Configure Timer1 for interrupts
TCCR1A = 0;
TCCR1B = (1 << WGM12) | (1 << CS11);
TCNT1 = 0;
OCR1A = (timerInterval * 2000) - 1;
TIFR1 |= _BV(OCF1A);
TIMSK1 = _BV(OCIE1A);
}

void loop() {
  if (timerTriggered) {
    timerTriggered = false;
    interruptCounter++;

    if (interruptCounter == (maxInterrupts - 1)) {
      frequencyIndex++;
      if (frequencyIndex >= sizeof(frequencyHopping)) {
        frequencyIndex = 0;
      }
      radio.setChannel(frequencyHopping[frequencyIndex]);
    }
  }

  if (interruptCounter == maxInterrupts) {
    radio.stopListening();

    radio.write(&dataToSend, sizeof(dataToSend));
  }
}

```

```
    if (radio.isAckPayloadAvailable()) {
        radio.read(&dataReceived, sizeof(dataReceived));
        Serial.print("Received data: ");
        Serial.print(dataReceived.value1);
        Serial.print(", ");
        Serial.print(dataReceived.value2);
        Serial.print(", ");
        Serial.print(dataReceived.value3);
        Serial.print(", ");
        Serial.print(dataReceived.value4);
        Serial.print(", ");
        Serial.print(dataReceived.value5);
        Serial.print(", ");
        Serial.println(dataReceived.value6);
    }

    radio.startListening();
    interruptCounter = 0;
} else {
    // Update data for the next transmission
    dataToSend.value1++;
    dataToSend.value2++;
    dataToSend.value3++;
    dataToSend.value4++;
    dataToSend.value5++;
}
}

ISR(TIMER1_COMPA_vect) {
    timerTriggered = true;
}
```