

Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень

Програмно-апаратний засіб автоматичного контролю будинку за допомогою
мережевих систем на базі Arduino.

Назва теми

КвРКІ 180228.18.02.03 ПЗ
Шифр

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 125 «Кібербезпека»

Шифр, назва

Освітня програма «Кібербезпека»

Назва

Виконав: студент IV курсу, група KI-18-2

Гула 07.06.22
Підпис, дата

В.Р. Гула

Ініціали, прізвище

Керівник

[Підпис]
Підпис, дата

13.06.22

Ю.В. Хмельницький

Ініціали, прізвище

Нормоконтролер

[Підпис]
Підпис, дата

13.06.22

С.В. Мостовий

Ініціали, прізвище

До захисту допускаю:
Зав. кафедри комп'ютерної
інженерії та системного
програмування

[Підпис]
Підпис, дата

Ю.П. Кльоц

Ініціали, прізвище

« 15 » 06 2022 р.

Хмельницький 2022

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КІБЕРБЕЗПЕКИ

Освітній рівень БАКАЛАВР

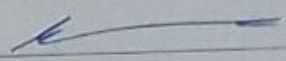
Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЯ ПРОГРАМА «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Ю.П.Кльоц


" 01 " 03 2022 року

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Гулі Вадиму Руслановичу

(Прізвище, ім'я, по батьков студента)

1. Тема проекту (роботи): Програмно-апаратний засіб автоматичного контролю будинку за допомогою мережевих систем на базі Arduino

Керівник роботи Хмельницький Юрій Владиславович к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджено наказом ректора університету від 1.03. 2022 року, № 18

2. Строк подання студентом проекту (роботи) на кафедру: 30.05.2022 рр

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) Здійснити огляд, провести аналіз та дослідження існуючих рішень по реалізації програмно-апаратного засобу автоматичного контролю будинку на базі Arduino, спроектувати мережу приладів у будинку.

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

«Схема апаратних з'єднань». «Інтерфейс програмно-апаратного засобу».
«Блок-схема програми». «Схема апаратних з'єднань».

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання Видав	Завдання Прийняв
Нормо контроль	Мостовий С.В., викладач кафедри КБ	—	<i>С.В. Мостовий</i>
Плагіат	Мостовий С.В., викладач кафедри КБ	—	<i>С.В. Мостовий</i>

7. Дата видачі завдання 06.03. 2022 р.

КАЛЕНДАРНИЙ ПЛАН

Пор. №	Назва етапу (розділу) кваліфікаційної роботи	Строк виконання етапу роботи	Примітка
1.	Вступ. Огляд існуючих методів, засобів.	1 декада Лютий	Виконано
2.	Обґрунтування вибраного варіанту.	2 декада. Лютий	Виконано
3.	Опис характеристики та роботи .	3 декада. Лютий	Виконано
4.	Розробка організаційної структури	1 декада. Березень	Виконано
5.	Розробка схеми розташування станцій	2 декада. Березень.	Виконано
6.	Підготовка ескізів креслень.	3 декада. Березень	Виконано
7.	Розробка частини по захисту	1 декада . Квітень	Виконано
8.	Розрахункова частина.	2 декада . Квітень	Виконано
9.	Висновки.	3 декада. Квітень.	Виконано
10.	Погодження з консультантами.	1 декада. Травень	Виконано
11.	Оформлення графічного матеріалу.	1 декада. Травень	Виконано
12.	Оформлення пояснювальної записки.	2 декада. Травень	Виконано
13.	Попередній захист кваліфікац. роботи.	3 декада. Травень	Виконано
14.	Подання роботи на плагіат	3 декада. Травень	Виконано
15.	Захист кваліфікаційної роботи	1 декада . Червень	Виконано

Студент *Григор*
(підпис)

В.Р. Гула
(Ініціали, прізвище)

Керівник роботи

[Підпис]
(підпис)

Ю. В. Хмельницький
(прізвище та ініціали)

№ р я д к а	ф о р м а т	Позначення	Найменування	К і л · л и с т і в	№ ек з	П р и м і т к а
			<u>Текстові документи</u>			
1		КвРКІ 180228.18.02.03 ПЗ	Пояснювальна записка	61		
			<u>Графічні матеріали</u>			
2		КвРКІ 180228.18.02.03 Е8	Схема апаратних з'єднань	1		
3		КвРКІ 180228.18.02.03 Е8	Блок-схема програми	1		
4		КвРКІ 180228.18.02.03 Е8	Інтерфейс програмно-апаратного засобу	1		

КвРКІ 180228.18.02.03 ВП				
Зм	Арк	№ докум	Підпис	Дата
Розробив		Гула	<i>Гула</i>	07.06.22
Перевір.		Хмельницький	<i>Хмельницький</i>	13.06.22
Н.контр.		Мостовий	<i>Мостовий</i>	15.06.22
Затв.		Кльоц	<i>Кльоц</i>	15.06.22
Відомість проєкту				
ХНУ, КІ-18-2				
Літера				
Аркуш				
Аркушів				
У 1 1				

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмно-апаратний засіб автоматичного контролю будинку за допомогою мережевих систем на базі Arduino».

Автор роботи: Гула Вадим Русланович

Керівник роботи: Хмельницький Юрій Владиславович

Пояснювальна записка: 60 с., 32 рис., 4 дод., 41 джерело.

Графічна частина: 8 презентаційних слайдів.

МЕРЕЖЕВІ СИСТЕМИ, АВТОМАТИЧНИЙ КОНТРОЛЬ БУДИНКУ, ARDUINO, ІНТЕРНЕТ РЕЧЕЙ, МІКРОСХЕМА.

Мета кваліфікаційної роботи полягає у проектуванні, а також реалізації системи керування пристроями всередині будинку, та їх відслідковування в режимі реального часу на базі Arduino.

Об'єкт дослідження у даній роботі – це програмно-апаратний засіб автоматичного контролю будинку за допомогою мережевих систем на базі Arduino.

Предметом дослідження виступають саме мережеві системи автоматичного контролю будинку на базі Arduino.

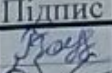

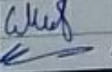
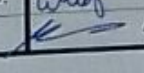
Практична цінність роботи виходить із реалізованого програмно-апаратного засобу автоматичного контролю будинку на базі Arduino, а також у мережевій системі, яка об'єднує прилади в будинку.

Gzaff

09.06.2022 р.

ЗМІСТ:

ВСТУП.....	5
1 ДОСЛІДЖЕННЯ МЕРЕЖНИХ СИСТЕМ ТА ПОСТАНОВКА ЗАДАЧІ.....	6
1.1 Концепція системи автоматичного керування будинком.....	6
1.2 Невеликий огляд мережної системи контролю будинку.....	7
1.3 Дослідження основної мети та обсягу роботи.....	10
1.4 Концепція Інтернету речей та її використання у роботі.....	10
1.5 Аналіз предметної області і виявлення наявних проблем і завдань.....	11
1.6 Порівняльний аналіз переваг та недоліків існуючих рішень.....	12
1.7 Методологічні підходи до вирішення задачі за темою дослідження.....	16
1.8 Висновки.....	18
2 ПРОЄКТУВАННЯ ПРОГРАМНО-АПАРАТНОГО ЗАСОБУ АВТОМАТИЧНОГО КОНТРОЛЮ БУДИНКУ.....	19
2.1 Обґрунтування вибору апаратних ресурсів та програмного забезпечення.....	19
2.2 Вимоги до системного обладнання.....	25
2.3 Вимоги до програмного забезпечення.....	25
2.4 Проектування локальної мережі будинку.....	27
2.5 Висновки.....	31

КвРКІ 180228.18.02.03 ПЗ								
Зм.	Арк	№докум.	Підпис	Дата	Програмно-апаратний засіб автоматичного контролю будинку за допомогою мережних систем на базі Arduino.	Літера	Аркуш	Аркушів
Виконав		Гула В.Р.		07.08.22				
Перевір.		Хмельницький Ю.В.		13.01.22			2	61
Н.контр. Затвер.		Мостовий С.В. Кльонц Ю.П.	 	15.06.22 15.06.22		ХНУ, КІ-18-2		

3 РЕАЛІЗАЦІЯ ПРОГРАМНО-АПАРАТНОГО ЗАСОБУ АВТОМАТИЧНОГО КОНТРОЛЮ БУДИНКУ НА БАЗІ ARDUINO ТА СТВОРЕННЯ МЕРЕЖНОЇ СИСТЕМИ.....	32
3.1 Проведення процедури прошивки Wi-Fi модуля.....	32
3.2 Створення локального хост-серверу.....	39
3.3 Проведення апаратного з'єднання релейних модулів із мікросхемою.....	44
3.4 Проведення апаратного з'єднання дисплею LCD1602 із мікросхемою.....	48
3.5 Проведення апаратного з'єднання додаткових комплектуючих пристрою.....	53
3.6 Тестування роботи пристрою.....	55
3.7 Висновки.....	57
ВИСНОВКИ.....	59
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	61
Додаток А (Обов'язковий) Копія графічної частини.....	64
Додаток Б Копія програмного коду.....	67

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

GPRS – General Packet Radio Service

GSM – Global System for Mobile Communications

SQL – Structured Query Language

IoT – Internet of Things

PIC - Peripheral Interface Controller

LCD - Liquid Crystal Display

B - ВОЛЬТ

IDE - Integrated Drive Electronics

HTML - HyperText Markup Language

IP - Internet Protocol

LAN – Local Area Network

WAN – Wide Area Network

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		4

ВСТУП

Основна ідея проєкту автоматизації будинку полягає в тому, щоб мати змогу контролювати всі електронні прилади, які під'єднані до загальної мережі. Зараз, завдяки шаленому темпу розвитку інтернет технологій і його поширенню по всьому світі є можливість втілювати в життя навіть такі, здавалося б неймовірні проєкти. Запит на таку ідею виник нещодавно через появу більш потужних смартфонів та їх можливості під'єднуватись до інших приладів за допомогою загальної мережі в будинку. Ця ідея має спільні риси з концептом "Інтернет речей". Мережа Інтернет дозволяє приладам в будинку швидко під'єднуватись і давати відгуки на різні запити від користувачів. До того ж такий варіант мережі є недорогим, адже використовується спільна мережа, швидкість роботи якої залежить лише від обраного тарифу Інтернет послуг. Зазвичай для контролю над приладами в будинку нам потрібен певний інтерфейс та сам контроллер. Однією з найкращих особливостей такої системи є доступ до керування будинком з будь-якої точки світу за наявності підключення до Інтернету вашого пристрою. Інші існуючі системи, які працюють використовуючи Bluetooth, GPRS, Інфрачервоні промені, або GSM є не такими надійними і мають багато своїх недоліків через проблеми з передачею даних на великі відстані.

Мета роботи полягає у проєктуванні, а також реалізації систем керування пристроями всередині будинку, їх відслідковування в режимі реального часу на базі Arduino.

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		5

1 ДОСЛІДЖЕННЯ МЕРЕЖНИХ СИСТЕМ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Концепція системи автоматичного керування будинком

Автоматизація домашніх приладів, а також їх контроль може значно полегшити життя людини в будь-якій точці світу. Здійснюється така автоматизація за допомогою локальної мережі, або через дистанційне керування. Все це відбувається для контролю побутової техніки і певних її функцій. В цьому нам допомагає штучний інтелект, який відсилає дані для прийняття рішень самим приладам, що під'єднані до мережі [8]. Така концепція є простою, пристрої в нашому будинку можуть передавати будь-які дані один одному, тобто взаємодіяти між собою через мережу де б вони не знаходились. Просто уявіть собі як можна використати такі системи автоматизації будинку, наприклад в такий спосіб можна керувати світлом в будинку, вмикати або вимикати його дистанційно і за вашим бажанням, також керувати іншими приладами в будинку через смартфон або певний мережевий інтерфейс з доступом до цієї мережі будинку.

В той час, як раніше Інтернет використовували лише для підключення комп'ютерів між собою, зараз багато приладів мають свій невеликий комп'ютер всередині, що має змогу здійснювати обмін інформацією через Інтернет [11]. Варто зазначити, що в цьому проєкті також використовується такий важливий елемент, як веб-сервер MySQL, який вміло поєднується з мікроконтролерною технологією задля реалізації системи керування будинком.

З втіленням у життя такого проєкту людина більше не ризикує забути вимкнути праску, або інший прилад, що може спричинити пожежу. З повним контролем над домашніми пристроями людині не потрібно сильно хвилюватись, адже вона може легко перевірити що забула вимкнути через свій смартфон навіть якщо вона далеко від дому [4]. Це допоможе запобігти багатьом катастрофам, втратам людських життів або їх майна, які можуть спричинити не вимкнуті завчасно електроприлади. Саме тому для власників сучасних електроприладів така можливість є дуже важливою, адже навіть якщо людина згадала про

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		6

працюючий пристрій, у неї не буде миттєвого засобу це виправити, тільки повертатись швидко назад до будинку. Погодьтеся, що простіше просто натиснути на кнопку на смартфоні, ніж скасовувати всі свої плани і руйнувати собі день. Також в наявності є функція керування потужностями, з якими діють прилади в будинку для забезпечення їх захисту від перегріву та подальшого загоряння. Для людей з обмеженими можливостями, такими як інваліди, або літні люди присутні свої додаткові переваги у зручності. Не треба ходити до вимикачів на стінах щоб увімкнути світло в кімнаті, адже все це можна зробити за допомогою одного смартфона. Загальний концепт мережі, яка керує приладами у будинку зображений на рисунку 1.1.

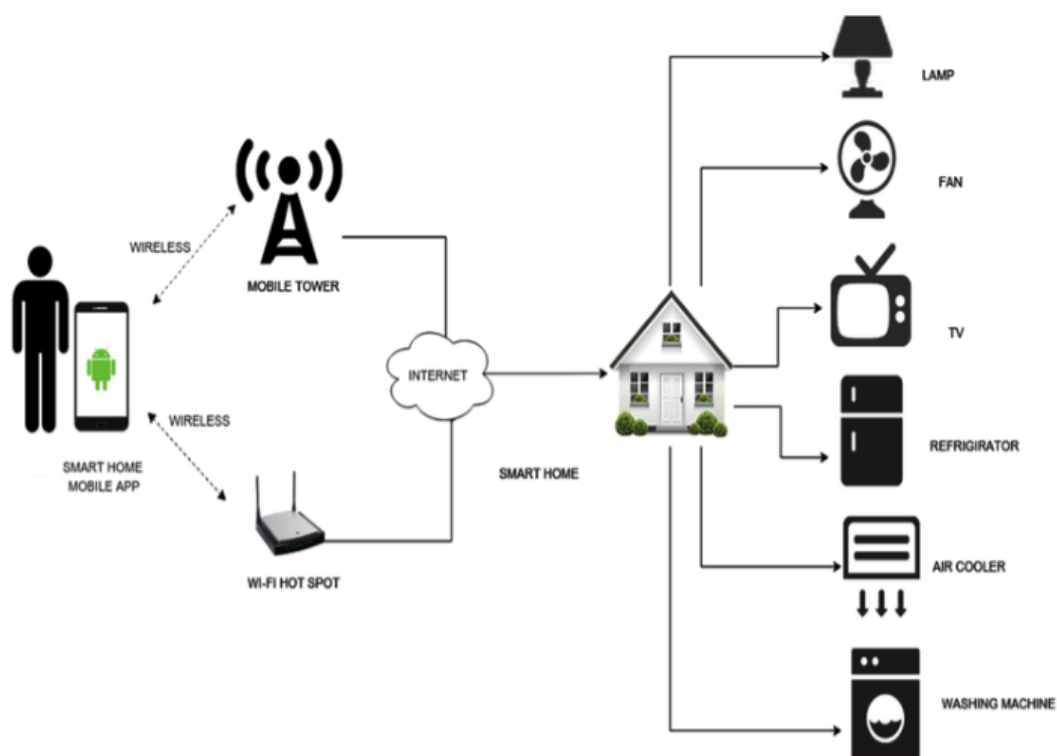


Рисунок 1.1 – Концепт керування приладами у будинку за допомогою мережі

1.2 Огляд мережної системи контролю будинку

Сама ідея автоматизації не є новою. У 1898 році схожий принцип створив Нікола Тесла, коли він зміг контролювати невеликий човен, посылаючи на нього

					КвРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		7

електромагнітні радіохвилі за допомогою бездротового пульта дистанційного керування. Це спричинило справжній вибух і появу різних побутових приладів, таких як тостери, пральні машини, праски та багато інших. І хоча на той час ще було далеко до теперішньої автоматизації в 1966 році Джим Сазерленд розробив свою систему домашньої автоматизації, яка називалась Echo 4. Така система могла вмикати, а також вимикати пристрої, керувати температурою і складати список покупок. Нажаль ця система не була комерціалізована.

Згодом внаслідок появи більшої кількості мікропроцесорів прийшло і значне зниження цін на різні електронні компоненти, що спричинило великий інтерес у будівельників як ідею втілення будинку, що має електричну систему, яка контролюється з певного центрального управління. Через це з'явилися ідеї нових приладів, які б працювали на штучному інтелекті. Внаслідок поширення електроприладів і їх постійного використання людьми зросло і споживання енергії, а також через людей, які забували вимикати прилади, коли залишали свою домівку. Ці, а також різні інші фактори призвели до створення системи автоматизації будинку, як системи, яка буде контролювати прилади в будинку.

Перші спроби створити таку систему ґрунтувались на інфрачервоних та GSM променях. Проблемою інфрачервоного випромінювання було обмеження у відстані і потреба знаходитись поблизу, або навіть у деяких випадках у зоні видимості приладу, який потрібно було контролювати. Щодо GSM системи керування, то вона використовувала спосіб передавання коротких повідомлень до керуючих пристроїв. Недоліком такого способу був значний час передачі повідомлень, що могло сильно затримати реакцію пристроїв та їх спрацьовування. Мережеве керування позбавлено такого недоліку.

Пристрої, які здійснюють моніторинг, автоматизацію, а також контроль, використовуються в певних приладах, таких, як система автоматизації будинку, контролю температури, тиску, швидкості, та інших. Дослідження про це здійснював М. Кассім, який запропонував слідкувати за системою температури за допомогою веб-моніторингу [7]. Надихнувся він тим, як у будівельній сфері застосовували бездротову систему моніторингу температури, яка застосовувалась у будівництві для отримання конкретної температури будівничих матеріалів та

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		8

подальшого контролю їх температурного стану. Його колеги, які продовжили його справу запропонували свій варіант системи, яка може здійснювати віддалений доступ, обслуговування обладнання та моніторинг через мережу за допомогою веб-браузера. Така система, на відміну від системи на основі GSM, яка використовує короткі повідомлення для керування пристроями, керує пристроями у режимі реального часу, що є значно ефективнішим, ніж GSM, адже відправлення короткого повідомлення може зайняти багато часу.

На рисунку 1.2 зображена блок-схема системи автоматичного контролю будинку на базі Arduino.

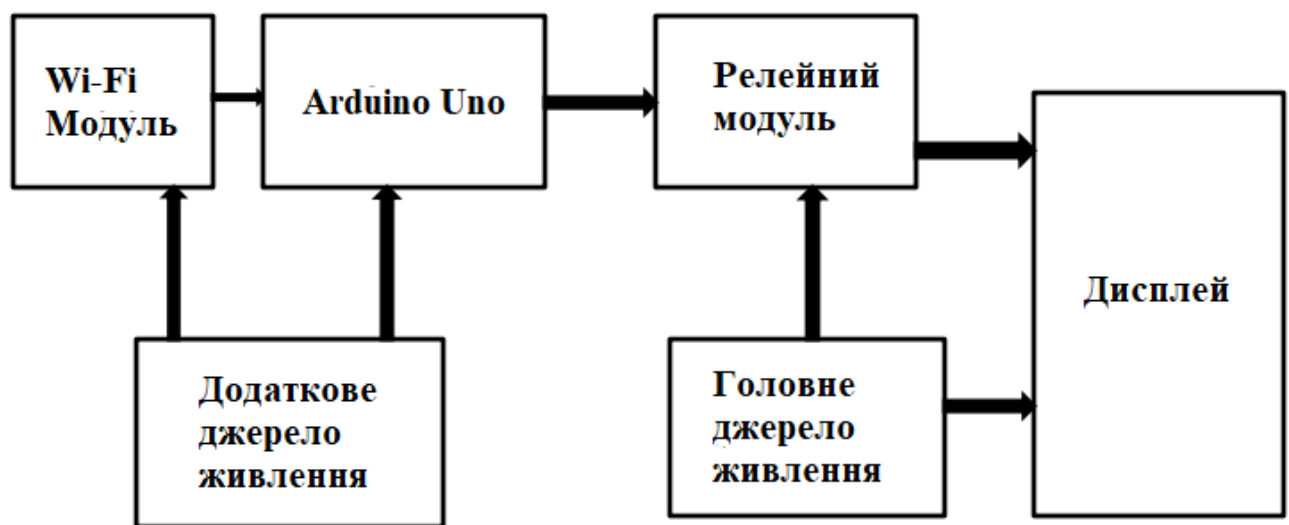


Рисунок 1.2 – Блок-схема системи автоматичного контролю будинку на базі Arduino

У системах автоматизації використання бездротових технологій має певні переваги, які неможливо отримати, використовуючи дротову мережу. Такими перевагами є:

- 1) Менша вартість встановлення, через відсутність кабелів;
- 2) Можливість під'єднатись до пристроїв з будь-якого місця за наявності мережі;
- 3) Більш захищена мережа;
- 4) Можливість зміни обсягів мережі, та універсальність.

1.3 Дослідження основної мети та обсягу роботи

Як вже було зазначено раніше, головною метою цього проєкту є створення працюючого прототипу системи автоматизації будинку, що базується на мережевих технологіях [15]. Така система має бути надійною, а також доступною різним користувачам, чи іншим власникам. Також ця система може використовуватись у готелях, або освітніх закладах.

Головними у цій системі будуть такі пункти:

- 1) Можливість увійти до системи будинку з будь-якого місця, де є Інтернет мережа;
- 2) Бачити прилади через інтерфейс системи, а також їх статус;
- 3) Мати можливість вмикати та вимикати прилад за необхідності.

1.4 Концепція Інтернету речей та її використання у роботі

Інтернет речей, або IoT (Internet of Things) є концепцією мережі, яка працює через пристрої, які поєднані між собою [1]. Пристрої мають у собі програмне забезпечення, а також датчики, вбудовані в них. За допомогою цього вони мають можливість передавати, а також обмінюватись даними між комп'ютерними системами і реальним світом [3]. Головною ідеєю Інтернету речей є здатність підключення будь-яких речей, якими користується людина, головне, щоб ці речі мали в собі спеціальні датчики, які будуть приймати сигнали.

Саме з'єднання речей між собою здійснюється за допомогою певних технологій, а саме:

- 1) Технологія накопичення даних та подальшого їх оброблення з допомогою комп'ютера, що вбудований у пристрій;
- 2) Технологія обміну даними без використання кабелів;
- 3) Технологія спеціальних сенсорів, які відстежують зміни у пристрої;
- 4) Технологія бездротової передачі даних із використанням таких мереж, як Bluetooth чи Wi-Fi;

					КВРКІ 180228.18.02.03 ПЗ	Арк
						10
Зм.	Арк.	№докум.	Підпис	Дата		

5) Технологія ідентифікації та відслідковування пристрою.

1.5 Аналіз предметної області і виявлення наявних проблем і завдань

Будь-яка система може мати свої певні проблеми і наша також не виключення, але таке явище є цілком нормальним, тож потрібно відповідально підійти до пошуку рішень цих проблем.

Через специфіку обраної теми роботи проєкт може мати наступні проблеми:

- 1) Проблеми із сумісністю пристроїв мережі;
- 2) Обмежені функції та можливості;
- 3) Перегрівання пристроїв;
- 4) Фінансові затрати на обладнання;
- 5) Проблеми із безпекою мережі.

Тепер, коли було конкретно визначено основні проблеми, які можуть виникнути можна перейти до їх обговорення та методів їх вирішення.

Перша проблема полягає у тому, що без належного центрального пристрою, який би здійснював керування всіма іншими пристроями, а також забезпечував їх належну роботу система може мати проблеми, внаслідок яких робота мережі буде ненадійною. Вирішити цю проблему можна, якщо перед створенням системи передбачити один центральний хаб, який буде керувати пристроями, та регулювати їх роботу.

Друга проблема виникає у випадку, якщо користувач підійшов до питання підбору пристроїв у системі безвідповідально, та не врахував конкретні цілі та функції, які він очікує від системи. Вирішується ця проблема на етапі планування, а саме підбором пристроїв з саме тими функціями, які безпосередньо потрібні для користувача.

Третя проблема є дуже поширеною серед усіх пристроїв. Перегрівання пристрою, внаслідок якого відбувається вимкнення пристрою є механізмом, що вберігає пристрій від пошкоджень внаслідок нагрівання елементів або деталей пристрою. Причиною перегрівання є пил, найпоширеніший чинник, або неналежна вентиляція. Вирішення цієї проблеми досягається шляхом звичайного

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		11

прибирання приміщення від пилу, а також уникання розміщення пристроїв в одному місці. Таке розміщення погано впливає на вентиляцію і спричиняє їх перегрівання.

Четверта проблема залежить лише від фінансових можливостей користувача. Якщо є великий обсяг коштів і немає обмежень у придбаннях, то ця проблема не є суттєвою. Якщо ж фінанси є обмеженими, то така проблема вирішується пошуком більш дешевих альтернатив, які будуть задовольняти бюджет.

П'ята проблема є безпековою і її не можна недооцінювати, адже безпека мережі дуже важлива для функціонування всієї системи, а будь-які втручання в неї з боку як зовнішніх так і внутрішніх чинників можуть вивести її з ладу. Нажаль цій проблемі часто не приділяють належної уваги.

Вирішується проблема безпеки чітким контролем за доступом до керування мережею, тобто не допусканням до неї сторонніх осіб. Також важливо мати антивірусну програму, яка забезпечить захист від шпигунських програм, вірусів та іншого. Також хорошим рішенням буде запровадити систему автентифікації користувачів із різними рівнями допусків як для адміністрації так і для звичайних користувачів. Це покращить перевірку користувачів, які користуються мережею.

Важливо не допускати різних недоліків у системі, які можуть спростити роботу шахраям, потрібно уникати простих паролів до мереж, відкритих портів та іншого.

1.6 Порівняльний аналіз переваг та недоліків існуючих рішень

Ідею автоматизації будинку частково, або в більшій мірі вже втілювали у життя, адже із поширенням нових технологій амбіції щодо автоматизації багатьох речей тільки зростали [13]. Наведемо певний перелік вже існуючих варіантів пристроїв, обладнання та іншого:

1. Amazon Echo – Дуже зручна та популярна система автоматизації, яка дозволяє контролювати будь-який пристрій, що синхронізується з цією системою. Також може телефонувати у відеоформаті. Недоліками можна зазначити певні

					КвРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		12

проблеми із відтворенням різних музикальних композицій, зникнення зв'язку із системою у рідкісних випадках, а також повільний відгук на команди. Зображена дана система на рисунку 1.3.



Рисунок 1.3 – Система автоматизації Amazon Echo [38]

2. Philips – Якщо мова йде про освітлення, то компанія Philips є одним з тих брендів, які мають високу довіру клієнтів. Ця компанія надає доступ до спеціальних концентраторів, які контролюють освітлення у приміщенні.

Таким чином за допомогою одного пристрою можна контролювати до п'ятдесяти приладів освітлення у будинку.

Окрім того встановлення такого концентратора не є дуже складним завданням, все доволі зрозуміло. Також є можливість синхронізації пристроями контролю будинку типу Alexa або Google. Якщо говорити про недоліки, то ціна може бути доволі дорога. Один із таких приладів освітлення зображений на рисунку 1.4.



Рисунок 1.4 – Система освітлення Philips HUE [36]

3. Aqara – Ця система автоматизації також дозволяє контролювати освітлення, але додатково має багато інших можливостей. Система має в собі спеціальні сенсори, які розміщуються біля вікон та дверей і стежать чи заходив хтось у будівлю чи ні.

Також ці сенсори можуть реагувати на людину, яка заходить у приміщення вмиканням світла, або спеціальним сигналом. Інтерфейс пристрою є простим та інтуїтивно зрозумілим, а сама система надійною та якісною.

Головним недоліком є те, що Aqara не сумісна із пристроями інших компаній, тобто працювати вона буде лише з такими самими приладами Aqara. Зображена така система на рисунку 1.5.



Рисунок 1.5 – Система автоматизації Aqara [37]

4. Hubitat – Особливістю цього пристрою є його сумісність із дуже великою кількістю інших приладів, що забезпечує їхню спільну та злагоджену роботу разом, без різних ускладнень із підключенням.

Важливо також зазначити, що він працює через локальний процесор, що означає, що він може працювати без доступу до Інтернету.

Окрім цього дозволяє контролювати освітлення, температуру, сигналізацію, а також поливні системи води.

Щодо недоліків, то вони полягають у складності налаштування, а також у неможливості працювати без Інтернет з'єднання тих пристроїв, які на це не розраховані. Зображений пристрій на рисунку 1.6.



Рисунок 1.6 – Пристрій Hubitat Elevation Home Automation Hub [2]

Тепер, переглянувши найпоширеніші альтернативи можна більш чіткіше враховувати всі недоліки інших систем автоматизації і зосередитись на тому, щоб їх усунути у своєму проєкті, або хоча б пом'якшити.

Наведемо перелік поширених недоліків автоматизованих систем:

- повільний відгук на команди від користувача;
- дорога вартість пристроїв;
- несумісність із пристроями інших виробників;
- складність налаштування;
- обмежені умови використання;
- залежність від Інтернету та інших чинників.

1.7 Методологічні підходи до вирішення задачі за темою дослідження

Проаналізувавши та врахувавши усі недоліки та проблеми, які можуть виникнути під час виконання даного проєкту маємо змогу визначитися із підходами до вирішення цих проблем.

Для того, щоб краще керувати пристроями у системі буде задіяно спеціальний керувальний прилад, який буде контролювати усе розгалуження системної мережі.

					КВРКІ 180228.18.02.03 ПЗ	Арк
						16
Зм.	Арк.	№докум.	Підпис	Дата		

Фінансовий підхід до вирішення задачі також важливий, тож підбір усіх комплектуючих та пристроїв буде здійснюватись із урахуванням бюджету, а їх якість має бути оптимальною для системи. Недопустимо виходити за рамки бюджету, але й вибір серед альтернатив комплектуючих має бути на перевагу тих, якість яких є задовільною, а ціна не надто дорогою.

Щодо питання безпеки мережі, то воно буде вирішуватись за допомогою антивірусного програмного забезпечення, а також обмеженням доступу до всіх функцій мережі для звичайних користувачів.

Всі інші технічні питання потребують певних навичок, наприклад:

- розуміння принципів роботи автоматизованих систем;
- розуміння алгоритмів роботи системи;
- знання принципів роботи мікроконтролерів на Arduino;
- розуміння принципів роботи мережі;
- вміння підбирати найоптимальніші варіанти або рішення.

Апаратна обчислювальна платформа Arduino є доволі дешевим і поширеним варіантом, тож у забезпеченні його комплектуючих не має бути жодних проблем. Саме за ці якості користувачі і обирають її, коли мають бажання створити власну систему автоматизації будинку. Сама плата є стійкою до різних пошкоджень, не боїться раптових вимкнень світла, що можливо у процесі виробництва. У випадку знеструмлення поновлення роботи над мікроконтролером здійснюється звичайним підключенням до джерела живлення.

Платформа Arduino є універсальною, а вибір плат дуже широким. Завжди можна знайти те, що підходить саме тобі. Саме через ці чинники і була обрана апаратна обчислювальна платформа Arduino, адже її можливості цілком задовольняють даний проєкт і задачі, що постають в ньому.

Щодо обраного мікроконтролера, то кращим варіантом буде мікроконтролер Arduino Uno. Цей мікроконтролер є простим у використанні, добре взаємодіє із різними давачами, сигналами, тощо. До цього всього він ще й потужний. Arduino Uno являє собою одноплатовий комп'ютер, код якого є відкритим та безкоштовним.

					КВРКІ 180228.18.02.03 ПЗ	Арк
						17
Зм.	Арк.	№докум.	Підпис	Дата		

Система мережі буде створюватись за допомогою програмного середовища, в ньому ж буде здійснюватись його моделювання

1.8 Висновки

Згідно із проведеними розрахунками всіх проблем та рішень у створенні програмно-апаратного засобу автоматичного контролю будинку можна зробити висновок, що серйозними проблемами у виконанні роботи є питання безпеки, а також вартість комплектуючих.

Виявивши проблеми маємо можливість підібрати максимально ефективні варіанти їх вирішення, а саме:

- 1) Встановлення спеціального програмного забезпечення, яке буде здійснювати захист мережі;
- 2) Здійснення контролю за придбанням необхідного обладнання, задля більш оптимальної витрати коштів.

Подбавши про ці проблеми, можна продовжувати роботу, націлену на реалізацію програмно-апаратного засобу, а саме приступити до його проектування.

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		18

2 ПРОЄКТУВАННЯ ПРОГРАМНО-АПАРАТНОГО ЗАСОБУ АВТОМАТИЧНОГО КОНТРОЛЮ БУДИНКУ

2.1 Обґрунтування вибору апаратних ресурсів та програмного забезпечення

Кращим варіантом буде використати апаратну обчислювальну платформу Arduino Uno [23]. Вона найкраще підійде у даній роботі через свою простоту та надійність. Мікросхема Arduino Uno зображена на рисунку 2.1.

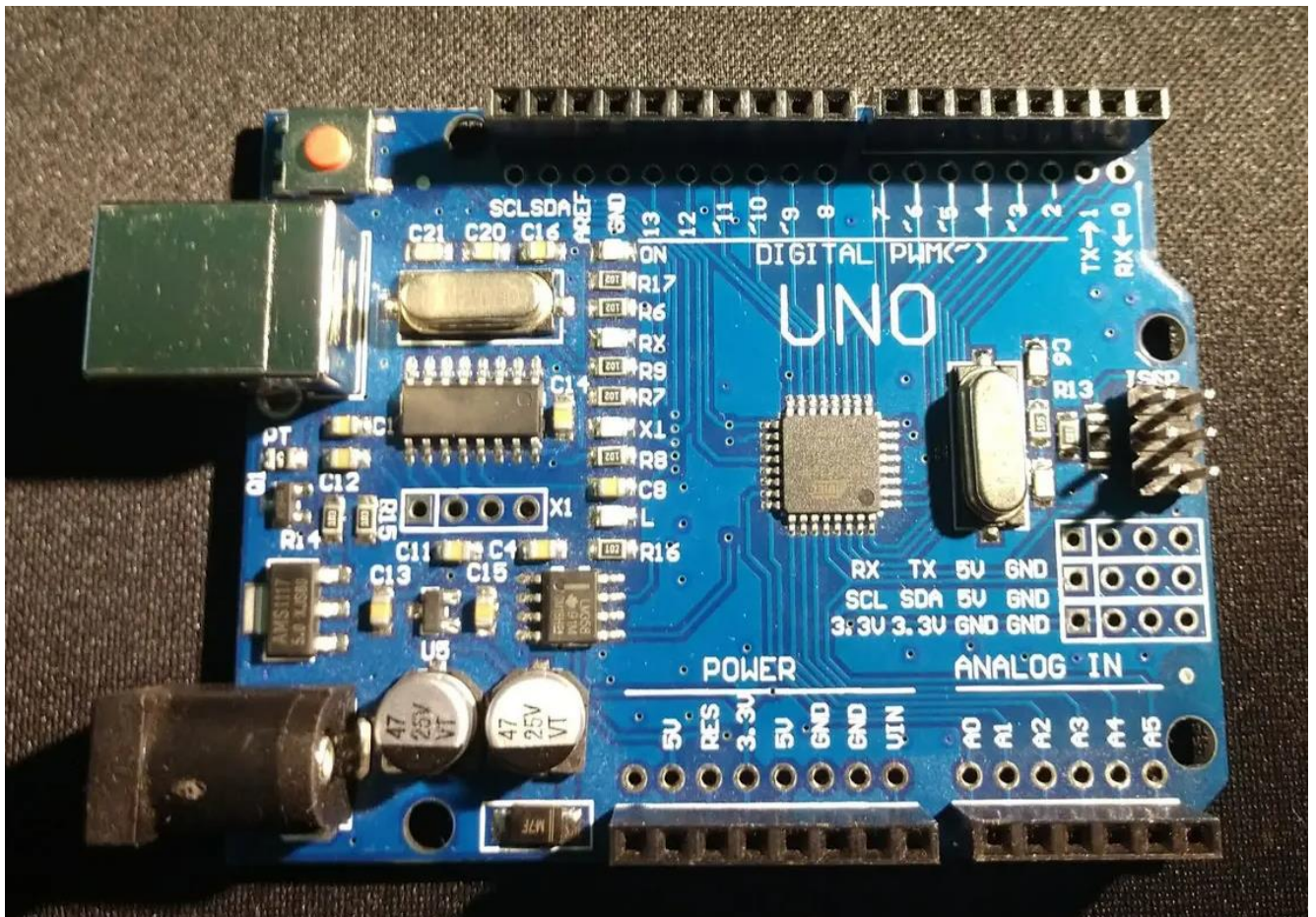


Рисунок 2.1 – Мікросхема Arduino Uno

Такий мікроконтролер привабливий тим, що має хорошу потужність. Також він прекрасно поводить себе у з'єднанні із багатьма сенсорами, індикаторами, давачами, тощо. Як варіант можна було б обрати мікроконтролер PIC, який дає можливість додати певні елементи у випадках, коли такі елементи потрібні, але для економії коштів краще оберемо Arduino Uno [40].

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		19

Мікроконтролер Arduino Uno створений на базі мікроконтролера ATmega328P, має в собі безліч речей, які можуть забезпечити якісну роботу з ним.

Також використаємо у проєкті такий пристрій, який називається рідкокристалічний дисплей (LCD). Потрібен він для того, щоб візуально відображати інформацію на своєму дисплеї. Такий пристрій часто використовується у різних портативних комп'ютерах та цифрових годинниках. На рисунках 2.2 та 2.3 зображений пристрій LCD1602.

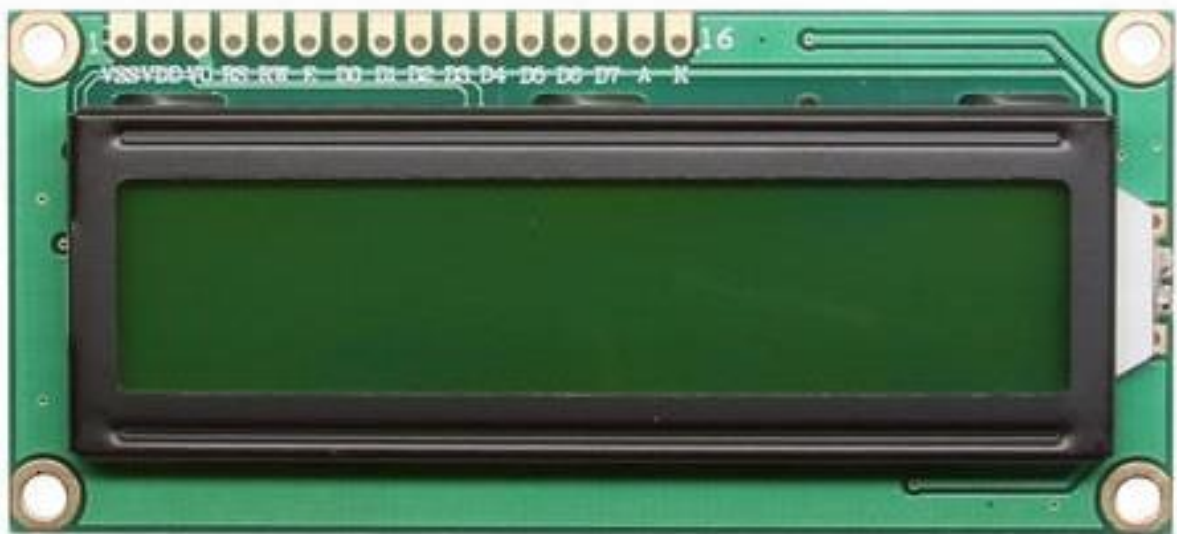


Рисунок 2.2 – Дисплей LCD1602 (Вигляд зверху)



Рисунок 2.3 – Дисплей LCD1602 (Вигляд знизу)

LCD1602 є електронним модулем, що базується на драйвері HD44780 від компанії Hitachi. Цей модуль має змогу працювати як у 4-бітному режимі, так і у 8-бітному, а також має 16 контактів. Для під'єднання до нього використовується макетна плата. Розмір дисплея складає 2,6 дюйма, а необхідна напруга живлення має сягати 5 В.

Наступним елементом, який є дуже важливим для створення повноцінного пристрою, є Wi-Fi модуль ESP8266 (ESP-01) [35]. Потрібен він для того, щоб прилади могли під'єднуватись до нього через бездротову мережу Wi-Fi. Зображений цей модуль на рисунку 2.4.



Рисунок 2.4 – Wi-Fi модуль ESP8266 (ESP-01)

Створений цей модуль за зразком часто використовуваного чіпсета ESP8266EX. Його структура являє собою плату, на якій розташована мікросхема флешпам'яті, об'єм якої сягає 2МБ. Також на платі розміщено сам чіп, два світлодіоди, невеличка антена, а також кварцовий резонатор. За допомогою флешпам'яті на платі зберігається встановлене програмне забезпечення, яке при ввімкненні починає завантажуватись автоматично у чіп. Модуль має перевагу у

тому, що його прошивання може бути здійснене багатьма мовами програмування на вибір, а отже це додаткова зручність для налаштування.

Ще один важливий модуль, а саме релейний модуль Tolako 5V допоможе здійснювати керування за приладом [39]. Сила струму, при якій працює цей релейний модуль сягає від 15 до 20 мА. Він зображений на рисунках 2.5 та 2.6.



Рисунок 2.5 – Релейний модуль Tolako 5V (Вигляд зверху)

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ 180228.18.02.03 ПЗ

Арк

22

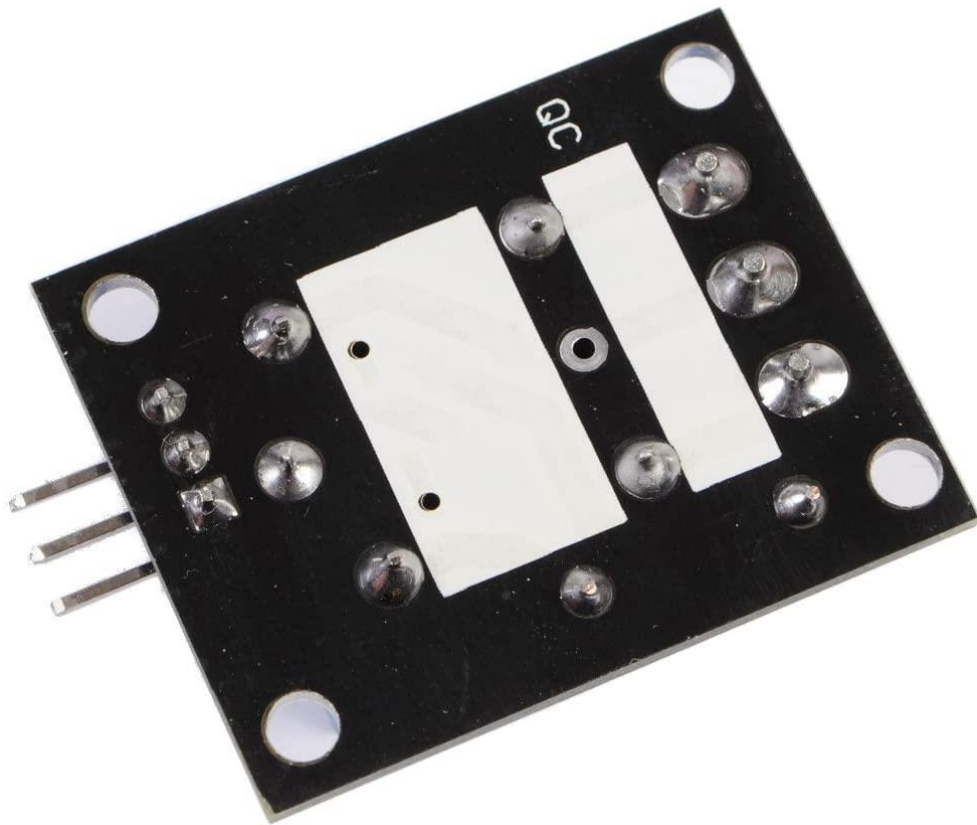


Рисунок 2.6 – Релейний модуль Tolako 5V (Вигляд знизу)

Живиться цей модуль або від зовнішніх джерел живлення, таких як батарейки, акумулятори чи блоки живлення, або від головного керуючого пристрою. Присутня також світлодіодна індикація реле, яка показує стан кожного з них. Інтерфейсом модуля можна керувати за допомогою таких контролерів як Arduino, PIC, або через комп'ютер.

Перевірка роботи програмно-апаратного засобу буде здійснюватись з урахуванням всіх заходів безпеки.

Індикаторами, які будуть показувати результати перевірки пристрою на справність роботи будуть звичайні невеличкі лампочки [9], потужність яких складає 15 Вт. Зображена така лампочка на рисунку 2.7.

Зм.	Арк.	№докум.	Підпис	Дата



Рисунок 2.7 – Лампочка

При проведенні тестування роботи пристрою в залежності від показників індикаторів можна буде визначати наявний стан пристрою. Це або стан, коли пристрій увімкнений, або коли він вимкнений. З основними приладами лампочки з'єднуються за допомогою безпосереднього фізичного контакту, через металеві частини корпусу.

Для забезпечення живлення пристрою будуть використовуватись кабелі [17], на кожний із входів та виходів, а також пряма штепсельна вилка на 220В для під'єднання електроприладу до електричної мережі через розетку. Дана штепсельна вилка зображена на рисунку 2.8.

Зм.	Арк.	№докум.	Підпис	Дата



Рисунок 2.8 – Пряма штепсельна вилка

2.2 Вимоги до системного обладнання

Тепер, після того, як було перераховано усе обладнання, необхідне у проектуванні програмно-апаратного засобу автоматичного контролю будинку, можна сформулювати повний його перелік нижче:

- 1) Апаратна обчислювальна платформа Arduino Uno;
- 2) Дисплей LCD1602;
- 3) Wi-Fi модуль ESP8266 (ESP-01);
- 4) Релейний модуль Tolako 5V;
- 5) Макетна плата;
- 6) Лампочки;
- 7) Кабелі ;
- 8) Розетка;
- 9) Пряма штепсельна вилка.

2.3 Вимоги до програмного забезпечення

					КВРКІ 180228.18.02.03 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		25

Для того, щоб здійснювати моніторинг та контроль різноманітних пристроїв у будинку буде розроблена спеціальна система, яка буде надавати користувачу можливість забезпечити відстежування їх роботи. Працювати це буде через Інтернет мережу, з можливістю використання браузера.

По суті Веб-сторінка може виконувати роль інтерфейсу, з яким користувач може працювати та здійснювати керування. Для цього Веб-сторінку необхідно створити на локальному хост-сервері, який під'єднаний до мікроконтролера через послідовне з'єднання.

Сам мікроконтролер здійснює керування пристроями, внаслідок отримання чітких вхідних сигналів від користувача. Стан пристроїв можна відстежувати віддалено, в режимі реального часу. Також можливо змінювати їх стан так само віддалено, за бажанням користувача.

Для реалізації програмного забезпечення, яке буде здійснювати управління пристроєм буде використовуватись програма, що має назву Arduino IDE. Встановлено останню версію даної програми, а також зроблено всі її необхідні оновлення. Інтегроване середовище розробки Arduino є кросплатформовим застосунком, написане на мові програмування Java. Воно включає в себе редактор програмного коду з різними функціями. Також в ньому присутні механізми для легкого компілювання коду, та подальшого його завантаження на плату Arduino за допомогою лише одного кліку миші.

Дане програмне забезпечення містить область для повідомлень, текстову консоль, панель інструментів з кнопками для загальних функцій та ієрархією операційних меню.

Arduino IDE працює на багатьох сучасних операційних системах, наприклад Windows, MacOS, або Linux та інших. Підтримує мови C та C++, а також використовує особливі правила для структурування програмного коду. Програма має в собі широку бібліотеку, яка містить багато важливих елементів для здійснення процедур введення та виведення, а програмний код, який написав користувач може бути легко запущений або як макет проєкту, або як повноцінний проєкт.

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		26

Приклад інтерфейсу цього програмного забезпечення зображений на рисунку 2.9.

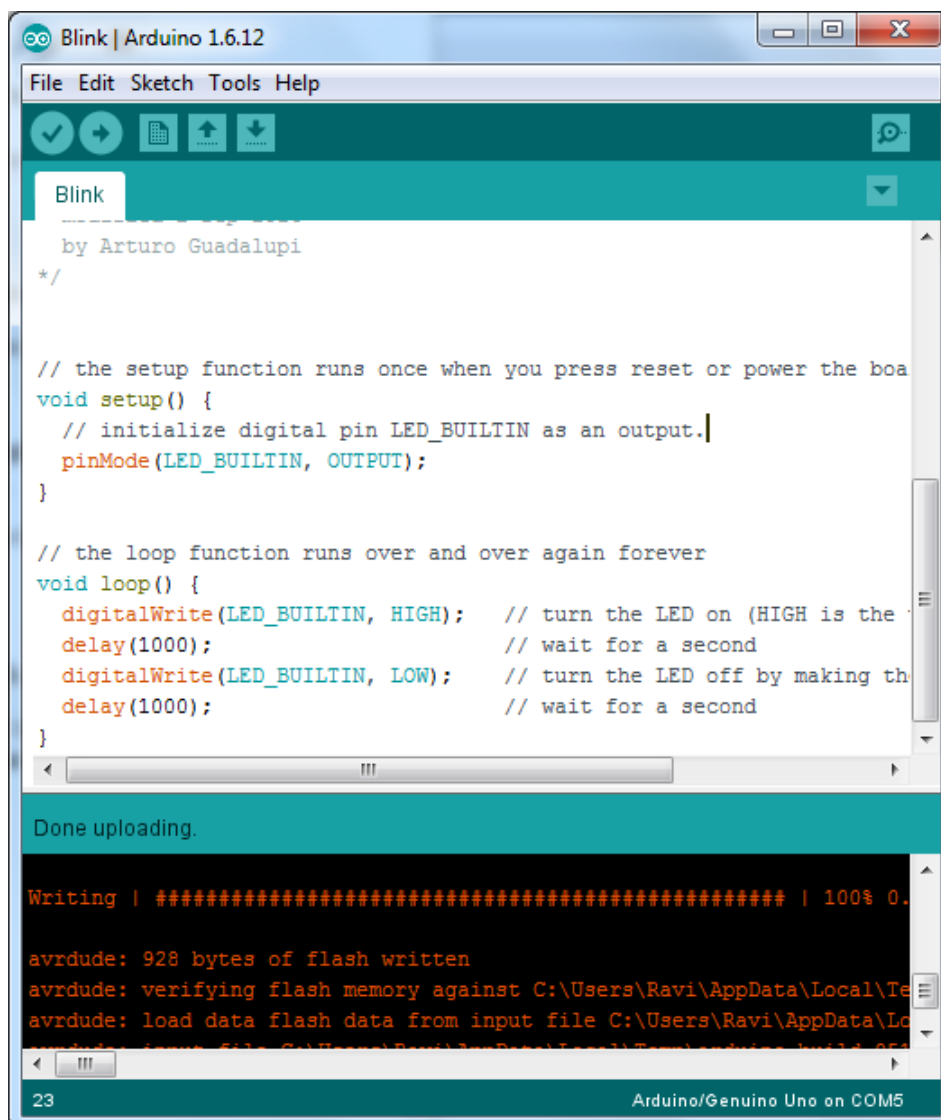


Рисунок 2.9 – Інтерфейс програмного середовища Arduino

2.4 Проектування локальної мережі будинку

Якщо потрібно прив'язати декілька приладів до одного, який би здійснював керування ними, тоді потрібно створити локальну мережу. Локальна мережа, або LAN – це сполучення кількох приладів, що під'єднані до спільної мережі, в якій вони можуть взаємодіяти між собою. Така мережа не є обширною і діє на маленькій території. Подібною до локальної мережі є також глобальна мережа WAN, яка може діяти на значно більшій території.

Зм.	Арк.	№докум.	Підпис	Дата

Що варто зазначити, то це перевагу локальної мережі у швидкості її роботи, адже невеликі відстані дозволяють оперативно надсилати дані, а також приймати їх. На рисунку 2.10 зображено приклад того, як виглядає спільна локальна мережа комп'ютерів.

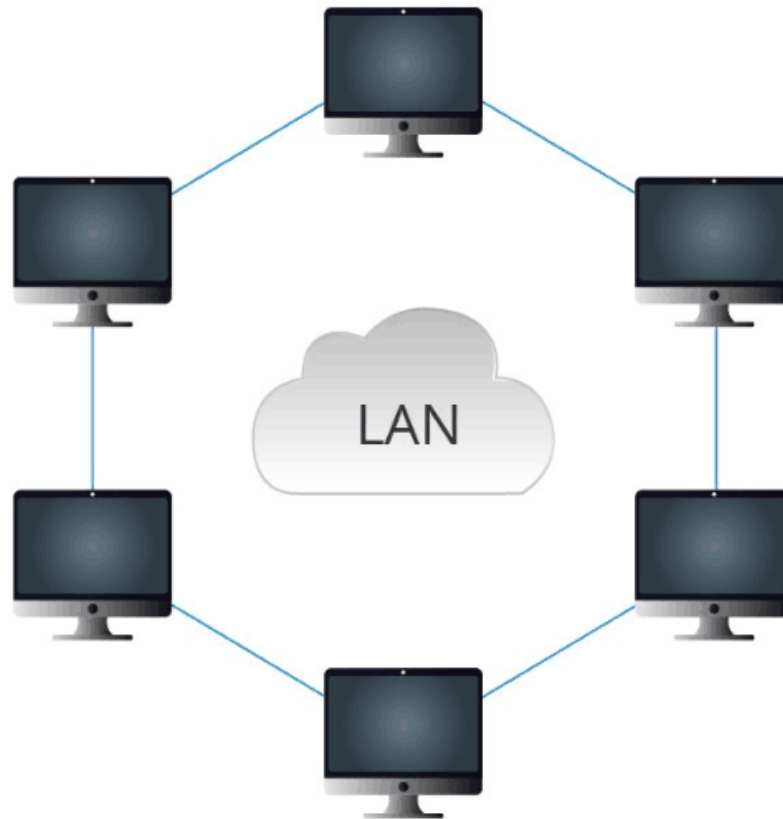


Рисунок 2.10 – Локальна мережа комп'ютерів

В даній роботі використання спільної мережі для пристроїв дуже важливе, адже без неї не вдасться досягнути необхідної мети. Програмно-апаратний засіб автоматичного контролю будинку буде мати керівну роль над усіма іншими приладами, які доєднались до локальної мережі.

Щодо варіантів сполучення приладів у мережі, то це може бути як дротове з'єднання, так і бездротове, з використанням мережі Wi-Fi.

Використання мережі Wi-Fi є найпростішим варіантом того, як можна долучити прилади до однієї спільної мережі.

Якщо будинок має підключені Інтернет послуги, а також Wi-Fi роутер, який має змогу поширювати дію Інтернет мережі на необхідну відстань, то це є хорошим варіантом того, як можна створити локальну мережу.

Загальний принцип, за яким будуть працювати прилади у локальній мережі буде полягати у одному програмно-апаратному засобі, який буде мати можливість змінювати стан всіх приладів.

Усі прилади, які мають можливість під'єднуватись до мережі, будуть підпадати під контроль. Керуючий пристрій за бажанням користувача зможе або активувати роботу приладу, або деактивувати.

Щодо того, як саме буде керуватись стан роботи приладів у мережі, то для цього буде використовуватись спеціальна сторінка, на яку буде відправляти IP адреса, яка буде попередньо згенерована. Через програмування Wi-Fi модулю, буде створено спеціальний інтерфейс, який міститиме у собі кнопки для вмикання, чи вимикання приладів.

Перевагою такого рішення буде те, що користувач, маючи смартфон, або персональний комп'ютер зможе перейти за спеціальною IP адресою, використовуючи для цього будь-який Інтернет браузер. Коли користувач перейде на сторінку, для нього відкриється інтерфейс керування приладами, що є в мережі.

Залежно від того, які кнопки буде натискати користувач, прилади будуть змінювати свій стан. На рисунку 2.11 зображено приклад схематичного плану різних кімнат у будівлі, саме за таким планом буде побудований інтерфейс сторінки.

Wi-Fi роутер буде розповсюджувати Інтернет сигнал на всю площу будинку так, щоб прилади, які розташовані у кімнатах мали достатній сигнал і могли без зайвих труднощів під'єднатись до мережі.

Важливо пам'ятати, що не всі маршрутизатори можуть мати достатню площу для покриття всієї території будівлі, тож підбір маршрутизатора має індивідуальний характер і залежить від площі будинку.

Простіше кажучи, перед тим, як придбати маршрутизатор, необхідно врахувати розміри будинку та кімнати, у яких будуть розташовуватись прилади.

					КВРКІ 180228.18.02.03 ПЗ	Арк
						29
Зм.	Арк.	№докум.	Підпис	Дата		



Рисунок 2.11 – План кімнат будинку, у яких будуть розміщуватись прилади

Ще одна важлива деталь, це особливості моделі маршрутизатору. Окрім того, що необхідно врахувати розміри будинку, також варто звернути увагу на потужність маршрутизатора, адже його характеристики можуть бути зовсім різні.

Ті маршрутизатори, які мають кращу якість та більш широку площу досяжності сигналу, можуть коштувати значно більше, тож потрібно обирати індивідуальний варіант, який задовольнить потреби користувача.

Щодо питання того, як локальна мережа буде взаємодіяти із програмно-апаратним засобом автоматичного контролю будинку, то ця функція буде покладена на Wi-Fi модуль ESP8266.

За допомогою програмного забезпечення Arduino IDE буде здійснено процедуру налаштування даного модуля.

Внаслідок цього програмно-апаратний засіб, після повної збірки всіх його комплектуючих, в тому числі і Wi-Fi модулю, отримає змогу так само, як і інші прилади в будинку приєднуватись до локальної мережі.

При проектуванні мережі буде використано фіксовану кількість приладів у кімнатах будинку.

За бажанням список приладів, за якими можна здійснювати контроль буде збільшуватись, або зменшуватись, в залежності від бажання користувача.

2.5 Висновки

Отже, в цьому розділі було обґрунтовано та пояснено вибір необхідних апаратних засобів, програмного забезпечення, системного обладнання та мережної системи. Також було наведено весь перелік обладнання, яке буде застосовуватись у даному проєкті.

					КВРКІ 180228.18.02.03 ПЗ	Арк.
						31
Зм.	Арк.	№докум.	Підпис	Дата		

3 РЕАЛІЗАЦІЯ ПРОГРАМНО-АПАРАТНОГО ЗАСОБУ АВТОМАТИЧНОГО КОНТРОЛЮ БУДИНКУ НА БАЗІ ARDUINO ТА СТВОРЕННЯ МЕРЕЖНОЇ СИСТЕМИ

3.1 Проведення процедури прошивки Wi-Fi модуля

Для того, щоб краще налаштувати Wi-Fi модуль ESP8266 під конкретні потреби даного проєкту, потрібно провести його прошивку. Підготовка цього модулю до прошивки зображена на рисунку 3.1.

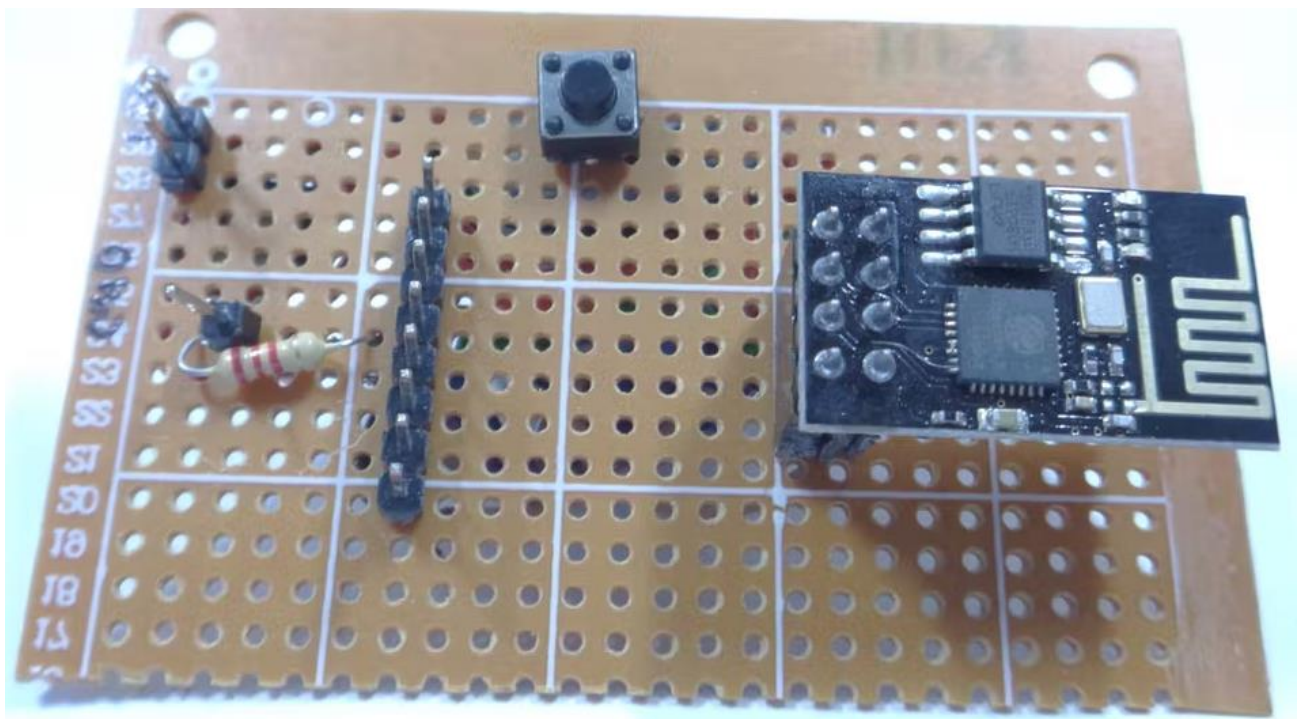


Рисунок 3.1 – Підготовка Wi-Fi модулю ESP8266 до прошивки

З'єднаємо апаратну обчислювальну платформу Arduino Uno із Wi-Fi модулем за схемою, яка показана на рисунку 3.2. Не забуваємо при цьому чітко дотримуватись визначеної схеми, інакше є ризик пошкодження обладнання, або певних його елементів.

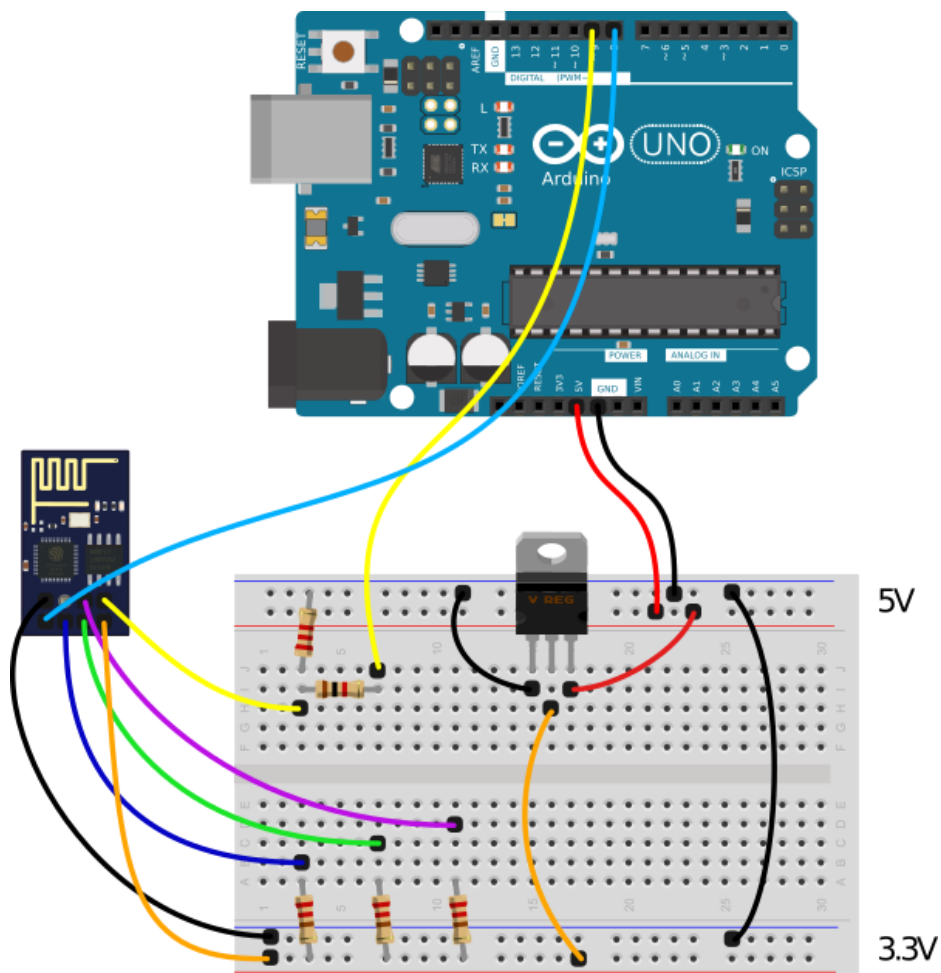


Рисунок 3.2 – Схема апаратних з'єднань Arduino із ESP8266

Тепер, коли було проведено фізичне з'єднання, за допомогою програмного забезпечення Arduino IDE, яке зображене на рисунку 3.3 можемо розпочати процедуру прошивки [24].

Arduino IDE можна завантажити безкоштовно і він працюватиме як на Windows, так і на OS X і Linux. Це дозволяє писати комп'ютерні програми, які потім завантажуються на Arduino за допомогою USB кабелю. Ваш Arduino виконуватиме інструкції на основі взаємодії із зовнішнім світом.

Коли ви відкриєте Arduino IDE, він повинен виглядати дуже схожим, як на рисунку 3.3. У Arduino IDE розділена панель інструментів, у верхній частині з кнопками де найчастіше використовувані функції; вікно коду або ескізу в центр, де ви будете писати або переглядати свої програми; і додаткове вікно для виведення внизу. Відображається у вікні послідовний вихід комунікаційних

Зм.	Арк.	№докум.	Підпис	Дата

повідомлень між вашим ПК та Arduino, і також перерахує будь-які помилки, якщо ваш ескіз не оброблюється належним чином.

Важливо також звертати увагу на бібліотеки, у середовищі Arduino бібліотека — це невеликий фрагмент коду, який виконується як конкретна функція. Замість того, щоб повторно вводити цей самий код до ваших проєктів, ви можете додати команду, яка запозичує код з бібліотеки.

Цей ярлик економить час і полегшує підключення до таких елементів, як давач, дисплей або модуль. Arduino IDE включає в себе ряд вбудованих бібліотек, наприклад як бібліотека LiquidCrystal, що дозволяє легко спілкуватися з LCD дисплеями — і в Інтернеті є ще багато інших. Для створення проєктів у бібліотеці, вам потрібно буде імпортувати такі бібліотеки: RFID, NewPing, IRRemote та DHT.

На сайті ви знайдете всі необхідні бібліотеки. Після того як ви завантажите бібліотеки, вам потрібно буде їх встановити. Щоб встановити бібліотеку в Arduino версії 1.0.6 і вище, дотримуйтесь наведених нижче кроки:

1) Виберіть “Sketch”, далі “Включити бібліотеку”, далі “Додати бібліотеку .ZIP”.

2) Перейдіть до завантаженого ZIP-файлу та виберіть його. Для старших версії Arduino, вам потрібно буде розпакувати файл бібліотеки а потім покласти всю папку та її вміст у спеціальну теку для файлів бібліотеки.

Якщо маєте бажання встановити встановити бібліотеку вручну, перейдіть до ZIP-файлу, що містить файл бібліотеки та розпакуйте його.

Наприклад, якщо було встановлено бібліотеку до клавіатури у стиснутому файлі під назвою keypad.zip, необхідно розпакувати keypad.zip, який розгорнеться в папку з іменем keypad, яка, у свою чергу, буде містити такі файли, як keypad.cpp і keypad.h. Коли файл ZIP буде розгорнутий, потрібно перетягнути папку клавіатури у папку бібліотеки у операційній системі, а потім перезапустити програмне середовище Arduino IDE.

Далі виконуємо запуск програми Arduino IDE. В головному меню обираємо пункт «File». У списку, що відкрився обираємо наступний пункт під назвою «New». Ця опція слугує для того, щоб створити новий проєкт. Якщо ж було напрацьовано якийсь інший проєкт, який було попередньо збережено, тоді

					КВРКІ 180228.18.02.03 ПЗ	Арк
						34
Зм.	Арк.	№докум.	Підпис	Дата		

потрібно використати опцію «Open» і вибрати свій збережений проєкт, знайшовши його на диску.

Створивши новий проєкт, можемо починати написання програмного коду, який і допоможе правильно прошити Wi-Fi модуль. Така процедура здійснюється автоматично, програма самостійно опрацює введені дані і в результаті дасть відповідь на екрані стосовно успішності процедури прошивки.

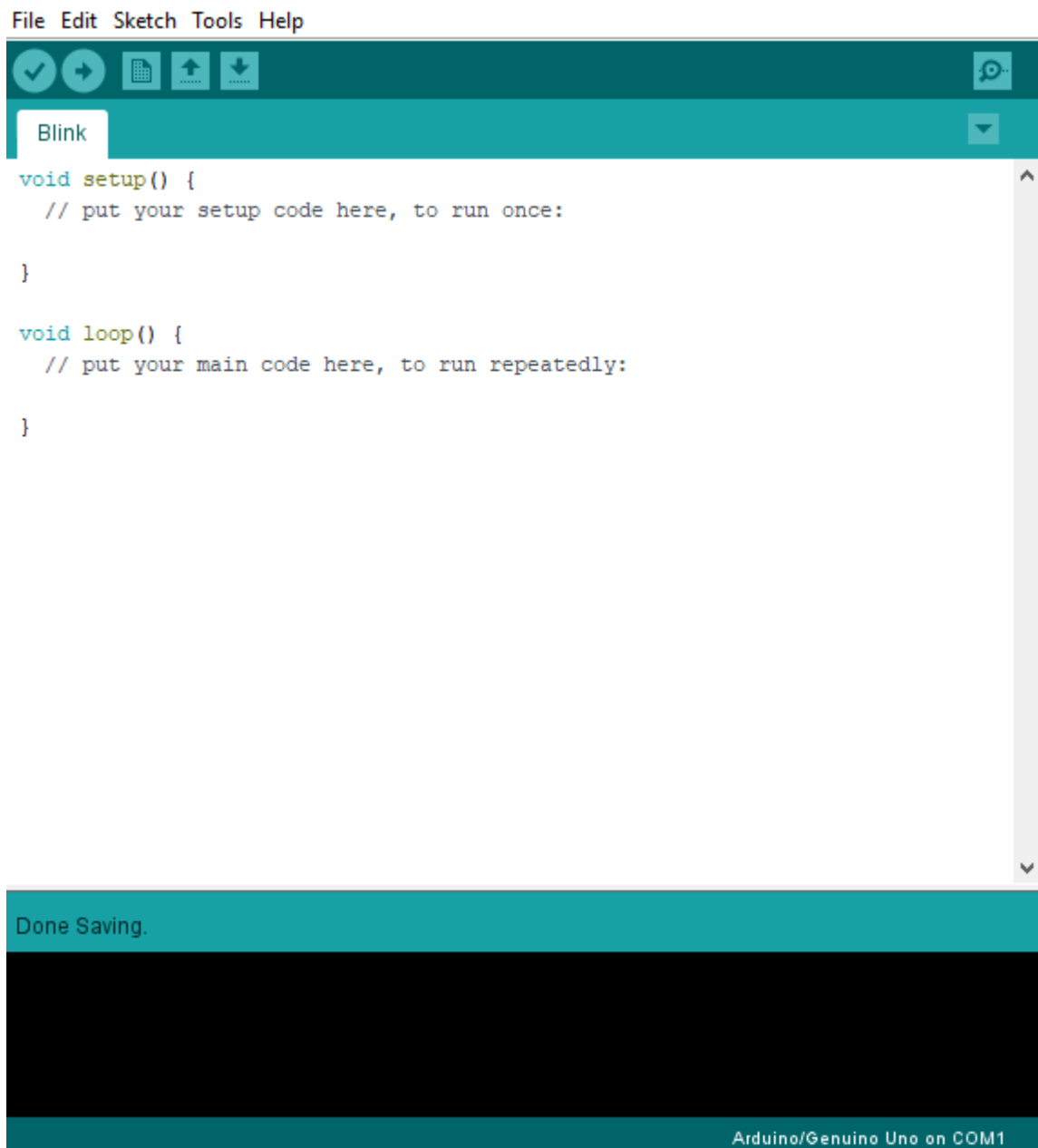


Рисунок 3.3 – Головне меню програми Arduino IDE

Прошивається Wi-Fi модуль за допомогою наступного програмного коду. Спочатку оголошується serial-порт, до якого під'єднаний модуль.

```
#define WIFI_SERIAL Serial1
```

```
void setup()
```

```
{
```

Далі необхідно відкрити порт, який здійснює моніторинг дій програмного забезпечення.

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

```
while (!Serial)
```

```
{
```

Тепер необхідно почекати до тих пір, поки не побачимо необхідний порт на екрані. Для того, щоб краще відслідковувати процеси, які відбуваються в програмі можна написати наступну команду.

```
}
```

```
Serial.print("Serial init OK\r\n");
```

Наступним кроком буде введення команди, за допомогою якої встановлюється з'єднання із Wi-Fi модулем із заданою швидкістю.

```
WIFI_SERIAL.begin(115200);
```

```
}
```

```
void loop()
```

```
{
```

У випадку, коли дані, надходять з Wi-Fi модулю надсилаємо їх в порти комп'ютера.

```
if (WIFI_SERIAL.available())
```

```
{
```

```
Serial.write(WIFI_SERIAL.read());
```

```
}
```

У випадку, якщо дані надходять з комп'ютеру, тоді надсилаємо їх у Wi-Fi модуль.

```
if (Serial.available())
```

```
{
```

					КВРКІ 180228.18.02.03 ПЗ	Арк
						36
Зм.	Арк.	№докум.	Підпис	Дата		

```
WIFI_SERIAL.write(Serial.read());  
}  
}
```

На рисунку 3.4 показано, як саме виглядає програмний код прошивки у програмному середовищі Arduino IDE



```
File Edit Sketch Tools Help  
Blink  
#define WIFI_SERIAL Serial1  
void setup()  
{  
Serial.begin(9600);  
while (!Serial) {  
}  
Serial.print("Serial init OK\r\n");  
WIFI_SERIAL.begin(115200);  
}  
  
void loop()  
{  
if (WIFI_SERIAL.available()) {  
Serial.write(WIFI_SERIAL.read());  
}  
if (Serial.available()) {  
WIFI_SERIAL.write(Serial.read());  
}  
}  
  
Done Saving.
```

Рисунок 3.4 – Програмний код прошивки у Arduino IDE

Тепер, коли було написано необхідний програмний код для прошивки, потрібно провести тестування, щоб перевірити як працює Wi-Fi модуль. Для цього треба зайти в меню, що має назву «Tools», та обрати плату Arduino Uno. Перед цим обов'язково перевірте, чи під'єднана плата до комп'ютера. Далі знову заходимо у меню «Tools» і обираємо там пункт «Serial Port», в якому підбираємо

той порт, який під'єднаний до плати. Натискаємо на клавішу «Upload», та очікуємо на відгук програми. В процесі цього можливо, що індикатори на платі почнуть реагувати і будуть світитись.

Після того, як програма завершить завантажуватись, вона виведе на інтерфейс відповідне повідомлення, та покаже усі результати. Завершення процесу тестування зображено на рисунку 3.5.



The screenshot displays the Arduino IDE interface. At the top, the menu bar includes 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for saving, undo, redo, and uploading. The sketch name 'Blink' is shown in the top right corner. The main workspace contains the following C++ code:

```
#define WIFI_SERIAL Serial1
void setup()
{
  Serial.begin(9600);
  while (!Serial) {
  }
  Serial.print("Serial init OK\r\n");
  WIFI_SERIAL.begin(115200);
}

void loop()
{
  if (WIFI_SERIAL.available()) {
    Serial.write(WIFI_SERIAL.read());
  }
  if (Serial.available()) {
    WIFI_SERIAL.write(Serial.read());
  }
}
```

At the bottom of the IDE, a status bar shows '37' and 'Arduino/Genuino Uno or CCM4'. A message box at the bottom of the sketch area displays the following information:

```
Done uploading
Sketch uses 930 bytes (2%) of program storage space. Maximum is 32256 bytes.
Global variables use 9 bytes (0%) of dynamic memory, leaving 2039 bytes for local variables.
```

Рисунок 3.5 – Завершення процесу тестування Wi-Fi модулю

Перевагою даного програмного забезпечення є його невеликі потреби у використанні пам'яті комп'ютера, що дозволяє здійснювати такі тестування швидко і без сильного навантаження на систему.

Тестування роботи Wi-Fi модулю пройшло успішно, всі індикатори та світлодіоди реагують належним чином. Апаратне з'єднання модулю ESP8266 із апаратною обчислювальною платформою Arduino Uno показано на рисунку 3.6.

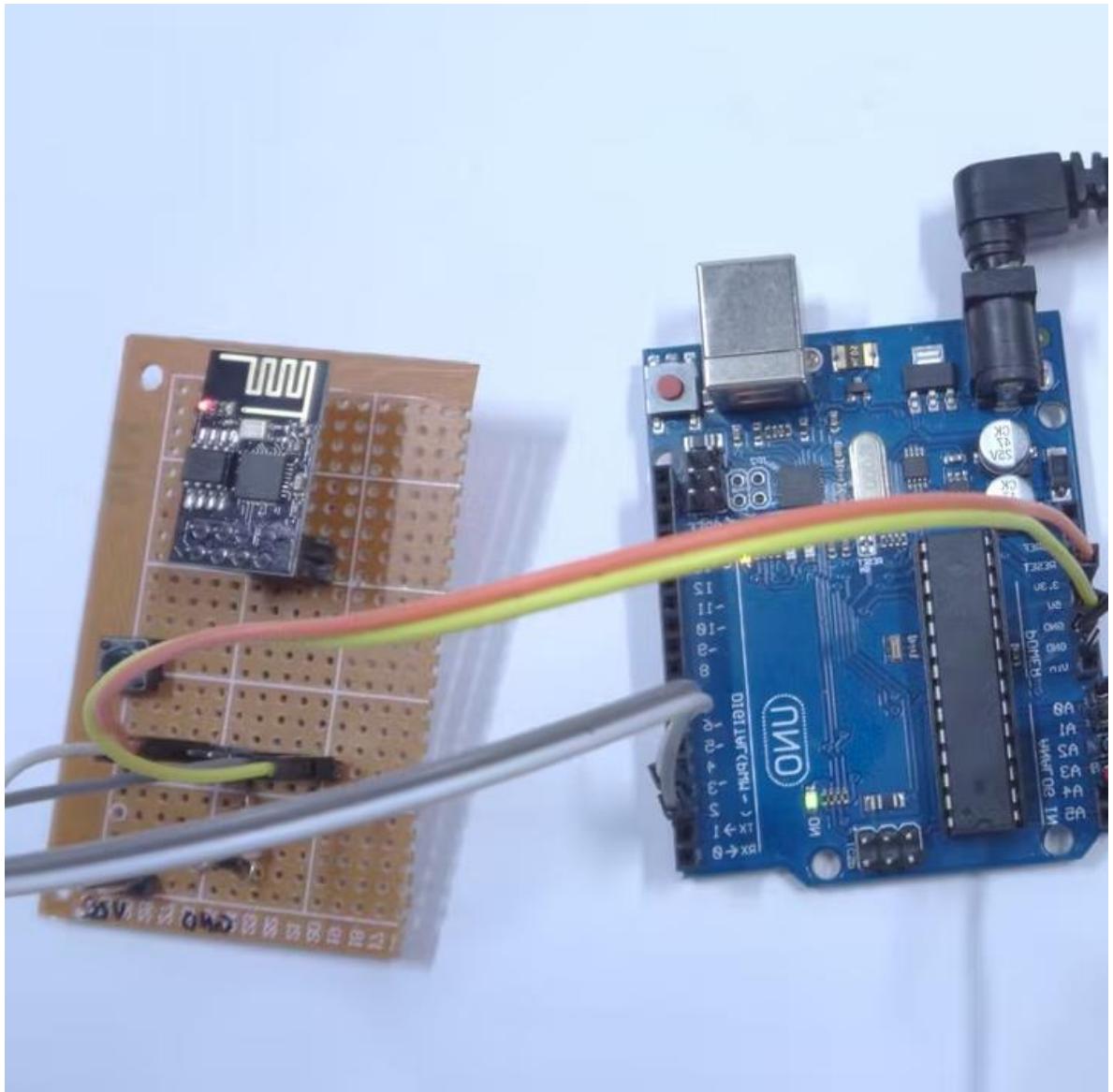


Рисунок 3.6 – Апаратне з'єднання ESP8266 із Arduino Uno

3.2 Створення локального хост серверу

Зм.	Арк.	№докум.	Підпис	Дата

Можна сказати, що вся система складається з двох частин: сервера і клієнта. У нашому випадку сервер є веб-інтерфейсом, що складається з кнопок та інтерфейсу користувача (UI), який буде дозволяти увімкнути, або вимкнути пристрій. Він складається з файлів програмного коду, файлів HTML і файлів “.txt” для зберігання даних.

Сервер зазвичай зберігає інформацію про натиснуту кнопку на сторінці та у файлі “.txt”. У цьому проєкті інтерфейс користувача складається з п’яти пристроїв, кожен має по дві кнопки для ввімкнення та вимкнення. Натискання кнопок ініціює виконання файлу головної програми, яка називається button.php.

Ця програма служить інтерфейсом прикладного програмування для зберігати даних в текстовому файлі під назвою buttonStatus.txt.

Зв’язок з віддаленим користувачем встановлюється за допомогою Інтернету, який вимагає IP адреса для спілкування. Веб-сервер створюється за допомогою програмного забезпечення Python та мережі Інтернет. Інтерфейс створений за допомогою програми Dreamweaver.

За допомогою веб-інтерфейсу, який виконує роль графічного інтерфейсу користувач приймає отримані інструкції в базу даних сервера. Згідно з даними, які отримані з бази даних веб-сервера, мікроконтролер посилає сигнали на свій вихідний порт прив'язаний до схеми драйвера для сигналу. Таким чином вмикається або вимикається потрібний пристрій.

Блок для схема драйвера відповідає за посилення сигналу, отриманого від вихідних контактів мікроконтролера, до рівня, який може активувати або знеструмити реле. За станом с реле, підключена до нього лампочка або ввімкнеться, або вимкнеться. Блок живлення буде забезпечити необхідні регульовані рівні напруги на всі компоненти системи.

Для того, щоб мати змогу запускати програми з різних пристроїв потрібно визначити хост сервер. До прикладу адміністратор мережі може використовувати свій комп’ютер для запуску веб-сервера на одній системі та використовувати спеціальну програму, щоб здійснювати віддалений доступ на іншій системі.

Така програма буде запускатися з комп’ютера, який є відмінним від локального хосту, тобто локальний хост є стандартною назвою хоста, що

					КВРКІ 180228.18.02.03 ПЗ	Арк
						40
Зм.	Арк.	№докум.	Підпис	Дата		

призначається автоматично [32]. Наступні кроки будуть полягати у тому, щоб за допомогою програмного коду та певної IP адреси створити локальний хост сервер, щоб мати змогу під'єднуватись до нього іншими пристроями та керувати ними. Для цього у програмі Arduino IDE, здійснивши з'єднання апаратної обчислювальної платформи Arduino Uno та Wi-Fi модулю ESP8266 потрібно вводити наступний програмний код.

Перейдемо до безпосереднього створення головного серверу.

Спочатку потрібно увімкнути основний файл. Далі створюється ім'я та пароль для процедури авторизації. Вибір імені та паролю здійснює сам користувач.

```
#include <LocalServer.h>
const char* ssid = "LocalServerVadym";
const char* password = "Badroom123";
```

Також запровадимо власний сервер, порт якого буде відповідати цифрам 70. Саме ці цифри будуть відповідати за порт, через який будуть проходити необхідні дані. Саме підключення до серверу також буде здійснюватись через цей порт. Тепер необхідно оголосити змінну, тип якої є рядковим. Це здійснюється спеціально для запуску HTML

```
LocalServer server(70);
String header;
```

Оголосимо наступні змінні, які будуть відповідати за поточний статус пристроїв, тобто вони будуть показувати чи є пристрої увімкненими, чи ні.

```
String Device1State = "off";
String Device2State = "off";
String Device3State = "off";
String Device4State = "off";
```

Далі оголосимо змінні цифрових контактів ESP8266. За допомогою них різні пристрої можуть під'єднуватись.

```
const int Device1 = 17;
const int Device2 = 16;
const int Device3 = 15;
```

```
const int Device4 = 14;
```

Наступна команда відповідає за час, за який Wi-Fi модуль ESP8266 з'єднається із мережею. Важливо провести тестування і перевірити оптимальний варіант часу з'єднання, щоб мати можливість підкоригувати його.

```
unsigned long currentTime = millis();
```

```
unsigned long previousTime = 0;
```

```
const long timeoutTime = 3000;
```

```
void setup()
```

```
{
```

```
Serial.begin(115200);
```

Тепер треба оголосити усі релейні модулі у режим вихідних контактів.

```
pinMode(Device1, OUTPUT);
```

```
pinMode(Device2, OUTPUT);
```

```
pinMode(Device3, OUTPUT);
```

```
pinMode(Device4, OUTPUT);
```

Після цього потрібно обов'язково вимкнути усі релейні модулі. Це робиться для того, щоб під час початкового запуску під'єднані пристрої не вмикалися самі собою.

```
digitalWrite(Device1, LOW);
```

```
digitalWrite(Device2, LOW);
```

```
digitalWrite(Device3, LOW);
```

```
digitalWrite(Device4, LOW);
```

Наступний програмний код здійснює саме підключення до Wi-Fi мережі через встановлене раніше ім'я мережі та її пароль.

```
Serial.print("Connecting to ");
```

```
Serial.println(ssid);
```

```
LocalServer.begin(ssid, password);
```

```
while (WiFi.status() != WL_CONNECTED) {
```

```
delay(700);
```

```
Serial.print(".");
```

```
}
```

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		42

Для запуску сервера потрібно набрати на клавіатурі локальну IP адресу.

```
Serial.println("");
```

```
Serial.println("WiFi connected.");
```

Щоб отримати IP адресу додамо ще одні рядки програмного коду. Їх суть полягає у тому, що при підключенні модулю ESP8266 до мережі Wi-Fi з'являється необхідна нам адреса. Саме її і використаємо для того, щоб відкрити сторінку сервера.

```
Serial.println("IP address: ");
```

```
Serial.println(WiFi.localIP());
```

```
server.begin();
```

```
}
```

```
void loop()
```

```
{
```

Увесь набраний нами програмний код у Arduino IDE зображено на рисунку

3.7.



```
File Edit Sketch Tools Help
Blink
#include <LocalServer.h>
const char* ssid = "LocalServerVadym";
const char* password = "Badroom123";
LocalServer server(70);
String header;
String Device1State = "off";
String Device2State = "off";
String Device3State = "off";
String Device4State = "off";
const int Device1 = 17;
const int Device2 = 16;
const int Device3 = 15;
const int Device4 = 14;
unsigned long currentTime = millis();
unsigned long previousTime = 0;
const long timeoutTime = 3000;
void setup() {
  Serial.begin(115200);
  pinMode(Device1, OUTPUT);
  pinMode(Device2, OUTPUT);
  pinMode(Device3, OUTPUT);
  pinMode(Device4, OUTPUT);
  digitalWrite(Device1, LOW);
  digitalWrite(Device2, LOW);
  digitalWrite(Device3, LOW);
  digitalWrite(Device4, LOW);
  Serial.print("Connecting to ");
  Serial.println(ssid);
  LocalServer.begin(ssid, password);
  while (LocalServer.status() != WL_CONNECTED) {
    delay(700);
    Serial.print(".");
  }
}
```

Рисунок 3.7 – Програмування локального серверу у Arduino IDE

Після програмування локального серверу, можливе відображення певних команд, які свідчать про наявний статус серверу. Наведемо нижче їх перелік та пояснення їх значення:

- 1) WL_CONNECTED: з'являється при підключенні до мережі Wi-Fi
- 2) WL_NO_SHIELD: з'являється у випадку, якщо захист у мережі Wi-Fi відсутній
- 3) WL_IDLE_STATUS: тимчасовий статус, який призначається під час виклику функції WiFi.begin(). Статус залишається активним, поки не закінчиться кількість спроб з'єднання, або не буде встановлено з'єднання
- 4) WL_NO_SSID_AVAIL: статус з'являється у випадку, коли SSID недоступний
- 5) WL_SCAN_COMPLETED: призначається після завершення сканування мереж
- 6) WL_CONNECT_FAILED: призначається, коли підключення не вдається для всіх спроб
- 7) WL_CONNECTION_LOST: з'являється у тому випадку, коли підключення зникло
- 8) WL_DISCONNECTED: з'являється у тому випадку, коли користувач від'єднався від мережі

3.3 Проведення апаратного з'єднання релейних модулів із мікросхемою

Тепер потрібно під'єднати релейні модулі Tolako до мікросхеми Arduino Uno та запрограмувати їх у середовищі Arduino IDE. Для цього знадобиться релейний модуль, Arduino (у нашому випадку Arduino UNO), лампочка змінного струму та дріт живлення з 2-контактним штекером.

Спочатку береться дріт живлення та відокремлюється нульовий дріт та дріт живлення, що знаходиться всередині нього. Під'єднується червоний дріт до

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		44

лампочки, а нейтральний (чорний або синій) дріт під'єднується до розімкнутого роз'єму на релейному модулі.

Далі підключають другий дріт лампочки до роз'єму COM на релейному модулі. Ось так під'єднується реле з лампочкою змінного струму для перемикання.

Тепер потрібно подати живлення до релейного модуля та подати йому сигнал від Arduino, який повідомляє реле вмикати та вимикати лампочку. Для цього потрібно підключити контакт зі знаком мінус (-) на модулі реле до землі (GND) на Arduino, а плюсовий (+) контакт релейного модуля до джерела живлення 5 В на платі Arduino. Третій контакт, що залишився на модулі реле, є командним контактом, який необхідно підключити до контакту вводу-виводу на Arduino. Для цієї мети ми розглядаємо контакт 7. Ви можете вибрати будь-який цифровий контакт введення-виводу, який вам подобається, але переконайтеся, що запрограмували свій Arduino UNO відповідним чином.

Після підключення обладнання, як описано, потрібно написати простий програмний код, який встановлює контакт 7 як вихідний контакт і посилає сигнал на включення та вимкнення лампочки через модуль реле. Ось як можна написати код для релейного модуля:

```
void setup()
{
  pinMode(7, OUTPUT);
}
```

Дані команди визначають шляхи виводу даних.

```
void loop()
{
  digitalWrite(7, HIGH);
  delay(1000);
  digitalWrite(7, LOW);
```

Тепер встановлюємо затримку на реагування.

```
  delay(1000);
}
```

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		45

Таке явище, як затримка, життєво необхідне у випадках, коли іноді нашому коду потрібно чекати на якісь речі або події, які займають певний проміжок часу.

Наприклад, користувач може захотіти увімкнути світлодіод на кілька секунд, а потім вимкнути його. В такому випадку керувати світлодіодом за допомогою команди `digitalWrite()` не дуже зручно, тож яка команда відповідає за затримку на кілька секунд?

Один простий метод це - створити порожній цикл. Порожній цикл являє собою петлю, яка насправді нічого не робить, окрім, як займається витрачанням часу. Наприклад, якщо є знання того, що користувач просто збільшує, тестує і розгалужує в циклі, який займає мікросекунду, тоді є можливість написати таку функцію:

```
unsigned long endmsec = msec * 1000;
```

Зауважте, що тут вказується кількість мілісекунд, яку має бути витрачена. Оскільки кожна ітерація циклу займає одну мікросекунду, то вона множить на 1000, щоб отримати мілісекунди.

Важливим тут також є модифікатор `volatile`. Він говорить компілятору не оптимізувати програмний код надто сильно, тому що міг бути змінений програмний код, що виконується в іншому місці (наприклад, у перериванні).

В іншому випадку компілятор може зрозуміти, що він може досягти того ж кінцевого результату, ігноруючи цикл і виконуючи просте додавання. Проблема з цією функцією полягає в тому, що результуюча затримка сильно залежить від використовуваного мікроконтролера та його тактової частоти. Якщо користувачу просто потрібна швидка й невибаглива затримка, тоді це спрацює добре, але набагато точніша затримка доступна лише з функцією `delay()` та її братньою функцією `delayMicroseconds()`.

Після завантаження коду на плату Arduino та підключення схеми, як показано вище, можна помітити, що реле автоматично вмикає та вимикає лампочку через кожну секунду, коли отримує команду від Arduino.

Схема підключення релейного модулю до мікросхеми показана на рисунку 3.8.

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		46

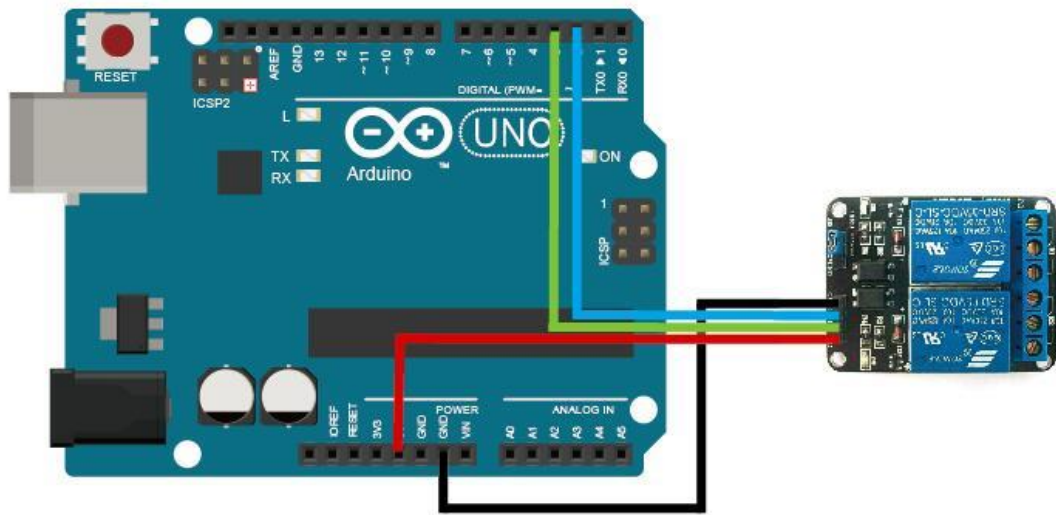


Рисунок 3.8 – Схема підключення релейного модулю Tolako 5V

Маючи схему, розпочнемо безпосереднє апаратне з'єднання, яке зображене на рисунку 3.9.

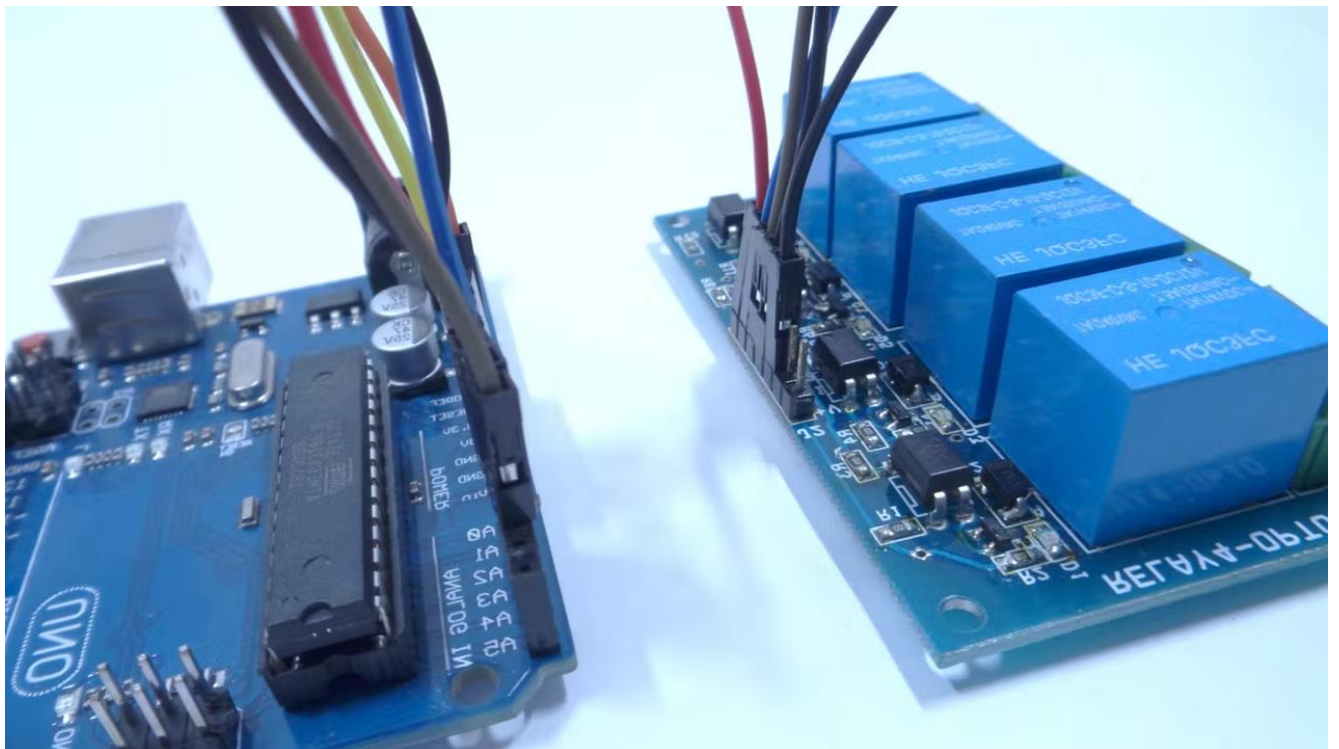


Рисунок 3.9 – Апаратне з'єднання релейного модулю Tolako із Arduino Uno

Зм.	Арк.	№докум.	Підпис	Дата

Обов'язково потрібно здійснювати процедуру з'єднання, чітко дотримуючись інструкції, адже релейний модуль - це електричний вимикач, який використовує електромагніт для переміщення перемикача зі свого положення.

Замість того, щоб людина рухала перемикач це робить звичайний електромагніт.

Для того, щоб увімкнути релейний модуль потрібна відносно невелика кількість енергії, але реле може керувати тим, що споживає набагато більше енергії. Наприклад, реле використовується для управління кондиціонером у домі.

Варто зазначити, що блок змінного струму, який міститься у реле, працює від 220В змінного струму при силі струму близько 30 А. Це 6600 Вт. Котушці, яка здійснює керування реле, потрібно лише кілька ват, щоб з'єднати контакти. Вихідні контакти нормально розімкнуті (тобто не під'єднані).

При пропусканні струму вхідна котушка створює магнітне поле, яке замикає перемикач. Зазвичай пружина тягне перемикач знову, як тільки він розмикається, і як тільки живлення від'єднується від котушки.

3.4 Проведення апаратного з'єднання дисплею LCD1602 із мікросхемою

Наступним етапом буде апаратне з'єднання дисплею LCD1602 із мікросхемою Arduino Uno. Щоб досягти цього, необхідно дотримуватись схеми, на якій зображено, як саме здійснюється з'єднання. Дана схема зображена на рисунку 3.10.

Необхідно підготувати рідкокристалічний дисплей, бо він, ймовірно, потребуватиме особливого підходу до з'єднання із мікросхемою. Дисплей має бути з 16 отворами і окремими смужками штифтів.

Візьміть смужку шпильок і відламайте 16 ряд шпильок. Вставте штифти коротшою стороною у 16 отворів дисплею. П'яти їх потрібно буде на місці: спочатку припаяйте крайній правий і крайній лівий контакти, щоб з'єднати їх затримайте їх на місці і зачекайте, поки вони схопляться.

Потім кожен шпильку необхідно припаяти по черзі, притримуючи паяльну суміш і паяльник до кожного штиря. Пам'ятайте, що притримування заліза до

					КВРКІ 180228.18.02.03 ПЗ	Арк
						48
Зм.	Арк.	№докум.	Підпис	Дата		

шпильок занадто довго пошкодить їх; вам тільки потрібно припаяти їх на пару секунд.

Помістіть рідкокристалічний дисплей у макетну плату, вставивши заголовок шпильками в отвори макетної плати.

Також вставимо потенціометр в макетну плату та використаємо макетну плату та перемичку для під'єднання до дисплею, мікросхеми Arduino та потенціометру.

Є три GND підключення від рідкокристалічного модуля, тому використовуйте макетну плату GND залізниці, щоб здійснити ці з'єднання.

Центральний контакт потенціометра 50 кОм під'єднується до контакту дисплея 3. Потенціометр контролює контрастність екрана. Повертайте його, поки не побачите символи на екрані. Тепер підключіть один із зовнішніх контактів до GND, а інший до +5V

Рідкокристалічні дисплеї з підсвічуванням будуть мати вбудовані резистори, але якщо у вас дисплей без підсвічування, вам слід вставити резистор 220 Ом між РК 15 і +5 В.

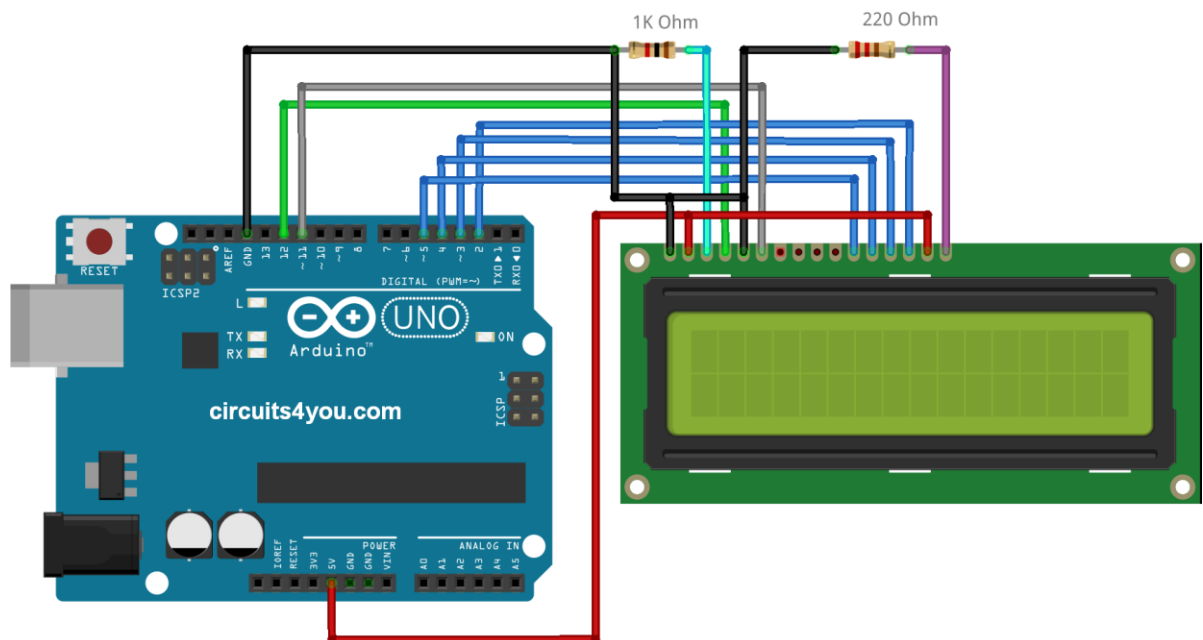


Рисунок 3.10 - Схема апаратних з'єднань Arduino із LCD1602

Зм.	Арк.	№докум.	Підпис	Дата

За наявності схеми, можемо розпочати з'єднання. На рисунку 3.11 зображено результат такого з'єднання.

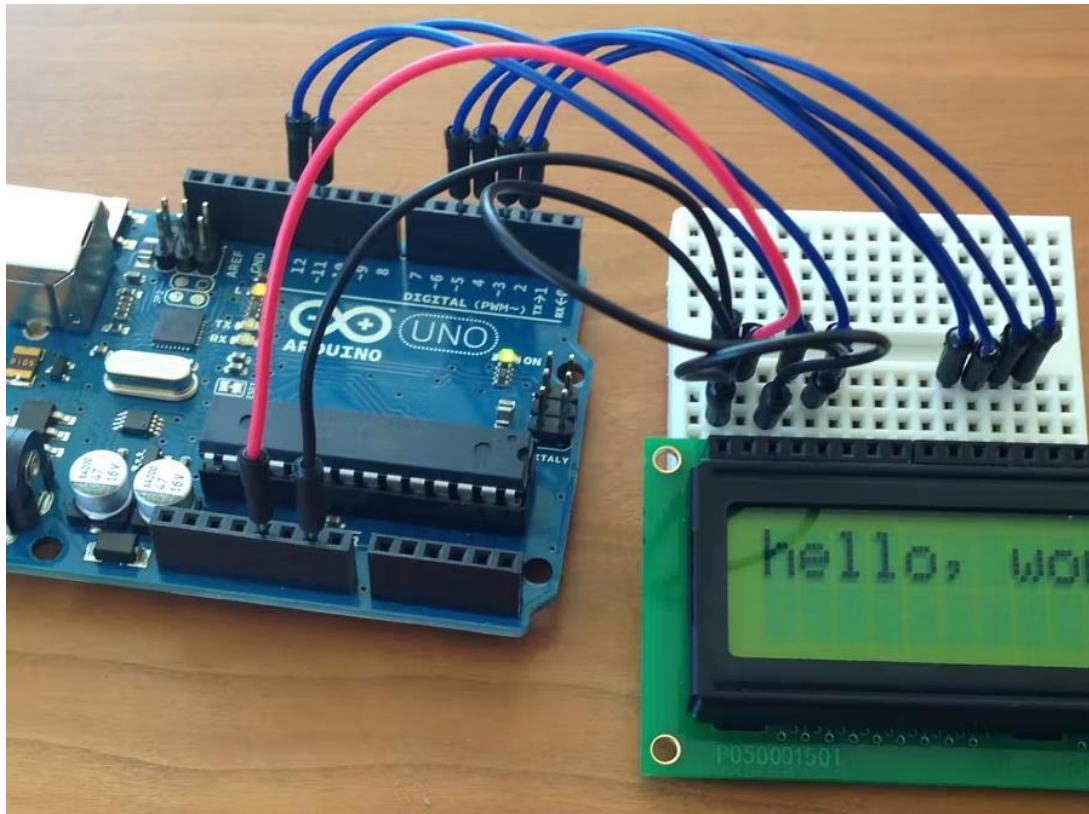


Рисунок 3.11 - Апаратне з'єднання дисплею LCD1602 із Arduino Uno

Тепер можемо почати роботу у програмному середовищі Arduino IDE. Знову відкриваємо програму, обираємо даний пристрій та починаємо вводити програмний код.

Для початку вводим команди, які здійснюють ініціалізацію LCD дисплею.

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

```
void setup()
```

```
{
```

Встановимо число рядків та стовпчиків для LCD дисплею та розпочнемо ініціалізацію з заданою швидкістю.

```
lcd.begin(16, 2);
```

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

```
}
```

```
void loop()
```

```
{
```

Наступні команди відповідають за затримку при появі символів на екрані, а також за читання цих символів.

```
if (Serial.available())
```

```
{
```

```
delay(100);
```

```
lcd.clear();
```

```
while (Serial.available() > 0)
```

```
{
```

Вводимо команду, для виведення символів на екран дисплею.

```
lcd.write(Serial.read());
```

```
}
```

```
}
```

Тепер необхідно зафіксувати рядки та стовпчики на оптимальному місці, для відображення символів.

```
{
```

```
lcd.setCursor(0, 2);
```

```
lcd.print(millis()/1000);
```

```
}
```

```
}
```

Набраний програмний код у середовищі Arduino IDE зображено на рисунку 3.12.

```
✓ ↻ 📄 ⬆️ ⬇️ 🔍
Blink
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
|
void setup(){
  lcd.begin(16, 2);
  Serial.begin(9600);
}

void loop()
{
  if (Serial.available())
  {
    delay(100);
    lcd.clear();
    while (Serial.available() > 0)
    {
      lcd.write(Serial.read());
    }
  }

  {
    lcd.setCursor(0, 2);
    lcd.print(millis()/1000);
  }
}
```

Рисунок 3.12 – Програмування дисплею LCD1602

Щодо взаємодії дисплею 1602 із іншими пристроями, то його програма повинна взаємодіяти із зовнішнім світом за допомогою введення та виведення пристроїв, які безпосередньо взаємодіють із людиною.

Один з найпоширеніших пристроїв для 1602 - це підключений рідкокристалічний дисплей. Деякі з найпоширеніших рідкокристалічних дисплеїв, підключених до 1602 — це дисплеї розмірами 16x2 і 20x2. Це означає 16 символів на рядок, по 2 рядки і 20 символів у рядку на 2 рядки відповідно.

На щастя, існує дуже популярний стандарт, який дозволяє нам взаємодіяти з величезною більшістю рідкокристалічних дисплеїв, незалежно від їх виробника.

Стандарт називається HD44780U, який відноситься до мікросхеми контролера, який отримує дані від зовнішнього джерела (у нашому випадку,

LCD1602) і зв'язується безпосередньо з рідкокристалічним дисплеєм. Стандарт 44780 вимагає 3 лінії для здійснення керування, а також 4 або 8 ліній введення, або виводу для шини даних. Користувач може вибрати, чи буде рідкокристалічний дисплей працювати з 4-розрядною шиною даних або 8-розрядною шиною даних. Якщо використовується 4-розрядна шина даних, рідкокристалічному дисплею буде потрібно всього 7 рядків даних (3 лінії керування, а також додаткові лінії для шини даних). Якщо використовується 8-розрядна шина даних, рідкокристалічному дисплею буде потрібно всього 11 рядків даних (3 лінії керування плюс 8 ліній для шини даних).

Три контрольні лінії називаються EN, RS і RW. Рядок EN називається «Увімкнути». Цей контрольний рядок використовується для повідомлення РК-дисплею про те, що ви надсилаєте це дані. Щоб надіслати дані на рідкокристалічний дисплей, ваша програма повинна переконатися, що цей рядок не працює, а потім встановити дві інші лінії керування, або помістити дані на шину даних. Коли інші рядки є повністю готовими, приведіть контрольну лінію, яка відповідає за увімкнення до готовності і дочекайтеся мінімальної кількості часу, необхідного для таблиці даних дисплея.

3.5 Проведення апаратного з'єднання додаткових комплектуючих пристрою

Після того, як було забезпечено належний програмний рівень комплектуючих пристрою, можна перейти до етапу безпосередньої збірки усіх елементів пристрою.

Для кращого з'єднання дисплею, було використано чотиризначний семисегментний послідовний дисплей, який має вбудовану інтегральну схему для управління світлодіодами і може бути підключений до Arduino лише за допомогою трьох дротів. Вибираючи дисплей, переконайтеся, що він має вхід RX, щоб ви могли керувати ним лише за допомогою одного дроту.

Під'єднайте семисегментний послідовний дисплей RX до контакту мікросхеми Arduino за номером 3, підключіть VCC до +5V і підключіть GND до

					КВРКІ 180228.18.02.03 ПЗ	Арк
						53
Зм.	Арк.	№докум.	Підпис	Дата		

Arduino GND через макетну плату. В процесі цього можливо знадобиться зняти частину дроту, щоб зробити з'єднання.

Вставте потенціометр в макетну плату і підключіть лівий контакт до +5V, центральний контакт до контакту Arduino A0, а правий контакт до GND. Підключіть червоний дріт до контакту 4 і чорний провід від Arduino до GND. Вставте дві кнопки в макетну дошку шпильками A і B з одного боку центрального розриву і шпильками D і C на інший бік, відповідно до конфігурації на мікросхемі.

Далі з'єднуються кнопки. Щоб створити кнопку Arm, підключіть контакт C першої кнопки до GND і контакт D до контакту 5 на мікросхемі Arduino. Щоб створити кнопку запуску, підключіть контакт C іншої кнопки до GND і контакт D до Контакт Arduino 6. Вставте червоний світлодіод у макетну плату, де з коротшим мінусом ніжка підключена до контакту B кнопки Arm.

З'єднайте іншу ніжку до резистора 220 Ом і підключіть іншу сторону резистора до +5V. Потім вставте зелений світлодіод із підключеною мінусовою ніжкою до контакту кнопки запуску, де плюсова ніжка підключена до +5 В через резистор 220 Ом. Підключіть запальник.

В даній роботі використовується жовтий світлодіод в якості запальника індикатора на даний момент. Вставте його в макет, з від'ємним струмом, де ніжка підключена до GND, а додатня ніжка підключена до контакту Arduino 7 через резистор 220 Ом. Коли зворотний відлік досягає 0, контакт 7 встановлюється на запуск і спрацьовує запальник.

Замість того, щоб насправді запалювати запобіжник, запалюється жовтий світлодіод, який буде відображати запалювання. Переконайтеся, що ваше налаштування відповідає принциповій схемі.

З'єднуємо усі комплектуючі між собою за допомогою дротів, за схемами, які були вказані раніше на рисунках 3.2 та 3.10. На рисунку 3.13 зображено, як саме здійснено з'єднання.

					КВРКІ 180228.18.02.03 ПЗ	Арк
						54
Зм.	Арк.	№докум.	Підпис	Дата		

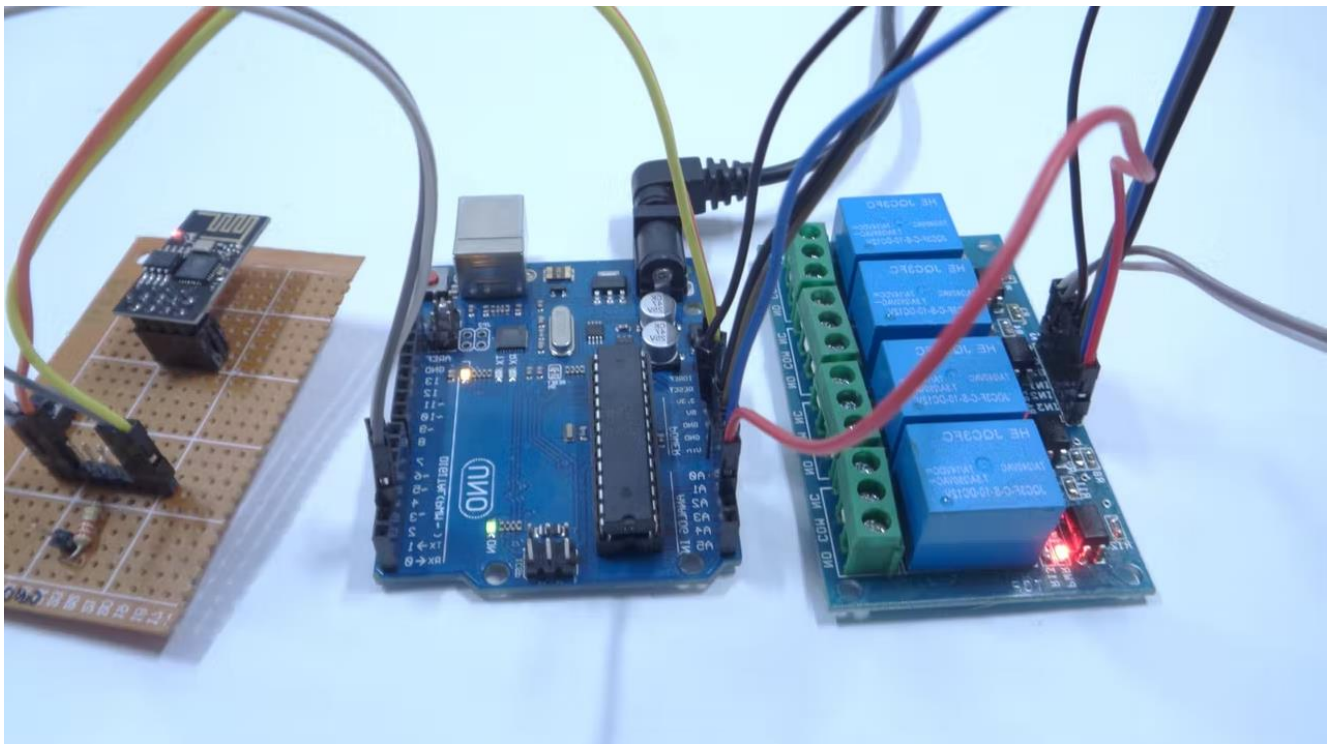


Рисунок 3.13 – Апаратне з'єднання комплектуючих пристрою

3.6 Тестування роботи пристрою

Фінальним етапом буде проведення тестування програмно-апаратного засобу на його здатність працювати належним чином. Таке тестування є обов'язковим, адже є висока необхідність у перевірці пристрою на наявність проблем із використанням. Під час тестування важливо дотримуватись заходів безпеки та поводитись із пристроєм обережно.

Спочатку вмикаємо пристрій у електричну мережу за допомогою розетки. Після того, як пристрій був увімкнений, на його дисплеї повинні з'явитись символи, що позначають його стан, як увімкнений. Наступний крок – це налагодження локальної мережі будинку. Для цього необхідно обрати будь який пристрій, що може під'єднуватись до мережі. У нашому випадку це смартфон. Тепер потрібно за допомогою Інтернет браузера зайти за попередньо встановленою адресою мережі. Зовнішній вигляд сторінки інтерфейсу керування приладами у будинку зображено на рисунку 3.14.

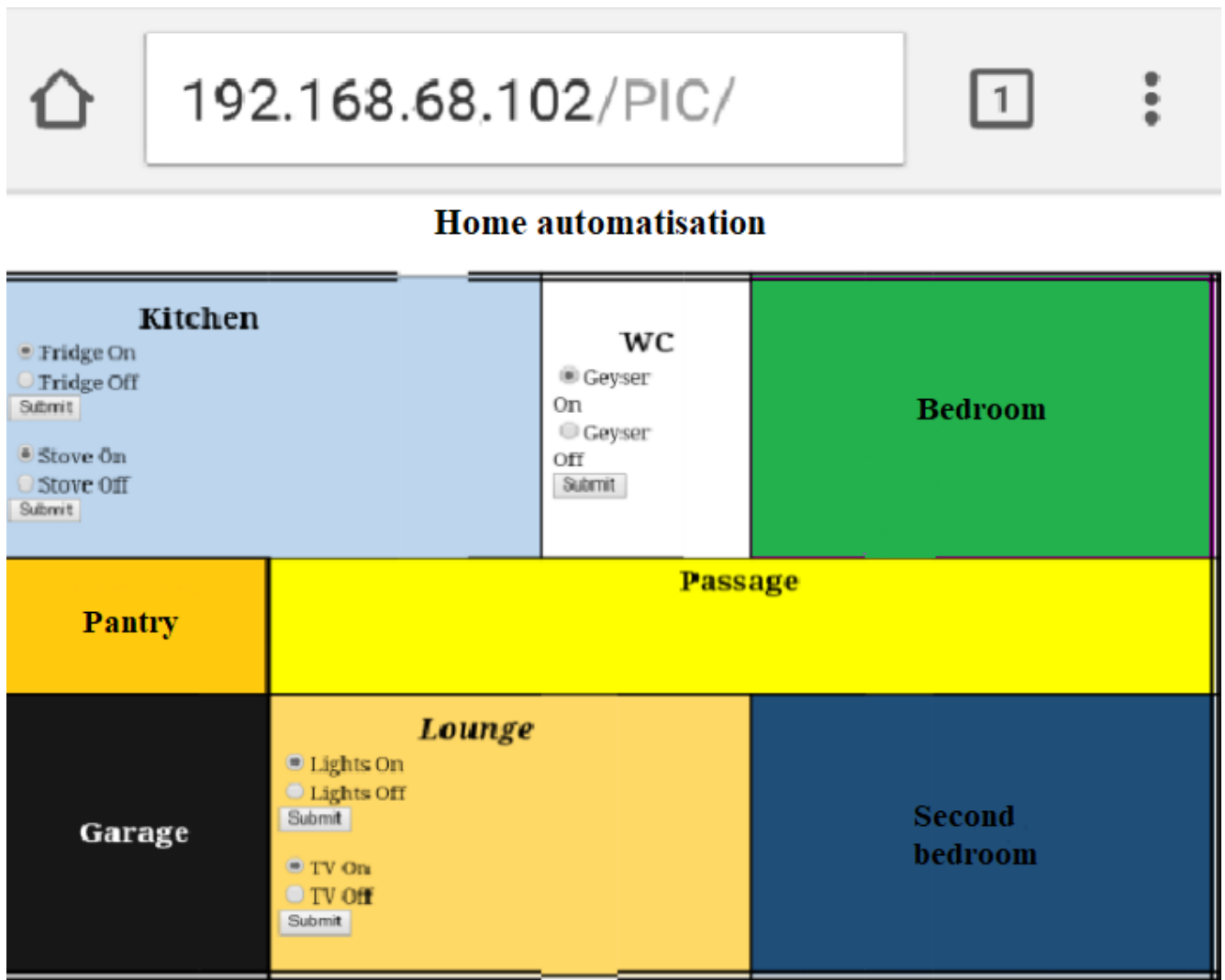


Рисунок 3.14 – Вигляд інтерфейсу керування приладами на сторінці браузера

Після того, як було відкрито сторінку керування пристроями у мережі, потрібно обрати всі пункти по черзі. Із натисканням на різні опції, лампочки будуть реагувати на вхідну команду і почнуть світитись.

Таким чином необхідно провести тестування усіх трьох опцій, які відповідають за стани роботи приладів. Якщо усі три лампочки відреагували на вхідні команди неналежним чином, тоді потрібно перевірити пристрій на наявність несправностей, або перевірити налаштування локальної мережі в будинку. На рисунку 3.15 показано процес тестування роботи пристрою.



Рисунок 3.15 – Процес тестування роботи пристрою

Як можемо бачити, тестування роботи відбулось успішно, усі елементи та комплектуючі пристрою відреагували належним чином та без збоїв. Таким чином можна констатувати, що програмно-апаратний засіб автоматичного контролю будинку пройшов тестування та готовий до експлуатації.

3.7 Висновки

У цьому розділі було успішно реалізовано програмно-апаратний засіб автоматичного контролю будинку. Цей розділ включав у себе важливі етапи реалізації приладу, а саме:

- 1) Апаратне з'єднання комплектуючих, які необхідні для роботи приладу;
- 2) Програмування та прошивання комплектуючих, задля забезпечення їх оптимальної роботи;
- 3) Створення спеціальної локальної мережі будинку, до якої можуть приєднуватись інші прилади;
- 4) Тестування роботи програмно-апаратного засобу.

Можемо констатувати, що всі етапи були успішно пройдені. Завдяки вдалому плануванню, проєктуванню та оптимальному підходу вдалось реалізувати даний проєкт.

					КВРКІ 180228.18.02.03 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		58

ВИСНОВКИ

Програмно-апаратний засіб автоматичного контролю будинку – це проект, який полягає у автоматизації багатьох буденних речей навколо користувача. Його ідеєю є контроль електричних приладів, шляхом долучення їх до загальної мережі будинку. Користувач за допомогою програмно-апаратного засобу може здійснювати контроль над ними, вмикати, чи вимикати їх за власним бажанням.

Головна мета даної роботи полягає у тому, щоб спроектувати, а потім реалізувати програмно-апаратний засіб автоматичного контролю будинку, який буде працювати за допомогою мережевих систем. Реалізація засобу буде відбуватись на базі Arduino.

Об'єкт дослідження у даній роботі – це програмно-апаратний засіб автоматичного контролю будинку за допомогою мережевих систем на базі Arduino.

Предметом дослідження виступають саме мережеві системи автоматичного контролю будинку на базі Arduino.

Перший розділ кваліфікаційної роботи містить у собі інформацію про дослідження мережних систем, її концепцію та структуру. Також у розділі присутня концепція Інтернету речей, яка за своєю суттю є близькою до теми роботи. Розділ ставить питання вирішення проблем, які можуть виникнути під час реалізації проекту, а також проводить пошуки схожих рішень, які вже були створені, їх недоліків та переваг.

Другий розділ кваліфікаційної роботи займається проектуванням програмно-апаратного засобу. Цей розділ обґрунтовує використання тих, чи інших апаратних ресурсів, комплектуючих, матеріалів та програмних засобів. У фінальній частині другого розділу складається список із вимог до обладнання та програмного забезпечення, а потім складається план, який включає в себе головні пункти по реалізації проекту.

Третій розділ кваліфікаційної роботи відповідає за безпосередню реалізацію програмно-апаратного засобу. У ньому детально зображується процес створення пристрою, програмування його комплектуючих, з'єднання їх поміж собою, а

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		59

також тестування їх роботи. Перш за все за допомогою програмного забезпечення Arduino IDE було здійснено процедуру прошивки Wi-Fi модуля ESP8266. Після цього в будинку було створено локальну мережу, з метою під'єднання до неї усіх необхідних приладів.

Наступним етапом була збірка усіх комплектуючих в один пристрій, до мікросхеми Arduino Uno за допомогою кабелів приєднали релейний модуль Tolako, а також дисплей LCD1602. Після збірки роботу пристрою було успішно протестовано.

Практична цінність роботи виходить із реалізованого програмно-апаратного засобу автоматичного контролю будинку на базі Arduino, а також у мережевій системі, яка об'єднує прилади в будинку.

					КВРКІ 180228.18.02.03 ПЗ	Арк
						60
Зм.	Арк.	№докум.	Підпис	Дата		

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Javed A. Building Arduino Projects for the Internet of Things. New York City: Apress, 2016. 233 p.
2. Hubitat. URL: <https://hubitat.com/> (дата звернення: 14.05.2022).
3. Internet of Things. URL: https://en.wikipedia.org/wiki/Internet_of_things (дата звернення: 14.05.2022).
4. Home automation. URL: https://www.indianelevator.com/product-category/domains/?gclid=Cj0KCQjwqPGUBhDwARIsANNwjV6-0lCV68ZHwt367SaZ6ocJuf9nBTPIe1XLxqZYWuwte1ExsvMHHKUaAsuqEALw_wcB (дата звернення: 16.05.2022).
5. Thomasnet. URL: <https://www.thomasnet.com/articles/automation-electronics/best-home-automation-systems/> (дата звернення: 17.05.2022).
6. Shovic J. Raspberry Pi IoT Projects: Prototyping Experiments for Makers. New York City: Apress, 2016. 202 p.
7. Lea P. Internet of Things for Architects: Architecting IoT solutions by implementing sensors, communication infrastructure, edge computing, analytics, and security. Birmingham: Packt Publishing, 2018. 389p.
8. Smart Home Automation. URL: <https://www.savant.com/> (дата звернення: 16.05.2022).
9. Green Light. URL: <https://www.lamps2udirect.com/standard-incandescent-light-bulbs/15-watt-es-e27mm-green-decorative-golfball-light-bulb/144203> (дата звернення: 17.05.2022).
10. Solanki V., Garcia Diaz V., Davim J. P. Handbook of IoT and Big Data. Florida: CRC Press, 2019. 312 p.
11. Hassan Q. F., Khan A. R., Madani S. A. Internet of Things: Challenges, Advances, and Applications. Florida: CRC Press, 2017. 318 p.
12. Home Assistant. URL: <https://www.home-assistant.io/> (дата звернення: 16.05.2022).

					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		61

13. Best Home Automation Systems. URL: <https://www.whichhomeautomation.com/blog/9-best-home-automation-systems-for-a-smart-home/> (дата звернення: 14.05.2022).
14. Geddes M. Arduino Project Handbook, Volume 2: 25 Simple Electronics Projects for Beginners. San Francisco: No Starch Press, 2017. 190 p.
15. What is Home Automation. URL: <https://www.safewise.com/home-security-faq/how-does-home-automation-work/> (дата звернення: 15.05.2022).
16. Cables. URL: <https://cables.g1/> (дата звернення: 15.05.2022).
17. Electrical Cables. URL: <https://www.screwfix.com/c/electrical-lighting/cable/cat8960001> (дата звернення: 15.05.2022).
18. Knight S. Arduino for Beginners: Step-By-Step Guide to Arduino (Arduino Hardware & Software). Amazon Digital Services LLC – KDP Print US, 2018. 121 p.
19. Bhattacharjee S. Practical Industrial Internet of Things Security. Birmingham, United Kingdom: Packt Publishing Ltd 2018. 234 p.
20. Jadot F. Smart Control Harnesses the Industrial Internet of Things. *Machine and Process Management* 2016. June 30. 7 p.
21. Матвієнко М. П. Комп'ютерна схемотехніка: навчальний посібник. Київ: ТОВ "Центр навчальної літератури", 2012. 143 с.
22. Матвієнко М. П. Архітектура комп'ютерів: навчальний посібник. Київ: ТОВ "Центр навчальної літератури", 2012. 184 с.
23. Arduino Uno R3 Technical Details. URL: <http://static6.arrow.com/arrowpdfconversion/97b354c3cd11851042ebd1fd252fbd6b6def5a73/adafruit2877arduino-uno.pdf> (дата звернення: 16.05.2022).
24. Arduino programming notebook. URL: <http://engineering.nyu.edu/gk12/amps-cbri/pdf/ArduinoBooks/Arduino%20Programming%20Notebook.pdf> (дата звернення: 17.05.2022).
25. MiMi Smart Керування освітленням. URL: https://www.smarthouse.ua/ua/upravlenie_osvecsheniem.html (дата звернення: 16.05.2022).
26. Караван Smart House дистанційне керування будинком. URL: <http://hifidom.com.ua/statti/smarthome/distcontrol> (дата звернення: 18.05.2022).

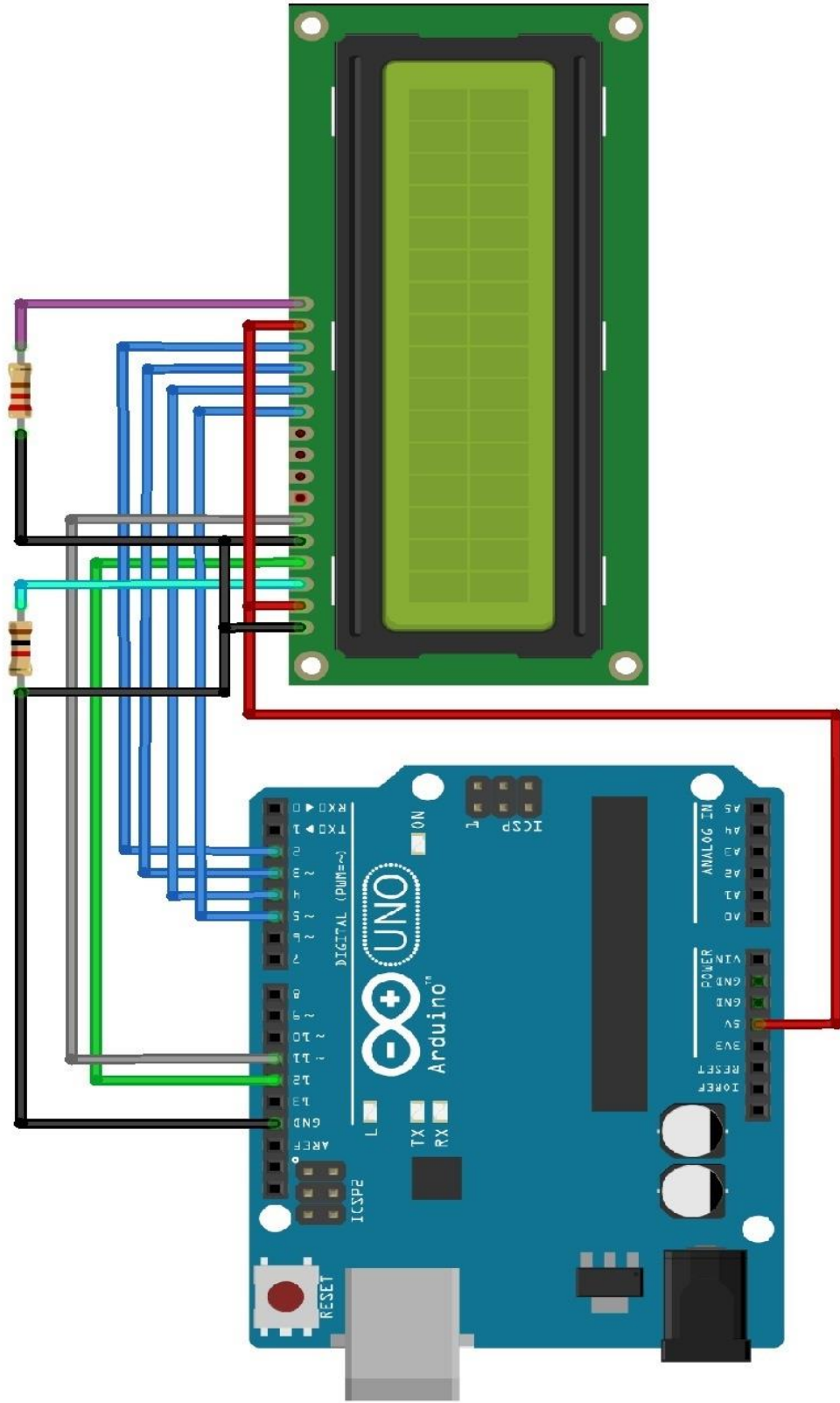
					КВРКІ 180228.18.02.03 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		62

27. Розумний будинок – з чого він складається та чи потрібен вам. URL: <https://nachasi.com/tech/2018/06/25/smart-house-faq/> (дата звернення: 20.05.2022).
28. TinyCircuits. URL: <https://tinycircuits.com/> (дата звернення: 20.05.2022).
29. Smart Home With Arduino. URL: <https://www.instructables.com/Smart-home-with-arduino/> (дата звернення: 20.05.2022).
30. Arduino. URL: <https://www.arduino.cc> (дата звернення: 20.05.2022).
31. Побудова системи "Розумний будинок" на базі Arduino. URL: <https://card-file.onaft.edu.ua/handle/123456789/10929> (дата звернення: 22.05.2022).
32. Програмування мікроконтролерів. З чого почати. URL : <http://eprints.zu.edu.ua/25722/1/Кузьменко%20С.%20В..pdf> (дата звернення: 23.05.2022).
33. BroadLink. URL: <https://broadlink.com.ua> (дата звернення: 23.05.2022).
34. Link Labs, Thread vs ZigBee. URL: <https://www.link-labs.com/blog/thread-vs-zigbee-for-iot-engineers> (дата звернення: 26.05.2022).
35. Wi-Fi module ESP8266. URL: <https://arduino.ua/prod980-wifi-modul-esp8266> (дата звернення: 26.05.2022).
36. Philips. URL: <https://www.philips.ua/> (дата звернення: 26.05.2022).
37. Aqara. URL: <https://www.aqara.com/en/home.html> (дата звернення: 26.05.2022).
38. Amazon Echo. URL: <https://www.amazon.com/smart-home-devices/b?ie=UTF8&node=9818047011> (дата звернення: 26.05.2022).
39. Tolako. URL: <https://www.tolako.com/> (дата звернення: 26.05.2022).
40. Arduino Uno. URL: <https://doc.arduino.ua/ru/hardware/Uno> (дата звернення: 26.05.2022).
41. LCD Display. URL: <https://arduino.ua/prod169-lcd-1602-simvolnii-displei-16x2-sinii> (дата звернення: 26.05.2022).

Додаток А

(Обов'язковий)

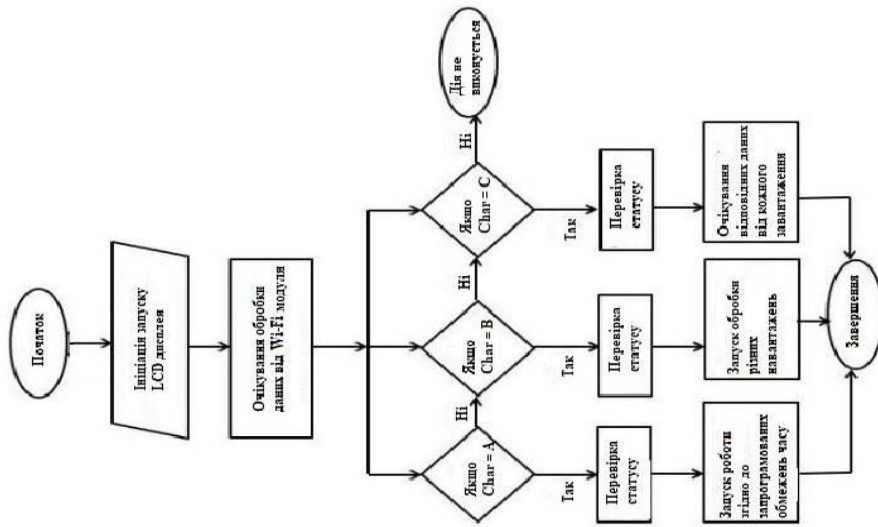
Копія графічної частини



КвРКІ.180228.18.02.03 Е8

КвРКІ.180228.18.02.03 Е8		Листопад	Місяць	Машинний
Схеми апаратних з'єднань		Листопад	Дата	
Зад. Діаг.	Мі. Вислук.	Листопад	Дата	
Розробник	Голов. Д.Р.			
Перевір.	Модельніця	В.Ю.В.		
М. Коляда	Маслова	С.В.		
Зима	Калач	Ю.П.		
				ХНУ, КІ-18-2

Блок-схема роботи програми



КВРКІ.180228.18.02.03.Е8		Листопад	Місяць	
Блок-схема програми		Листопад	Місяць	
Зм. Дан.	№ докум.	Листопад	Місяць	
Розробив	Розв. В.Р.			
Перевір.	Комп'ютерна КС/В			
	Архив	2	Архівує?	
П. Кошик	Методич. С.Б.			
Замк.	Катег. Ю.П.			
		ХНУ, КІ-18-2		

Интерфейс сторінки керування пристроями в будинку



КвРКІ.180228.18.02.03 Е8		Ліпень	Місяць
Интерфейс програмно-апаратного засобу		Листопад	
Знак	Адрес	№ поверху	Підпис
Розробник	Угода В.Р.	Метельницький Ю.Б.	
Проєктер	Метельницький Ю.Б.		
№ комп'ютера	Масштабний С.В.		
Ванна	Класифікація		
	ХНУ, КІ-18-2		

Додаток Б

Копія програмного коду

Лістинг коду для прошивки Wi-Fi модулю ESP8266

```
#define WIFI_SERIAL Serial1
void setup()
{
  Serial.begin(9600);
  while (!Serial)
  {
  }
  Serial.print("Serial init OK\r\n");
  WIFI_SERIAL.begin(115200);
}
void loop()
{
  if (WIFI_SERIAL.available())
  {
    Serial.write(WIFI_SERIAL.read());
  }
  if (Serial.available())
  {
    WIFI_SERIAL.write(Serial.read());
  }
}
```

Лістинг коду для програмування локального серверу

```
#include <LocalServer.h>
const char* ssid = "LocalServerVadym";
const char* password = "Badroom123";
LocalServer server(70);
```

```

String header;
String Device1State = "off";
String Device2State = "off";
String Device3State = "off";
String Device4State = "off";
const int Device1 = 17;
const int Device2 = 16;
const int Device3 = 15;
const int Device4 = 14;
unsigned long currentTime = millis();
unsigned long previousTime = 0;
const long timeoutTime = 3000;
void setup()
{
Serial.begin(115200);
pinMode(Device1, OUTPUT);
pinMode(Device2, OUTPUT);
pinMode(Device3, OUTPUT);
pinMode(Device4, OUTPUT);
digitalWrite(Device1, LOW);
digitalWrite(Device2, LOW);
digitalWrite(Device3, LOW);
digitalWrite(Device4, LOW);
Serial.print("Connecting to ");
Serial.println(ssid);
LocalServer.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
delay(700);
Serial.print(".");
}
Serial.println("");

```

```
Serial.println("WiFi connected.");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());
server.begin();
}
void loop()
{
```

Лістинг коду для програмування LCD дисплею

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup()
{
  lcd.begin(16, 2);
  Serial.begin(9600);
}
void loop()
{
  if (Serial.available())
  {
    delay(100);
    lcd.clear();
    while (Serial.available() > 0)
    {
    }
  }
  {
    lcd.setCursor(0, 2);
    lcd.print(millis()/1000);
  }
```

Ім'я користувача:
Кафедра кібербезпеки

ID перевірки:
1011481290

Дата перевірки:
07.06.2022 09:17:43 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
07.06.2022 09:31:35 EEST

ID користувача:
100008300

Назва документа: Плагіат Кваліфікаційна робота Гула В.Р. КІ-18-2

Кількість сторінок: 57 Кількість слів: 8278 Кількість символів: 65475 Розмір файлу: 8.59 MB ID файлу: 1011358702

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

8.84%
Схожість

Найбільша схожість: 3.62% з джерелом з Бібліотеки (ID файлу: 1008383715)

2.54% Джерела з Інтернету

111

Сторінка 59

7.97% Джерела з Бібліотеки

89

Сторінка 59

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Підозріле форматування

13
сторінок

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 1.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 11%**

ID: 104645 Название: Програмно-апаратний засіб автоматичного контролю будинку за допомогою мережевих систем на базі Arduino Добавлено в БД: 2022-06-07 Авторы: Гула Вадим Русланович Руководители: Хмельницький Ю.В. Консультанты: Опоненты:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	46243	455	1037 (2%)	18 (4%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

5. Негативні сторони проекту: У роботі при оцінці параметрів реалізації та забезпечення роботи програмно-апаратного засобу автоматичного контролю будинку недостатньо приділено уваги технічній стороні втілення сучасних підходів

6. Оцінка графічного оформлення та пояснювальної записки роботи: Графічне оформлення виконане відповідно до теми кваліфікаційної роботи із дотриманням усіх стандартів. У загальному графічне оформлення виконане на достатньому технічному рівні. Пояснювальна записка відповідає нормам для її оформлення та вимогам

7. Відгук про роботу в цілому: В загальному кваліфікаційна робота заслуговує позитивної оцінки. Весь матеріал кваліфікаційної роботи структурований, чіткий та послідовний. Усі розділи роботи послідовні та логічні, що дозволяє чітко розуміти викладений матеріал в рамках тематики кваліфікаційної роботи. У пояснювальній записці багато рисунків, схем та наглядних пояснень. Графічний матеріал дозволяє наочно побачити доцільність та ефективність рішень, які були прийняті за основу для досягнення поставленої задачі проектування.

8. Інші зауваження

9. Оцінка дипломної роботи Розглянувши позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що робота заслуговує оцінки « добре ».

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи)

Огнєвич Олександр Вікторович
доцент кафедр. ТМТТ

« 10 » 06 2022 .

(підпис)

Завідувачу кафедри кібербезпеки
к.т.н., доц. Кльоцу Ю.П.

Гули Вадима Руслановича
ПІБ здобувача вищої освіти

студента ФІТ, 4 курсу, групи КІ-18-2

ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений (а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

13.06.2022

дата

В. Гули

підпис

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КІБЕРБЕЗПЕКИ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Програмно-апаратний засіб автоматичного контролю будинку за допомогою мережевих систем на базі Arduino

Автор: В.Р. Гула

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: Ю.В. Хмельницький, к.т.н, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) запозичення розміщені в розділах аналізу існуючих аналогів та технологій, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 3) окремі виявлені збіги є загальноживаними шаблонами, що використовуються при оформленні текстової документації, а саме шаблони рамок
- 4) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту, використання абревіатур.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 8,84%, з урахуванням наведених обґрунтувань, відповідає характеру дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КБ

Ю.В. Хмельницький

С.М. Лисенко

Ю.П. Кльоц