

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління

КОНДРАТЮК Георгій Георгійович

**Програмно-апаратний модуль домашньої автоматизації з
використанням IoT / Software and hardware module for home automation
using IoT**

Спеціальність 122 – Комп'ютерні науки
Освітньо-професійна програма – Комп'ютерні науки

Дипломний проект

Виконав студент групи КН-41
Г.Г. Кондратюк

Науковий керівник:
к.т.н., доцент О.Р. Осолінський

Дипломний проект допущено до
захисту

«___»_____ 2023 р.

Завідувач кафедри
_____ М.П. Комар

Тернопіль – 2023

Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління
Освітній ступінь «бакалавр»
Спеціальність 122 – Комп'ютерні науки
Освітньо-професійна програма – Комп'ютерні науки

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ М.П. Комар
« ____ » _____ 2022 р.

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЕКТ СТУДЕНТУ
КОНДРАТЮК Георгій Георгійович
(прізвище, ім'я, по батькові)

1. Тема проекту: Програмно-апаратний модуль домашньої автоматизації з використанням IoT / Software and hardware module for home automation using IoT

керівник проекту к.т.н., доцент. О.Р. Осолінський

затверджені наказом по університету від 08 грудня 2022 р. № 491.

2. Строк подання студентом закінченого проекту 01 червня 2023 р.

3. Вихідні дані до проекту: технічне завдання.

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

- опис з Інтернету речей та автоматизації ;
- опис систем автоматизації розумного будинку;
- постановка задачі;
- розробка архітектури модуля;
- розробка алгоритмічного забезпечення модуля;
- розробка інформаційного забезпечення модуля домашньої автоматизації;
- аналіз програмного та апаратного забезпечення системи;
- розробка програмного забезпечення для апаратних модулів та користувацького інтерфейсу;
- проведення тестування модуля.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
- Архітектура модуля домашньої автоматизації з використанням IoT;
 - Алгоритм роботи модуля автоматизації розумного будинку з використанням Singspeak.

6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
Н. контроль	к.т.н., доцент О.Р. Осолінський		

7. Дата видачі завдання 08 грудня 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту	Строк виконання етапів проекту	Примітка
1	Інтернет речей та системи автоматизації розумних будинків	30.12.2022	
2	Розробка архітектури програмно-апаратно модуля домашньої автоматизації	24.03.2023	
3	Програмно-технологічне забезпечення програмно-апаратно модуля домашньої автоматизації	12.05.2023	
4	Повне завершення та оформлення дипломного проекту	01.06.2023	

Студент _____ Г.Г.Кондратюк
(підпис)

Керівник проекту _____ О.Р. Осолінський
(підпис)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту: 64 с., 38 рис., 3 додатки, 33 джерела.

Метою дипломного проекту є розробка програмно-апаратного модуля домашньої автоматизації з використанням IoT.

В роботі використовувалися методи аналізу, порівняння, логічного узагальнення результатів, методи інтелектуального аналізу даних.

Було розроблено архітектуру програмно-апаратного модуля домашньої автоматизації, який включає в себе базову станцію для управління та передачі даних від сателітних станцій.

Також описано алгоритми функціонування та розроблено відповідне програмне забезпечення, як для апаратного так і для серверного забезпечення. Також було розроблено графічний інтерфейс для мобільного застосунку, хоча користувач може так само управляти модулем через веб інтерфейс.

Ключові слова: IOT, АВТОМАТИЗАЦІЯ, РОЗУМНИЙ БУДИНОК, ARDUINO, RF-МОДУЛЬ.

ABSTRACT

The bachelor's thesis report: 64 pages, 38 figures, 3 appendices, 33 sources.

The aim of the diploma project is to develop a software and hardware module for home automation using IoT.

The methods used in the work were analysis, comparison, logical generalization of results, methods of data mining.

The architecture of the home automation software and hardware module was developed, which includes a base station for controlling and transmitting data from satellite stations.

The algorithms of functioning were also described and the corresponding software for both hardware and server software was developed. A graphical interface for a mobile application has also been developed, although the user can also operate the module via a web interface.

Keywords: IOT, AUTOMATION, SMART HOME, ARDUINO, RF MODULE.

ТЕХНІЧНЕ ЗАВДАННЯ

1. НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ

1.1 Програмно-апаратний модуль домашньої автоматизації з використанням IoT.

1.2 Область застосування – системи розумних будинків.

2. ОСНОВА ДЛЯ РОЗРОБЛЕННЯ

Основою для розроблення є завдання на дипломний проект, затверджене кафедрою інформаційно-обчислювальних систем і управління факультету комп'ютерних інформаційних технологій Західноукраїнського національного університету.

3. ПРИЗНАЧЕННЯ РОЗРОБЛЕНОГО КОМПЛЕКСУ

Метою дипломного проекту є розробка програмно-апаратного модуля домашньої автоматизації з використанням IoT.

4. ДЖЕРЕЛА РОЗРОБЛЕННЯ

Джерелами даної розробки є матеріали навчальної і реферативної літератури, технічна документація, науково-дослідні статті, журнали, Інтернет.

5. ТЕХНІЧНІ ВИМОГИ

5.1 Основні функціональні вимоги до програмної системи:

- збір сенсорних даних;
- бездротова передача даних;
- віддалений доступ;
- збереження та обробка даних на хмарному сервісі;

5.2 Вимоги до апаратних засобів:

- Arduino;

- Модуль термістора NTC LM393;
- Інфрачервоний датчик руху HC-SR501
- Сенсор диму MQ-2
- Модуль радіозв'язку nRF24L01
- Модуль ESP8266

5.3 Вимоги до програмних засобів:

- Arduino Software (IDE);
- Сервіс ThingSpeak;

6. ПОРЯДОК КОНТРОЛЮ

6.1 Представлення дипломного проекту на попередній захист.

6.2 Представлення дипломного проекту на захист.

Завдання прийняв до виконання _____ Г.Г.Кондратюк
(підпис) (прізвище та ініціали)

Керівник дипломного проекту _____ О.Р. Осолінський
(підпис) (прізвище та ініціали)

ЗМІСТ

Вступ.....	9
1 Інтернет речей та системи автоматизації розумних будинків.....	10
1.1 Інтернет речей та автоматизація.....	10
1.2 Система автоматизації розумного будинку.....	11
1.3 Аналіз існуючих рішень в області іот для автоматизації будинків	13
1.4 Постановка задачі	18
2 Розробка архітектури програмно-апаратно модуля домашньої автоматизації	20
2.1 Архітектура модуля	20
2.2 Алгоритмічне забезпечення модуля.....	25
2.3 Інформаційне забезпечення модуля домашньої автоматизації.....	31
3 Програмно-технологічне забезпечення програмно-апаратно модуля домашньої автоматизації	35
3.1 Аналіз програмного та апаратного забезпечення системи	35
3.2 Програмне забезпечення для апаратних модулів та користувацького інтерфейсу	42
3.3 Тестування модуля.....	48
Висновки	53
Список використаних джерел	54
Додаток А лістинг програми радіочастотної комунікації.....	58
Додаток Б Архітектура модуля домашньої автоматизації з використанням IoT	63
Додаток В Алгоритм роботи модуля автоматизації розумного будинку з використанням Singspeak	64

					<i>ДП.КН.8091600.064.ПЗ</i>			
Змн.	Арк.	№ докум.	Підпис	Дата	Програмно-апаратний модуль домашньої автоматизації з використанням IoT	Літ.	Арк.	Акрушів
Розроб.		Кондратюк Г.Г.					8	64
Перевір.		Осолінський О.Р.						
Реценз.								
Н. Контр.		Осолінський О.Р.						
Затверд.		Комар М.П.				ЗУНУ.ФКІТ.КН-41		

ВСТУП

Актуальність теми дослідження. Домашня автоматизація відноситься до автоматичного способу управління побутовими приладами. Існують різні системи, що використовуються для домашньої автоматизації, які базуються на різних мікроконтролерах і приймають різні параметри для моніторингу та управління побутовими приладами.

Такі системи, що забезпечують можливість ефективного управління побутовими приладами за допомогою датчиків IoT та інших комунікаційних пристроїв. Можна контролювати побутову техніку за допомогою мобільних пристроїв, ноутбуків або через Інтернет з будь-якої точки світу.

Дані системи використовуються для управління різними лампами, вентиляторами, побутовою технікою, електродвигунами, кондиціонерами, системами опалення тощо, якими легко керувати за допомогою веб-пристроїв або пристроїв з доступом до Інтернету.

Всі ці типи систем стають все більш популярними завдяки меншій вартості впровадження та гнучкій функціональності, які можна легко налаштувати відповідно до потреб кожного, тому всі системи Інтернету речей користуються великим попитом і мають велику цінність.

Метою дипломного проекту є розробка програмно-апаратного модуля домашньої автоматизації з використанням IoT.

Об'єктом дослідження є системи домашньої автоматизації з використанням IoT.

Предметом дослідження процесів управління розумними приладами в будинку.

Методи та технології розробки. Технологією розробки було обрано, C++ для мікроконтролерів, Хмарні сервіси для IoT.

Практичне значення одержаних результатів. В результаті проведеної роботи буде програмно-апаратний модуль домашньої автоматизації для контролю різних побутових приладів, що працює в середовищі IoT.

					ДП.КН.8091600.064.ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

1 ІНТЕРНЕТ РЕЧЕЙ ТА СИСТЕМИ АВТОМАТИЗАЦІЇ РОЗУМНИХ БУДИНКІВ

1.1 Інтернет речей та автоматизація

З плином часу індустрія мобільних додатків стрімко розвивається. У цьому столітті кількість користувачів мобільних телефонів зросла неймовірно. Зростання використання телефонів вимагає від розробників створення зручних для користувача мобільних додатків. Поєднання суті мобільної розробки з Інтернетом речей (ІОТ) полегшує користувачам управління або моніторинг апаратних пристроїв за допомогою мобільних додатків. Речі, які працюють в рамках ІОТ, мають здатність відчувати, збирати та обмінюватися даними через мережу з будь-якої частини світу, де ці дані використовуються для різних цілей. Інтернет речей створює всі можливості для того, щоб об'єкти працювали самостійно [1].

Інтернет речей (ІОТ) визначається як середовище, де всі пристрої взаємодіють один з одним без взаємодії "людина-людина" або "людина-комп'ютер" [2]. ІОТ передбачає, що речі, які підключені до інтернету, можуть взаємодіяти, спілкуватися та обмінюватися даними між собою. Завдяки підключенню об'єктів до Інтернету з'являється незліченна кількість можливостей, які можна реалізувати [3]. Зараз вони набувають все більшого значення, оскільки стають частиною системи. Це також включає в себе професійний спосіб збору та аналізу даних через бездротове з'єднання, що робить зв'язок між об'єктами швидшим. в управлінні та автоматизації об'єктів, використання інтернету речей забезпечило набагато більшу гнучкість, що робить цю технологію повсюдною. ІОТ полегшив наше життя, запровадивши концепцію "розумних будинків". Системи автоматизації розумних будинків дали користувачам можливість керувати своїми електроприладами одним дотиком на мобільному телефоні. Більше того, Розумні будинки також допомагають вам у підтримці домашньої безпеки. Він надішле на ваш телефон сповіщення, якщо буде виявлено проникнення злоумисника. Інтернет речей

					ДП.КН.8091600.064.ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

вже впроваджений у багатьох інших сферах, і він виконує свою роботу добре організовано. Підтримка інтернету речей вже надається в багатьох додатках, що також корисно при створенні великих вбудованих систем. Багато автоматизованих систем для медичної, комерційної, транспортної та великої промисловості адаптовані до концепції інтернету речей [4]. Після впровадження систем "розумного будинку", які розробляються з використанням ІОТ, дослідження ІОТ з різною реалізацією набули значного поширення і продовжуються дотепер [5]. Розпочалася нова ера комунікаційних технологій з удосконаленнями, що стали можливими завдяки цьому [6].

Однією з головних переваг впровадження "розумного будинку" на основі ІОТ є те, що користувач може економити електроенергію та енергоресурси. Крім того, за допомогою розумного будинку користувач може контролювати споживання енергії приладами на щоденній основі. Як тільки користувачі дізнаються про стан споживання, вони докладуть зусиль, щоб зменшити його, що в кінцевому підсумку призведе до зменшення рахунків за електроенергію [7].

1.2 Система автоматизації розумного будинку

Домашня автоматизація вперше була використана і впроваджена в цивілізованих країнах за допомогою інфрачервоного випромінювання, Bluetooth і подібних технологій. У нас вона не набула великої популярності з кількох причин. Радіус дії занадто обмежений, швидкість роботи дуже низька, а також дуже висока вартість, тому ніхто не може легко встановити цю автоматизовану систему [8]. Але після стрімкого розвитку сучасних технологій, коли підключення пристроїв здійснюється за допомогою Інтернету, що також допомагає будувати ці системи за допомогою Інтернету, їх розвиток зростає дуже швидкими темпами через широкий спектр використання, швидку реакцію, низьке енергоспоживання та багато інших

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

особливостей, що загалом тепер відоме як "Автоматизація розумного будинку" [9].

Системи автоматизації розумного будинку визначаються як впровадження системи в домашнє середовище, щоб полегшити своїм клієнтам життя, забезпечуючи комфорт, зручність та енергоефективність S. Послуги, які вона надає, забезпечують більшу зону комфорту. Також покращується швидкість реагування. Це повністю змінює домашнє середовище, оскільки дозволяє здійснювати інтелектуальну комунікацію між людьми, системами та речами. Перевага підключення розумних пристроїв до Інтернету створює простий спосіб для пристроїв моніторингу та контролю його стану [10]. Розумна домашня автоматизація забезпечує управління побутовими пристроями в режимі реального часу. Смартфони, які ми всі носимо в кишенях або в руках, є дуже потужними інструментами, за допомогою яких ми залишаємося на зв'язку з усім світом і це робить наше життя простішим

Доступ до Інтернету є скрізь, у кожному будинку є власна мережа Wi-Fi, а у кожної людини - власний смартфон. Завдяки цим можливостям використання систем автоматизації розумного будинку стає дуже поширеним. Зі збільшенням рівня злочинності управління безпекою в наших будинках стає дуже складним. Роль, яку відіграє ІОТ в цій сфері, дуже значна, він перетворює звичайний дверний дзвінок в розумний дзвінок разом з вбудованою відеокамерою, так що власник будинку може отримати відео людини, яка підходить до дверей, і він може керувати дверима за допомогою свого мобільного телефону [11]. Автоматизація розумного будинку відноситься до високотехнологічної функціональності і забезпечує розкішний комфорт, який був недоступний в минулому. Системи розумного будинку є енергоефективними, забезпечують повну підтримку автоматизації побутових пристроїв і корисні для підтримки або економії енергоспоживання, що допомагає досягти певного рівня комфорту [12]. До ІОТ ці системи були реалізовані за допомогою Bluetooth-пристроїв, але вони не стали дуже відомими, оскільки мали доступ лише в межах певного заданого діапазону.

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

Інтернет речей долає це обмеження і надає людині чудовий спосіб контролювати кожен аспект домашніх пристроїв лише одним кліком за допомогою програмного додатку з будь-якого місця. Ці типи систем також застосовуються в багатьох готелях, ресторанах, будинках, офісах і т.д., щоб полегшити життя клієнтам.

Після того, як бездротові технології та дешеві датчики стали частиною інтернету речей, вартість розумних будинків знизилася, що сприяло розвитку цих систем [13]. З розвитком технологій можливості для вдосконалення та покращення системи завжди будуть існувати. Система на основі інтернету речей забезпечує інтелектуальний спосіб комунікації між речами та користувачами. Розумна автоматизація відстежує всю систему за допомогою програмного додатку, який допомагає користувачам контролювати свої пристрої, навіть якщо він/вона відсутній у місці, де встановлена апаратна система. Якщо користувач забув вимкнути всі домашні пристрої і пішов з дому, він не зможе дізнатися стан пристроїв і не зможе змусити їх вимкнутися. Але використовуючи систему автоматизації розумного будинку, користувач отримає сповіщення або може контролювати стан своїх пристроїв за допомогою програмного додатку, що допомагає подолати непотрібне використання енергії. Таким чином, ці системи також допомагають подолати непотрібне використання електроенергії.

1.3 Аналіз існуючих рішень в області IoT для автоматизації будинків

Інтернет речей стає однією з нових технологій останніх років. Це розширення інтернет-сервісів, яке змінило спосіб життя людини, забезпечивши зв'язок з будь-ким і всюди [14]. IoT - це комбінація електронних пристроїв, пов'язаних з датчиками, виконавчими механізмами, програмним забезпеченням та Wi-Fi, які дозволяють цим об'єктам обмінюватися інформацією [15]. Деякі роботи, пов'язані з автоматизацією розумного будинку, розглянуті нижче.

					ДП.КН.8091600.064.ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

У цій статті автор запропонував дизайн розумного будинку з використанням Raspberry Pi та комп'ютерного зору. Raspberry pi займається керуванням пристроями, записом з відеокамер та визначенням руху, в той час як методи комп'ютерного зору виявляють присутність зловмисника. Таким чином, щоразу, коли виявляється постороння особа, raspberry надсилає попередження у вигляді повідомлення або іншого сповіщення [16]. Однак, ця ж методика може бути реалізована за допомогою мікроконтролера, що зменшує вартість системи, або може мати інші обмеження.

В даній роботі [17] автор запропонував дружній розумний будинок, який працює на основі пуш-сповіщень та серверів, як персональних помічників. Основними компонентами, що використовуються в архітектурі, є Arduino-мікроконтролер і телефон на базі Android. Дизайн цієї системи можна розділити на три частини: веб-сервер Pi керує загальною мережею та обслуговуванням бази даних. Крім того, він надсилає запит до хмари та надсилає відповідь через пуш-сповіщення в результаті введення користувачем даних. Масив датчиків з контролером та Arduino, з'єднаний з усіма датчиками, включає: пасивний інфрачервоний датчик руху (PIR), датчик освітленості (LDR), датчик температури та вологості (DHT11), магнітний герконовий датчик, датчик дощових крапель (LM393), датчик вологості ґрунту, датчик газу (MQ5). Привід Pi, що містить модуль релейного масиву на 12 В, працює як перемикач, який може надсилати та отримувати повідомлення одночасно. Вся система залежить від датчиків, немає ніякої взаємодії з людиною. Якщо датчики вийшли з ладу, система також перестане працювати.

В іншій роботі автор запропонував дизайн розумного будинку, в якому використовується датчик КТ, який веде облік одиниць, спожитих побутовими приладами, і регулярно оновлює базу даних. Ці дані про споживання енергії будуть надсилатися користувачеві у вигляді сповіщень кожні 15 днів, що містять детальну інформацію про споживання енергії. Ця програма в режимі реального часу допомагає користувачеві вести моніторинг і вживати певних заходів [18]. Ця система вимагає великого обсягу пам'яті, оскільки вона

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

зберігає записи на регулярній основі, тому після певного обсягу зберігання, це може призвести до вичерпання обсягу пам'яті. Крім того, було використано датчик навантаження, з'єднаний з Arduino, який відстежує рівень скрапленого газу, і коли балон досягає певного рівня, агентство з бронювання газу отримує повідомлення, яке надсилається системою, а на мобільний телефон користувача надсилається сповіщення, яке генерує та прикріплює унікальний ідентифікатор до кожного бронювання [16].

Метою даної роботи є розробка системи моніторингу розумного будинку, яка працює на основі колектора даних, що збирає дані на основі різних датчиків. Різні блоки датчиків даних згруповані в один блок, наприклад, аудіодатчики, відеодатчики та датчики диму згруповані в один блок даних. Всі датчики в групі з'єднані через бездротову мережу. Ці системи надають допомогу користувачеві в разі надзвичайних ситуацій. Припустимо, якщо в будинку виникла пожежа, збирач даних отримує дані від датчика диму, записує звук і негайно надсилає тривогу на мобільний телефон користувача разом з відео. Є випадки, в яких датчик не буде працювати належним чином, тому, як тільки користувач отримає сповіщення з центру обробки даних на мобільний телефон, він повинен перевірити відео вручну і прийняти рішення відповідно до цього. Розумний будинок також працює на аудіо датчику, який надсилає повідомлення користувачеві, якщо чути якийсь ненормальний шум, наприклад, крик когось в будинку, або якщо виявлено злодія [19] [20].

У цій статті автор запропонував розумний будинок на основі ZigBee Wi-Fi шлюзу. Датчики та виконавчі механізми розгортаються та отримують сигнали віддалено через бездротову мережу, блок управління працює через графічний інтерфейс, а також шлюз. Шлюз - це міст, який встановлює зв'язок між різними протоколами, що призводить до розвитку розумних будинків шляхом впровадження різних датчиків і виконавчих механізмів [21]. Домашня автоматизована система на основі ZigBee працює тільки на обмеженій пам'яті для обміну даними між приладами, оскільки має протокол з низьким енергоспоживанням.

					ДП.КН.8091600.064.ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

1.3.1 Функціональний опис

Інтернет речей створює взаємодію між користувачем та об'єктом або об'єктом та об'єктом. З технічної точки зору, фізичний апаратний об'єкт отримує запит від пристрою користувача через мережу з певними інструкціями, і цей апаратний об'єкт виконує дію відповідно до отриманих інструкцій.

Фізичний апаратний об'єкт може складатися з будь-якої апаратно-обчислювальної платформи, наприклад, raspberry pi, Arduino, RFID тощо. Всі плати мають свої переваги та недоліки. Вони мають майже однакову робочу архітектуру, але відрізняються за можливостями, вартістю та швидкістю роботи. Впровадження цих апаратних плат дозволило вивести технологію інтернету речей на дуже високий рівень. З розвитком бездротових технологій та перевагами Wi-Fi мережі в автоматизації, з'являється підтримка різних технологічних середовищ зв'язку, кожне з яких має свої унікальні технічні характеристики та ефективність роботи [22].

1.3.1.1 Raspberry pi

Raspberry pi - це свого роду міні-комп'ютер. Він має всі функції, які має комп'ютер, навіть має деякі додаткові можливості керування пристроями, які підключаються до його GPIO виводів. Вперше він був виготовлений у Великобританії компанією Raspberry Pi Foundation. Його головною особливістю є те, що він може запускати декілька програм, оскільки він відноситься до комп'ютерів загального призначення [23]. Він використовує технологію ARM (Advanced Instruction Set Computing Machine - обчислювальна машина з розширеним набором інструкцій), яка зменшує споживання енергії [24]. Цей же принцип роботи raspberry pi також допомагає у розробці великих вбудованих систем. Використовуючи raspberry і їх робоче використання зростає з кожною оновленою версією.

Raspberry pi 4 - це модель серед доступних А, В і В+, яка містить процесор з 512 МБ і має 24 виводи GPIO (General Purpose Input Output), що

					ДП.КН.8091600.064.ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

означає, що 24 електронні пристрої, такі як світло, вентилятори, можуть управлятися за допомогою кожного релейного модуля, який працює як перемикач між виводами GPIO на платі Raspberry pi. Вона підтримує операційну систему Raspbian, яка має такі ж робочі можливості, як і операційна система Linux.

1.3.1.2 Arduino

Arduino - це ще один тип апаратної плати з відкритим вихідним кодом, який, як правило, вважається мікроконтролером, що забезпечує програмно-апаратну взаємодію. Він гнучкий і дуже простий у використанні, але має низьку швидкість обробки, якщо порівнювати зі швидкістю обробки raspberry pi. Він запускає лише одну програму одночасно. В основному використовує мови програмування C або C++.

Плата Arduino складається з різних типів обладнання, всі плати мають 14 цифрових виводів, крім Mega2560 Arduino, яка має 54 виводи, які працюють так само, як і виводи GPIO raspberry pi [25]. Релейна підтримка перемикачів надається цифровим вхідним вихідним контактам Arduino, які з'єднані з побутовими пристроями.

1.3.1.3 Модуль ZigBee

Модуль ZigBee є ще одним непрямим засобом використання для системи автоматизації розумного будинку. Він створює та керує зв'язком між інтерфейсом програми користувача та функціями системи. Зв'язок здійснюється шляхом передачі повідомлення між пристроями. Він відповідає за встановлення зв'язку між користувачами за допомогою відповідного інтерфейсу програми та функціональності системи [26]. В основному розглядається, як альтернативний варіант Bluetooth або Wi-Fi. На відміну від функціоналу Bluetooth, який працює за принципом "точка-точка", модуль ZigBee працює за протоколом комірчастої мережі, який транслює всю передачу даних. Це забезпечує набагато кращу стабільність зв'язку.

					ДП.КН.8091600.064.ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

1.4 Постановка задачі

Як видно з аналізу робіт, зараз присутні багато різних варіацій таких систем від дуже дешевих до таких які коштують десятки тисяч доларів. Така тенденція показує те, що інтерес в цій області не спадає. Нижче приведено яким критеріям повинна відповідати модуль автоматизації:

- Функціональна сумісність. Принадність системи автоматизації полягає в її здатності об'єднувати різноманітні електронні пристрої разом, щоб вони могли працювати як одна уніфікована система. Забезпечення злагодженої, залежно від "відкритості" системи автоматизації..

- Віддалений доступ. Можливість швидко і легко змінювати налаштування. Найчастіше це буде потрібно, коли власника немає вдома, тому можливість віддалено повідомляти про критичні і не тільки зміни в системі домашньої автоматизації є однією з основних функцій.

- Можливість розширення. Так як технології продовжуватимуть розвиватися, виводячи на ринок абсолютно нове покоління продуктів. У майбутньому може постати завдання додавання до мережі автоматизації нових пристроїв з приміщеннями приміщення. З цих причин важливо, щоб система домашньої автоматизації дозволяла легко розширюватись по вертикалі для включення додаткових продуктів і по горизонталі для підтримки додаткових приміщень.

- Можливість модернізації. Рушійною силою системи автоматизації є програмне забезпечення. Чим досконаліше це програмне забезпечення, тим більше система може зробити. Такі системи повинні мати можливість оновлюватись, як користувачем (дешеві рішення), так і в автоматичному режимі (готові рішення).

- Різноманітність користувацьких інтерфейсів. Залежно від динаміки родини яка живе у будинку, бюджету та уподобань, система або модуль повинен підтримувати декілька варіантів підключення користувацьких

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

інтерфейсів, наприклад, через веб-сторінку, або ПЗ на смартфоні або планшеті.

– Енергоменеджмент. Системи автоматизації повинні мати можливість управління енергоспоживанням електроприладів для економії ресурсів та коштів, дана вимога не є обов'язковою, але повинна бути можливість додавання такого функціоналу.

Отже, розробка програмно-апаратного модуля домашньої автоматизації з використанням IoT, який би був не дорогим і мав можливість як горизонтального так і вертикального розширення є актуальною задачею.

Метою дипломного проекту є розробка програмно-апаратного модуля домашньої автоматизації з використанням IoT з метою покращення комфорту мешканців будинку, забезпечення безпеки, віддаленого керування та економії електроенергії.

Для досягнення мети потрібно вирішення такі завдань:

1. Визначити, як вплинув Інтернет речей на системи автоматизації розумних будинків
2. описати Інтернет речей та автоматизацію
3. описати системи автоматизації розумного будинку
4. провести аналіз існуючих рішень в області ІОТ для автоматизації будинків.
5. розробити архітектуру модуля
6. розробити алгоритмічне забезпечення модуля
7. розробити інформаційне забезпечення модуля домашньої автоматизації
8. Розробити програмне забезпечення для апаратних модулів та користувацького інтерфейсу
9. провести тестування модуля

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

2 РОЗРОБКА АРХІТЕКТУРИ ПРОГРАМНО-АПАРАТНО МОДУЛЯ ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ

2.1 Архітектура модуля

Принадність системи автоматизації полягає в її здатності об'єднувати різні електронні пристрої разом, щоб вони могли працювати як одна уніфікована система. Забезпечення злагодженої роботи цих пристроїв може бути простим або складним, залежно від "відкритості" системи автоматизації. Чим більш відкритою є система, тим легше освітлювальним приладам, термостатам, аудіо- та відеообладнанням, пристроям безпеки, моторизованим жалюзям та іншій електроніці взаємодіяти один з одним. Хорошим прикладом інтероперабельності є вимкнення світла, повернення термостатів назад, коли натискається кнопка "до побачення" на клавіатурі або коли датчик руху помічає, що людина вийшла з кімнати.

З конкретною кінцевою метою вирішити завдання в розділі 1.4 адаптивності та функціональності в огляді літератури, було сплановано та актуалізовано гнучку, адаптивну та мінімально трудомістку систему дистанційного керування та моніторингу будинку.

Запропонований апаратно-програмний модуль складається з базової станції та сателітних станцій. Базова станція являє собою мікроконтролерну плату Arduino Mega, підключену до модуля Wi-Fi і, таким чином, здатну отримати доступ до Інтернету. Вона має модуль радіочастотного приймача, за допомогою якого зв'язується з іншими станціями (також відомими як віддалені вузли).

Сателітні станції - це мікроконтролерні плати ArduinoUno з датчиками: Датчик освітленості, датчик температури, датчик руху, ємнісні сенсори, датчик вологості і датчик газу. Сателітні станції зв'язуються з базовою станцією через модуль радіочастотного приймача. Ці станції мають реле, які можуть бути підключені до електроприладів, в даному випадку це виключно

					ДП.КН.8091600.064.ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

світло та вентилятор. Функції модуля: Запропонований модуль домашньої автоматизації має можливості моніторингу різних електроприладів у будинку.

Сателітна станція (рис. 2.1 (б)) за допомогою відповідних датчиків вимірює температуру в приміщенні, виявляє рух, витік газу та дотик і може вмикати/вимикати електроприлади (наприклад, світло, вентилятор тощо) за допомогою реле. У цьому модулі базова станція та сателітна станція працюють у конфігурації "ведучий-ведений", де базова станція надає необхідні команди сателітним станціям для виконання дій на основі ручного або автоматичного режиму, обраного користувачем, як показано на рисунку 2.1.

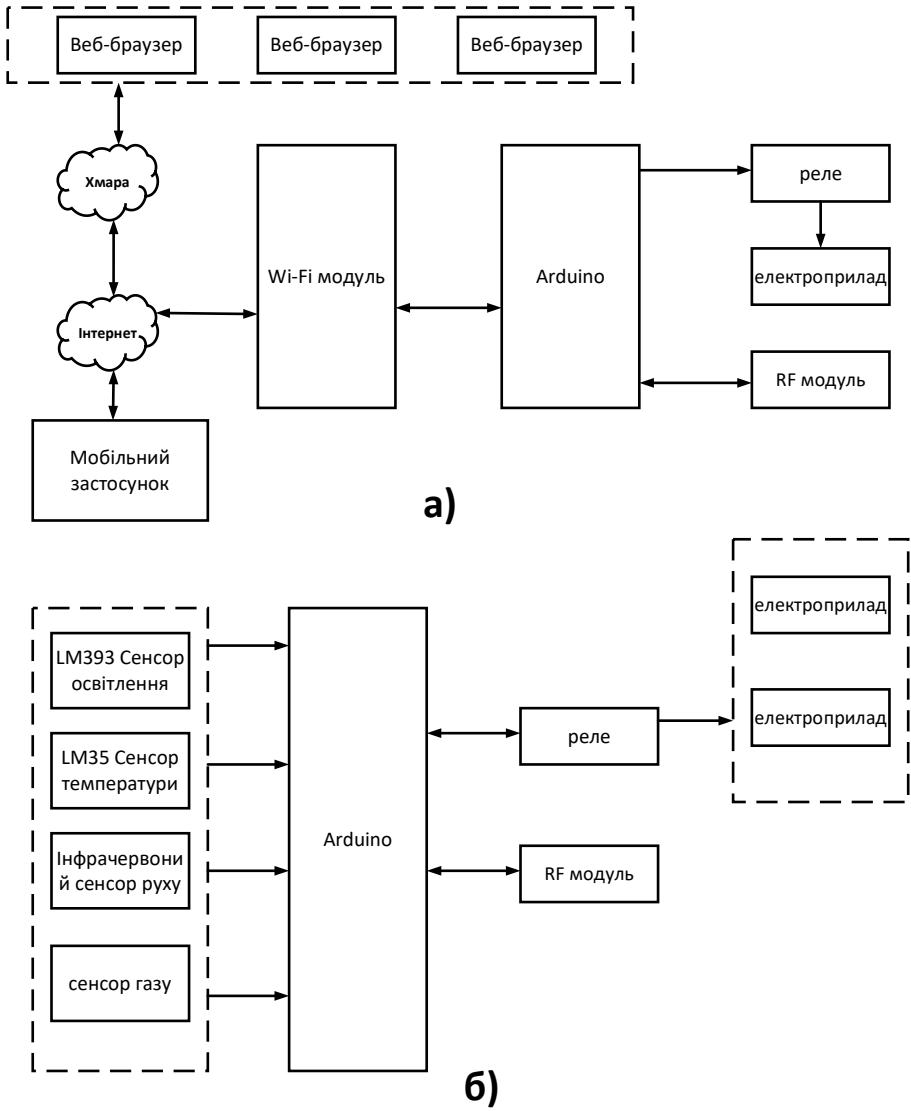


Рисунок 2.1 -Архітектура модуля

Управління системою здійснюється за допомогою смартфона на базі Android, або веб-інтерфейсом який виступає в ролі клієнта, а дані передаються через Socket-програму. Додаток надає інтерфейс для кожної з трьох станцій, за допомогою якого користувач має доступ до управління та моніторингу їх віддалено.

2.1.1 Робота системи

Після ввімкнення системи Базова станція через модуль Wi-Fi підключається до вказаної точки доступу Wi-Fi і встановлює TCP-з'єднання з "thingspeak.com" та Android-додатком. Вона отримує команди від програми, запущеної на смартфоні Android, який підключений до тієї ж точки доступу або до іншої точки за допомогою переадресації портів. Вона також встановлює радіочастотний зв'язок із сателітними станціями через радіомодулі, використовуючи їхні відповідні адреси. Базова станція отримує дані датчиків та умови в приміщенні від супутникових станцій по черзі через радіоканал, коли користувач надсилає команду, і ці дані надсилаються до хмарної платформи ThingSpeak Cloud Platform з частотою 15 секунд за допомогою модуля Wi-Fi, використовуючи заданий ключ API. Показання всіх цих датчиків можна отримати з облікового запису ThingSpeak і побачити в додатку, а також на веб-сайті.

Існує два режими, за допомогою яких можна керувати електроприладами: ручний режим та автоматичний режим

2.1.2 Автоматичний режим

Режим на основі датчиків. сателітна станція, отримуючи команду ручного керування від базової станції, вмикає або вимикає відповідний електроприлад вручну. При отриманні команди автоматичного керування освітленням, вона вмикає світло, якщо датчик освітленості вказує на відсутність світла в приміщенні, в іншому випадку світло вимикається.

					ДП.КН.8091600.064.ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

При отриманні команди AUTO для вентилятора, він вмикає вентилятор, якщо датчик температури виявляє температуру вище встановленого порогу; в іншому випадку вентилятор вимикається. Супутникова станція безперервно зчитує аналогові або цифрові значення, які виводяться з датчиків, і такі дані, як температура, перетворюються з необробленого аналогового виходу в градуси Цельсія. Ці дані датчиків надсилаються на базову станцію за запитом.

2.1.3 Система безпеки та моніторингу

На супутниковій станції PIR-датчик руху, газовий датчик і ємнісний сенсорний датчик забезпечують функцію моніторингу та безпеки системи. За допомогою PIR-сенсора і датчика газу здійснюється безперервний моніторинг приміщення на предмет руху і витоку зрідженого газу. Ємнісний сенсорний датчик здатний виявити навіть найменший дотик і може бути використаний для виявлення вторгнення.

Поточний стан цих датчиків надсилається на базову станцію, через яку він потрапляє в хмару, а користувачеві надсилається повідомлення (наприклад, твіт) у разі виникнення аварійної ситуації в системі домашньої автоматизації або спрацювання тривоги на сателітній станції.

Індикація цих тривог відображається в додатку для Android у вигляді 1 або 0, де 1 означає, що тривога спрацювала, а 0 - ні. Дані з датчиків PIR, LPG і сенсорного датчика передаються в хмару через базову станцію.

2.1.4 Альтернативний підхід до архітектури

Також передбачено іншу архітектуру, яка може використовувати NodeMCU, як альтернативу, тому що засоби програмування є однакові, а даний модуль усуває додаткове комунікаційне обладнання.

Ця модель системи домашньої автоматизації містить сервер, актуатори, датчики та мікроконтролери. Внутрішній сервер буде налаштований на управління, моніторинг сенсорних пристроїв. Запропонована система домашньої автоматизації буде дистанційно управлятися за допомогою

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23

бездротових технологічних комунікаційних пристроїв, таких як смартфони, планшети та інші бездротові пристрої, віддалено через Інтернет.

У запропонованій системі (рисунок 2.1) домашньої автоматизації можна дистанційно контролювати, керувати температурою в приміщенні, автоматичним включенням і виключенням вентиляторів, автоматичним включенням і виключенням світла, автоматичним витоком газу, виявленим датчиками, системою кондиціонування повітря і т.д. Автоматично контролюються і управляються системою домашньої автоматизації. Дана альтернативна система домашньої автоматизації без участі людини контролює витік газу, вмикання та вимикання вентиляторів, вмикання та вимикання світла, перевіряє та контролює температуру та рівень вологості в приміщенні за допомогою комунікаційних пристроїв, пов'язаних з IoT.

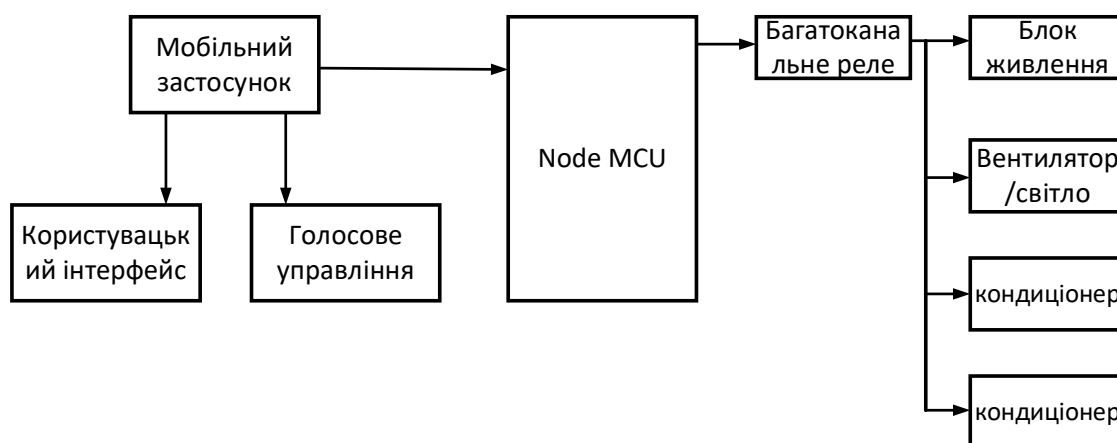


Рисунок 2.2 – Альтернативний варіант архітектури

В цій архітектурі NodeMCU [27] є мозком цієї системи і виконує різні процеси для системи домашньої техніки. NodeMCU підключається, спілкується з різними датчиками і збирає інформацію в режимі реального часу для системи домашньої автоматизації. Вони містять два Node MCU. NodeMCU (Node Micro Controller Unit) - це програмне і апаратне забезпечення з відкритим вихідним кодом, яке дозволяє створити дуже дешеву систему, побудовану на мікросхемі ESP8266[28].

2.2 Алгоритмічне забезпечення системи

Як було сказано вище про архітектуру домашньої автоматизації, робота системи в основному базується на архітектурі. Програмне забезпечення відповідає за надання користувацького інтерфейсу користувачам. Користувач повинен увійти в систему, після чого він/вона може керувати або контролювати стан пристроїв.

Програмний додаток отримує запит, згенерований користувачем, і передає його на апаратну плату. Додаток за допомогою IP-адреси отримує віддалений доступ до апаратної плати і виконує свої дії відповідно до введеного користувачем запиту. Робота програмного додатку визначається за допомогою блок-схеми на рисунку 2.3.



Рисунок 2.3 – Блок-схема алгоритму програмного додатку

Скрипти пишуться та зберігаються для кожного вхідного та вихідного контакту апаратної плати, які підключаються до кожного окремого пристрою, такого як світло, вентилятор тощо, за допомогою одного релейного модуля, який працює як перемикач між пристроєм та контактом апаратної плати.

Здебільшого скрипти написані на мові python. Але підтримка JavaScript також доступна на багатьох апаратних платах, залежно від того, яку операційну систему вони використовують. Як тільки користувач натискає на будь-яку кнопку пристрою з програмного додатку, програма викликає і запускає відповідний скрипт плати, який виконує команду, в результаті чого стан пристрою змінюється відповідно до виконаної користувачем дії. Таку ж саму функцію виконують і всі інші контакти вводу-виводу апаратної плати. На рисунку 2.4 показана вся внутрішня робота системи автоматизації розумного будинку.

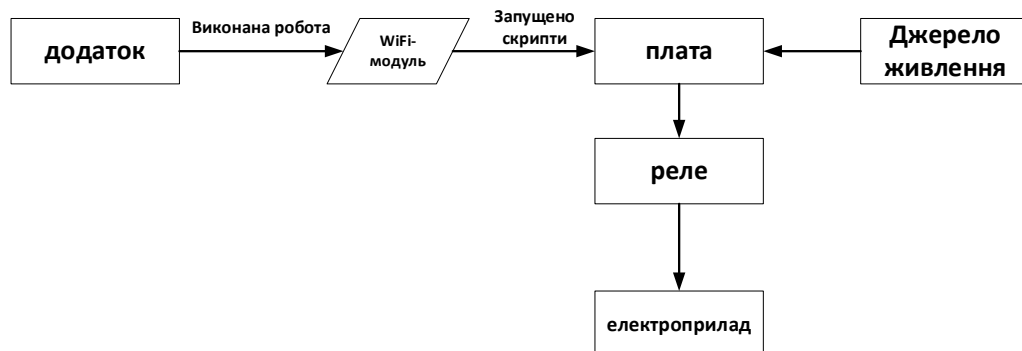


Рисунок 2.4 - Внутрішня робота модуля домашньої автоматизації

Користувачі додатків також можуть керувати або контролювати пристрої розумного будинку, якщо він недоступний в межах певного домену або мережі. Але обов'язковою умовою для апаратної плати є наявність певної статичної IP-адреси, через яку можна отримати віддалений доступ до апаратної плати з будь-якої точки світу. Завдяки цьому Додаток може запускати будь-які скрипти, що зберігаються в апаратній платі, які відповідають виконуваним користувачами діям. Але це вимагає доступу авторизованих користувачів, інакше це буде потік для системи, який може

призвести до шкідливого результату. Для аутентифікації необхідний певний рівень підтримки безпеки.

Алгоритм роботи системи автоматизації розумного будинку показано на рисунку 2.5. Як показано на малюнку, початковою вимогою є підключення до Інтернету для доступу до вашого розумного будинку. Можна отримати доступ до розумного будинку або через веб-сервіс, або через singspeak асистента.

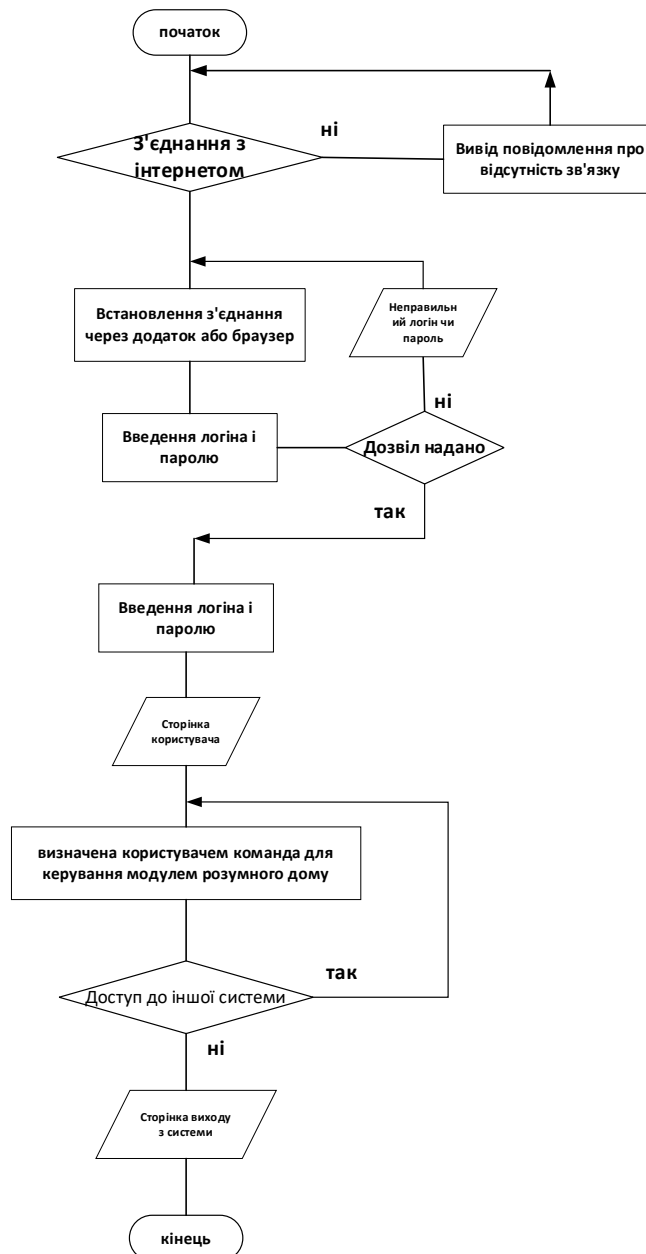


Рисунок 2.5 – Алгоритм роботи модуля автоматизації розумного будинку з використанням singspeak асистента

На рисунку 2.6 представлено схему потоків даних базової станції запропонованої архітектури (див.рис. 2.1)

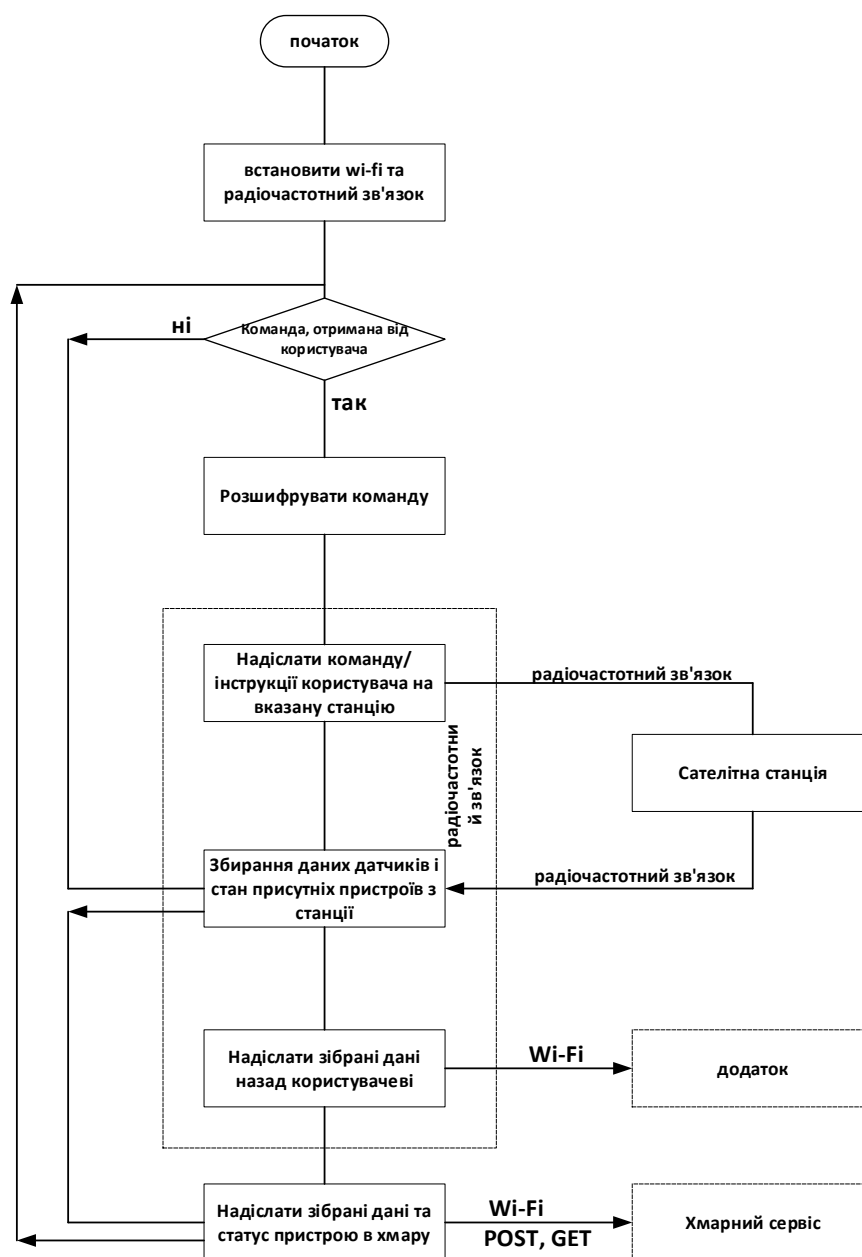


Рисунок 2.6 - схема потоків даних базової станції

Спочатку базова станція встановлює два типи з'єднання – це зв'язок по Wi-Fi для роботи з хмарою та мережею інтернет, а друге встановлює з сателітними модулями через радіочастотний зв'язок.

Далі базова станція перевіряє чи не прийшла команда від користувача, якщо ні то система переводиться в режим очікування і буде чекати на активні підключення.

Якщо ж є підключення і отримана команда, далі базова станція розшифровує команду для того , щоб передати її вже по радіочастотному каналу до сателітної станції.

Після надсилання команди базова станція чекає на відповідь від сателіта. Після отримання даних про параметри контрольованого об'єкта та стану самого реле та сателіта, модуль відсилає дані назад до базової станції, а базова станція вже надсилає оброблені дані користувачеві на додаток через TCP/IP протокол.

Крім того базова станція відсилає дані на хмарний сервіс через REST API. Таке дублювання потрібно тому, що є два варіанти використання модуля – коли користувач знаходиться безпосередньо в будинку і коли користувач знаходиться віддалено.

На рисунку 2.7 представлено загальну схему потоків даних роботи модуля вцілому.

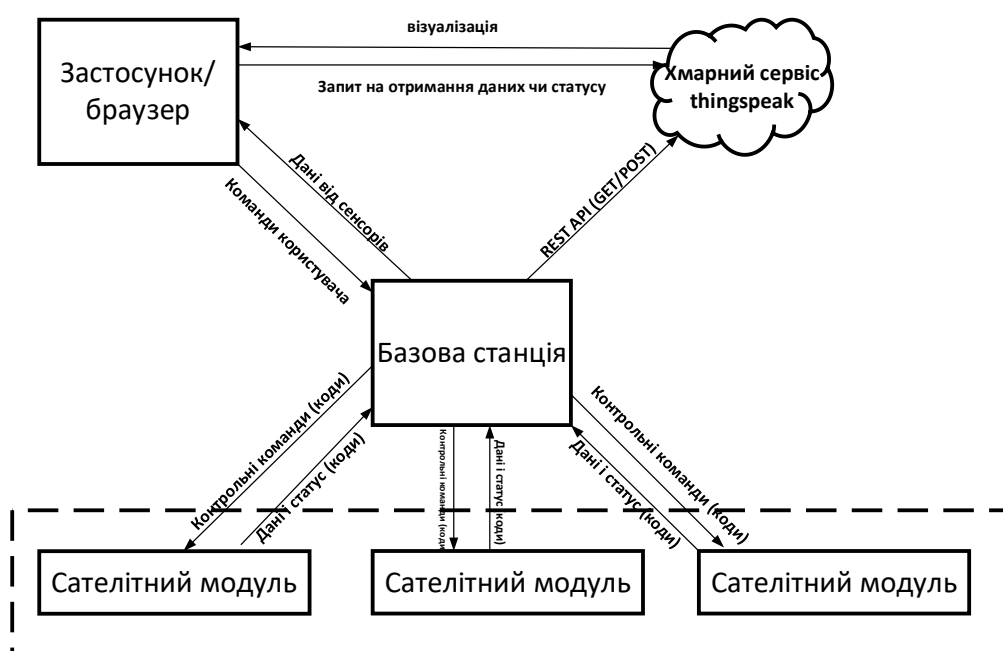


Рисунок 2.7 - Загальна схема потоків даних роботи модуля

На рисунку 2.8 показано алгоритм функціонування сателітного модуля.

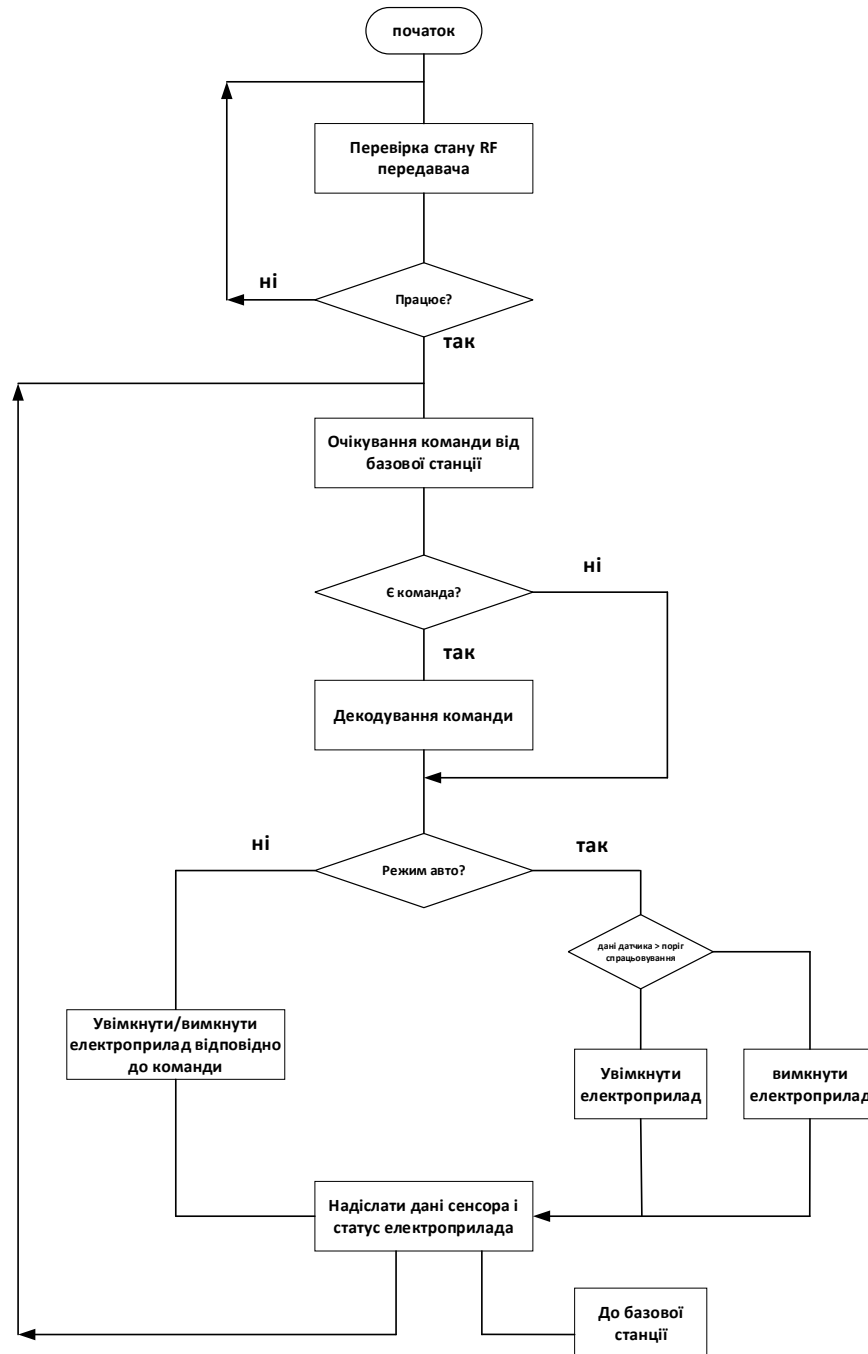


Рисунок 2.8 Алгоритм роботи сателітного модуля

В загальному сателіт працює наступним чином. Після ввімкнення самого мікроконтролера, він починає перевіряти чи працює контролер зв'язку, який знаходиться на тій самій платі що і головний МК. Після того, як визначено що передавач функціонує, модуль заходить в цикл опитування надходження нових команд.

Коли є команда від базової станції, модуль декодує команду і визначає в якому режимі він має працювати. Якщо це ручний режим то контролер надсилає до реле безпосередньо ту команду яка прийшла не перевіряючи сенсор. Якщо ж приходить команда з режимом авто, то контролер перевіряє спочатку стан реле і сенсора а тоді відповідно до стану міняє його на протилежний. На наступному кроці мікроконтролер формує відповідь (пакет) де прописано стан електроприладу та команда, яка була остання і відсилає по радіоканалу відповідь до базової станції, після чого переходить знову в режим очікування нової команди.

2.3 Інформаційне забезпечення модуля домашньої автоматизації

Діаграма варіантів використання автоматизації розумного будинку (рисунок 2.9) описує різні функціональні можливості автоматизації розумного будинку. Перший варіант використання - це управління енергоспоживанням. Це передбачає моніторинг використання енергії різними приладами та пристроями в будинку і надання рекомендацій щодо зменшення енергоспоживання.

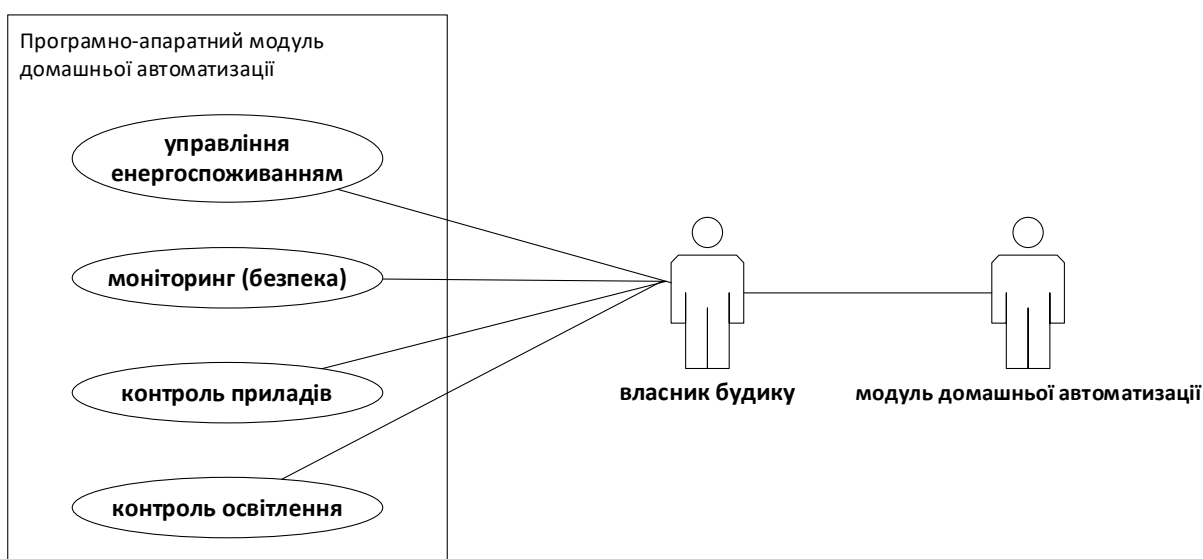


Рисунок 2.9 - Діаграма варіантів використання автоматизації розумного будинку

Система автоматизації розумного будинку також може контролювати використання енергоємних приладів, таких як кондиціонери або обігрівачі, щоб зменшити споживання енергії.

Другий варіант використання - моніторинг системи безпеки. Це передбачає моніторинг домашньої системи безпеки за допомогою датчиків і камер, щоб забезпечити безпеку будинку. Система розумної домашньої автоматизації може сповіщати власників будинків про будь-які порушення безпеки, наприклад, про відчинення дверей чи вікна або виявлення руху всередині будинку. Крім того, систему можна запрограмувати на автоматичне замикання дверей або ввімкнення світла, коли власник будинку відсутній, щоб відлякати грабіжників.

Третій варіант використання - це контроль температури. Це передбачає контроль температури в будинку за допомогою розумних термостатів, які можна запрограмувати на регулювання температури відповідно до уподобань або розкладу власника будинку. Система автоматизації розумного будинку також може регулювати температуру в залежності від погоди за вікном або присутності в будинку, зменшуючи споживання енергії, підтримуючи при цьому комфортну температуру.

Четвертий варіант використання - керування освітленням. Це передбачає керування освітленням будинку за допомогою розумних вимикачів, які можна запрограмувати на ввімкнення або вимкнення за розкладом домовласника або присутності в будинку. Система також може регулювати яскравість або колір світла залежно від часу доби або активності в будинку.

2.3.1 Діаграма послідовності

Діаграма послідовності - це діаграма взаємодії, яка показує, як процеси взаємодіють один з одним і який їхній порядок. Діаграма послідовності показує взаємодію об'єктів, розташованих у часовій послідовності. Вона зображує об'єкти та класи, що беруть участь у сценарії, а також послідовність повідомлень, якими обмінюються об'єкти, необхідних для виконання

					ДП.КН.8091600.064.ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

функціональності сценарію. На діаграмі нижче (рисунок 2.10) показано, як команди передаються від одного модуля до іншого модуля для керування приладами. Користувач надсилає команди через андроїд додаток або веб-браузер, який перетворює їх спеціальні кодові послідовності, а потім надсилає на базову станцію для керування пристроями.

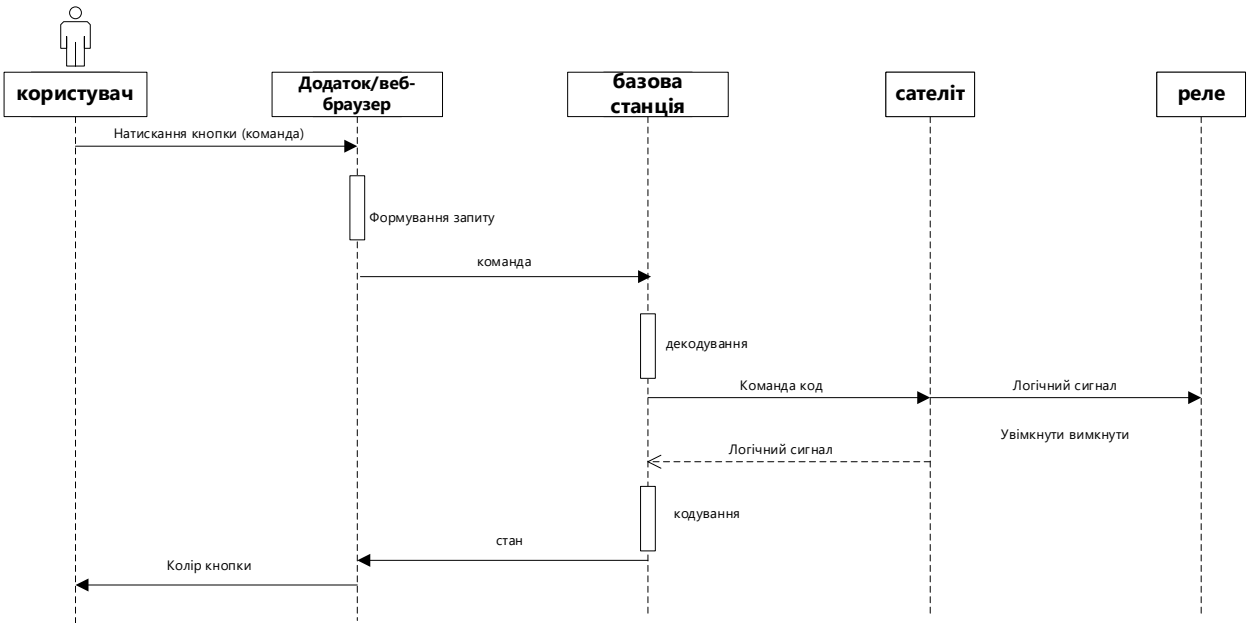


Рисунок 2.10 - Діаграма послідовності автоматизації

2.3.2 Діаграми потоків даних (DFD)

Дана діаграма являє собою графічне представлення системи, яке показує входи в систему, обробку на входах, виходи системи, а також внутрішні сховища даних. DFD ілюструє серію перетворень або обчислень, що виконуються над об'єктами або системою.

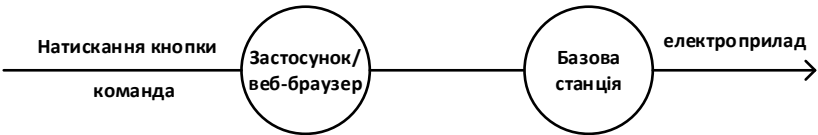


Рисунок 2.11 - Діаграма потоків даних

На рисунку нижче показано DFD рівня 1, де команди користувача, що подаються додатку для Android чи веб-браузеру, перетворюються на текстові REST API запити та команди, які передаються на базову станцію, потім сигнали з базової станції передаються супутникам надсилаються на прилади, якими керують.

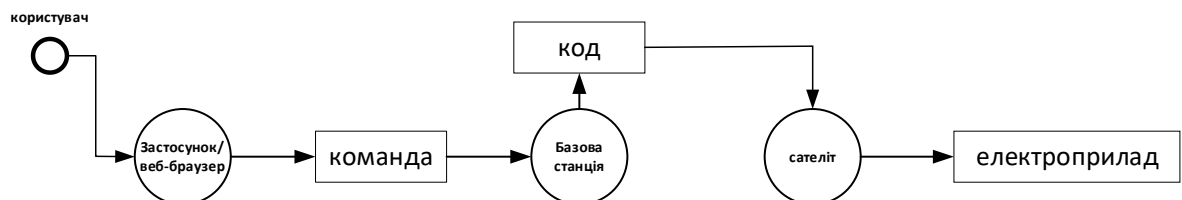


Рисунок 2.12 – Діаграма DFD рівня 1

Також на рисунку 2.13 показано DFD 2-го рівня, де команди, перетворені в запити за допомогою додатку або веб-браузера, передаються на апаратний модуль. База даних перевіряється щосекунди, щоб побачити, чи база даних була оновлена або змінена, після чого відповідно до команд здійснюється керування вентилятором або лампочкою.

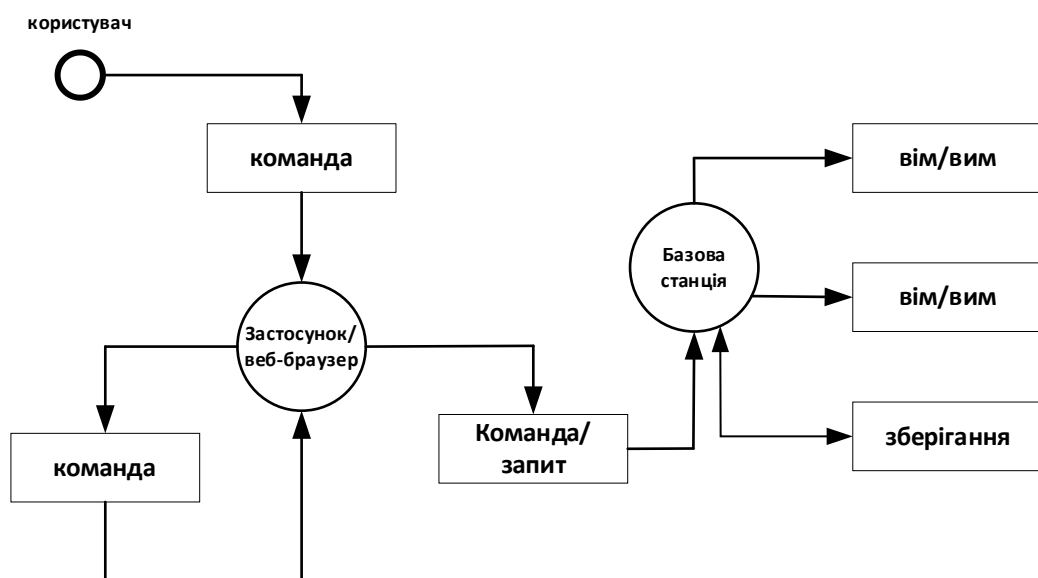


Рисунок 2.13 – Діаграма DFD рівня 2

З ПРОГРАМНО-ТЕХНОЛОГІЧНЕ ЗАБЕЗПЕЧЕННЯ ПРОГРАМНО-АПАРАТНО МОДУЛЯ ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ

3.1 Аналіз програмного та апаратного забезпечення системи

Так як модуль складається з двох частин, а саме апаратної і програмної частини. Нижче буде описано та підібрано відповідний інструментарій для реалізації модуля.

3.1.1 Сенсор освітлення LM393

LM393[29] - це інтегральна мікросхема з двома вбудованими операційними підсилювачами (рисунок 3.1), які використовують одне джерело живлення для виконання різних завдань. Крім того, вони також можуть застосовувати в своїй роботі роздільне живлення.

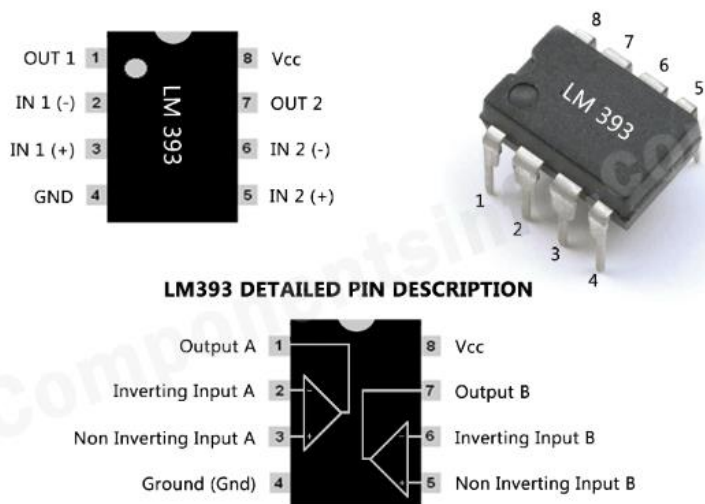


Рисунок 3.1 - Сенсор освітлення LM393

Характеристики мікросхеми LM393 мають наступні параметри і значення:

- Одиарна напруга живлення - від 2В до 36В,
- Диференціальна напруга і/р - 36В,
- Корпус - DIP і SOIC 8 pin,

- Струм стоку - 0,4 мА,
- Температура зберігання - від -65°C/Вт до 150°C/Вт,
- Температура виводів - 260°C,
- Розсіювана потужність - 660 мВт,
- Роздільне живлення - від $\pm 1\text{В}$ до $\pm 18\text{В}$, і

На рисунку 3.2 представлено збірку на базі фоторезистивного сенсора, LM393 та плати Arduino/

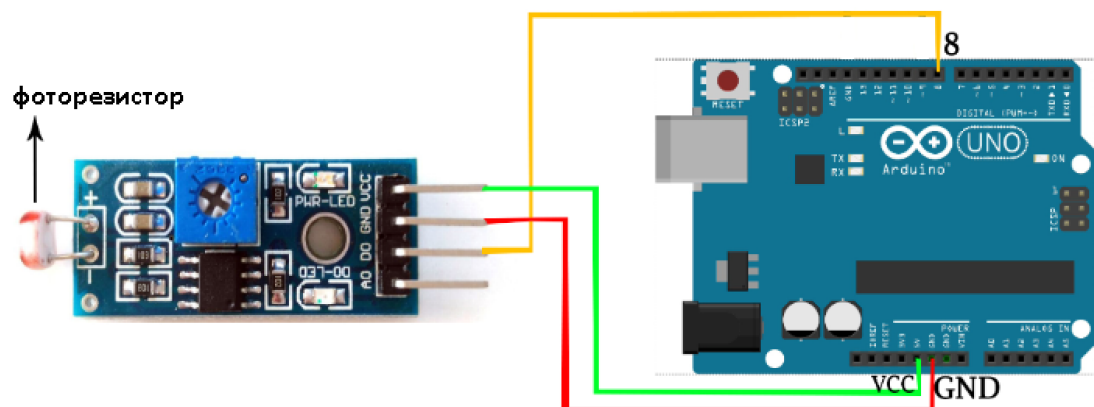


Рисунок 3.2 – Схема підключення LM393 до Arduino

3.1.2 Модуль термістора NTC LM393

Терморезисторний модуль NTC (рисунок 3.3) має датчик температури NTC, який дає єдиний цифровий вихід, що дозволяє швидко виявити перегрів і зниження температури. Поріг виявлення температури регулюється за допомогою потенціометра.

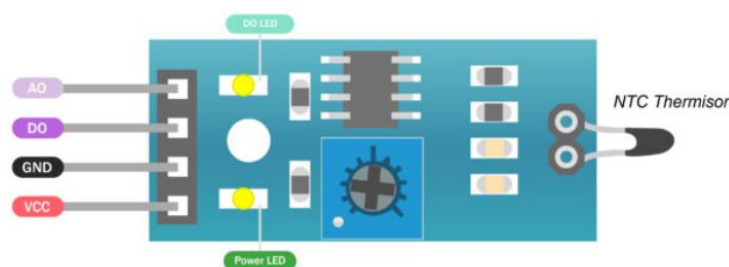


Рисунок 3.3 - Модуль термістора NTC LM393

Вихід DO можна (див. рис. 3.2, 3.3) безпосередньо підключити до мікроконтролера, і він видає напругу високого або низького рівня в залежності від зміни температури навколишнього середовища. Також вихід DO можна використовувати безпосередньо для керування релейним модулем або як перемикач термостата для керування зовнішнім обладнанням, наприклад, вентилятором охолодження.

3.1.3 Інфрачервоний датчик руху HC-SR501 для Arduino

HC-SR501 [30] - це PIR-сенсор (рисунок 3.4), що означає пасивний інфрачервоний датчик. Він використовує інфрачервоні хвилі для виявлення об'єкта. Це пристрій автоматичного контролю, має велику чутливість і високу надійність. Застосовується в пристроях автоматичного керування, де необхідно здійснювати виявлення руху.



Рисунок 3.4 - Пасивний інфрачервоний датчик

3.1.4 Сенсор диму MQ-2

MQ-2— це можна сказати універсальний датчик, тому що може виявляти не тільки газ або дим, але й зріджений газ, алкоголь, пропан, водень, метан і чадний газ. Модуль датчика виявлення газу MQ-2 має чотири виводи VCC, GND, Aout і Dout, які можна використовувати для отримання необхідної інформації з датчика, Схема підключення датчика виявлення газу MQ-2 наведена нижче.

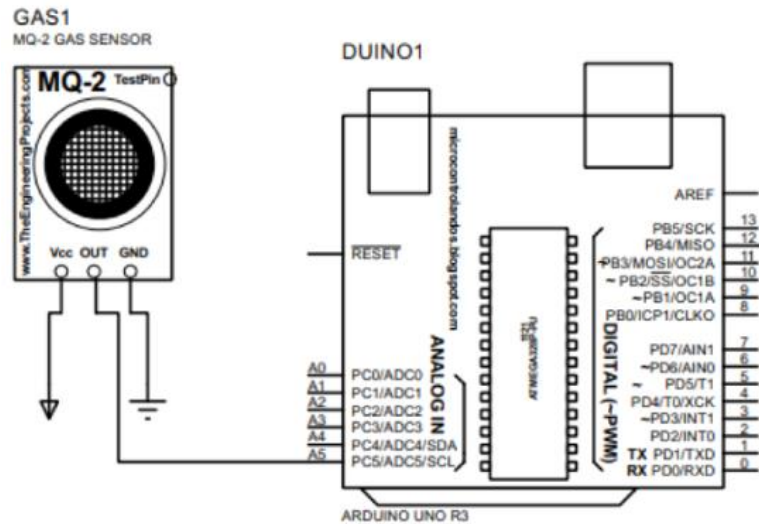


Рисунок 3.5 – Підключення сенсору газу

3.1.5 Модуль радіозв'язку nRF24L01

NRF24L01 [31] - це бездротовий радіочастотний прийомопередавач (рисунок 3.6), де кожен модуль може надсилати та приймати дані. Оскільки він працює в діапазоні 2,4 ГГц ISM, технологія дозволена для використання в інженерних цілях майже у всіх країнах. Кожен модуль має діапазон адрес 125 для зв'язку з іншими 6 модулями, а також дозволяє декільком бездротовим пристроям зв'язуватися один з одним у визначеному місці. Тому mesh-мережі та інші типи мереж використовують цей модуль.

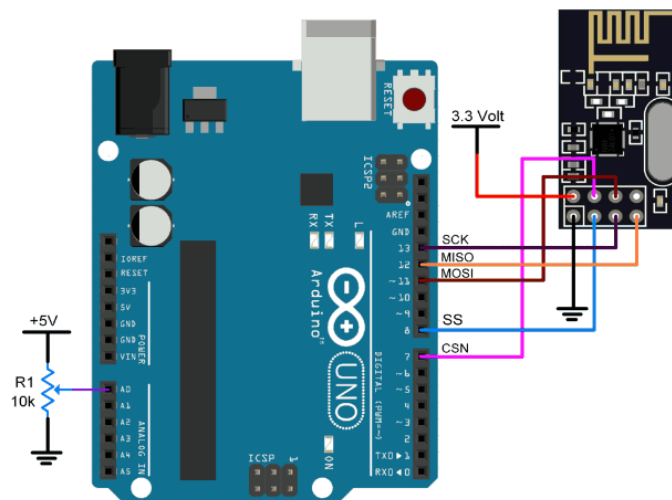


Рисунок 3.6 – Схема підключення nRF24L01 та Arduino

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		38

Для організації зв'язку по мережі Інтернет було підбрано 2 варіанти - WiFi Плата NodeMCU ESP8266 (CP2102) або ESP8266 s Arduino (рисунок 3.7).

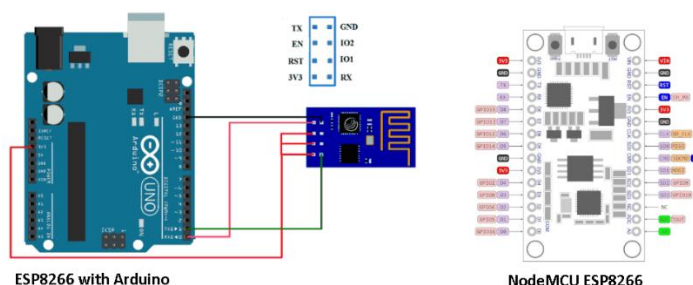


Рисунок 3.7 - ESP8266 та його реалізації

Як видно з рисунку, по суті великої різниці немає яку реалізацію вибирати і все буде залежати від ціни проекту та кількості обладнання.

3.1.6 Вибір програмного забезпечення

Для програмування усіх частин що пов'язані з радіопередачею, мережевою роботою та отриманням даних з сенсорів, буде використовуватись середовище Arduino cc. (рисунок 3.8).

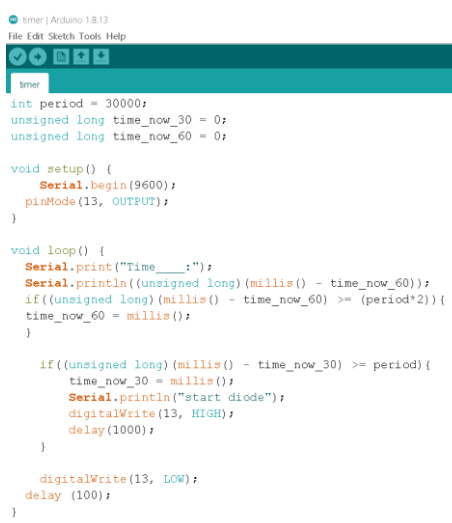


Рисунок 3.8 - Вікно середовища Arduino Software (IDE)

Інтегроване середовище розробки Arduino - або програмне забезпечення Arduino (IDE) - містить текстовий редактор для написання коду, область повідомлень, текстову консоль, панель інструментів з кнопками для загальних функцій і ряд меню. Воно підключається до апаратного забезпечення Arduino для завантаження програм і комунікації з ними.

Програми, написані за допомогою Arduino Software (IDE), називаються скетчами. Скетчі пишуться у текстовому редакторі і зберігаються з розширенням .ino. Консоль відображає текст, виведений програмним забезпеченням Arduino (IDE).

Наприклад для того щоб почати писати код в даному середовищі під nodeMCU потрібно тільки вибрати в середовищі відповідну бібліотеку або завантажити її після чого вибрати тим контролера і далі можна вже програмувати не Arduino а ESP32

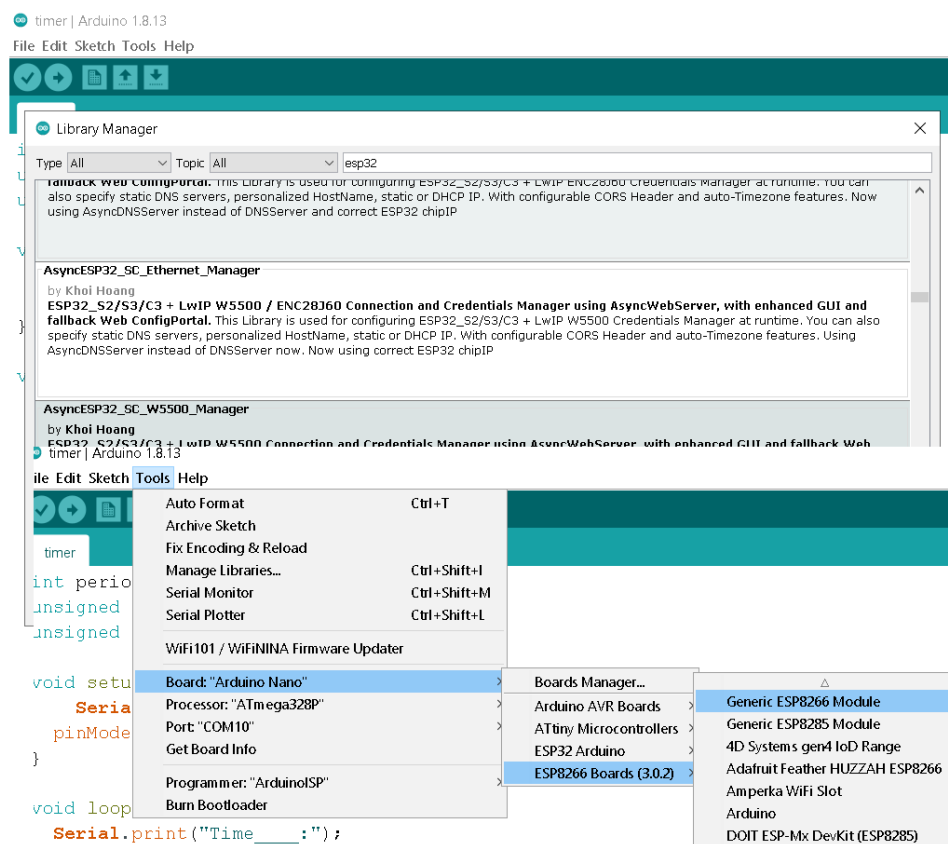


Рисунок 3.9 – Налаштування модулів в Arduino (IDE)

Для зберігання даних і аналізу було вибрано хмарний сервіс ThingSpeak (рисунок 3.10)

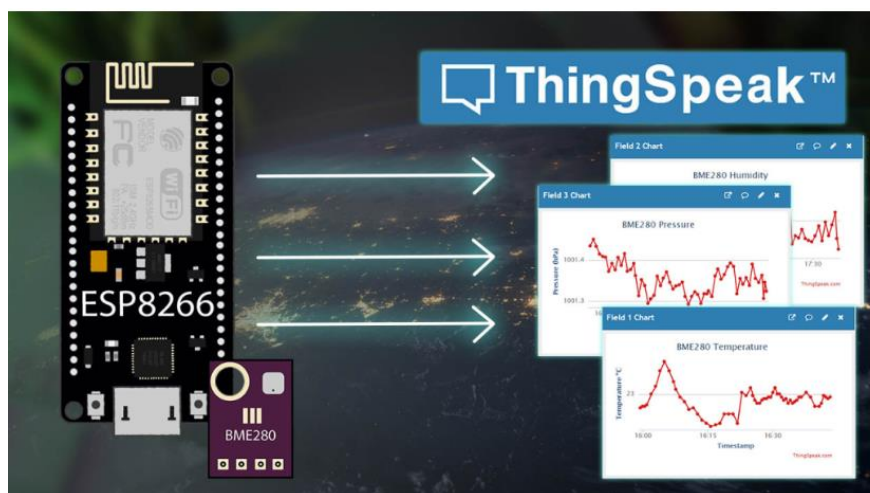


Рисунок 3.10 - Сервіс ThingSpeak

Програмне забезпечення ThingSpeak IoT [32] - це сервіс аналітичної платформи, який дозволяє користувачам візуалізувати, об'єднувати та аналізувати потоки даних в режимі реального часу.

Компанії використовують програмне забезпечення ThingSpeak IoT для реєстрації додатків, відстеження місцезнаходження, моніторингу соціальних мереж, онлайн-аналізу та обробки даних в режимі реального часу. Професіонали використовують платформу для створення прототипів та концепцій Інтернету речей, які потребують аналітики.

Програмне забезпечення ThingSpeak IoT пропонує рішення для декількох вертикальних застосувань, таких як домашня автоматизація, охорона навколишнього середовища, здоров'я, промисловість та моніторинг і управління автопарком.

Програмне забезпечення дозволяє користувачам надсилати дані з датчиків у хмару, аналізувати та візуалізувати їх за допомогою MATLAB, перш ніж запускати певні реакції.

Програмне забезпечення ThingSpeak IoT зберігає всю інформацію, яку користувачі надсилають у хмару, в одному центральному місці в режимі

онлайн. Аналітики мають доступ до своїх даних як для онлайн, так і для офлайн використання.

3.2 Програмне забезпечення для апаратних модулів та користувацького інтерфейсу

Для роботи з сенсором освітлення на базі LM393 реалізовано наступний код (рисунок 3.11).

```
1  const int sensorPin = A0;
2  const int ledPin = 13;
3  void setup() {
4      pinMode(ledPin, OUTPUT);
5      Serial.begin(9600);
6  }
7  void loop() {
8      int sensorValue = analogRead(sensorPin);
9      int brightness = map(sensorValue, 0, 1023, 0, 255);
10     analogWrite(ledPin, brightness);
11     Serial.println(sensorValue);
12     delay(500);
13 }
```

Рисунок 3.11 – Код для температурного сенсора

У цьому коді датчик освітленості LM393 підключено до аналогового виводу A0 Arduino. Значення датчика зчитується за допомогою функції `analogRead()`, яка повертає значення від 0 до 1023, що представляє аналогову напругу. Потім це значення відображається в діапазон яскравості від 0 до 255 за допомогою функції `map()`. Крім того, значення датчика виводиться на послідовний порт за допомогою функції `Serial.println()` для моніторингу.

Код для модуля термістора NTC LM393 дуже схожий на попередній, але є деякі зміни в плані обробки результатів від АЦП. У РТС-термісторів опір зростає з підвищенням температури, як правило, через посилення збудження теплової решітки.

Фактичні значення опору конкретного NTC-термістора отримують шляхом множення відношення $RT/R25$ (табличне значення) на значення опору

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

при 25 °С, яке вказано в технічному паспорті. Щоб виміряти опір NTC-термістора, спочатку вимірюється напруга з дільника напруги. Рівняння дільника напруги наступне.

$$V_{out} = V_{in} * [R2 / (R1 + R2)] \quad (1)$$

Оскільки відомо V_{in} , $R1$ і V_{out} , можна розрахувати значення NTC-термістора $R2$ за допомогою наступного рівняння

$$R2 = (V_{out} * R1) / (V_{in} - V_{out}) \quad (2)$$

На рисунку 3.12 представлено код для температурного сенсора

```

1  #define ntc_pin A0
2  #define vd_power_pin 2
3  #define nominal_resistance 10000
4  #define nominal_temperature 25
5  #define samplingrate 5
6  #define beta 3950
7  #define Rref 10000
8  int samples = 0;
9  void setup(void) {
10     pinMode(vd_power_pin, OUTPUT);
11     Serial.begin(9600);
12 }
13 void loop(void) {
14     uint8_t i;
15     float average;
16     sample = 0;
17     digitalWrite(vd_power_pin, HIGH);
18     for (i=0; i< samplingrate; i++) {
19         samples += analogRead(ntc_pin);
20         delay(10);
21     }
22     digitalWrite(vd_power_pin, LOW);
23     average = 0;
24     average = samples / samplingrate;
25     Serial.print("ADC readings ");
26     Serial.println(average);
27     // Calculate NTC resistance
28     average = 1023 / average - 1;
29     average = Rref / average;
30     Serial.print("Thermistor resistance ");
31     Serial.println(average);
32     float temperature;
33     temperature = average / nominal_resistance;
34     temperature = log(steinhart);
35     temperature /= beta;
36     temperature += 1.0 / (nominal_temperature + 273.15);
37     temperature = 1.0 / temperature;
38     temperature -= 273.15;
39     Serial.print("Temperature ");
40     Serial.print(temperature);
41     Serial.println(" *C");
42     delay(1000);
43 }

```

Рисунок 3.12 Реалізація температурного сенсора

В даному коді спочатку визначаються потрібні піни і константи, включаючи пін для зчитування напруги від термістора (ntc_pin), пін для живлення термістора (vd_power_pin), номінальний опір термістора

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

(nominal_resistance), номінальна температура (nominal_temperature), час зчитування (samplingrate), коефіцієнт бета (beta) і опір опорного резистора (Rref). В функції setup() встановлюється режим піна vd_power_pin як вихідний, а також встановлюється швидкість передачі даних через Serial порт. У функції loop() зчитуються значення напруги від термістора. Перед зчитуванням значень термістора, підключеного до ntc_pin, пін vd_power_pin встановлюється в HIGH для живлення термістора. Проводиться серія зчитувань аналогового сигналу з ntc_pin протягом samplingrate разів. Значення зберігаються і додаються разом. Після цього пін vd_power_pin встановлюється в LOW для вимкнення живлення термістора. Знаходиться середнє значення отриманих зчитувань. Виконуються обчислення для знаходження опору термістора, використовуючи знайдене середнє значення. Обчислюється опір термістора за допомогою формули стандартної серії резисторів. Знайдений опір використовується для розрахунку температури за допомогою формули Стейнгарта-Гартнера. Результат виводиться на Serial монітор.

На рисунку 3.13 представлено код для сенсора руху.

```

1  int ledPin = 13;
2  int inputPin = 8;
3  int pirState = LOW;
4  int val = 0;
5  void setup() {
6      pinMode(ledPin, OUTPUT);
7      pinMode(inputPin, INPUT);
8      Serial.begin(9600);
9  }
10 void loop(){
11     val = digitalRead(inputPin);
12     if (val == HIGH)
13     {
14         digitalWrite(ledPin, HIGH);
15         if (pirState == LOW)
16         {
17             Serial.println("Motion detected!");
18             pirState = HIGH;
19         }
20     }
21     else
22     {
23         digitalWrite(ledPin, LOW);
24         if (pirState == HIGH)
25         {
26             Serial.println("Motion ended!");
27             pirState = LOW;
28         }
29     }
30 }

```

Рисунок 3.14 – Код програми роботи з сенсором руху

Спочатку визначаються змінні та піни. ledPin - пін для підключення світлодіода, inputPin - пін для зчитування сигналу з датчика руху PIR, pirState - змінна для відстеження стану датчика, val - змінна для збереження значення, отриманого з inputPin.

У функції setup() встановлюються режими пінів: ledPin встановлюється як вихідний для управління світлодіодом, inputPin встановлюється, як вхідний для отримання сигналу з датчика руху PIR. Також ініціалізується Serial порт з швидкістю передачі даних 9600 бод.

У функції loop() зчитується значення з inputPin за допомогою функції digitalRead(). Значення зберігається в змінну val.

Перевіряється, чи отримане значення val дорівнює HIGH, що вказує на виявлений рух.

Якщо значення val дорівнює HIGH, тоді світлодіод підключається, і якщо попередній стан датчика pirState був LOW (відсутність руху), то виводиться повідомлення "Motion detected!" через Serial порт, а змінна pirState встановлюється в HIGH.

Якщо значення val не дорівнює HIGH, тоді світлодіод вимикається, і якщо попередній стан датчика pirState був HIGH (рух був виявлений), то виводиться повідомлення "Motion ended!" через Serial порт, а змінна pirState встановлюється в LOW.

На рисунку 3.15 представлено код прийому передачі даних за допомогою радіочастотного зв'язку.

```

1  #include <SPI.h>
2  #include <nRF24L01.h>
3  #include <RF24.h>
4  RF24 radio(9, 10);
5  void setup() {
6      radio.begin();
7      radio.openWritingPipe(0xF0F0F0F0E1LL);
8  }
9  void loop() {
10     char text[] = "Hello, Arduino!";
11     radio.write(&text, sizeof(text));
12     delay(1000);
13 }
14
15 #include <SPI.h>
16 #include <nRF24L01.h>
17 #include <RF24.h>
18 RF24 radio(9, 10);
19 void setup() {
20     Serial.begin(9600);
21     radio.begin();
22     radio.openReadingPipe(1, 0xF0F0F0F0E1LL);
23     radio.startListening();
24 }
25 void loop() {
26     if (radio.available()) {
27         char text[32] = "";
28         radio.read(&text, sizeof(text));
29         Serial.println(text);
30     }
31 }

```

Рисунок 3.15 – Радіочастотна комунікація

Ці два коди використовуються для встановлення бездротового зв'язку між двома Arduino за допомогою радіочастотного модуля NRF24L01. Перший код (від #include до delay(1000);) виконується на передавачі (Transmitter) Arduino, а другий код (від #include до Serial.println(text);) виконується на приймачі (Receiver) Arduino.

Перший код (від #include до delay(1000);):

1. Включаються необхідні бібліотеки SPI, nRF24L01 і RF24.
2. Створюється об'єкт класу RF24 з ініціалізацією пінів для зв'язку з модулем NRF24L01 (9 і 10).
3. У функції setup() починається робота з модулем NRF24L01 за допомогою radio.begin(). Також встановлюється адреса для відправки даних за допомогою radio.openWritingPipe().
4. У функції loop() створюється масив тексту char text[], який містить "Hello, Arduino!".
5. За допомогою radio.write() дані з масиву text[] передаються через бездротовий модуль NRF24L01.
6. Після передачі затримка delay(1000) на 1 секунду перед повторним виконанням циклу.

Другий код (від #include до Serial.println(text);):

Включаються необхідні бібліотеки SPI, nRF24L01 і RF24.

1. Створюється об'єкт класу RF24 з ініціалізацією пінів для зв'язку з модулем NRF24L01 (9 і 10).
2. У функції setup() починається робота з модулем NRF24L01 за допомогою radio.begin(). Також встановлюється адреса для прийому даних за допомогою radio.openReadingPipe() і radio.startListening().
3. У функції loop() перевіряється наявність доступних даних для прийому за допомогою radio.available().
4. Якщо є доступні дані, то створюється масив тексту char text[32], який використовується для збереження отриманого тексту.

					ДП.КН.8091600.064.ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

5. За допомогою `radio.read()` дані з модуля NRF24L01 зчитуються у масив `text[]`.

6. Отриманий текст виводиться на послідовний порт `Serial.println(text)`.

7. Таким чином, перший код відправляє текстове повідомлення "Hello, Arduino!" з передавача на приймач через бездротовий модуль NRF24L01, а другий код приймає це повідомлення і виводить його на послідовний порт.

Для відправки даних на сервіс використовується базова станція, опис коду якої представлено нижче.

Спочатку підключаються відповідні бібліотеки роботи з ESP8266, WiFi – клієнта, бібліотеки для ThingSpeak

```
1 #include <ESP8266WiFi.h>
2 #include <WiFiClient.h>
3 #include <ThingSpeak.h>
```

Рисунок 3.16 – Підключення бібліотек

Потім визначаються облікові дані Wi-Fi та налаштування ThingSpeak:

```
4 const char* ssid = "Home";
5 const char* password = "*****";
6 const char* server = "api.thingspeak.com";
7 const char* apiKey = "xxxxxxxxxxxxxxxx";
8 const int sensorPin = A0;
```

Рисунок 3.17 – Налаштування підключення

Налаштування з'єднання Wi-Fi та ThingSpeak.

```
1 void setup() {
2   WiFi.begin(ssid, password);
3   while (WiFi.status() != WL_CONNECTED) {
4     delay(500);
5     Serial.print(".");
6   }
7   Serial.println("");
8   Serial.println("WiFi connected");
9   ThingSpeak.begin(client);
10  Serial.println("Setup complete");
11 }
```

Рисунок 3.18 – Wi-Fi та ThingSpeak

У функції `setup()` код підключається до мережі Wi-Fi, використовуючи надані облікові дані. Він чекає, поки з'єднання буде встановлено (`WiFi.status() == WL_CONNECTED`). Потім він ініціалізує ThingSpeak, викликаючи `ThingSpeak.begin(client)`. Це встановлює зв'язок з сервером ThingSpeak.

Надсилення даних датчиків до ThingSpeak.

```
void loop() {
  int sensorValue = analogRead(sensorPin);
  ThingSpeak.writeField(channelId, 1, sensorValue, apiKey);
  delay(20000);
}
```

Рисунок 3.19 – Дані від базової станції до сервісу

У функції `loop()` код зчитує значення датчика за допомогою `analogRead(sensorPin)`. Потім він надсилає значення датчика до ThingSpeak за допомогою `ThingSpeak.writeField()` з номером поля (1 у цьому випадку), значенням датчика та ключем API.

3.3 Тестування модуля

Після опису основних модулів було проведено кінцеве тестування.

На рисунку 3.20 показано результат роботи результату роботи коду для температурного сенсора NTC LM393.

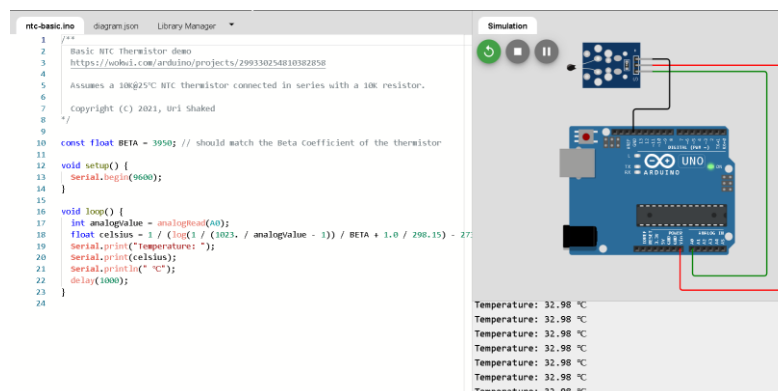


Рисунок 3.20 – Симуляція роботи температурного сенсора

Для симуляції радіозв'язку було вибрано пакет proteus та просимульовано роботу з газовим сенсором, як показано на рисунку 3.21.

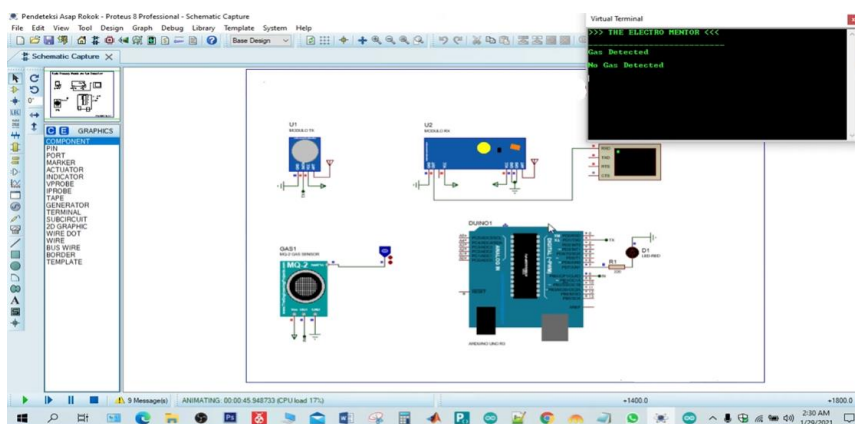


Рисунок 3.21 – Тестування радіозв'язку

Після тестування апаратної частини було протестовано інтерфейс користувача, який попередньо реалізувався на платформі thingspeak (рисунк 3.22).

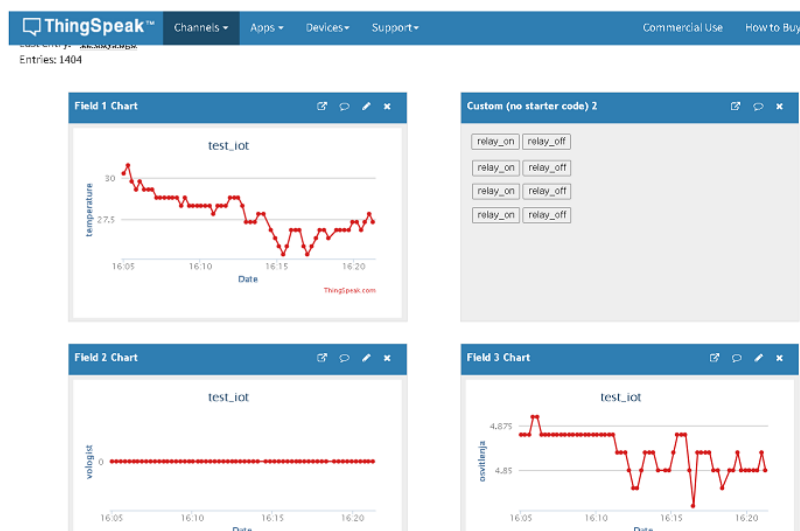


Рисунок 3.22- Тестування графічного інтерфейсу

Для отримання команд від кнопок які представлено вище використано можливість самого сервісу та додатка TalkBack. TalkBack дає змогу будь-якому пристрою реагувати на команди, що стоять у черзі. Наприклад, якщо у є двері, обладнані Wi-Fi і датчиком руху, можна поставити в чергу команди на

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

відчинення та зачинення дверей. На рисунку 3.23 показано результат роботи даного застосунку.

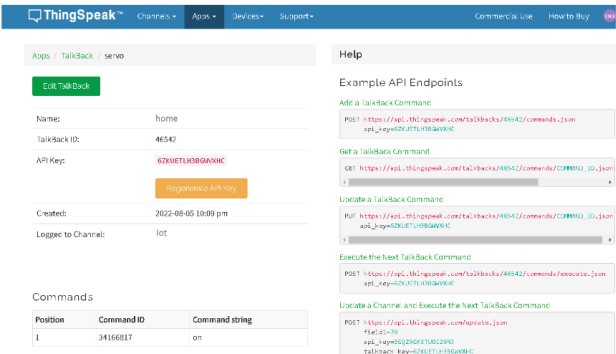


Рисунок 3.23 – Додавання команди до базової станції через веб інтерфейс

На головному екрані застосунку для смартфона є можливість відкрити інтерфейси кожної з сателітних станцій та базової станції, а також текстове поле для введення IP-адреси Wi-Fi модуля або зовнішньої IP-адреси домашнього роутера, якщо увімкнено переадресацію портів.

Інтерфейс базової станції має кнопки увімкнення/вимкнення підключених до неї пристроїв (рисунок 3.24).

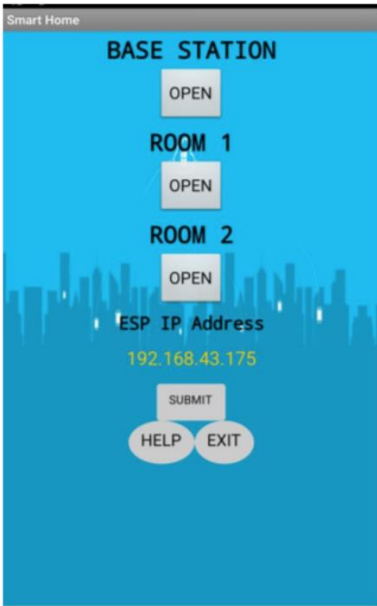


Рисунок 3.24 – Екран де працює сателітна станція

Інтерфейс сателітної станції також має кнопки ON/OFF (рисунок 3.25) для роботи станції в ручному режимі, а також кнопки AUTO для роботи станції в автоматичному режимі, коли керування приладами відбувається на основі показань датчиків. У запропонованому модулі натискання кнопки автоматичного увімкнення світла вмикає світло, якщо датчик освітленості вказує на відсутність світла в приміщенні, в іншому випадку світло вимикається. При натисканні кнопки автоматичного вмикання вентилятора вмикається вентилятор, якщо датчик температури зафіксував температуру більше 32С, в іншому випадку вентилятор вимкнений.

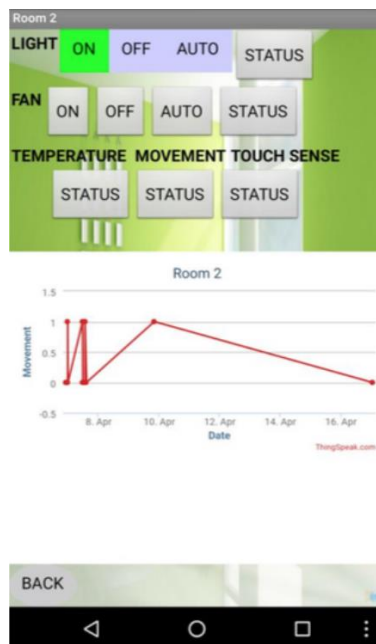


Рисунок 3.25 – Інтерфейс сателітної станції

При натисканні кнопки керування на будь-якому з інтерфейсів на базову станцію надсилається 3-байтовий код (3 цифри) між двома символами "\$". Перша цифра вказує на одну зі станцій, друга цифра вказує на пристрій на цій станції, а третя цифра вказує на дію (ON/OFF/AUTO). Ці дані з програми надходять через Інтернет до модуля Wi-Fi на базовій станції, де вони декодуються мікроконтролером, визначаються відповідні дії та надсилаються на супутникову станцію через модуль радіочастотного приймача.

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

При натисканні кнопки STATUS для будь-якого з параметрів, відповідні показники з датчиків, які були збережені на хмарній платформі ThingSpeak завантажуються і відображаються на екрані.

					ДП.КН.8091600.064.ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У результаті виконання дипломного проекту було проаналізовано сучасні підходи до автоматизації будинків, вплив який зробив Інтернет речей на автоматизацію. Розглянуто принципи побудови та функціонування розумних будинків та застосування в них автоматизації. Проведено аналіз існуючих рішень в області IoT для автоматизації будинків. Розглянуто та проаналізовано ключові модулі, які забезпечують автоматизацію розумного будинку.

Також було розроблено архітектуру програмно-апаратного модуля домашньої автоматизації, який включає в себе базову станцію для управління та передачі даних від сателітних станцій. Сателітні станції які за допомогою радіочастотного зв'язку обмінюються командами, станом та даними з базовою станцією, крім того сателіти виконують команди які приходять від базової станції у вигляді включити, або виключити прилад.

Для кожного з підмодулів було описано алгоритми функціонування та розроблено відповідне програмне забезпечення, як для апаратного так і для серверного забезпечення. Також було розроблено графічний інтерфейс для мобільного застосунку, хоча користувач може так само управляти модулем через веб інтерфейс.

Після збірки усіх компонентів було проведено тестування модуля вцілому. Після тестувань були виявлені деякі недоліки, а саме нестабільність роботи Arduino біля деяких електроприладів, обмеження покриття сателітів і базової станції, обмеження по кількості збережених даних на хмарному сервісі.

Для вирішення даних недоліків потрібно, використовувати захищені контролери біля електроприладів, які впливають на роботу МК, додати можливість кабельного з'єднання, наприклад 1Wire, використати Raspberri Pi для розгортання власного сервісу, thingspeak лишити в якості тунелю або шлюзу на зовні.

					ДП.КН.8091600.064.ПЗ	Арк.
						53
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Swathi, D., & Rekha, V. S. D. (2019). Home automation based on IoT using Google assistant. International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), 6(1), 1-6.
2. Khan, A., Al-Zahrani, A., Al-Harbi, S., Al-Nashri, S., & Khan, I. A. (2018, February). Design of an IoT smart home system. In 2018 15th Learning and Technology Conference (L&T) (pp. 1-5). IEEE.
3. Bugeja, J., Jacobsson, A., & Davidsson, P. (2018). Smart connected homes. Internet of things A to Z: Technologies and applications, 359-384.
4. Mahzan, N. N., Wahid, N., Zin, N. M., Enzai, N. I. M., Noh, K. S. S. K. M., Omar, S., ... & Salleh, M. A. M. (2020, April). A Design of Smart IoT-Based College Room Using Arduino. In Journal of Physics: Conference Series (Vol. 1529, No. 2, p. 022045). IOP Publishing.
5. Jabbar, W. A., Kian, T. K., Ramli, R. M., Zubir, S. N., Zamrizaman, N. S., Balfaqih, M., ... & Alharbi, S. (2019). Design and fabrication of smart home with internet of things enabled automation system. IEEE access, 7, 144059-144074.
6. Hoque, M. A., & Davidson, C. (2019). Design and implementation of an IoT-based smart home security system. Int. J. Networked Distributed Comput., 7(2), 85-92.
7. Al-Kuwari, M., Ramadan, A., Ismael, Y., Al-Sughair, L., Gastli, A., & Benammar, M. (2018, April). Smart-home automation using IoT-based sensing and monitoring platform. In 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018) (pp. 1-6). IEEE.
8. Islam, M. M., Farook, M. N., Mostafa, S. M. G., & Arafat, Y. (2019, May). Design and implementation of an IoT based home automation. In 2019 1st international conference on advances in science, engineering and robotics technology (ICASERT) (pp. 1-5). IEEE.

9. Eini, R., Linkous, L., Zohrabi, N., & Abdelwahed, S. (2019, April). A testbed for a smart building: design and implementation. In Proceedings of the Fourth Workshop on International Science of Smart City Operations and Platforms Engineering (pp. 1-6).

10. Agarwal, K., Agarwal, A., & Misra, G. (2019, December). Review and performance analysis on wireless smart home and home automation using iot. In 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 629-633). IEEE.

11. Kumar, S., Swetha, S., Kiran, V. T., & Johri, P. (2018, September). IoT based smart home surveillance and automation. In 2018 International Conference on Computing, Power and Communication Technologies (GUCON) (pp. 786-790). IEEE.

12. Al-Kuwari, M., Ramadan, A., Ismael, Y., Al-Sughair, L., Gastli, A., & Benammar, M. (2018, April). Smart-home automation using IoT-based sensing and monitoring platform. In 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018) (pp. 1-6). IEEE.

13. Chen, C. C., Chen, L. H., & Guo, M. H. (2019, March). Integration of building interface and smart sensor network to control indoor air pollution through internet of things. In Proceedings of the 8th International Conference on Informatics, Environment, Energy and Applications (pp. 15-20).

14. Singh, H., Pallagani, V., Khandelwal, V., & Venkanna, U. (2018, March). IoT based smart home automation system using sensor node. In 2018 4th International Conference on Recent Advances in Information Technology (RAIT) (pp. 1-5). IEEE.

15. Sisavath, C., & Yu, L. (2021). Design and implementation of security system for smart home based on IOT technology. Procedia Computer Science, 183, 4-13.

					ДП.КН.8091600.064.ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

16. Nadafa, R. A., Hatturea, S. M., Bonala, V. M., & Naikb, S. P. (2020). Home security against human intrusion using Raspberry Pi. *Procedia Computer Science*, 167, 1811-1820.
17. Hassan, R. J., Zeebaree, S. R., Ameen, S. Y., Kak, S. F., Sadeeq, M. A., Ageed, Z. S., ... & Salih, A. A. (2021). State of art survey for iot effects on smart city technology: challenges, opportunities, and solutions. *Asian Journal of Research in Computer Science*, 8(3), 32-48.
18. Verma, A., Prakash, S., Srivastava, V., Kumar, A., & Mukhopadhyay, S. C. (2019). Sensing, controlling, and IoT infrastructure in smart building: a review. *IEEE Sensors Journal*, 19(20), 9036-9046.
19. Qiu, T., Chen, N., Li, K., Atiquzzaman, M., & Zhao, W. (2018). How can heterogeneous internet of things build our future: A survey. *IEEE Communications Surveys & Tutorials*, 20(3), 2011-2027.
20. Raj, J. S., & Ananthi, J. V. (2019). Automation using IoT in greenhouse environment. *Journal of Information Technology*, 1(01), 38-47.
21. Sarah, A., Ghozali, T., Giano, G., Mulyadi, M., Octaviani, S., & Hikmaturokhman, A. (2020, August). Learning IoT: basic experiments of home automation using ESP8266, Arduino and XBee. In *2020 IEEE International Conference on Smart Internet of Things (SmartIoT)* (pp. 290-294). IEEE.
22. Eleyan, A., & Fallon, J. (2020, October). IoT-based home automation using android application. In *2020 International Symposium on Networks, Computers and Communications (ISNCC)* (pp. 1-4). IEEE.
23. Ghael, H. D., Solanki, L., & Sahu, G. (2020). A Review Paper on Raspberry Pi and its Applications. *International Journal of Advances in Engineering and Management (IJAEM)*, 2(12), 4.
24. Hockney, R. W., & Jesshope, C. R. (2019). *Parallel Computers 2: architecture, programming and algorithms*. CRC Press.
25. Banzi, M., & Shiloh, M. (2022). *Getting started with Arduino*. Maker Media, Inc.

26. Alobaidy, H. A., Mandeep, J. S., Nordin, R., & Abdullah, N. F. (2020). A review on ZigBee based WSNs: concepts, infrastructure, applications, and challenges. *Int. J. Electr. Electron. Eng. Telecommun*, 9(3), 189-198.

27. Parihar, Y. S. (2019). Internet of things and nodemcu. *journal of emerging technologies and innovative research*, 6(6), 1085.

28. Mesquita, J., Guimarães, D., Pereira, C., Santos, F., & Almeida, L. (2018, September). Assessing the ESP8266 WiFi module for the Internet of Things. In *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)* (Vol. 1, pp. 784-791). IEEE.

29. Putra, R. A. (2019). Pemanfaatan LM393 IR Sensor Module Sebagai Pengukur Kecepatan Rotasi Berbasis Mikrokontroler. *Jurnal Hadron*, 1(1), 12-15.

30. Wahyuni, R., Rickyta, A., Rahmalisa, U., & Irawan, Y. (2021). Home security alarm using Wemos D1 and HC-SR501 sensor based telegram notification. *Journal of Robotics and Control (JRC)*, 2(3), 200-204.

31. Bakri, A. B., Adnan, R., & Ruslan, F. A. (2019, April). Wireless hand gesture controlled robotic arm via nrf24l01 transceiver. In *2019 IEEE 9th Symposium on Computer Applications & Industrial Electronics (ISCAIE)* (pp. 16-22). IEEE.

32. Mohamad, A. A., Jumaa, N. K., & Majeed, S. H. (2019). ThingSpeak cloud computing platform based ECG diagnose system. *International Journal of Computing and Digital Systems*, 8(01), 11-18.

33. Комар М.П., Саченко А.О., Васильків Н.М., Гладій Г.М., Коваль В.С. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за першим (бакалаврським) рівнем вищої освіти. – Тернопіль: ЗУНУ, 2021. – 56 с.

					ДП.КН.8091600.064.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

Додаток А

Лістинг програми радіочастотної комунікації

```

/***** RF RECEIVER LINK SETUP *****/
#include <VirtualWire.h>
//int RF_RX_PIN = 2;
#undef int
#undef abs
#undef double
#undef float
#undef round
/***** RF RECEIVER LINK SETUP *****/

/***** ALARM LED BUZZER SETUP *****/
#define BUZZER_PIN 6
#define LED_PIN 7
#define ERROR_LED 8
/***** ALARM LED BUZZER SETUP *****/

/***** LED SETUP *****/
#define FAN_LED 3
#define DOOR_LED 4
#define BULB_LED 5
boolean bBulb = false;
boolean bFan = false;
int iFan = 0;
int iDoor = 0;
/***** LED SETUP *****/

/***** MOTOR SETUP *****/
#define DOOR_MOTOR_A A0
#define DOOR_MOTOR_B A1
#define FAN_MOTOR_A A2
/***** MOTOR SETUP *****/

void setup()
{
    Serial.begin(9600);
    Serial.println("setup");
    // Initialise the IO and ISR
    vw_set_ptt_inverted(true); // Required for DR3100
    //vw_set_rx_pin(RF_RX_PIN); // Setup receive pin.
    vw_setup(2000); // Transmission speed in bits per second.
    vw_rx_start(); // Start the PLL receiver.

    // Set pinmode for LED, BUZZERS
    pinMode(BUZZER_PIN, OUTPUT);
    pinMode(LED_PIN, OUTPUT);
    pinMode(ERROR_LED, OUTPUT);
    pinMode(FAN_LED, OUTPUT);
    pinMode(DOOR_LED, OUTPUT);
}
```

					ДП.КН.8091600.064.ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    pinMode(BULB_LED, OUTPUT);
    bBulb = false;
    bFan = false;
    iDoor = 0;
}

void loop()
{
    uint8_t buf[VW_MAX_MESSAGE_LEN];
    uint8_t buflen = VW_MAX_MESSAGE_LEN;
    if(vw_get_message(buf, &buflen)) // non-blocking I/O
    {
        delay(500);
        ProcessCommand ( (char *) buf);
        delay(500);
    }

    if(bBulb == true)
    {
        digitalWrite(BULB_LED, HIGH);
    }

    if(bBulb == false)
    {
        digitalWrite(BULB_LED, LOW);
    }

    if(iFan == 1)
    {
        digitalWrite(FAN_LED, HIGH);
        RunFan();
    }

    if(iFan == 2)
    {
        StopFan();
        digitalWrite(FAN_LED, LOW);
        iFan = 0;
    }

    if(iDoor == 1 )
    {
        digitalWrite(DOOR_LED, HIGH);
        OpenDoor();
        digitalWrite(DOOR_LED, LOW);
        iDoor =0;
    }
    else if ( iDoor == 2)
    {
        digitalWrite(DOOR_LED, HIGH);
        CloseDoor();
        digitalWrite(DOOR_LED, LOW);
        iDoor =0;
    }
}

```

```

    }
    else
    {
        iDoor = 0;
        digitalWrite(DOOR_LED, LOW);
    }

    /*Serial.println("Fan, Door, Lights");
    Serial.print(iFan);
    Serial.print(",");
    Serial.print(iDoor);
    Serial.print(",");
    Serial.print(bBulb);*/
}

void ProcessCommand(char msg[])
{
    if (strlen(msg) == 0) { ErrorAlarm(); return; }

    Serial.println("Message Received-->");
    Serial.println(msg);
    // check if the string doesn't have valid string
    if ( compareTag(msg, '@') == false || compareTag(msg, '$')
== false) { ErrorAlarm(); return; }

    SucessAlarm();
    Serial.println("Command Received-->");
    Serial.println(msg);

    char cTag[5] = "";
    int index = 0;
    boolean batFound = false;
    boolean dollarFound = false;
    int iLoop = IndexOf(msg, '@');

    if(iLoop == -1){ ErrorAlarm(); return;}

    iLoop++;
    while (iLoop < strlen(msg))
    {
        if(msg[iLoop] == '$'){break;}
        else{cTag[index] = msg[iLoop]; index++;}

        iLoop ++;
    }

    String sTag = String(cTag);

    if      (sTag.equals("BON"))      {      bBulb      =      true;
Serial.println("Compared BON - 1");}

```

					ДП.КН.8091600.064.ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        if (sTag.equals("BOFF")) { bBulb = false;
Serial.println("Compared BOFF - 0");}

        if (sTag.equals("FON")) { iFan = 1; bFan = true;
Serial.println("Compared FON -1");}

        if (sTag.equals("FOFF")) { iFan = 2; bFan = false;
Serial.println("Compared FOFF - 0");}

        if (sTag.equals("DON")) { iDoor = 1;
Serial.println("Compared DON - 1" ); }

        if (sTag.equals("DOFF")) { iDoor = 2;
Serial.println("Compared DON - 2"); }

    }

    int IndexOf(char msg[], char tag)
    {
        if (strlen(msg) ==0) return false;
        boolean bFlag = false;
        int iIndex = -1;

        for( int i=0; i<strlen(msg); i++)
        {
            if(msg[i]== tag)
                iIndex = i;
        }

        return iIndex;
    }

    boolean compareTag (char msg[], char tag)
    {
        if (strlen(msg) ==0) return false;
        boolean bFlag = false;

        for( int i=0; i<strlen(msg); i++)
        {
            if(msg[i]== tag)
                bFlag = true;
        }

        return bFlag;
    }

    void RunFan()
    {
        Serial.println("Running Fan");
        analogWrite(FAN_MOTOR_A, 150);
    }

    void StopFan()

```

					ДП.КН.8091600.064.ПЗ	Арк.
						61
Змн.	Арк.	№ докум.	Підпис	Дата		

```

{
    Serial.println("Stopping Fan");
    analogWrite(FAN_MOTOR_A, 0);
    delay(50);
}

void OpenDoor()
{
    Serial.println("Open Door");
    analogWrite(DOOR_MOTOR_A, 150);
    analogWrite(DOOR_MOTOR_B, 0);
    delay(1000);
    analogWrite(DOOR_MOTOR_A, 0);
    analogWrite(DOOR_MOTOR_B, 0);
}

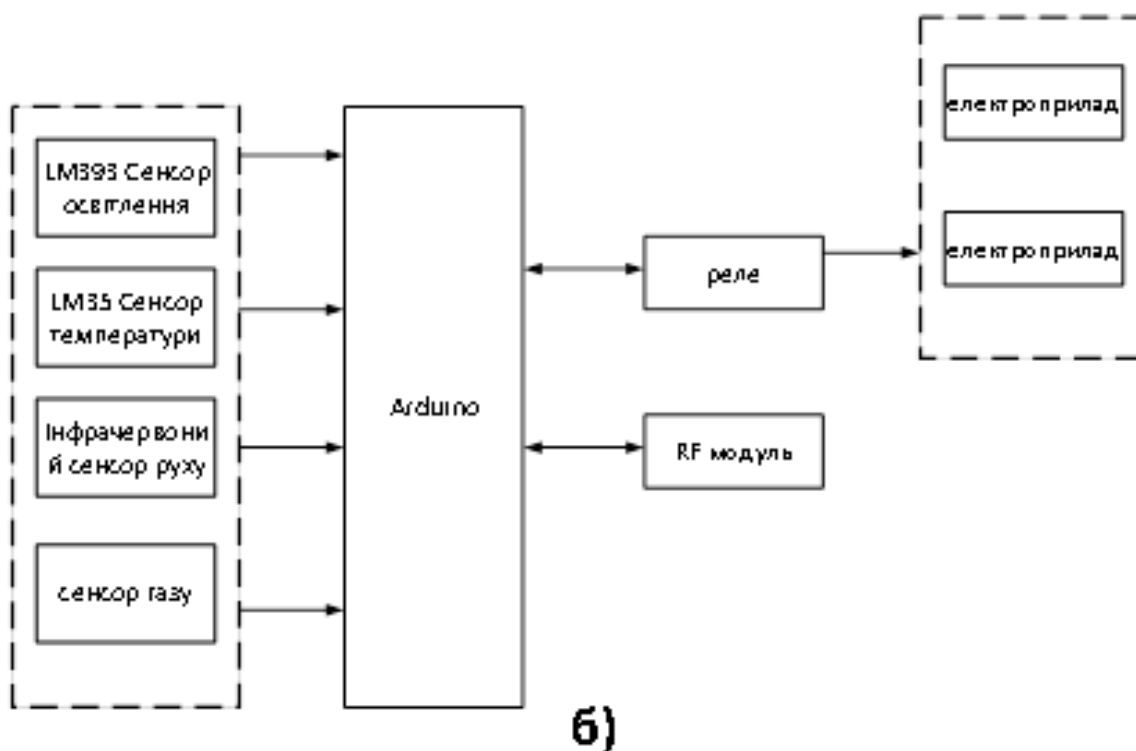
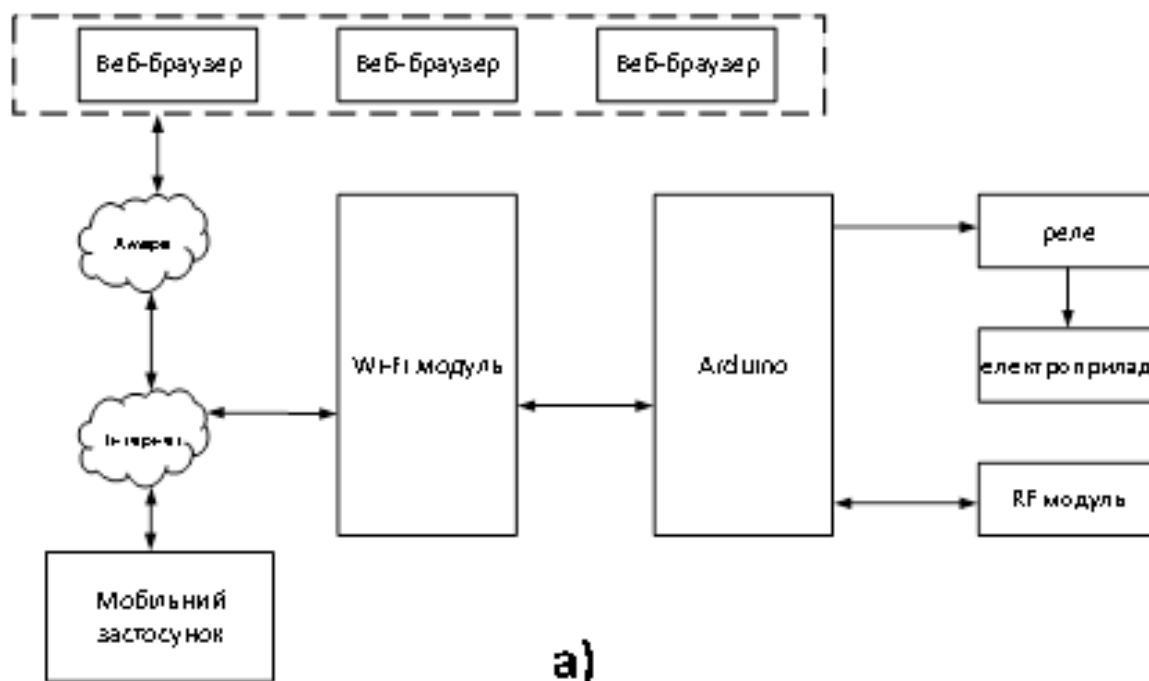
void CloseDoor()
{
    Serial.println("Close Door");
    analogWrite(DOOR_MOTOR_A, 0);
    analogWrite(DOOR_MOTOR_B, 150);
    delay(1000);
    analogWrite(DOOR_MOTOR_A, 0);
    analogWrite(DOOR_MOTOR_B, 0);
}

void SucessAlarm()
{
    digitalWrite(BUZZER_PIN, HIGH);
    digitalWrite(LED_PIN, HIGH);
    delay(500);
    digitalWrite(BUZZER_PIN, LOW);
    digitalWrite(LED_PIN, LOW);
}

void ErrorAlarm()
{
    digitalWrite(ERROR_LED, HIGH);
    digitalWrite(BUZZER_PIN, HIGH);
    delay(500);
    digitalWrite(ERROR_LED, LOW);
    digitalWrite(BUZZER_PIN, LOW);
}

```

Додаток Б
Архітектура модуля домашньої автоматизації з використанням IoT



Додаток В Алгоритм роботи модуля автоматизації розумного будинку з використанням singspeak

