

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**  
ФАКУЛЬТЕТ АЕРОНАВІГАЦІЇ, ЕЛЕКТРОНІКИ ТА ТЕЛЕКОМУНІКАЦІЙ  
КАФЕДРА ЕЛЕКТРОНІКИ, РОБОТОТЕХНІКИ І ТЕХНОЛОГІЙ  
МОНІТОРИНГУ ТА ІНТЕРНЕТУ РЕЧЕЙ

ДОПУСТИТИ ДО ЗАХИСТУ  
Завідувач кафедри

\_\_\_\_\_ Шутко В.М.

«\_\_» \_\_\_\_\_ 2023 р.

**КВАЛІФІКАЦІЙНА РОБОТА**  
ЗДОБУВАЧА ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА  
ЗІ СПЕЦІАЛЬНОСТІ 171 «ЕЛЕКТРОНІКА»  
ОПП«ЕЛЕКТРОННІ СИСТЕМИ»

**Тема:** «Управління зовнішнім освітленням на мікроконтролері Atmega»

Виконавець:

студентка групи ЕС-407Б      Василюк Анна Олександрівна

Керівник:

к.т.н. доцент      \_\_\_\_\_ Морозова І.В.

Нормоконтролер:      \_\_\_\_\_ Сініцин Р.Б.

**НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ**  
**ФАКУЛЬТЕТ АЕРОНАВІГАЦІЇ, ЕЛЕКТРОНІКИ ТА ТЕЛЕКОМУНІКАЦІЙ**  
**КАФЕДРА ЕЛЕКТРОНІКИ, РОБОТОТЕХНІКИ І ТЕХНОЛОГІЙ**  
**МОНІТОРИНГУ ТА ІНТЕРНЕТУ РЕЧЕЙ**  
**171 «ЕЛЕКТРОНІКА», ОПП «ЕЛЕКТРОННІ СИСТЕМИ»**

ДОПУСТИТИ ДО ЗАХИСТУ  
Завідувач кафедри  
\_\_\_\_\_ Шутко В.М.  
«\_\_» \_\_\_\_\_ 2022 р.

**ЗАВДАННЯ**

**на виконання дипломної роботи**

Василюк Анна Олександрівна

(П.І.Б., випускника)

1. Тема дипломної роботи: «Управління зовнішнім освітленням на мікроконтролері Atmega» затверджена наказом ректора від «\_\_» 2023 р. № 398/ст.
2. Термін виконання роботи: з «\_\_» \_\_\_\_\_ 2023р. по «\_\_» \_\_\_\_\_ 2023р.
3. Вихідні дані до роботи: розробити систему управління зовнішнім освітленням.
4. Зміст пояснювальної записки: 1 Теоретичні основи розробки; 2 Вибір технічних засобів; 3 Апаратно-програмна реалізація.
5. Перелік обов'язкового графічного (ілюстрованого) матеріалу: таблиці, рисунки, зображення сенсорів, модулів, пристрою, код програми.
6. Календарний план-графік

№ п/п	Завдання	Термін виконання етапів	Відмітка про виконання
1.	Затвердження теми бакалаврської роботи	13.03.2023	01.03.2023

2.	Вивчення літератури	01.04.2023	31.03.2023
3.	Теоретичні основи розробки	20.04.2023	19.04.2023
4.	Вибір технічних засобів	01.05.2023	30.05.2023
5.	Апаратно-програмна реалізація	14.05.2023	13.05.2023
6.	Оформлення та усунення недоліків дипломної роботи	19.05.2023	18.05.2023

Дата видачі завдання: \_\_\_\_\_

Керівник дипломної роботи \_\_\_\_\_

(підпис керівника) (П.І.Б.)

Морозова І.В.

Завдання прийняв до виконання \_\_\_\_\_

(підпис випусника)

Василюк А.О.

(П.І.Б.)

## РЕФЕРАТ

Пояснювальна записка до дипломної роботи “Управління зовнішнім освітленням на мікроконтролері Atmega”:

72 сторінки, 25 рисунків, 10 таблиць, 22 джерел посилань.

**Актуальність теми** полягає у забезпеченні простоти управління зовнішнім освітленням. Інтелектуальне управління освітленням, оптимізуючи енергозбереження в бізнес-будівлі або забезпечуючи комфорт користувача в приватному будинку, дає змогу розумно передавати такі вимоги між освітлювальними приладами та навколишнім середовищем без участі людини.

**Мета роботи** - створення ефективної електронної системи управління зовнішнім освітленням, яка дає змогу переглядати показання датчиків, які відповідають за адаптивне освітлення, керувати параметрами системи та гнучко налаштовувати автоматизацію.

**Об'єктом розробки** є процеси, які відбуваються в електронних системах управління зовнішнім освітленням.

**Метод дослідження** – програмування на ПЕОМ з використанням програмного середовища Arduino IDE для аналізу обробки датчиків системи, докладні описи використаних методів і алгоритмів.

Результати проведеної роботи – розроблено інтелектуальну систему управління зовнішнім освітленням.

Матеріали даної дипломної роботи можуть бути використані для проведення наукових досліджень, у навчальному процесі, а також з можливістю використання в практичній діяльності процесів управління зовнішнім освітленням.

Ключові слова: МІКРОКОНТРОЛЕР, ДАТЧИКИ ОСВІТЛЕННЯ, ARDUINO NANO, АВТОМАТИЗАЦІЯ.

## ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

- IoT – (англ. Internet of Things) Інтернет речей;
- RFID – Radio Frequency Identification;
- GPS – Global Positioning System;
- RF – Radio Frequency (радіочастоти);
- AI – Artificial Intelligence;
- ПДУ – Пульт дистанційного керування;
- I2C – Inter-Integrated Circuit (схема внутрішнього зв'язку);
- SPI – Serial Peripheral Interface (послідовний периферійний інтерфейс);
- PCB – Printed Circuit Board (друкована плата);
- UART – Universal Asynchronous Receiver/transmitter (універсальний асинхронний приймач/передавач);
- LED – Light-emitting diode (світлодіод);
- IDE – Integrated Development Environment (середовище розробки);
- COM – Communication Port (послідовний порт);
- RDM – Remote Device Management (віддалене керування пристроями);
- NEC – протокол інфрачервоного управління.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ .....	5
ВСТУП .....	8
РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ ТА ОСНОВНІ КОНЦЕПЦІЇ СИСТЕМ УПРАВЛІННЯ ЗОВНІШНІМ ОСВІТЛЕННЯМ .....	10
1.1 Поняття та теоретичні аспекти зовнішнього освітлення .....	10
1.2 Огляд існуючих систем управління освітленням .....	12
1.2.1 Sunco Lighting Bulbs .....	12
1.2.2. Yeelight Smart .....	14
1.2.3 Gledepto Smart .....	16
1.3 Процесорний блок .....	18
1.3.1 Вибір мікропроцесора .....	18
1.3.2 Платформа Arduino .....	19
1.3.3 Апаратна частина платформи Arduino .....	19
1.3.4 Версії платформи Arduino .....	21
1.3.5 Огляд мікроконтролера Arduino Nano .....	23
1.3.6 Середовище програмування – Arduino IDE .....	29
1.4 Проблеми реалізації системи управління зовнішнім освітленням .....	30
1.5 Постановка задачі .....	33
РОЗДІЛ 2. ПРОЕКТУВАННЯ СИСТЕМИ УПРАВЛІННЯ ЗОВНІШНІМ ОСВІТЛЕННЯМ .....	34
2.1 Сучасні протоколи безпроводного зв'язку для побудови системи управління зовнішнім освітленням та їх порівняльний аналіз .....	34
2.1.1 Технологія безпроводної мережі Wi-Fi .....	34
2.1.2 Технологія бездротової мережі Z-Wave .....	37
2.1.3 Технологія бездротової мережі ZigBee .....	40

2.1.4	Технологія бездротової мережі Bluetooth .....	43
2.2	Розробка структурної схеми системи .....	46
2.3	Вибір апаратного забезпечення .....	48
2.3.1	Блок реле .....	48
2.3.2	Bluetooth модуль HC-06 .....	50
2.3.3	Датчик освітленості GY-302 .....	52
2.3.4	Блок живлення .....	54
<b>РОЗДІЛ 3. АПАРАТНО-ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ</b>		
<b>УПРАВЛІННЯ ЗОВНІШНІМ ОСВІТЛЕННЯМ .....</b>		
3.1	Програмний код та його оптимізація .....	55
3.2	Інструкція по користуванню приладом .....	59
3.3	Розробка друкованої плати для управління зовнішнім освітленням .....	61
<b>ВИСНОВКИ .....</b>		<b>63</b>
<b>СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ .....</b>		<b>64</b>
<b>ДОДАТОК А .....</b>		<b>1</b>
<b>ДОДАТОК Б .....</b>		<b>1</b>

## ВСТУП

Розумне освітлення, також відоме як адаптивне освітлення – це додавання інтелекту або логіки до світильників, щоб вони могли автоматично спілкуватися та взаємодіяти з оточенням, мешканцями та іншими пристроями.

Розумне освітлення забезпечує гнучкість інфраструктури освітлення та автоматизацію для керування широким спектром операцій освітлення, від базового ввімкнення/вимкнення до складного адаптивного освітлення.

Сучасним розвитком технології є додавання до джерел світла електричних компонентів керування та підтримки. Електронні блоки зазвичай доповнюють сервісні функції продукту, підвищують автоматизацію, спрощують використання, покращують дизайн, зменшують споживання електроенергії та іноді допомагають точніше виконувати основні завдання.

Коли необхідно підсвітити предмети інтер'єру чи екстер'єру, зазвичай виправдано використання світлодіодів RGB.

Деякі декоративні елементи, які підсвічуються в сучасних помешканнях, часто служать як прикрасою, так і освітленням. Зараз доступно кілька систем світлодіодних кольорових освітлювачів. Деякі з них використовують безліч кольорових світлодіодних стрічок з багатьма каналами. В інших ситуаціях може бути використана група світлодіодних матриць зі складним управлінням. Однак використання таких технологій може бути дорогим і складним.

Основними недоліками комерційних систем автоматизації управління освітленням є складність монтажу та завищена вартість таких систем, а також неможливість додавання власних функцій контролю і змін параметрів.

Інтелектуальне управління освітленням, оптимізуючи енергозбереження в бізнес-будівлі або забезпечуючи комфорт користувача в приватному будинку, дає змогу розумно передавати такі вимоги між



освітлювальними приладами та навколишнім середовищем без участі людини.

Система управління зовнішнім освітленням з використанням сучасних елементів може виконувати наступні завдання:

1. Контроль та управління всіма пристроями системи з одного місця (з однієї кімнати або навіть з персонального гаджета).
2. Окреме регулювання освітленості в кожній зоні освітлення (або для кожного освітлювального приладу).
3. Автоматичне вмикання та вимикання груп освітлення в різних місцях.
4. Самостійна побудова сценаріїв роботи системи та освітлювальних приладів для різних ситуацій (наприклад, прихід гостей, режим відпустки, щоденний режим використання, режим незаконного доступу тощо).
5. Економія електроенергії за рахунок самостійного регулювання системою інтенсивності освітлення та режиму роботи лампи.

Інтелектуальні системи освітлення доступні для роботи в проектах будь-якого розміру та взаємодії з використанням різноманітних протоколів через різні канали зв'язку, починаючи від одного світильника зі структурою з'єднання «точка-точка» до світильників, згрупованих у mesh-мережу.

Як наслідок, освітлення споживає значну частину загальної потужності, яка використовується в багатьох будівлях, які виконують різноманітні функції, наприклад промислові, житлові та адміністративні. А оскільки економія електроенергії в системах освітлення істотно впливає на споживання електроенергії, питання її раціонального способу експлуатації є критичним.

## **РОЗДІЛ 1. ТЕОРЕТИЧНІ АСПЕКТИ ТА ОСНОВНІ КОНЦЕПЦІЇ СИСТЕМ УПРАВЛІННЯ ЗОВНІШНІМ ОСВІТЛЕННЯМ**

### **1.1 Поняття та теоретичні аспекти зовнішнього освітлення**

Розумне освітлення стає загальноприйнятим терміном у міру розвитку технологій. Технологія розумного освітлення вважалася б перемиканням або затемненням освітлення за визначеним користувачем розкладом або при виявленні руху десять років тому. Те, що зараз називають розумним освітленням, – це інтелектуальна інфраструктура, що складається з освітлювальних вузлів, датчиків та інших інтелектуальних пристроїв, які об'єднані між собою через Інтернет-платформу.

У своїй основній формі розумний світильник – це програмований освітлювальний прилад, який може з'єднуватись з пультом дистанційного керування через дротову або бездротову мережу. Далі, розглянемо основні концепції розумного зовнішнього освітлення:

1. Локалізоване інтелектуальне освітлення – фокусується на додаванні двостороннього зв'язку між контролером освітлення та освітлювальним приладом без урахування масштабованості чи сумісності.

Лише підключене освітлювальне обладнання можна налаштовувати, контролювати стан, та програмувати дистанційно. Локалізоване інтелектуальне освітлення часто є запатентованим рішенням, яке працює на власній екосистемі спеціалізованого обладнання та програмного забезпечення. Локалізована система освітлення є мережею інтелектуальних світильників, які використовують Remote Device Management (RDM)

(механізм двостороннього зв'язку на основі DMX512) або власні протоколи Ethernet.

Локалізований інтелектуальний освітлювальний пристрій – це будь-яка бездротова інтелектуальна система освітлення, яка забезпечує зв'язок «точка-точка» лише на короткій відстані. Розумні світильники, які використовують оригінальний протокол Bluetooth є прикладами таких продуктів.

2. Мережеве інтелектуальне освітлення: робить крок далі, стандартизуючи протоколи зв'язку для підвищення масштабованості та сумісності систем інтелектуального освітлення, одночасно полегшуючи агонію, введення в експлуатацію та з'єднання.

Програми для смартфонів, бездротові сенсорні мережі та рішення WPAN (бездротова персональна мережа) призвели до широкого поширення розумних світильників, які забезпечують просте встановлення та експлуатацію. Системи розумного освітлення, які спілкуються одна з одною за допомогою стандартизованих бездротових протоколів, таких як ZigBee, Z-Wave, Wi-Fi, Bluetooth, Mesh і Thread, надзвичайно масштабовані та усувають перешкоди завдяки сумісності. Мережа освітлення цього типу легко розширюється від кількох вузлів освітлення до інтегрованих інсталяцій з десятків тисяч розумних пристроїв, включаючи не лише розумні лампочки, але й датчики розташування та датчики навколишнього середовища.

3. Інтелектуальне освітлення на базі IoT: додає нові функції до PAN (персональні мережі) і VAN (ділові мережі) (мережі на тілі). Інтернет речей виводить розумне освітлення на новий рівень.

Він містить обчислювальну потужність, вбудоване програмне забезпечення, підключення та архітектуру на основі IP для зв'язку та взаємодії з усіма розумними пристроями та Інтернетом. IoT реалізує весь потенціал підключеного освітлення, збираючи дані з пов'язаних датчиків і перетворюючи їх на свіжі ідеї та практичну інформацію за допомогою надзвичайно безпечної, масштабованої хмарної платформи. Інтелектуальне освітлення з підтримкою Інтернету речей перевершує інтелектуальне

освітлення на основі мережі з точки зору стабільності, масштабованості, надійності, сумісності та діапазону для з'єднань «точка-точка». Освітлювальне обладнання в додатках IoT або підключається безпосередньо до Інтернету, або за допомогою локальних або глобальних мереж.

## **1.2 Огляд існуючих систем управління освітленням**

Найкращі смарт-лампи яскраві, прості в установці та сумісні з усіма основними системами розумного будинку. Наприклад, світло вмикається, коли двері відчиняються або датчик руху зчитує, коли хтось йде і світло вмикається та вимикається, щоб створити враження, що хтось є вдома. Крім того, оскільки вони повністю світлодіодні, вони споживають набагато менше енергії, ніж звичайні лампи розжарювання та галогенні лампи. Вони достатньо надійні, щоб приєднатися до мережі дому. Коли їх пов'язано, користувач може керувати ними за допомогою смартфона або голосового помічника, наприклад Siri, Alexa або Google Assistant. Нижче проведемо порівняльний аналіз існуючих смарт-ламп, щоб обрати найкращий варіант.

### **1.2.1 Sunco Lighting Bulbs**

Компанія Sunco, що базується в США, дотримується найвищих стандартів для всіх своїх товарів, перевіряючи їх на безпеку та продуктивність перед тим, як надати 5-річну гарантію. За допомогою простого програмного забезпечення від Sunco користувач може легко керувати кількома розумними лампочками одночасно або однією окремо, використовуючи лише смартфон або планшет, без потреби в концентраторі. Є можливість запрограмувати таймер для включення і виключення світла. Лампочки Sunco (рис. 1.1.) можна використовувати для покращення технології розумного дому. За допомогою голосу можна керувати яскравістю

та кольором із палітри додатку, а також є можливість змінити колірну температуру від 2700К до 5000К. Ці смарт-лампи сумісні з Google Assistant і Amazon Alexa (табл. 1.1.).



Рисунок 1.1 – Світлодіодні лампи Sunco Lighting

Таблиця 1.1 – Технічна характеристика лампи Sunco Lightning [10]

Характеристика	Опис
Бренд	Sunco Lightning
Тип	LED
Колір	RGB
Потужність	8 Вт

Діапазон колірної температури	2700K-5000K
Сертифікати	UL, RoHS
Цикли перемикання	30000
Напруга	120В

### 1.2.2. Yeelight Smart

Розумною лампочкою WiFi Yeelight (рис. 1.2) можна дистанційно керувати голосом з будь-якого місця, і вона сумісна з Alexa Echo (через Smart Life), Echo point і Google Home Assistant. Концентратор не потрібен. Користувач може вмикати та вимикати ці світильники за допомогою безкоштовного додатку на смартфоні чи планшеті.

Колір можна змінювати від 1700 К для ніжно-білого до 6500 К для білого денного світла (табл. 1.2), а також можна змінювати яскравість. Завдяки можливості вибору з 16 мільйонів різних відтінків і кольорів білого, ця лампочка може допомогти вам зробити простір більш креативним.

Ця розумна лампочка споживає на 60% менше енергії, ніж звичайна лампочка, яка використовує найновішу технологію освітлення LED.



Рисунок 1.2 – Світлодіодна лампа Yeelight Smart Bulb

Таблиця 1.2 – Технічна характеристика лампи Xiaomi Yeelight Smart

Характеристика	Опис
Бренд	Xiaomi Yeelight Smart
Тип	LED
Колір	RGB
Потужність	8,5 Вт
Діапазон колірної температури	1700К-6500К
Сертифікати	UL, RoHS
Цикли перемикання	30000
Напруга	100-240В
Яскравість (світловий потік)	800 люмен

### 1.2.3 Gledopto Smart

Gledopto – це розумна система освітлення, яка водночас є інтелектуальною та простою. Незалежно від уподобань користувачів, Gledopto дає змогу регулювати освітлення, інтегроване в розумну домашню мережу за допомогою програми для смартфона або голосового керування – це легко, зручно та перспективно.

Для голосового керування ввімкненням/вимкненням світлодіодного освітлення, зміною кольору та затемненням ця лампа (рис. 1.3) має бути сумісна зі звичайними шлюзами ZigBee (табл. 1.3). Для цієї світлодіодної лампи потрібне постійне джерело напруги. Прожектори Zigbee MR16 мають чудове розсіювання тепла та схвалені CE та RoHS, вони не дзижчать та в них відсутнє просочення випромінюючого шару.

Цей світильник, створений за передовими технологіями та ультрасучасним дизайном, можна використовувати як вбудований освітлювальний елемент або в новітніх світильниках. Він перевершить усі очікування з точки зору атмосфери та відображення кольорів, освітлення дому, офісу чи будь-якого іншого місця, яке вимагає більше світла. Світлодіоди SMD 5050 + 3030 і зовнішнє покриття алюмінієвої основи, які є недорогими та енергоефективними, дозволять заощадити до 80% витрат на електроенергію при зміні 16 мільйонів кольорів, що дозволить розумній лампі майже повністю окупити себе протягом терміну служби (50000 годин).





Рисунок 1.3 – Світлодіодна лампа Gledopto Mr16 RGB CCT ZigBee

Таблиця 1.3 – Технічна характеристика лампи Gledopto Smart

Характеристика	Опис
Бренд	Gledopto Smart
Тип	LED
Колір	RGB
Потужність	12 Вт

Продовження таблиці 1.3

Діапазон колірної температури	2700-6500K
Сертифікати	UL, RoHS
Цикли перемикання	30000
Напруга	100-240В
Індекс передачі кольорів	>80

### 1.3 Процесорний блок

#### 1.3.1 Вибір мікропроцесора

Розглянемо кілька версій МК, які можуть бути використані в цьому проєкті. Компактність і економічність є важливими факторами для МК, так само як і простота освоєння мікроконтролера і IDE.

1. Arduino Nano – це компактна плата з процесором 16 МГц і 32 Кб флеш-пам'яті. Завдяки зручності та простоті мови програмування, а також відкритій архітектурі та коду платформа користується надзвичайною популярністю у всьому світі. Це перевірений МК, який можна придбати за 1-2 долари для використання при розробці.

2. RASPBERRY PI ZERO насправді є маленькою версією Raspberry Pi Model A Plus, розміром лише з кредитну картку. Цей мікрокомп'ютер здатний запускати повний дистрибутив Linux або будь-яку іншу сумісну операційну систему. При цьому він має прийнятну ціну, коливається від 5 до 7 доларів. Він оснащений процесором 1 ГГц і 512 МБ оперативної пам'яті. Також є два порти microUSB, але немає Ethernet, Bluetooth або WiFi.

3. NodeMCU – це мікроконтролер із великою кількістю функціональних можливостей, особливо враховуючи його низьку ціну в 3 долари. На відміну від Arduino, який використовує 8-розрядний процесор ATMEGA на 16 МГц, цей містить чіпсет ESP8266 з 32-розрядним процесором Tensilica Xtensa LX106 80 МГц, Wi-Fi, 4 МБ вбудованої пам'яті та 20 КБ оперативної пам'яті.

4. STM32 від STMicroelectronics – це недорогий 32-розрядний мікроконтролер. Він використовує як середовище розробки Keli, так і програматор ST-Link. Пристрій працює на базі 32-бітного чіпа ARM Cortex

МЗ з тактовою частотою 24 МГц і 8 КБ оперативної пам'яті. STM32 має низьке енергоспоживання, цифрову обробку сигналу та інші переваги.

На основі вищенаведеного аналізу існуючих варіантів мікроконтролерів вирішено використовувати Arduino Nano, оскільки він є найменш дорогим, добре зарекомендував себе та має зручне середовище розробки для проєктування систем різноманітної складності.

### **1.3.2 Платформа Arduino**

Мікроконтролер Arduino був розроблений у 2005 році в Інституті проєктування Івреа в Мілані, Італія, як навчальний інструмент. Платформу створив Ернандо Барраган у 2004 році з відкритим кодом для художників і дизайнерів. Інструмент був розроблений таким чином, щоб його могли використовувати творчі люди з дуже невеликим досвідом роботи з комп'ютером. Ернандо віддав перевагу утиліті платформи як інструменту створення прототипів. Зовнішні датчики можуть використовуватися пристроями на базі Arduino для збору даних про навколишнє середовище, і реагуючи на дані датчика, вони можуть керувати різноманітними виконавчими пристроями (приводами). Спеціальна мова програмування, яка є скороченою формою C++/C, використовується для програмування мікроконтролера на платі. Для програмування використовується роз'єм USB без використання спеціальних програматорів. Середовище Arduino IDE служить середовищем для прямого програмування. Крім того, можна програмувати платформу іншими мовами, такими як C++, Python і Java.

### **1.3.3 Апаратна частина платформи Arduino**

Зараз доступно кілька різних платформ Arduino. Найпоширенішими є Arduino Nano Uno (через широкі можливості) і Arduino (за компактність і ціну). Зазвичай, на платі розміщені всі необхідні елементи:

1. мікроконтролер;
2. пам'ять;
3. порти підключення;
4. специфічні для кожної версії модулі;
5. індикацію датчиків та інше.

Плату Arduino Nano/Uno можна живити як 12,5, так і 3,3 вольтами. Оскільки можна мати велику кількість необхідних модулів і датчиків, кожна версія Arduino розроблена для конкретного застосування.

Плата Arduino складається з мікроконтролера Atmel AVR, а також елементів обв'язки для програмування та інтеграції з іншими пристроями. На багатьох платах наявний лінійний стабілізатор напруги +5В або +3,3В. Тактування здійснюється на частоті 16 або 8 МГц кварцовим резонатором.

На концептуальному рівні усі плати програмуються через RS-232 (послідовне з'єднання), але реалізація даного способу різниться від версії до версії. Новіші плати програмуються через USB, що можливо завдяки мікросхемі конвертера USB-to-Serial FTDI FT232R [9].

У версії платформи Arduino Uno як конвертер використовується контролер Atmega8 у SMD-корпусі. Дане рішення дозволяє програмувати конвертер таким чином, щоб платформа відразу розпізнавалася як миша, джойстик чи інший пристрій за вибором розробника зі всіма необхідними додатковими сигналами керування. У деяких варіантах, таких як Arduino Mini або неофіційній Boarduino, для програмування потрібно підключити до контролера окрему плату USB-to-Serial або кабель [10].

Плати Arduino дозволяють використовувати значну кількість виводів мікроконтролера як вхідні/вихідні контакти у зовнішніх схемах. Наприклад, у платі Decimila доступно 14 цифрових входів/виходів, 6 із яких можуть генерувати ШІМ сигнал, і 6 аналогових входів. Ці сигнали доступні на платі

через контактні майданчики або штирові роз'єми. Також існує багато різних зовнішніх плат розширення, які називаються «shields» («щити»), які приєднуються до плати Arduino через штирові роз'єми [13].

### 1.3.4 Версії платформи Arduino

Нижче розглянемо основні версії, які можуть бути виділені з лінійки Arduino:

1. Найновіша модель, Nano 33 BLE, має розміри 45x18 мм. Платформа ідеально підходить для гаджетів, які можна носити, оскільки містить 9-осьові інерційні датчики та можливість Bluetooth.

2. Arduino Uno – це широко використовувана плата мікроконтролерів з відкритим кодом на базі мікроконтролера ATmega328P.

У його склад входить все необхідне для зручної роботи з мікроконтролером: 14 цифрових входів/виходів (з них 6 можуть використовуватися в якості ШІМ-виходів), 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм для програмування всередині схеми (ICSP) і кнопка скидання. Для початку роботи з пристроєм досить просто подати живлення від AC/DC-адаптера або батарейки, або підключити його до комп'ютера за допомогою USB-кабелю [20].

3. Leonardo відрізняється від усіх попередніх плат тим, що ATmega32u4 має вбудований USB-зв'язок, що усуває потребу у додатковому процесорі. Це дозволяє Leonardo відображатися на підключеному комп'ютері як миша та клавіатура на додаток до віртуального (CDC) послідовного/COM-порту.

4. Мікропроцесор Atmel SAM3X8E ARM Cortex-M3 є основою пристрою Due. Має 54 цифрових входи (12 для використання сигналу ШІМ), 12 аналогових входів, сумісність USB для OTG, 4 апаратних приймача UART (які передають послідовні дані), 2 ЦАП, 2 роз'єми TWI та з'єднання SPI.

5. Перша платформа з вбудованим модулем Wi-Fi є Yun, яка поєднує в собі зручність МК і функціональність Linux. Arduino YN поєднує систему Wi-Fi з добре відомою Arduino Leonardo (на базі мікропроцесора ATmega32U4). Arduino Yún – це мікроконтролерна плата на основі ATmega32u4 (таблиця даних) і Atheros AR9331. Процесор Atheros підтримує дистрибутив Linux на основі OpenWrt під назвою OpenWrt-Yun.

Плата має вбудовану підтримку Ethernet і WiFi, порт USB-A, слот для карт micro-SD, 20 цифрових входів/виходів (з яких 7 можна використовувати як ШІМ-виходи і 12 як аналогові входи), кристал 16 МГц. осцилятор, з'єднання micro USB, роз'єм ICSP і 3 кнопки скидання [21].

6. Arduino TRE – Arduino TRE приблизно в 100 разів потужніший за Arduino Leonardo або Uno і базується на 1 ГГц процесорі Sitara AM335x. З використанням потужних програм Linux така продуктивність надає розробнику безліч можливостей для розробки різноманітних систем.

Мікропроцесор Sitara, який використовується в Arduino, дозволяє виконувати ресурсомістке настільне програмне забезпечення Linux, алгоритми та високошвидкісні інтерфейси зв'язку.

7. Micro – це плата мікроконтролера на основі ATmega32U4, розроблена спільно з Adafruit. Він має 20 цифрових входів/виходів (з яких 7 можна використовувати як ШІМ-виходи та 12 як аналогові входи), кристалічний генератор 16 МГц, з'єднання micro USB, роз'єм ICSP і кнопку скидання.

8. Arduino Nano V3.0 – невелика самодостатня плата, сумісна з макетними платами, яка побудована на мікроконтролері ATmega328. Вона в основному збігається за функціональністю з Arduino Duemilanove/Uno, але має інший форм-фактор. Arduino Nano не вистачає тільки роз'єму живлення і замість стандартного, використовує Mini-B USB кабель [22].

9. Основна плата LilyPad Arduino 328 – це мікроконтролер, запрограмований на Arduino, розроблений для легкої інтеграції в проекти електронного текстилю та носимих пристроїв.

10. Arduino Pro Mini — це плата мікроконтролера на основі ATmega328P. Він має 14 цифрових входів/виходів (6 з яких можна використовувати як ШІМ-виходи), 6 аналогових входів, вбудований резонатор, кнопку скидання та отвори для кріплення штифтів. Шестиконтактний роз'єм можна під'єднати до кабелю FTDI або комутаційної плати Sparkfun для забезпечення живлення через USB і зв'язку з платою.

Є кілька інших варіантів платформи Arduino, які відрізняються за можливостями та ціною. Наразі існує більше варіацій Nano, Mega та Uno.

### **1.3.5 Огляд мікроконтролера Arduino Nano**

Arduino Nano — це невелика, повна і зручна для макетної плати плата, заснована на ATmega328P, випущеному в 2008 році.

Вона пропонує ті самі підключення та характеристики плати Arduino Uno в меншому форм-факторі. Arduino Nano оснащено 30 роз'ємами вводу/виводу у конфігурації, подібній до DIP-30, яку можна програмувати за допомогою інтегрованого середовища розробки (IDE) Arduino Software, яке є спільним для всіх плат Arduino і працює як онлайн, так і офлайн. Плата може живитися від кабелю mini-USB типу B або від акумулятора 9 В. Зовнішній вигляд даної платформи представлено на рис. 1.4.

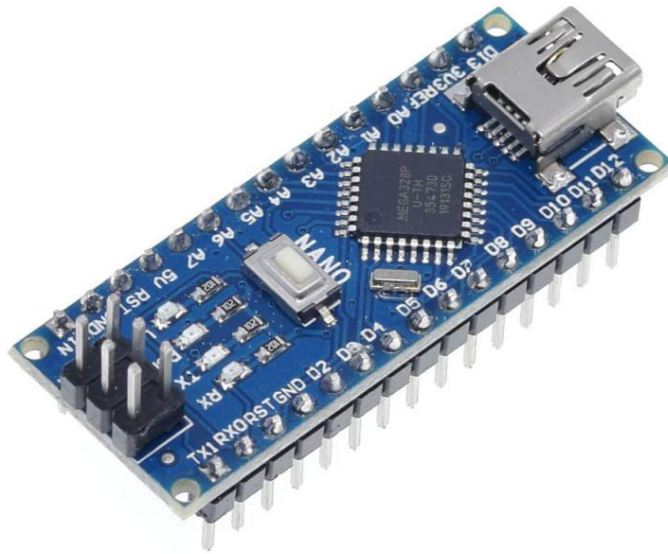


Рисунок 1.4 – Мікроконтролер Arduino Nano V3.0 [16]

Плата має щонайменше чотирнадцять контактів цифрових входів і виходів і до шести контактів аналогових входів, які використовуються для підключення мікроконтролера до відповідних електричних компонентів. Додаткові параметри мікроконтролера знаходяться у таблиці 1.3.

Таблиця 1.4 – Характеристики ArduinoNano V3.0

Мікроконтролер	ATmega328P
Тип корпусу	TQFP-32
Робоча напруга	5В
Вхідна напруга (рекомендована)	7-12В
Цифрових входів / виходів	14 (з яких 6 можуть бути використані як ШІМ)



Аналогових входів	8
Сила струму на входах / виходах	40мА
Сила струму для 3.3В виходу	50мА
Пам'ять	32 кБ, з яких 2кб використовується на бутлоадер
SRAM	2 кБ
EEPROM	1 кБ
Частота	16 МГц
Габарити	18 × 45 мм

Далі, розглянемо детальніше способи живлення плати. Живлення плати може здійснюватися двома способами:

1. Через mini-USB або micro-USB при підключенні до комп'ютера.
2. Через зовнішнє джерело живлення, що має напругу 6-20 В з низьким рівнем пульсацій [23].

Входи та виходи плати Arduino Nano також мають обмеження по напрузі та струму. Усі аналогові та цифрові з'єднання працюють від 0 до 5 вольт. Напруга буде обмежена захисними діодами при забезпеченні потужності вище заданих значень. Щоб запобігти вимкненню контролера в цій ситуації, сигнал повинен бути підключений через резистор.

Максимальний вхідний струм не повинен перевищувати 40 мА, а сумарний струм контактів не повинен перевищувати 200 мА.

Плата має 4 світлодіоди, які відображають стан сигналу. Мітки на них: TX, RX, PWR і L. На перших двох сигнал TX або RX активний, коли

світиться світлодіод, що вказує на низький рівень сигналу. Коли джерело живлення підключено, світлодіод PWR вмикається на 5 В.

Світлодіод загального призначення, який світиться, коли подається високий сигнал, є останнім світлодіодом. Приклад схеми живлення представлений на рисунку 1.5.

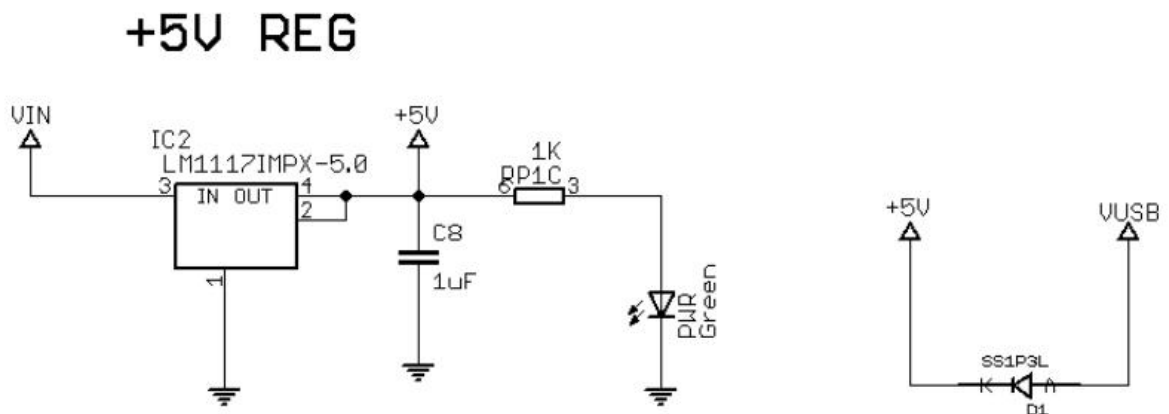


Рисунок 1.5 – Схема живлення Arduino Nano V3.0 [18]

Зараз створено кілька варіантів Arduino Nano. Версії 2.X і 3.0 відрізняються лише чіпом, на якому вони працюють. Версія 3.0 працює на пристрої ATmega328, але версія 2.X використовує чіп ATmega168, який має менше пам'яті та нижчу тактову частоту.

На рисунку 1.6 представлена детальна розпіновка плати Arduino Nano.

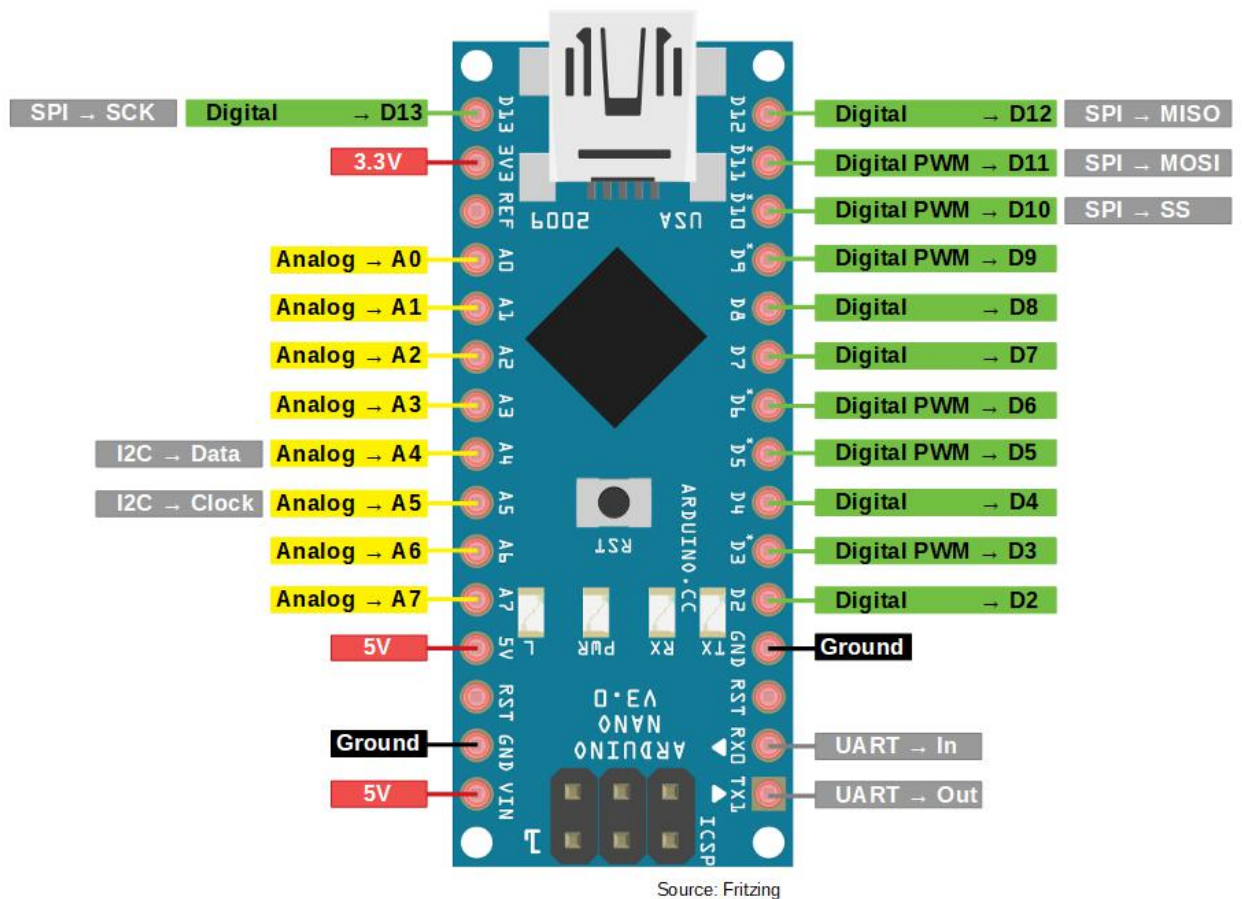


Рисунок 1.6 – Деталізована розпіновка плати Arduino Nano [20]

Нижче представлені основні піни живлення плати Arduino Nano:

1. VIN: Вхідний пін для підключення зовнішнього джерела живлення з напругою в діапазоні від 7 до 12 вольт;
2. 5V: Вихідний пін від регулятора напруги на платі з виходом 5 вольт і максимальним струмом 800 мА.
3. 3,3V: вихідний пін стабілізатора мікросхеми FT232R, який має максимальний струм 50 мА та вихідну напругу 3,3 В.

Нижче представлені порти виведення та введення плати Arduino Nano:

1. Цифрові входи / виходи: Піни 0-13. Один має логічний рівень 5 В, а нуль – 0 В. Максимальний вихідний струм – 40 мА. Контакти підключені до підтягуючих резисторів, які можуть бути включені програмним забезпеченням, навіть якщо вони за замовчуванням вимкнені;

2. ШІМ: Піни 3, 5, 6, 9, 10 і 11. Дозволяє виводити сигнал ШІМ аналогових значень. Розрядність ШІМ фіксована на рівні 8 біт і не змінюється.

3. АЦП: Піни А0-А7. Дозволяє цифрово виражати аналогову напругу. Розрядність АЦП фіксована на рівні 10 біт і не змінюється. Діапазон вхідної напруги від 0 до 5 В. Підвищення напруги призведе до руйнування МК.

4. UART: Піни 0 (RX) і 1 (TX). Через послідовний інтерфейс вони використовуються для підключення плати Arduino до комп'ютера або інших пристроїв. Відповідні Ground - 0 (RX) і 1 (TX) USB-UART на конвертері FT232R підключено. Для роботи потрібна бібліотека Serial.

5. SPI: Піни 11 (MOSI), 12 (MISO), 13 (SCK) і 10 (SS). Для зв'язку з периферійними пристроями через інтерфейс SPI.

6. Піни А4 (SDA) і А5 (SCL). Для використання інтерфейсу I2C для зв'язку з пристроями. Для роботи використовується бібліотека Wire.

Нижче представлені основні переваги Arduino Nano V3.0:

1. Невелика ціна у порівнянні з іншими платами;
2. Доступність специфікації і схеми приладу. Arduino.cc надає вільний доступ до відкритого коду і схеми пристрою.

3. Функціональність. У даному середовищі мова програмування може розширюватися за допомогою різноманітних C++ бібліотек.

Нижче представлені основні недоліки Arduino Nano V3.0:

1. Низький об'єм флеш-пам'яті для використання на великих проектах.
2. Низька частота процесора;

Arduino Nano – це універсальний мікроконтролер, який може використовувати як новачок у сфері електроніки, так і досвідчений інженер для розробки систем різноманітної складності.

### 1.3.6 Середовище програмування – Arduino IDE

Для платформи Arduino створено середовище розробки для написання скетчів (коду); вона відома як Arduino IDE. IDE сумісна з Windows, Mac OS і Linux. Інтегроване середовище розробки Arduino (IDE) – це кросплатформна програма (для Microsoft Windows, macOS і Linux), написана мовою програмування Java. Він походить від IDE для мов Processing і Wiring. Він містить редактор коду з такими функціями, як вирізання та вставлення тексту, пошук і заміна тексту, автоматичний відступ, зіставлення фігурних дужок і підсвічування синтаксису, а також забезпечує прості механізми компіляції та завантаження програм на плату Arduino одним клацанням миші. Він також містить область повідомлень, текстову консоль, панель інструментів із кнопками для загальних функцій та ієрархію робочих меню.

Скетч – це програма, написана за допомогою Arduino IDE. Скетчі зберігаються на комп'ютері розробки як текстові файли з розширенням .ino.

Мінімальна програма Arduino C/C++ складається лише з двох функцій:

1. `setup()`: Ця функція викликається один раз, коли запускається ескіз після ввімкнення чи скидання. Він використовується для ініціалізації змінних, режимів введення та виводу та інших бібліотек, необхідних у скетчі.

2. `loop()`: після завершення роботи функції `setup()` функція `loop()` повторно виконується в основній програмі. Він керує платою, доки плату не буде вимкнено або скинуто. Вона аналогічна функції `while(1)`.

На рисунку 1.7 представлений зовнішній вигляд Arduino IDE.

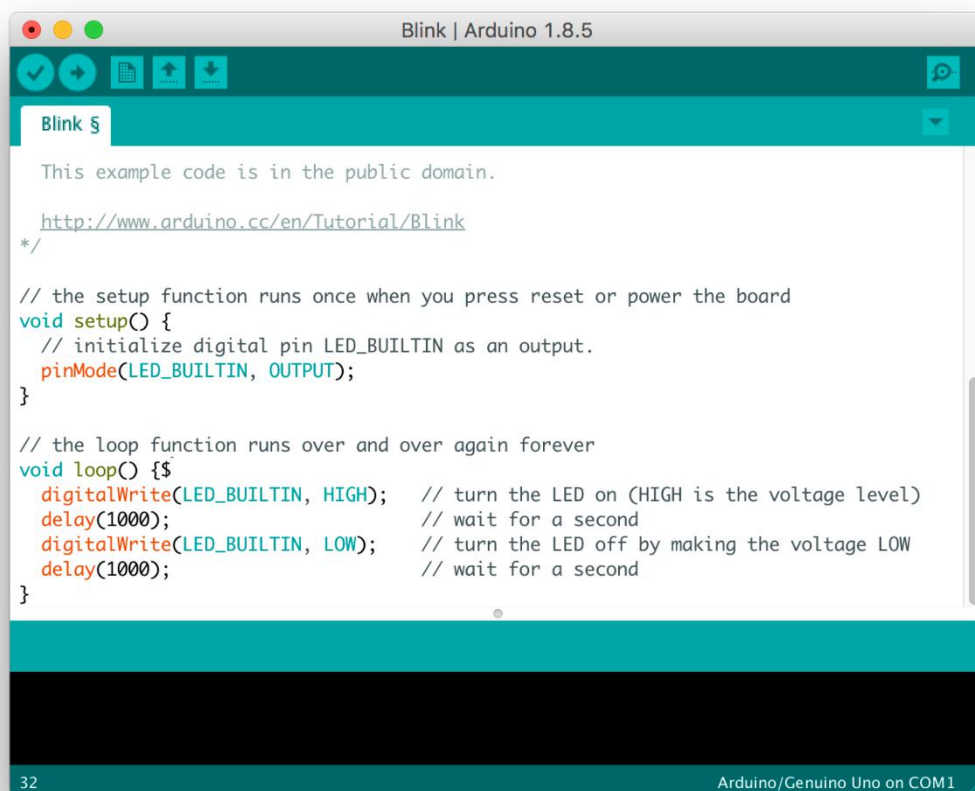


Рисунок 1.7 – Зовнішній вигляд Arduino IDE

14 вересня 2022 року Arduino IDE 2.0 було офіційно випущено як стабільну. Раніше, 18 жовтня 2019 року, початкова альфа-версія була випущена як Arduino Pro IDE. 1 березня 2021 року було випущено попередню бета-версію, яка перейменована на IDE 2.0. Система все ще використовує Arduino CLI (інтерфейс командного рядка), але вдосконалення включають більш професійне середовище розробки, підтримку автодоповнення та Git.

#### **1.4 Проблеми реалізації системи управління зовнішнім освітленням**

Окрім підвищення безпеки та комфорту, зовнішнє освітлення також споживає значну кількість енергії. Люди почуватимуться небезпечно, якщо

світло буде повністю вимкнене. Використання світлодіодного освітлення з регулюванням яскравості, яке зменшить споживання енергії без шкоди для безпеки, є відповідним рішенням цієї проблеми.

Інвестиції, управління та енергозбереження – це три змінні, які впливають на розгортання відповідної системи керування.

Найкращим вибором для того, щоб дотримуватися існуючих норм і почати скорочувати споживання енергії прямо зараз без необхідності значних коригувань або витрат на інфраструктуру, є використання автономних систем управління. Вони працюють окремо від решти системи та контролюють або регулюють освітлення на основі певних датчиків або програм. Системи керування мережею створені для досягнення більш ідеального рішення, оскільки вони забезпечують централізований контроль колекції точок освітлення. Для передачі керуючого сигналу або власного джерела живлення ламп така система потребує додаткової проводки. Однак ці системи відносно прості в установці та дозволяють швидко контролювати та керувати колекціями світлових точок, якщо необхідна інфраструктура вже є [17].

Системи керування точковою мережею не лише керують освітленням, але й дистанційно діагностують кожну точку освітлення. Ці системи забезпечують підвищену масштабованість, зберігаючи максимальну гнучкість. На рисунку 1.8 зображено приклад інтелектуальної системи керування зовнішнім освітленням.

Складність реалізації такої системи в даній кваліфікаційній роботі пов'язана з неможливістю роботи із зовнішнім освітленням через відсутність прав та необхідних технічних елементів. Таким чином, буде розглянуто та реалізовано спрощену версію для внутрішнього освітлення з найбільш суттєвими перевагами перед звичайним освітленням, такими як економічність, можливість регулювання кольору та яскравості і, мабуть, найголовніше, колірна температура підсвічування, яка зараз недооцінена.

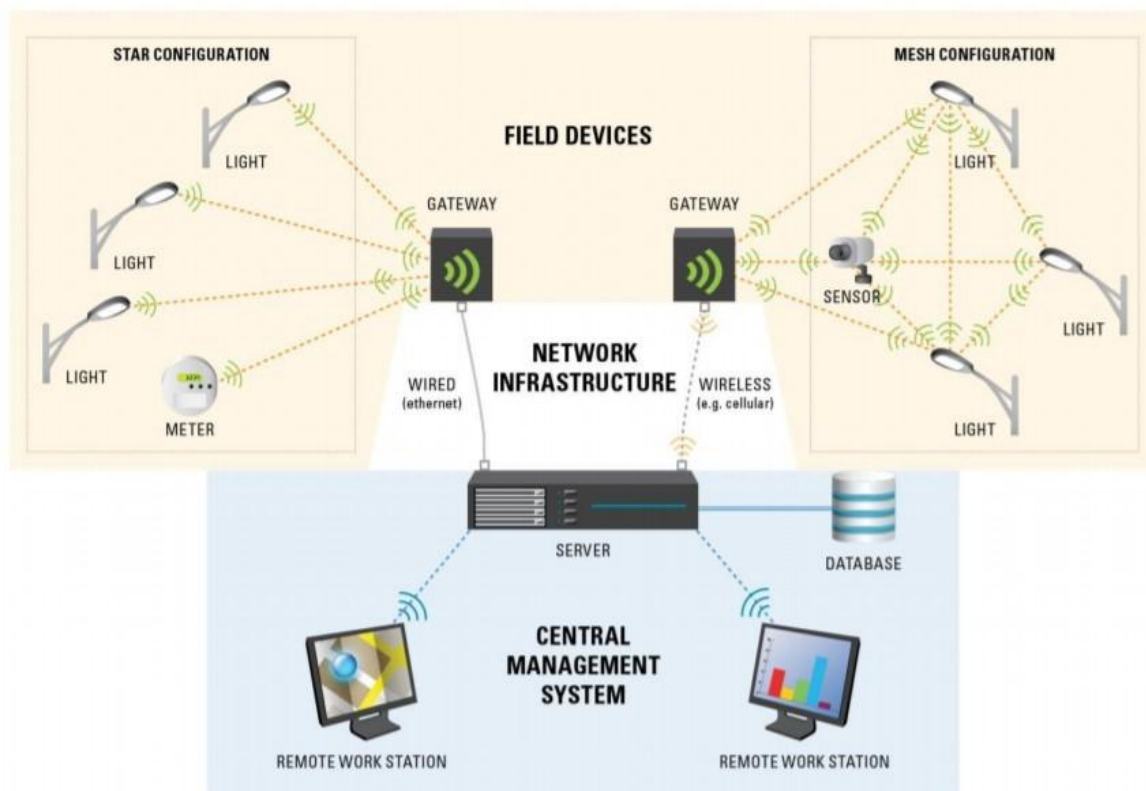


Рисунок 1.8 – Узагальнена схема системи управління зовнішнім освітленням за допомогою шлюзів

Підключені або інтегровані системи керування освітленням найпростіші в установці та можуть поєднуватися з різноманітними мережевими системами керування. Завдяки цьому можна не тільки дистанційно керувати освітленням, але й дистанційно діагностувати кожен точку світла. Це можливо за допомогою датчиків, вузлів керування, шлюзів, комунікацій тощо. Ці рішення економлять найбільше енергії, водночас забезпечують найбільшу гнучкість.

Можливо інтегрувати кілька систем управління, які адаптуються до кожного сектору міста, керуючи з єдиної платформи, з якої місто може трансформуватися. Ця платформа, яка працює вертикально та автономно, може поєднуватися з горизонтальною платформою міста та іншими службами, становлячи частину підключеного міста. Ця концепція багатообіцяюча через використання та кореляцію отриманих даних, що



дозволяє застосовувати сучасні аналітичні методології та обробку великих даних для сприяння прийняттю стратегічних рішень.

Нова інтелектуальна цифрова технологія має вирішальне значення для відповіді на запити міста та громади. Покращена інфраструктура освітлення сприяє з'єднанню, а також знижує експлуатаційні витрати на технічне обслуговування в екологічно відповідальний спосіб.

### **1.5 Постановка задачі**

В інтелектуальній системі керування зовнішнім освітленням здійснюється натисканням однієї кнопки на смартфоні або пульті дистанційного керування. Оскільки в цій системі не потрібні перемикачі, користувач може використовувати смартфон або пульт дистанційного керування, щоб змінювати параметри ламп, люстр, вуличних ліхтарів.

Для досягнення мети кваліфікаційної роботи необхідно виконати наступне:

1. Проаналізувати теоретичні аспекти та основні концепції створення електронної системи для управління зовнішнім освітленням.
2. Розглянути існуючі системи управління освітленням та провести порівняльний аналіз характеристик.
3. Розробити систему управління зовнішнім освітленням з використанням плати Arduino Nano на мікроконтролері ATmega328P.

## **РОЗДІЛ 2. ПРОЕКТУВАННЯ СИСТЕМИ УПРАВЛІННЯ ЗОВНІШНІМ ОСВІТЛЕННЯМ**

### **2.1 Сучасні протоколи безпроводного зв'язку для побудови системи управління зовнішнім освітленням та їх порівняльний аналіз**

Оскільки більшість конкуруючих бездротових протоколів мають спільні риси, може здатися, що вони надають порівнювані можливості та функції. При ближчому розгляді між ними можна виявити значні відмінності.

Оскільки кожна з технологій була розроблена для досягнення певної мети, вони функціонують по-різному [4]. Багато деталей того, як працює кожен протокол, стануть очевидними, якщо поглянути на них через призму стека (моделі) протоколів OSI [2, 7]. Стек протоколів OSI розглядає процес взаємодії елементів мережі як семирівневу структуру.

Далі, розглянемо основні сучасні протоколи безпроводного зв'язку для побудови системи управління зовнішнім освітленням та проведемо порівняльний аналіз їх характеристик.

#### **2.1.1 Технологія безпроводної мережі Wi-Fi**

Wi-Fi – це найпоширеніша технологія бездротової мережі сьогодні, до якої підключені мільйони людей. Цей стандарт забезпечує доставку майже половини інформації в Інтернеті [4].

Wi-Fi широко використовується в приватних будинках, квартирах, на підприємствах і в громадських місцях. Це найбільш часто використовуваний спосіб підключення смартфонів і комп'ютерів до мережі. Технічно можливо з'єднати численні пристрої для координації обміну інформацією за допомогою Wi-Fi, але ця потужна бездротова технологія вважається найгіршим рішенням для поняття IoT (інтернету речей) з усіх можливих варіантів.

Технологія Wi-Fi базується на стандарті бездротової мережі зв'язку IEEE 802.11x. Вони описують початкові два рівні моделі OSI, фізичний і канальний (рисунок 2.1). Коли мережа Wi-Fi реалізована в топології «Зірка», усі її вузли пов'язані з центральним пристроєм – маршрутизатором або комутатором – який виконує основні завдання адміністрування та інтерфейсу. Структуру такої мережі можна змінювати без шкоди для цілісності структури або передачі даних. Однак це рішення не забезпечує достатньої надійності, оскільки існує єдина точка відмови (комутатор), і якщо вона виходить з ладу, функціонування мережі стає неможливим [3, 4].

Технологія Wi-Fi використовує стандартизовані протоколи зв'язку, включаючи UDP і TCP для транспортного рівня (для моделі OSI) і IPv4 або IPv6 для мережевого рівня. У мережах Wi-Fi прикладний рівень, який забезпечує сумісність пристроїв, не визначений і реалізується на програмному рівні пристроями або програмами, які на них встановлені.

Переваги Wi-Fi як протоколу для створення мереж моніторингу включають той факт, що це надійне та широко використовуване рішення, яке вже давно використовується для розвитку локальних мереж. У переважній більшості випадків маршрутизатора нижчого цінового діапазону достатньо для покриття всієї квартири.

У великих будівлях можна розгорнути численні різні точки доступу або використовувати повторювачі сигналу для розширення зони покриття мережі. Крім того, основною перевагою Wi-Fi є доступність інфраструктури.

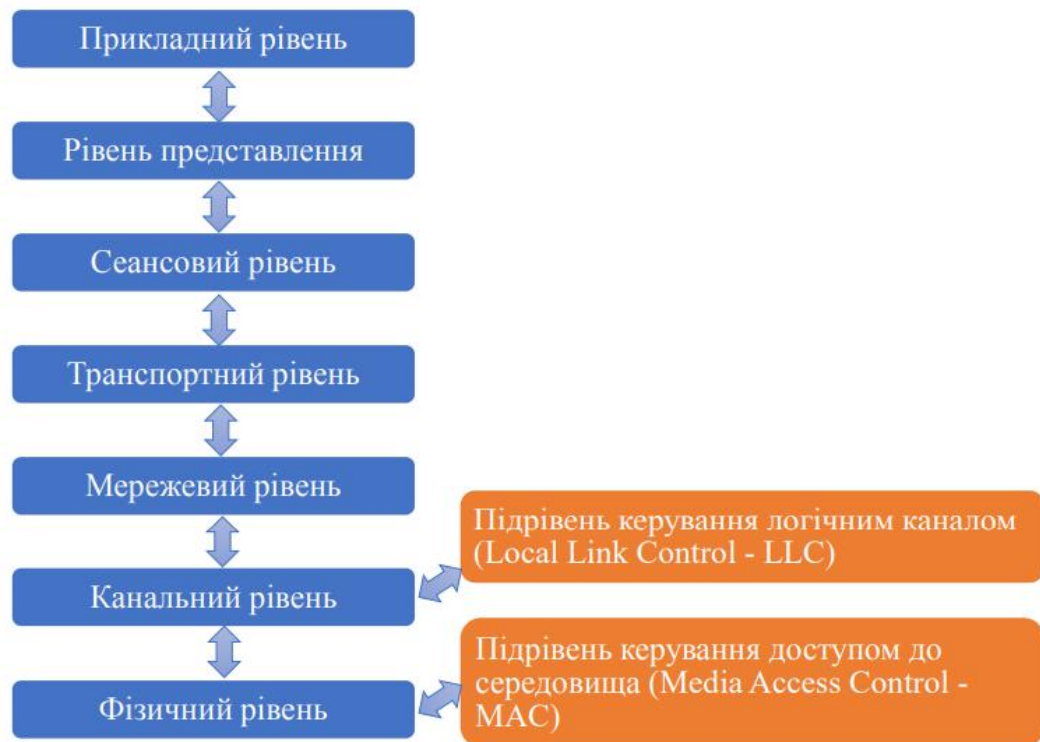


Рис 2.1 – Протокол Wi-Fi в моделі Open Systems Interconnection

Wi-Fi має багато недоліків при використанні в якості протоколу для розподілених систем автоматизації. Як стандарт для швидкого зв'язку, Wi-Fi споживає багато енергії. Якщо гаджет підключено до джерела живлення, енергоспоживання не є суттєвим фактором; але, коли важливо забезпечити автономність роботи від портативних джерел, таких як батареї або сонячні панелі, це стає проблемою. Розробити мережевий комунікаційний пристрій на основі Wi-Fi, який забезпечує як швидку реакцію, так і заряд акумулятора, який може забезпечувати постійне живлення протягом тривалих періодів часу, майже неможливо. Оскільки Wi-Fi настільки широко використовується, він не може добре працювати в автономних бездротових пристроях [4].

Значні обмеження також існують щодо архітектури бездротових мереж на основі Wi-Fi, оскільки, здається, існує єдина точка відмови, оскільки весь зв'язок (включаючи трафік) залежить від маршрутизатора, який діє як центральний вузол мережі. Збій маршрутизатора порушує роботу всієї

мережі автоматизації, оскільки не дозволяє окремим мережевим пристроям з'єднуватися один з одним. Найбільш вирішальним фактором при проектуванні систем автоматизації є питання сумісності пристроїв. Впровадити самоорганізовану мережу складно, оскільки стандарт Wi-Fi не визначає прикладний рівень мережевої архітектури OSI, що робить однорангову передачу даних неможливою.

### **2.1.2 Технологія бездротової мережі Z-Wave**

Маючи на своєму рахунку понад 100 мільйонів пристроїв у всьому світі, Z-Wave зараз є найпоширенішою технологією домашньої автоматизації [5]. Щоб забезпечити потенціал віддаленого керування датчиками та робочим обладнанням із найвищою ефективністю та надійністю в системах «розумного дому» та Інтернету речей, цей протокол бездротового зв'язку пропонує надзвичайно низьке енергоспоживання [8, 18].

Zensys створила компанію Z-Wave у 2003 році, а Sigma Designs придбала її в 2008 році. Після продажу напівпровідникового бізнесу Silicon Labs у 2017 році ця технологія вступила в період швидкого розвитку. Як провідний виробник модулів і компонентів Z-Wave, Silicon Labs ліцензує кожен модуль із підтримкою ZWave, забезпечуючи їх взаємодію один з одним [8]. Сьогодні цей протокол задовольняє найважливіші вимоги до систем автоматизації та є доступним і простим варіантом для розумних будинків та систем Інтернету речей. Як технологія Z-Wave перетворилася на потужний стандарт бездротового зв'язку для автоматизації та контролю.

Z-Wave охоплює кожен рівень семирівневого стеку протоколів моделі OSI, від фізичного до прикладного [1, 10, 19]. Ця функція може забезпечити неперевершений рівень сумісності датчиків і пристроїв керування для домашньої автоматизації, виробленої різними постачальниками. Z-Wave – це протокол, який фокусується на передачі коротких команд і повідомлень між

гаджетами, що призводить до мінімального навантаження на смугу радіочастот, де він працює, і знижує ймовірність втрати пакетів даних.

Мережева архітектура mesh, яку використовує протокол Z-Wave, є пористою (рисунок 2.2). Він розроблений таким чином, що дозволяє окремим елементам мережі, які служать ретрансляторами, перенаправляти повідомлення між вузлами, поки вони не будуть надіслані призначеним одержувачам [6]. За допомогою цього методу можна значно покращити радіус дії бездротової мережі, а також підвищити її надійність. Якщо один із компонентів мережі з будь-якої причини буде втрачено, мережа продовжуватиме функціонувати; однак повідомлення проходять іншим шляхом і проходять через інші вузли мережі [14].



Рисунок 2.2 – Використання мережі mesh Z-Wave для автоматизованої системи «розумний будинок»

Кожна логічна мережа Z-Wave може підтримувати до 232 пристроїв. Оскільки кілька мереж Z-Wave можуть працювати з частковим або повним перекриттям, не заважаючи одна одній, об'єднання багатьох мереж в одну екосистему є варіантом, якщо необхідно побудувати мережу з більшою

кількістю вузлів. Центральний контролер (базова станція) для мережі Z-Wave підключає нові пристрої до мережі та відключає ті, що не використовуються.

Він також виконує динамічну побудову карти маршрутизації в реальному часі, забезпечує безпечне з'єднання, реалізує створення сценаріїв роботи окремих компонентів системи та виконує інші завдання, пов'язані з організацією мережі. Цікавість Z-Wave полягала в тому, що він був розроблений як закритий протокол, на який поширюється багато патентів. Специфікація Z-Wave була частково доступна в 2012 році компанією Sigma Designs, а вся офіційна документація для цього протоколу була доступна для громадськості в 2016 році. Зокрема, були випущені класи команд і класи пристроїв. Security 2 – це опис вимог до шифрування мережі Z-Wave (S2). Алгоритми, що відповідають за стабільність роботи та маршрутизацію повідомлень, їх ретрансляцію та контроль прийому, детально описані на мережевому та транспортному рівнях, які на сьогодні ще закриті [5].

Перевага Z-Wave полягає в тому, що його пристрої є найбільш енергоефективними, надійними та безпечними з усіх варіантів домашньої автоматизації, доступних зараз на ринку. У цих мережах гарантовано буде використовуватися до 4 проміжних точок для передачі даних між різними пристроями. Враховуючи, що модулі Z-Wave забезпечують надійну передачу даних на відстані близько 40 метрів у прямій видимості, діапазон передачі сигналу в одній мережі достатній для реалізації масштабних проектів, таких як Інтернет речей і «розумний будинок». Ті самі механізми шифрування, що використовуються в системах онлайн-банкінгу, використовуються Z-Wave.

З 2017 року всі інтелектуальні пристрої Z-Wave тепер повинні бути сертифіковані за стандартом Security 2. Він також встановлює нові протоколи для додавання додаткових пристроїв, що використовують QR-коди, до розумної домашньої мережі, покращуючи вимоги до шифрування для потоку даних між вузлами [5, 19]. Недоліком Z-Wave є те, що через урядові обмеження для використання пристроїв малого радіусу дії в різних країнах виділені різні діапазони частот. Наприклад, це 868,42 МГц для Європи,

Китаю та інших азіатських країн. А оскільки технологія GSM використовує ці частоти в США і Мексиці, там Z-Wave працює на частоті 908,42 МГц.

### **2.1.3 Технологія бездротової мережі ZigBee**

На ринку комунікаційних мереж для систем автоматизації та моніторингу ZigBee є конкурентом Z-Wave [1, 7]. Подібно до Z-Wave, ZigBee є стандартом мережі з низьким енергоспоживанням і призначений для віддаленого моніторингу та керування розумним будинком та IoT.

Обидва стандарти використовують пористі сітчасті мережі та функціонують однаково. На перший погляд ці стандарти здаються еквівалентними з точки зору можливостей, але більш ретельний аналіз виявляє важливі відмінності між ZigBee і Z-Wave.

Тільки мережевий, транспортний і прикладний рівні моделі OSI визначаються набором протоколів ZigBee (рис. 2.8) [2, 21]. Він заснований на стандарті IEEE 802.15.4, який визначає, як побудувати найнижчі рівні бездротової мережі, орієнтовані на кінцевий пристрій. Стандарт IEEE 802.15.4 підтримується різними постачальниками мікросхем і використовується не лише ZigBee, але й багатьма іншими протоколами.

В якості робочого діапазону цей стандарт передбачає частоти 2,4 ГГц у всьому світі, 915 МГц в Америці та Австралії та 868 МГц у Європі [1].

Протокол зв'язку, який використовується Z-Wave та ZigBee, найбільше відрізняється: ZigBee використовує маршрутизацію повідомлень від джерела, а Z-Wave – від пункту призначення.

Тому для реалізації такої mesh-мережі на основі протоколу ZigBee потрібні три класи пристроїв: координатор (який встановлює та координує роботу мережі), маршрутизатор (який служить основним компонентом для трансляції та динамічної маршрутизації пакетів) і кінцеві пристрої (рис. 2.3).



Як наслідок, ZigBee та Z-мережі Wave не зовсім однакові з точки зору їх організації, але ZigBee все ще здатний самостійно відновлюватися після збоїв мережі та може перенаправляти пакети даних для забезпечення доставки [13].

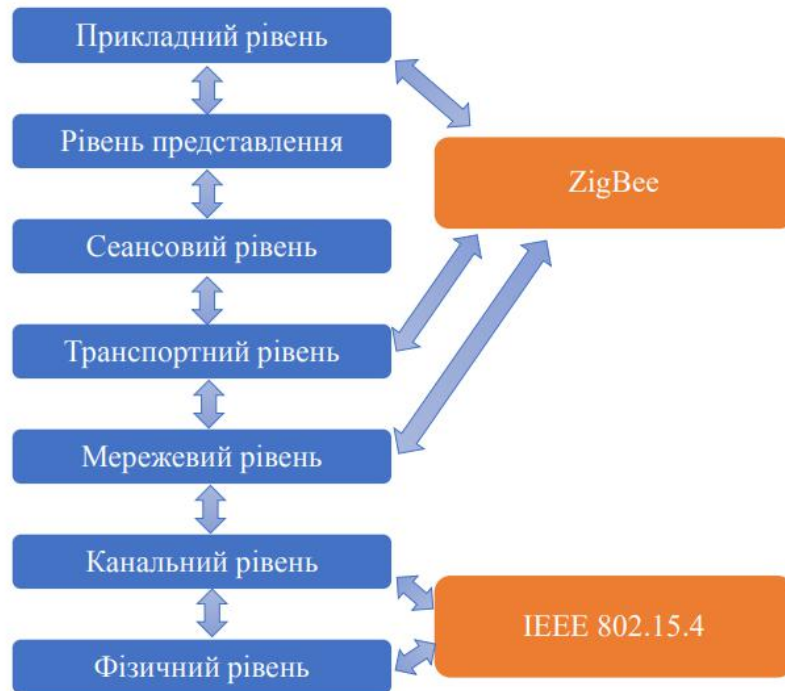


Рис 2.3 – Протокол Zigbee в моделі Open Systems Interconnection [15]

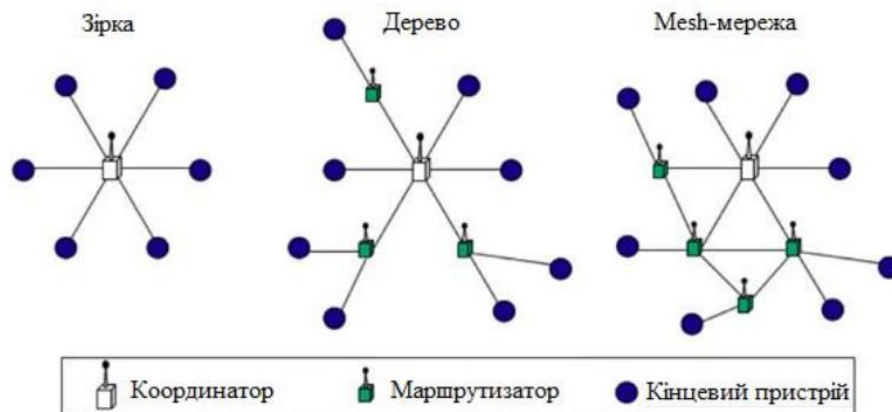


Рис 2.4 – Топологія протоколу ZigBee

Для виробників і розробників ZigBee – це відкрита технологія бездротового зв'язку, яка дуже зручна. Це дає більшу свободу вибору необхідного функціоналу та випуску додаткових пристроїв за меншими цінами (через відсутність обов'язкової сертифікації). Щоб покращити моніторинг, контроль та адміністрування використання послуг для своїх клієнтів, комунальні підприємства вмикають цей протокол у лічильниках електроенергії та води, тому ZigBee широко використовується в корпоративному секторі. Незважаючи на відносно невелику відстань передачі даних окремими модулями (10–20 м), ZigBee має високу масштабованість і може здійснювати покриття широкої зони покриття системи [16, 23]. Він також може взаємодіяти в одній мережі до 65 тисяч окремих компонентів.

Оскільки ZigBee є одноканальним рішенням, а частота 2,4 ГГц використовується спільно протоколами Wi-Fi і Bluetooth, воно не працює належним чином у сценаріях, коли є перешкоди від інших пристроїв поблизу мережі. Оскільки частотний спектр 2,4 ГГц з кожним роком використовується все більше і більше, не слід очікувати покращення ситуації. Крім того, той факт, що швидкість передачі даних для стандарту мережі IEEE 802.15.4 обмежена 250 Кбіт/с, не є ідеальним для ZigBee [11].

Щоб гарантувати, що пакети даних, які надсилаються між різними пристроями, належним чином захищені з точки зору безпеки, ZigBee використовує різноманітні методи.

Проблеми безпеки в мережах ZigBee не пов'язані з протоколом, а мають суб'єктивний характер, оскільки мережа використовує 128-бітний алгоритм AES для шифрування даних і безпечного підключення пристроїв.

Ще одним недоліком ZigBee є нездатність пристроїв різних виробників безпосередньо підключатися один до одного. Великі корпорації розробляють ексклюзивні рішення ZigBee, які можуть спілкуватися лише з іншими продуктами під їхнім брендом. Основною проблемою ZigBee є недосконала

взаємодія, однак існують цілі ізольовані екосистеми ZigBee різних компаній, таких як Xiaomi, Philips Hue та Fibaro [1, 7].

#### **2.1.4 Технологія бездротової мережі Bluetooth**

Сьогодні Bluetooth є основною технологією для підключення периферійних пристроїв і передачі даних у мережах персональних комп'ютерів (смартфони, ПК) (гарнітури, бездротові клавіатури, миші, принтери тощо). Поява Bluetooth версії 4.0, яка включає в себе варіант Bluetooth Smart з низьким споживанням енергії, поклала початок Інтернету речей. Основною метою розвитку цієї технології було нове покоління розумних гаджетів, багато з яких працюють від батарейок і вимагають підвищення енергоефективності [8].

Фізичні наскрізні прикладні рівні моделі OSI повністю покриваються Bluetooth, подібно до Z-Wave [2, 8]. Подібно до інших технологій зв'язку з низьким енергоспоживанням і низькою пропускну здатністю, Bluetooth Low Energy зосереджується на передачі даних у коротких пакетах, тому пристрої Bluetooth Low Energy підключаються один до одного лише тоді, коли потрібно надіслати або отримати дані. Перевагою Bluetooth Low Energy є низьке споживання енергії та швидкість передачі даних до 1 Мбіт/с.

Bluetooth Smart забезпечує надзвичайно тривалий час автономної роботи, оскільки підтримує сплячі вузли (пристрої, які в основному сплять, але час від часу прокидаються на короткий період, щоб завершити свою роботу).

Лише Bluetooth і Wi-Fi підтримуються практично всіма смартфонами, планшетами та ноутбуками з усіх бездротових технологій, які використовуються в Інтернеті речей [12].

Хоча Bluetooth забезпечує пряме з'єднання між пристроями, Wi-Fi передає всі повідомлення через централізовану точку доступу, що робить його непридатним для переважної більшості програм Інтернету речей.

Численні додаткові бездротові технології використовують ту саму частоту 2,4 ГГц, що й Bluetooth. Використання діапазону частот 2,4 ГГц є недоліком технології, оскільки, окрім перешкод, сигнал на цій частоті зникає набагато швидше, ніж на частотах нижче 1 ГГц.

Незважаючи на недоліки, кожна з перерахованих вище технологій займає особливе місце на ринку IoT і, в залежності від поставлених завдань, ідеально адаптована для розвитку певного класу мереж.

Основні переваги та недоліки бездротових мереж Wi-Fi, Bluetooth, Z-Wave та ZigBee підсумовано в таблиці 2.1.

Таблиця 2.1 – Порівняння безпроводних технологій для створення автоматизованої системи

Бездротовий протокол	Переваги	Недоліки	Наявність єдиного стандарту, щоб управляти всією технікою з однієї програми
Wi-Fi	<ol style="list-style-type: none"> <li>1. Широко використовується в мобільних і комп'ютерних пристроях;</li> <li>2. Має достатній радіус дії для взаємодії;</li> <li>3. Підтримує високу</li> </ol>	<ol style="list-style-type: none"> <li>1. Використовує велику кількість електроенергії;</li> <li>2. Відмовостійкість мережі не гарантується;</li> <li>3. Прикладний рівень OSI не</li> </ol>	Немає

	швидкість передачі даних	стандартизований	
--	--------------------------	------------------	--

Продовження таблиці 2.1

Z-Wave	<ol style="list-style-type: none"> <li>1. Лідер за використанням;</li> <li>2. Високий рівень відмовостійкості та масштабованості;</li> <li>3. Високий рівень захищеності;</li> <li>4. Низький рівень споживання електроенергії</li> </ol>	<ol style="list-style-type: none"> <li>1. Для Z-Wave використовуються різні частоти у різних країнах</li> </ol>	Повна сумісність
Zigbee	<ol style="list-style-type: none"> <li>1. Поширена технологія;</li> <li>2. Високий рівень відмовостійкості та масштабованості;</li> <li>3. Низький рівень споживання електроенергії</li> </ol>	<ol style="list-style-type: none"> <li>1. ZigBee використовує діапазон 2.4 ГГц, де існує велика кількість перешкод;</li> <li>2. Проблема з безпекою через недотримання вимог до сертифікації</li> </ol>	Немає
Bluetooth	<ol style="list-style-type: none"> <li>1. Стандарт охоплює всі</li> </ol>	<ol style="list-style-type: none"> <li>1. Bluetooth</li> </ol>	Немає

	<p>рівні моделі OSI;</p> <p>2. Зменшене споживання електроенергії (наприклад, у порівнянні з Wi-Fi).</p>	<p>використовує діапазон 2.4 ГГц, де існує велика кількість перешкод;</p> <p>2. Дуже малий радіус дії (до 10м);</p> <p>3. Bluetooth не використовує пористу топологію (як ZigBee).</p>	
--	--	--	--

На основі аналізу, проведеного в таблиці 2.1, для системи управління зовнішнім освітленням прийнято використовувати технології безпроводного зв'язку Bluetooth, оскільки цей стандарт охоплює всі рівні моделі OSI та має зменшене споживання електроенергії (наприклад, у порівнянні з Wi-Fi).

## 2.2 Розробка структурної схеми системи

На основі аналізу існуючих електронних систем керування зовнішнім освітленням, була обрана централізована архітектура для розробки та адміністрування системи з підвищеною ефективністю, оскільки вона дозволяє використовувати лише необхідні датчики та спрощує розробку, вимагаючи лише конфігурації центрального процесора.

На рисунку 2.5 представлено структурну схему системи управління зовнішнім освітленням з використанням мікроконтролера AtMega328P.

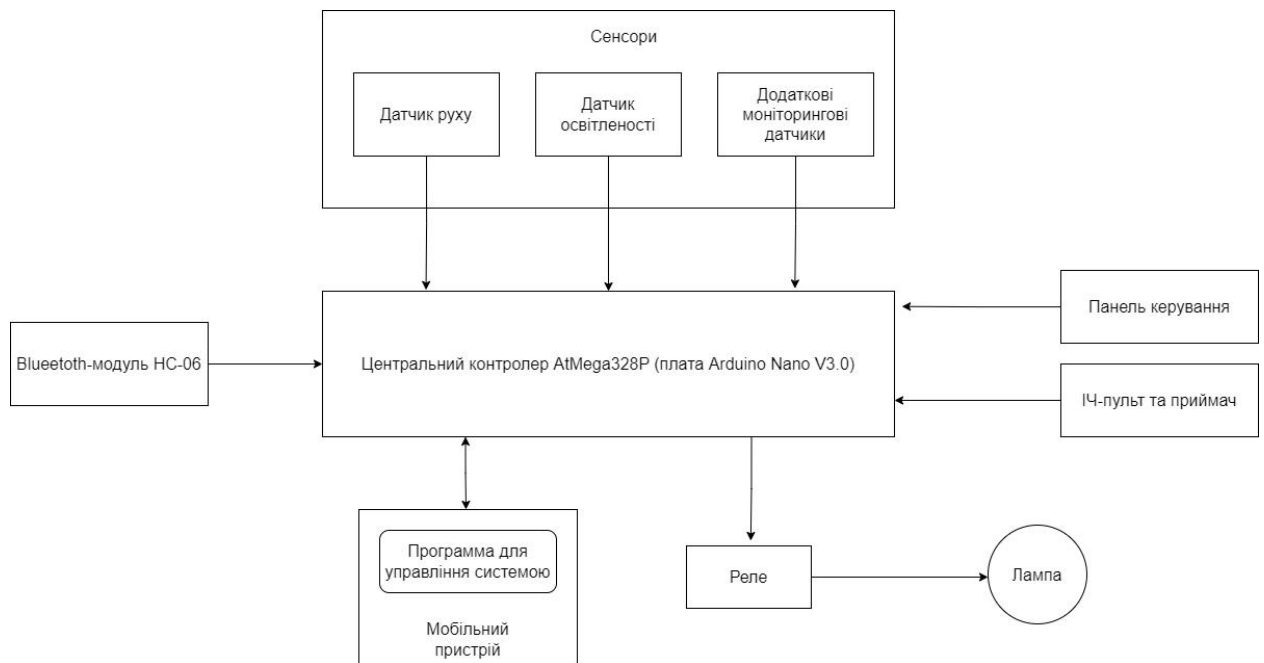


Рисунок 2.5 – Структура схема системи управління зовнішнім освітленням

Розглянемо зв'язок між кожним блоком і його з'єднання більш детально. Для повноцінної роботи приводів необхідний силовий компонент, наприклад реле або транзисторний перемикач. Оскільки вихід мікроконтролера може забезпечити лише 20 мА, цього недостатньо для живлення приводів і світлодіода. В результаті використовується силова частина, куди підключається зовнішнє джерело напруги як 5 В, так і 220 В. Реле має перевагу гальванічної розв'язки, що означає, що силова частина жодним чином не підключена до логічної частини, на відміну від транзисторного перемикача. Однак для живлення котушки реле необхідний достатній струм.

Для живлення модулів бажано використовувати зовнішнє джерело напруги, бажано 5 В. Таким чином буде збережено ходовий ресурс МК. Залежно від типу модуля датчики та виконавчі механізми будуть підключені до цифрових і аналогових контактів. Інтерфейс I2C, піни A4 і A5, будуть використовуватися для зв'язку між дисплеєм і іншими модулями. Цей інтерфейс може підтримувати до 128 модулів. Далі, детальніше

проаналізуємо технічні елементи та сенсори, які використовуються для розробки системи.

## **2.3 Вибір апаратного забезпечення**

### **2.3.1 Блок реле**

Реле – це пристрій, у якому вихідне значення  $Y$  швидко змінюється та приймає обмежений набір значень, коли вхідне значення  $X$  досягає певного значення. Найчастіше він має форму автоматизованого пристрою, який реагує на зміну вхідного параметра (температура, тиск, освітленість тощо) і, коли цей показник досягає певного значення, замикає або розмикає електричне коло. Коли потрібно керувати багатьма електричними лініями одним сигналом або коли потрібен сигнал малої потужності з ідеальною гальванічною розв'язкою, використовуються реле. У телефонних станціях і перших комп'ютерах, які могли виконувати логічні процеси, часто використовувалися реле.

Реле поділяють на: електромагнітні, електротеплові, індукційні.

Конструкція електромагнітного реле складається з котушки зі сталевим сердечником, набору рухомих контактів і регульованого електричного кола, яке замикається і розмикається.

Через відсутність механічних компонентів твердотільні реле є невеликими та надійними пристроями. Біполярні та MOSFET транзистори, тиристори та семістори є прикладами напівпровідникових компонентів, які виконують механічну роботу в цій системі. Її переваги перед іншими реле включають наступне:

1. Низький рівень шуму під час роботи.
2. Електроспоживання нижче на 95%.
3. Швидкодія твердотільного реле дуже висока (мілісекунди).



Для розробки системи керування зовнішнім освітленням буде використане саме електромагнітне реле, яке представлено на рисунку 2.6. Також, характеристики реле деталізовані в таблиці 2.2.



Рисунок 2.6 – Електромагнітне реле SRD-5VDC-SL-C [19]

Таблиця 2.2 – Технічні характеристики SRD-5VDC-SL-C

Максимальне навантаження	10A 250В змінного струму
Напруга живлення	5В
Номінальна напруга на котушці	5В постійного струму
Кількість контактів перемикання	2 (NO і NC)
Температура навколишнього середовища	-25 ... + 70 град. Цельсія
Опір ізоляції	$\geq 100$ МОм
Розміри	19x15x15 мм

Потужність котушки	0.36 Вт
Механічний термін служби	10,000,000

### 2.3.2 Bluetooth модуль HC-06

Модуль призначений для передачі та прийому даних Bluetooth. Коли до модуля подається живлення, червоний світлодіод починає блимати, а при підключенні до іншого пристрою він світиться червоним.

Розпіновка модулю наступна:

1. STATE – сюди дублюється сигнал з вбудованого світлодіода, коли модуль активний - світлодіод блимає, коли зв'язок встановлено - горить.
2. TXD – сюди модуль відправляє дані.
3. VCC – живлення 5В.
4. EN – вкл / викл, якщо подати сюди логічну одиницю, то модуль вимкнеться, якщо логічний нуль – буде працювати.

На рисунку 2.7 представлено зовнішній вигляд модуля, а основні параметри представлені в таблиці 2.3.

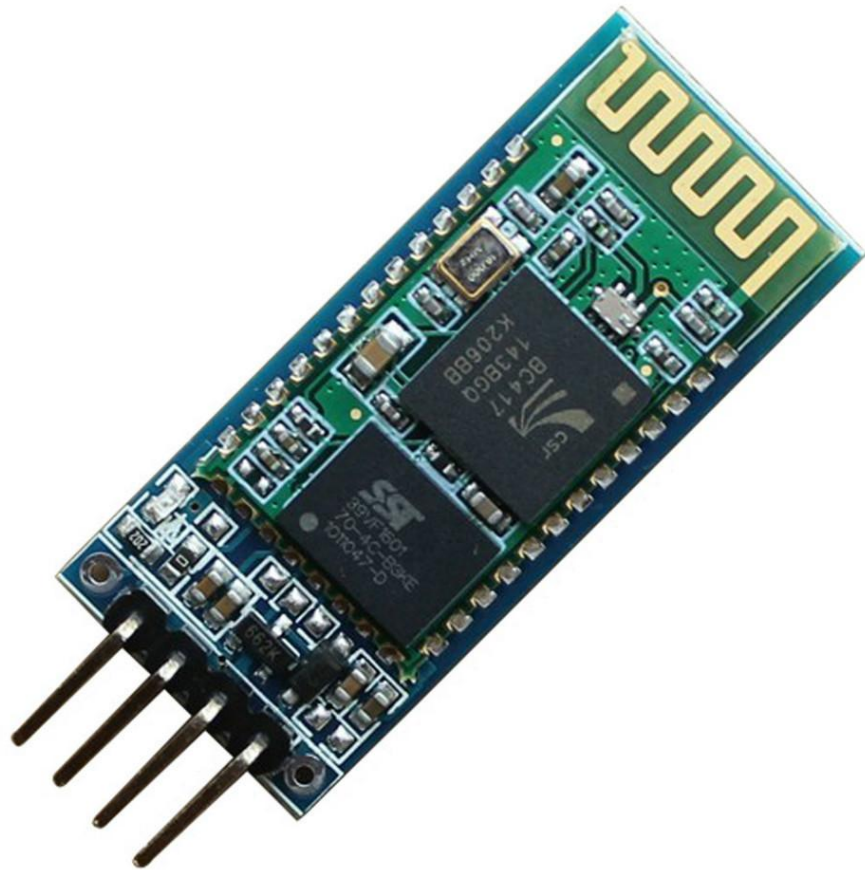


Рисунок 2.7 – Bluetooth-модуль HC-06

Таблиця 2.3 – Технічні характеристики HC-06

Напруга живлення	3.3-5V
Максимальна вхідна напруга	5V
Вихідна напруга логічної одиниці	3.3V
Максимальний струм споживання	45 мА
Швидкість передачі даних	1200–1382400 бод
Дальність зв'язку при прямій видимості	30 м
Розміри	19x15x15 мм

Схема підключення модулю HC-06 до Arduino Nano представлена на рисунку 2.8.

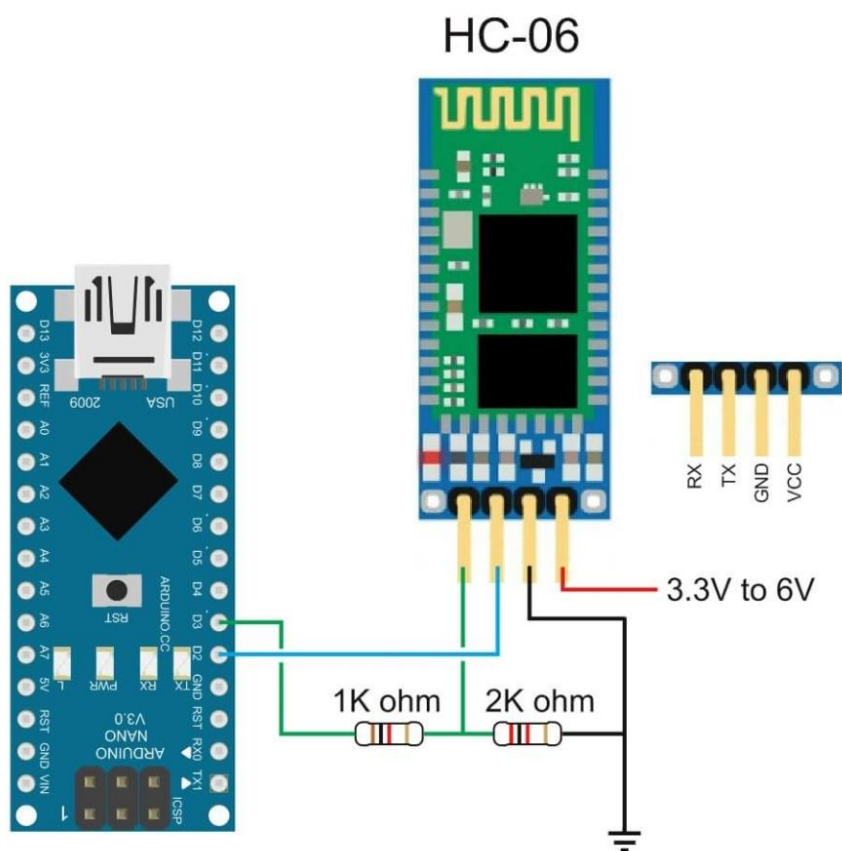


Рисунок 2.8 – Схема підключення модулю HC-06 до Arduino Nano

### 2.3.3 Датчик освітленості GY-302

Цифровий датчик освітленості GY-302 на чіпі BH1750 призначений для вимірювання фонового освітлення. Має високу чутливість і поширений послідовний інтерфейс I2C. Спектр чутливості збігається з кривою чутливості людського ока [21]. На рисунку 2.9 представлено зовнішній вигляд модуля, а основні параметри представлені в таблиці 2.4.

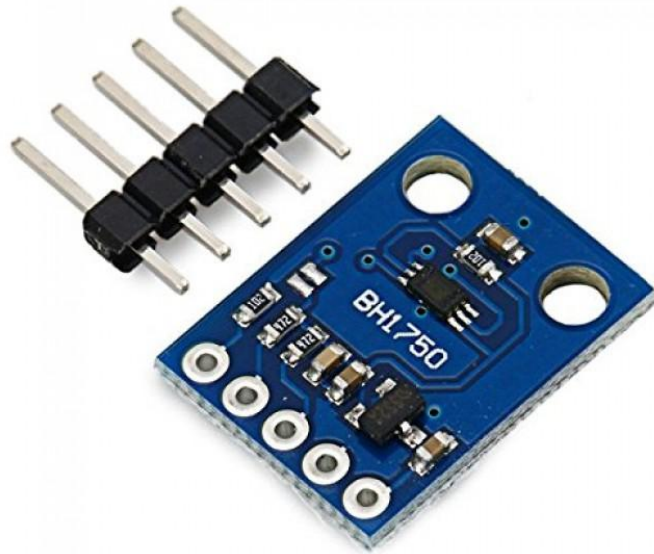


Рисунок 2.9 – Датчик освітленості GY-302

Таблиця 2.4 – Технічні характеристики GY-302

Оригінальний чіп	BH1750FVI ROHM
Точність вимірювання	1 люкс
Діапазон даних	0-65535 лк
Інтерфейс	I2C
Розміри	13.9 X 18.5 mm
Переваги	Спектральна характеристика близька до візуальної чутливості. Нечутливий до фонового світла. Прямий цифровий вихід, без додаткових складних обчислень.

Схема підключення датчика освітленості GY-302 BH1750FVI до Arduino Nano V3.0 представлена на рисунку 2.10.

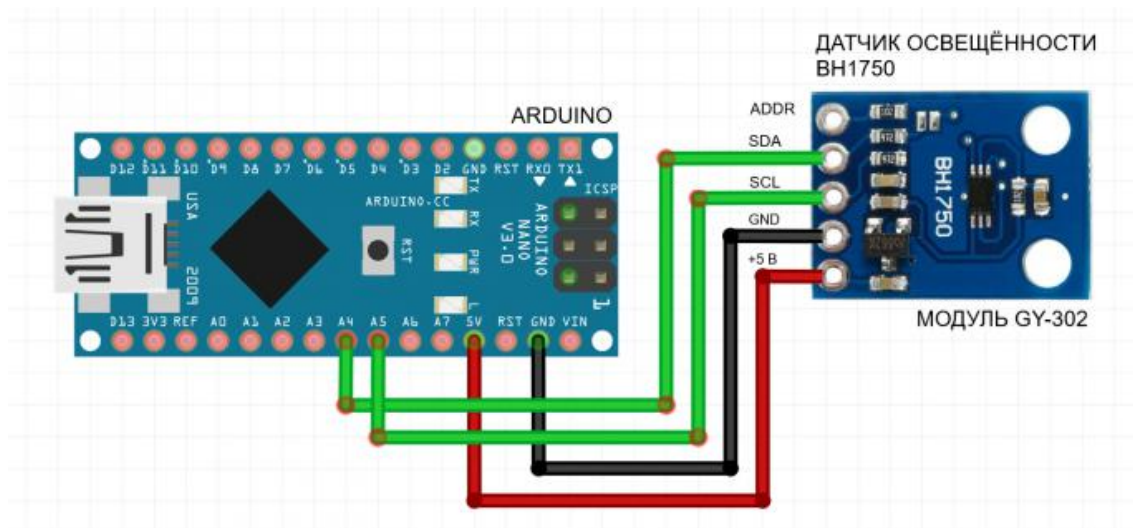


Рисунок 2.10 – Схема підключення модулю GY-302 до Arduino

### 2.3.4 Блок живлення

Для подачі живлення на Arduino та перетворення напруги з 220 В в 12 В використовується блок живлення. В таблиці 2.5 представлені основні характеристики блоку живлення, який був обраний для системи.

Таблиця 2.5 – Технічні характеристики SUNNY 12В

Виробництво	SUNNY Comp. Tech
Вхідна напруга	90 - 264 В
Вихідна напруга	12В
Енергоефективність	> 77.8%
Потужність	12Вт

## РОЗДІЛ 3. АПАРАТНО-ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ УПРАВЛІННЯ ЗОВНІШНІМ ОСВІТЛЕННЯМ

### 3.1 Програмний код та його оптимізація

Після завершення проектування концептуальної та електричної схеми системи управління зовнішнім освітленням, наступним етапом стало інтегрування архітектури в середовище Proteus Professional, а також розробка програмного коду для тестування працездатності відповідних модулів системи.

Для розробки програмного забезпечення використовуються наступні бібліотеки для роботи з мікроконтролерами:

1. `#include "Wire.h"` – для зв'язку по I2C інтерфейсу з модулями;
2. `#include <LiquidCrystal.h>` – для роботи з LCD дисплеєм;

Для налагодження роботи коду та для виводу необхідних повідомлень на монітор порту використовується метод `Serial.println()`.

В розробленому скетчі використовувалася велика кількість функцій, що дозволило зменшити обсяг коду в чотири рази та уникнути дублювання.

Підпрограми додаються після основного циклу `void loop()`. Найпоширенішими функціями є `MeasureAndRelay`, `MainLCDandMenu` і `setup`. Тоді як функція `clean` відповідає за очищення екрана дисплея, `select` - для кнопок панелі управління, а `select2` - для вибору значення параметра від 5 до 70 балів з кроком 5 пунктів. Команда `lcd.print("Please set value")` використовувалася для виведення символів і параметрів на дисплей, а команда `lcd.setCursor(0,0)` використовувалася для їх позиціонування.

На рисунку 3.1 представлений програмний код, який відповідає за роботу реле в системі управління зовнішнім освітленням.

```

95 void MeasureAndRelay() {
96     //Relay settings//
97     if (temp <= tempMin) {
98         digitalWrite(A1, 1);
99     } else if (temp >= tempMax) {
100         digitalWrite(A1, 0);
101     }
102
103     hum = myHumidity.readHumidity();
104     temp = myHumidity.readTemperature();
105     if (hum <= humMin) {
106         digitalWrite(A2, 1);
107     } else if (hum >= humMax) {
108         digitalWrite(A2, 0);
109     }
110
111     if (millis() - lastMillis > SAMPLE_TIME) {
112         light = analogRead(LIGHT_PIN);
113         light = map(light, 0, 910, 0, 100);
114         lastMillis = millis();
115         Serial.println("Temperature: ");
116         Serial.println(temp);
117         Serial.println("Humidity: ");
118         Serial.println(hum);
119         Serial.println("Light: ");
120         Serial.println(light);
121     }
122
123     if (light <= lightMin) {
124         digitalWrite(A3, 1);
125     } else if (light > lightMin) {
126         digitalWrite(A3, 0);
127     }
128     //
129 }

```

Рисунок 3.1 – Програмний код для обробки режимів роботи реле

Крім того, розроблено функції, які запобігають перевищенню налаштувань параметрів крайніх мінімальних і максимальних значень.

Далі, розглянемо програмний код для роботи контролера зовнішнього освітлення, який представлений на рисунку 3.2.



```

//////////Setting the lighting to trigger the relay/////
if (flagMenu == 4) {
  lcd.setCursor(0, 0);
  lcd.print("LIGHT ON/OFF   ");
  lcd.setCursor(0, 1);
  lcd.write(byte(2));
  lcd.setCursor(1, 1);
  lcd.print("VALUE:");
  lcd.setCursor(7, 1);
  lcd.print(lightMin);
  lcd.setCursor(10, 1);
  lcd.print("%");
  lcd.setCursor(15, 1);
  lcd.write(byte(1));

  if (butLeft.click()) {
    lightMin--;
    if (lightMin <= 0) lightMin = 0;
  }
  if (butRight.click()) {
    lightMin++;
    if (lightMin >= 100) lightMin = 100;
  }

  if (Serial.available() > 0) {
    int a = Serial.read();

    if(a == 'a') {irsend.sendNEC(0xFFB97F, 32); delay(40);} //Brightness+
    if(a == 'b') {irsend.sendNEC(0xFF11CF, 32); delay(40);} //Brightness-
    if(a == 'c') {irsend.sendNEC(0xFF718F, 32); delay(40);} //On
    if(a == 'd') {irsend.sendNEC(0xFFD92F, 32); delay(40);} //Off

    if(a == 'e') {irsend.sendNEC(0xFF817F, 32); delay(40);} //Red
    if(a == 'f') {irsend.sendNEC(0xFF09DF, 32); delay(40);} //Green
    if(a == 'g') {irsend.sendNEC(0xFF61BF, 32); delay(40);} //Blue
    if(a == 'h') {irsend.sendNEC(0xFFE91F, 32); delay(40);} //White

    if(a == 'i') {irsend.sendNEC(0xFFA64F, 32); delay(40);}
    if(a == 'j') {irsend.sendNEC(0xFF36DF, 32); delay(40);}
    if(a == 'k') {irsend.sendNEC(0xFF732F, 32); delay(40);}
    if(a == 'l') {irsend.sendNEC(0xFFE37F, 32); delay(40);}
  }
}

```

Рисунок 3.2 – Програмний код для налаштування зовнішнього освітлення

Як видно з рисунку 3.2, для роботи контролера зовнішнього освітлення використовуються умовні конструкції, які контролюють задані діапазони значень та встановлюють відповідні параметри.

Першим кроком є створення об'єкта передачі під назвою `Irsend`. Значення наступного байта з буфера послідовного підключення виділяється змінній `a`. Потім умовний оператор `if` визначає, чи відповідає надане значення кожній літері англійського алфавіту, від `a` до `l`, яке було попередньо введено для кожної кнопки програми смартфона та передано через виклик `BluetoothClient`. Код, попередньо зчитаний з пульта дистанційного керування, надсилається у форматі NEC об'єктом `irsend` в тілі оператора.

Далі, на рисунку 3.3 представлена функція `setup()`, яка ініціалізує розроблені символічні позначення. Функція `pinMode()` використовується для налаштування певної поведінки на вхід або вихід. Функція `digitalWrite()` використовується для запису значення `HIGH` або `LOW` на цифровий пін.

```
void setup() {  
  // put your setup code here, to run once:  
  Serial.begin(9600);  
  myHumidity.begin();  
  
  lcd.createChar(0, degr);  
  lcd.createChar(1, sun);  
  lcd.createChar(2, arrow);  
  lcd.createChar(3, lamp);  
  lcd.createChar(4, drop);  
  lcd.createChar(5, temprelay);  
  lcd.begin(16, 2);  
  
  pinMode(A0, INPUT);  
  pinMode(A1, OUTPUT);  
  pinMode(A2, OUTPUT);  
  pinMode(A3, OUTPUT);  
  pinMode(6, OUTPUT);  
  digitalWrite(A1, 0);  
  digitalWrite(A2, 0);  
  digitalWrite(A3, 0);  
  digitalWrite(6, 0);  
}
```

Рисунок 3.3 – Код головної функції ініціалізації

### 3.2 Інструкція по користуванню приладом

Розглянемо детальніше використання приладу для управління системою зовнішнього освітлення та налаштування параметрів. При першому запуску приладу відображається привітання (рис. 3.4).

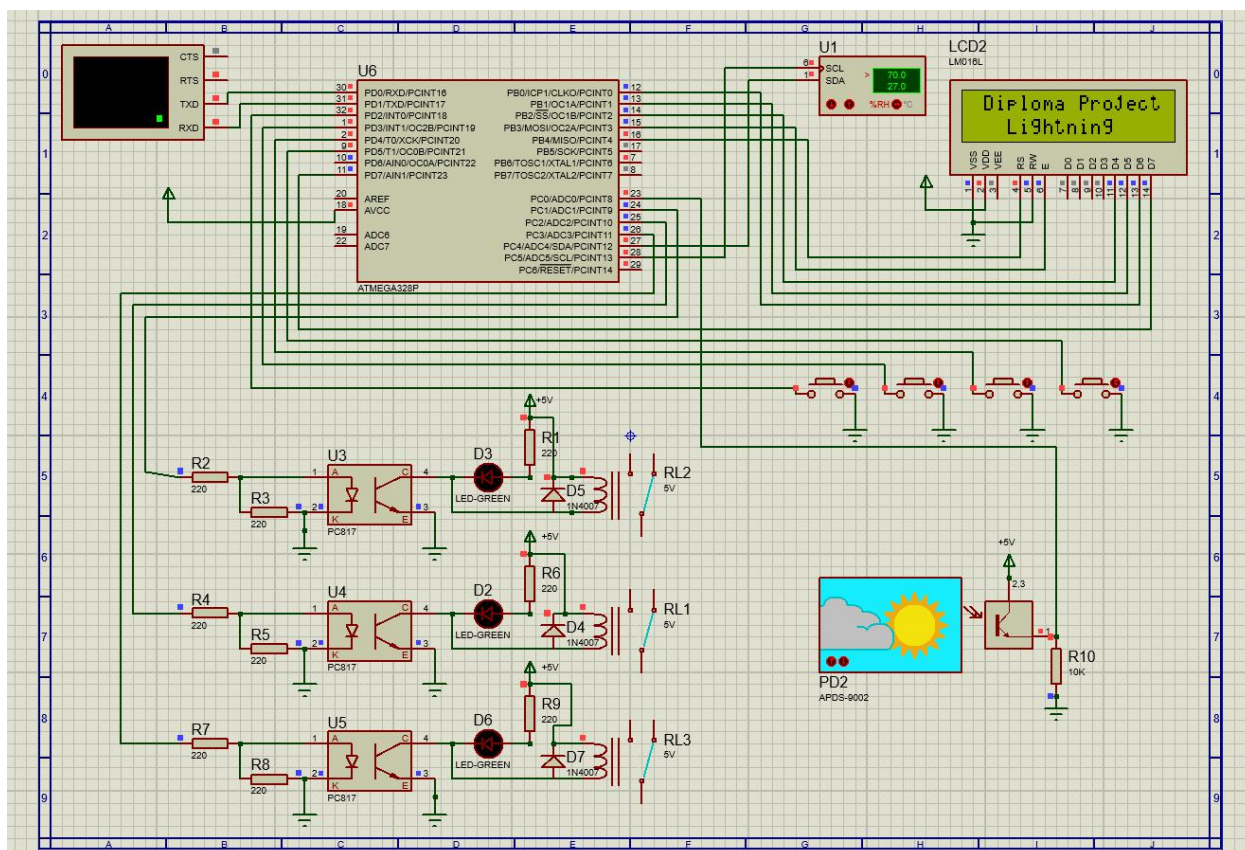


Рисунок 3.4 – Головний екран системи управління зовнішнім освітленням

Далі, через 5 секунд відбувається перехід на основний екран системи, де представлені основні параметри для налаштування (рис. 3.5).

На основному екрані системи управління зовнішнім освітленням представлені наступні індикатори:

1. L – освітленість приміщення (у відсотках).
2. Символьні позначення, які автоматично з'являються при досягненні відповідних значень.



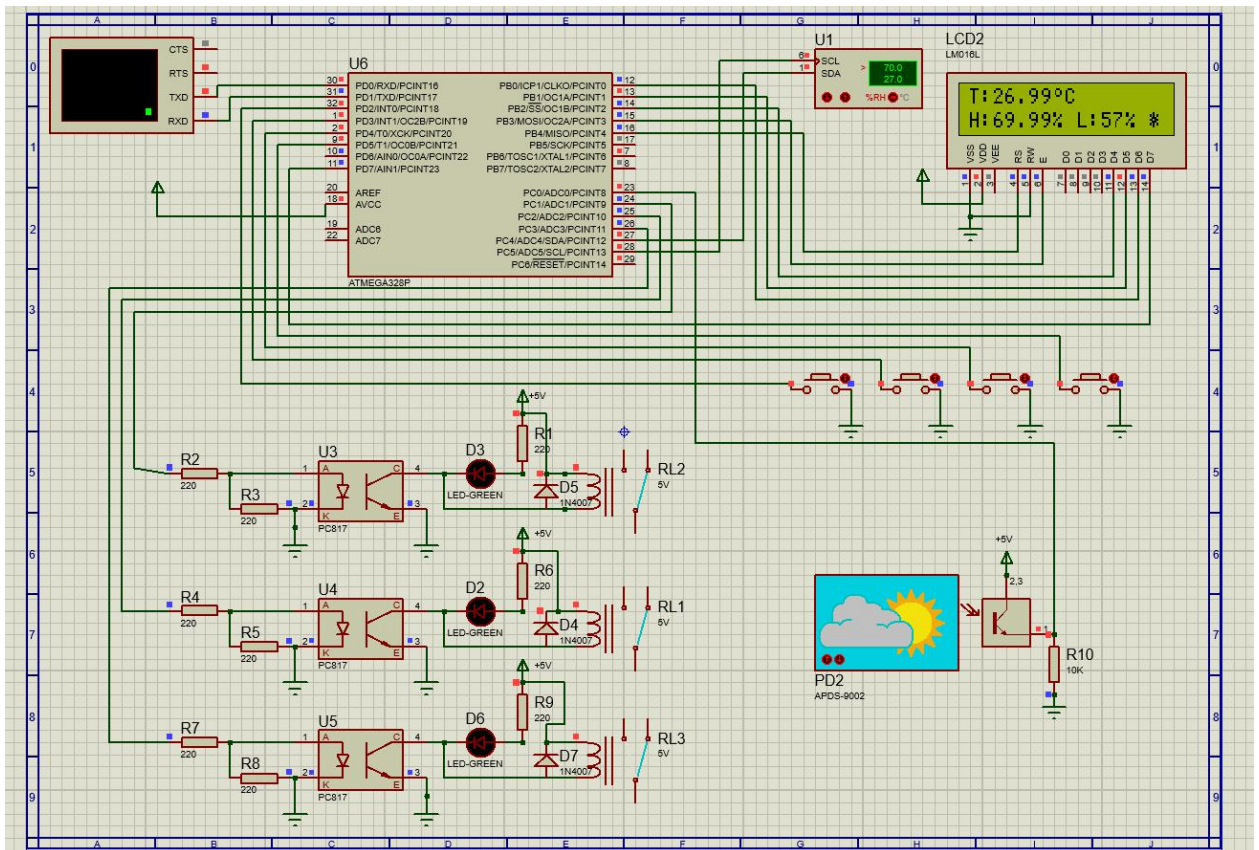


Рисунок 3.5 – Головний екран системи управління зовнішнім освітленням

При натисканні кнопки «Вибір», користувач системи переходить в режим налаштувань, де відображаються поточні мінімальні та максимальні значення параметрів системи. В меню з'являється стрілка вибору параметру, значення якого необхідно змінити. На рисунку 3.6 представлений вигляд налаштування мінімального значення для включення реле освітлення. Символьні позначення ідентифікують поточний стан реле освітлення. При активації освітлення на екрані з'являється позначення лампи, при дезактивації – позначення зникає до повторного тригера відповідного значення.

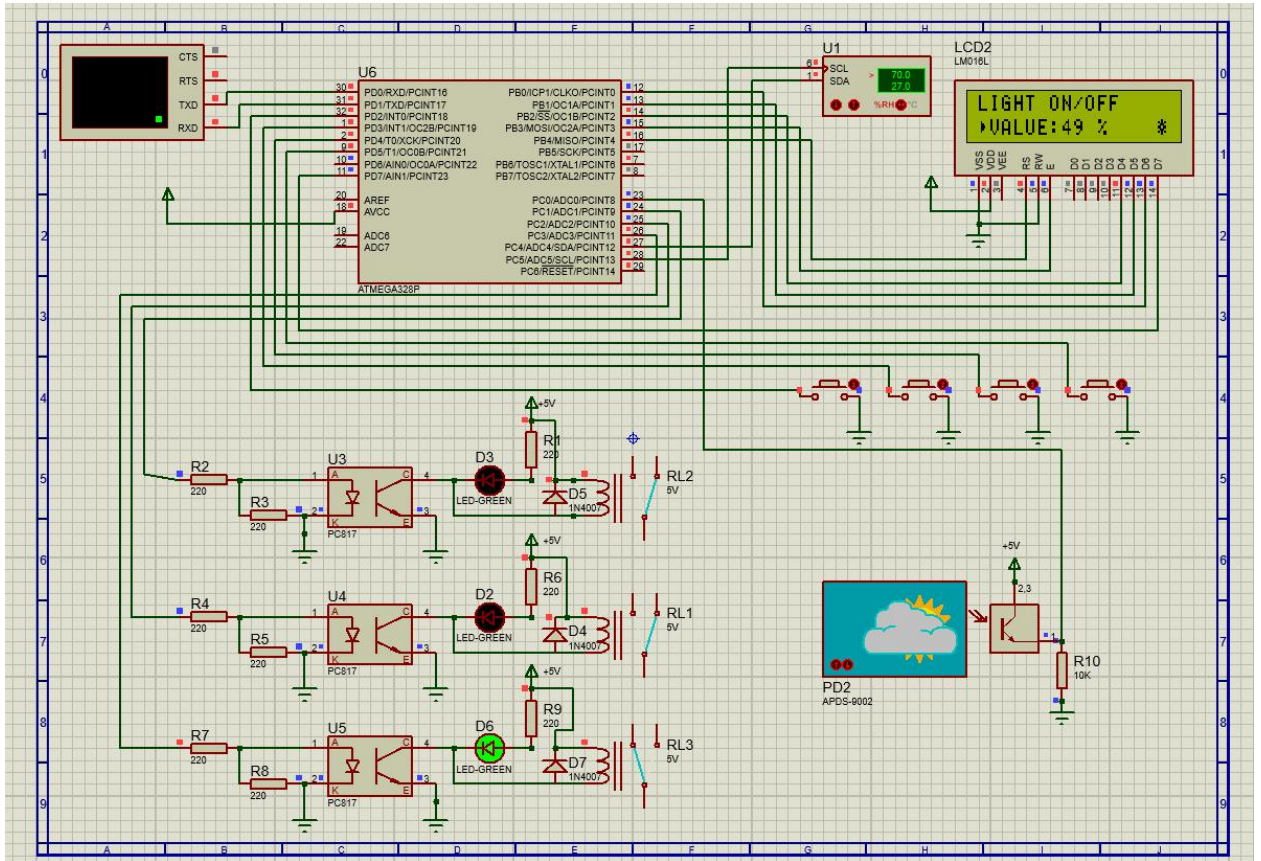


Рисунок 3.6 – Секція меню для налаштування реле активації освітлення

Відповідне значення використовується для контролювання процесу активації реле при низькому рівні освітленості в приміщенні, а також дезактивації при високому рівні освітленості в приміщенні.

### 3.3 Розробка друкованої плати для управління зовнішнім освітленням

Для побудови пристрою, схематичний опис якого наведено в додатку А, необхідно розробити друковану плату. Ця плата дозволить електрично з'єднати компоненти та забезпечити стабільну роботу кожного модуля.

Далі, на рисунку 3.7 представимо зовнішній вигляд розробленої друкованої плати для системи управління зовнішнім освітленням.

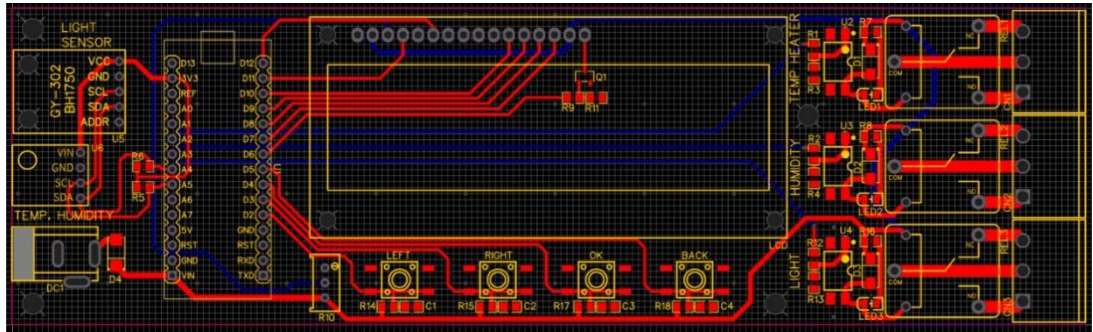


Рисунок 3.7 – Розведення доріжок РСВ для системи управління зовнішнім освітленням

На рисунку 3.8 представлений загальний вигляд друкованої плати системи управління зовнішнім освітленням з нанесеною маскою.

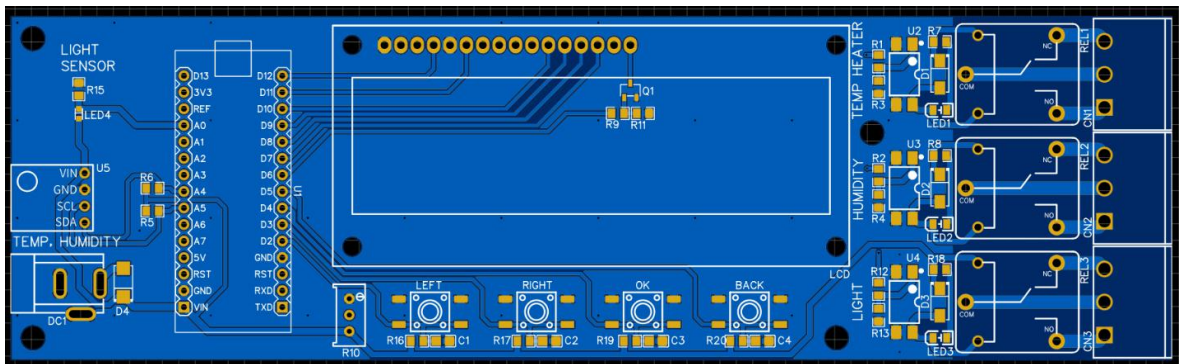


Рисунок 3.8 – Зовнішній вигляд друкованої плати з нанесеною маскою

На основі розробленої принципової схеми системи управління зовнішнім освітленням, було розроблено загальний вигляд друкованої плати, а також проведено моделювання розведення доріжок. Для моделювання друкованої плати використовувалося середовище розробки EasyEDA STD.

## ВИСНОВКИ

Отже, на основі дослідження проведеного у кваліфікаційній роботі, можна зробити наступні висновки.

1. У першому розділі було проаналізовано тему дослідження та розглянуто основні розробки у використанні технологій зовнішнього освітлення. За результатами дослідження були обрані компоненти для побудови системи управління зовнішнім освітленням. Центральним елементом цього пристрою є мікроконтролер, який отримує інформацію від датчиків, обробляє її та генерує відповідь через декілька виконавчих підсистем. Переваги використання Arduino Nano у створенні порівнянних систем, які використовують необхідні датчики, систематизовано.

2. В другому розділі проведено моделювання функціональної схеми системи управління зовнішнім освітленням приміщення. Розглянуто необхідні для розробки елементи: реле, датчик освітленості, блок живлення тощо.

3. В третьому розділі проведено моделювання друкованої плати пристрою. На основі розробленої принципової схеми системи управління зовнішнім освітленням, було розроблено загальний вигляд друкованої плати, а також проведено моделювання розведення доріжок. На основі проведеного проектування був розроблений та оптимізований програмний код, а також протестований на електричній схемі в середовищі Proteus Professional.

Для вдосконалення приладу в подальшому планується: заміна LCD-дисплею на сенсорний TFT-дисплей, реалізація можливостей встановлення зміни параметрів відповідно до різноманітних зовнішніх чинників, підключення Wi-Fi модулю для зручного керування параметрами, розробка застосунку для автоматизації управління можливостями пристрою.



## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Lee C. W. The influence of LED lighting on attention and long-term memory / C. W. Lee, J. H. Kim // International Journal of Optics. – 2020. – С. 1-6.
2. S. Folea, D. Bordenca, C. Hotea and H. Valean, "Smart home automation system using Wi-Fi low power devices". – 2012. – Automation Quality and Testing Robotics (AQTR) 2012 IEEE International Conference on, С. 569-574.
3. Patrascu M. Integrating Services and Agents for Control and Monitoring: Managing Emergencies in Smart Buildings. Service Orientation in Holonic and Multi-Agent Manufacturing and Robotics. / Patrascu., 2014. – 544 с.
4. Дужак І. О. Розумний будинок / І. О. Дужак. // Автоматизація технологічних і бізнес-процесів. Одеська національна академія харчових технологій. – 2013. – №13. – С. 31.
5. Полякова О.В. Класифікація функціональних складових елементів системи інтелектуального керування середовищем при проектуванні житла // Вісник Київського національного університету технологій та дизайну. Серія: Технічні науки. – 2016. – № 4. – С. 133–141.
6. S. Wadhvani, U. Singh, P. Singh, S. Dwivedi. Smart Home Automation and Security System using Arduino and IOT // International Research Journal of Engineering and Technology. 2018. V.5, №2. С. 1357 – 1359.
7. Understanding Building Automation and Control Systems. 2013. URL: [http://www.kmccontrols.com/products/Understanding\\_Building\\_Automation\\_and\\_Control\\_Systems.aspx](http://www.kmccontrols.com/products/Understanding_Building_Automation_and_Control_Systems.aspx) (дата звернення: 22.02.2023).
8. Z-Wave Vs. Zigbee. URL: <https://www.link-labs.com/blog/z-wave-vs-zigbee>. (дата звернення: 28.01.2023).
9. Arduino Platform Specification. Arduino CLI : веб-сайт. URL: <https://arduino.github.io/arduino-cli/0.21> (дата звернення: 10.12.2022).

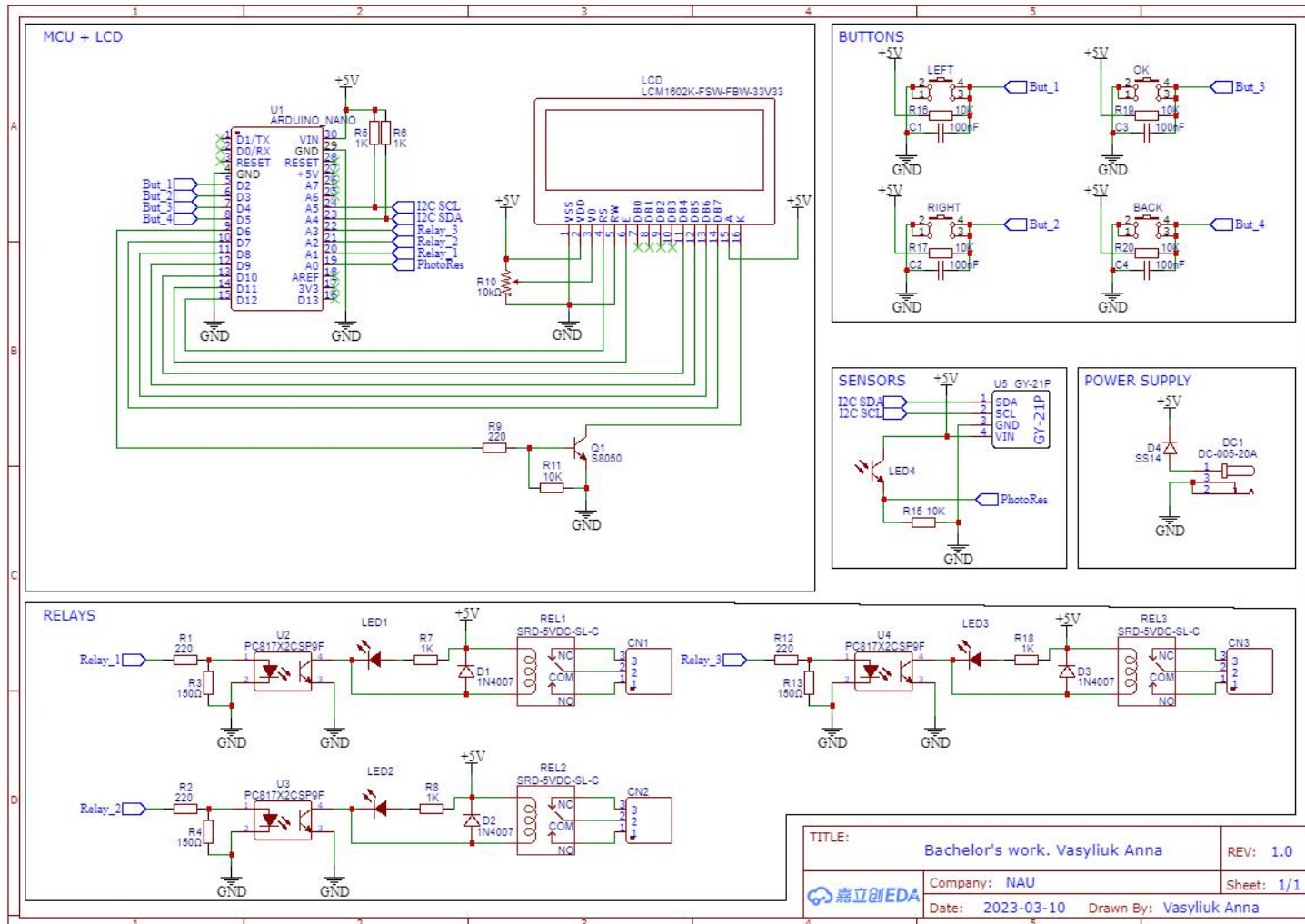


10. Sunco Lightning LED Bulbs. Sunco : веб-сайт. URL: <https://www.sunco.com/collections/led-bulbs> (дата звернення: 15.02.2023).
11. Zigbee Protocol. How does it work? Vesternet : веб-сайт. <https://www.vesternet.com/pages/what-is-zigbee> (дата звернення: 15.02.2023).
12. Bluetooth vs Wi-Fi. Comparison with charts. El-hub: веб-сайт. URL: <https://www.electronicshub.org/bluetooth-vs-wi-fi/> (дата звернення: 18.02.2023).
13. Arduino Hardware. Everything you need to know about Arduino. Circuitdigest : веб-сайт. URL: <https://circuitdigest.com/article/everything-you-need-to-know-about-arduino-uno-board-hardware> (дата звернення: 16.01.2023).
14. What is Z-Wave and why is it important? Z-Wave : веб-сайт. URL: <https://www.z-wave.com/> (дата звернення: 22.01.2023).
15. What is the OSI Model?. Forcepoint : веб-сайт. URL: <https://www.forcepoint.com/cyber-edu/osi-model/> (дата звернення: 24.01.2023).
16. Arduino Nano V3.0 AVR ATmega328P з розпаяними роз'ємами. Arduino. URL: <https://arduino.ua/prod166-arduino-nano-v3-0-avr-atmega328p-s-raspayannimi-razemami> (дата звернення: 25.01.2023).
17. Hajjaj, M., Miki, M., & Shimohara, K. “The Effect of Using the Intelligent Lighting System to Deduct the Power Consumption at the Office”, IEEE 7th Conference on Systems, Process and Control (ICSPC), 2019. — 66 p.
18. Power pins for Arduino. ArduinoDocs: веб-сайт. URL: <https://docs.arduino.cc/learn/electronics/power-pins> (дата звернення: 28.01.2023).
19. Реле 5В 10А SRD-5VDC-SL-C. Arduino. URL: <https://arduino.ua/prod393-rele-5v-srd-5vdc-sl-c> (дата звернення: 12.02.2023).
20. Arduino Uno Documentation. ArduinoDocs : веб-сайт. URL: <https://docs.arduino.cc/hardware/uno-rev3> (дата звернення: 16.02.2023).
21. Arduino Yun Documentation. ArduinoDocs : веб-сайт. URL: <https://docs.arduino.cc/retired/boards/arduino-yun> (дата звернення: 19.01.2023).
22. Arduino Nano board overview. JavaTPoint : веб-сайт. URL: <https://www.javatpoint.com/arduino-nano> (дата звернення: 19.01.2023).



# ДОДАТОК А

## Схема розробленого пристрою



## ДОДАТОК Б

Лістинг програмного коду системи управління зовнішнім освітленням

```
#include "Wire.h"
#include <LiquidCrystal.h>
#include "SparkFunHTU21D.h"
#include <EncButton.h>
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);
```

```
int light;
int lightMin = 50;
```

```
#define LIGHT_PIN A0
#define SAMPLE_TIME 100
static unsigned int lastMillis = 0;
byte degr[8] = {
    B00110,
    B01001,
    B01001,
    B00110,
    B00000,
    B00000,
    B00000,
};
```

```
byte sun[8] = {
    B00100,
    B10101,
    B01110,
```

```
B11011,  
B01110,  
B10101,  
B00100,  
};
```

```
byte arrow[8] = {  
    B00000,  
    B01000,  
    B01100,  
    B01110,  
    B01100,  
    B01000,  
    B00000,  
};
```

```
byte lamp[8] = {  
    B01110,  
    B10001,  
    B10001,  
    B10001,  
    B01010,  
    B01010,  
    B01110,  
};
```

```
byte drop[8] = {  
    B00000,  
    B00100,  
    B01110,
```

```
B11111,  
B11111,  
B11111,  
B01110,  
};
```

```
byte temprelay[8] = {  
    B11000,  
    B11000,  
    B00011,  
    B00100,  
    B00100,  
    B00100,  
    B00011,  
};
```

```
#define BUTTON_LEFT 2  
#define BUTTON_RIGHT 3  
#define BUTTON_OK 4  
#define BUTTON_BACK 5  
EncButton<EB_TICK, BUTTON_LEFT> butLeft;  
EncButton<EB_TICK, BUTTON_RIGHT> butRight;  
EncButton<EB_TICK, BUTTON_OK> butOk;  
EncButton<EB_TICK, BUTTON_BACK> butBack;
```

```
byte flagMenu = 1;  
byte flagTemp = 1;  
byte flagHum = 1;
```

```
void MeasureAndRelay() {
```

```

////////////////////Relay settings////////////////////
if (temp <= tempMin) {
    digitalWrite(A1, 1);
} else if (temp >= tempMax) {
    digitalWrite(A1, 0);
}

hum = myHumidity.readHumidity();
temp = myHumidity.readTemperature();
if (hum <= humMin) {
    digitalWrite(A2, 1);
} else if (hum >= humMax) {
    digitalWrite(A2, 0);
}

if (millis() - lastMillis > SAMPLE_TIME) {
    light = analogRead(LIGHT_PIN);
    light = map(light, 0, 910, 0, 100);
    lastMillis = millis();
    Serial.println("Temperature: ");
    Serial.println(temp);
    Serial.println("Humidity: ");
    Serial.println(hum);
    Serial.println("Light: ");
    Serial.println(light);
}

if (light <= lightMin) {
    digitalWrite(A3, 1);
} else if (light > lightMin) {

```

```

digitalWrite(A3, 0);
}
////////////////////////////////////
}

void MainLCDandMenu() {

////////////////////////////////////
if (flagMenu == 1) {
////////////////////////////////////
if (digitalRead(A1) == 1) {
    lcd.setCursor(11, 0);
    lcd.write(byte(5));
} else {
    lcd.setCursor(11, 0);
    lcd.print(" ");
}

if (digitalRead(A2) == 1) {
    lcd.setCursor(13, 0);
    lcd.write(byte(4));
} else {
    lcd.setCursor(13, 0);
    lcd.print(" ");
}

if (digitalRead(A3) == 1) {
    lcd.setCursor(15, 0);
    lcd.write(byte(3));
} else {

```



```

    lcd.setCursor(15, 0);
    lcd.print(" ");
}

if (temp > 99 && temp <= 121) {
    lcd.setCursor(0, 0);
    lcd.print("T:");
    lcd.setCursor(2, 0);
    lcd.print(temp);
    lcd.setCursor(8, 0);
    lcd.write(byte(0));
    lcd.setCursor(9, 0);
    lcd.print("C");
} else if (temp < 100 && temp >= 10) {
    lcd.setCursor(0, 0);
    lcd.print("T:");
    lcd.setCursor(2, 0);
    lcd.print(temp);
    lcd.setCursor(7, 0);
    lcd.write(byte(0));
    lcd.setCursor(8, 0);
    lcd.print("C ");
} else if (temp < 10 && temp >= 0) {
    lcd.setCursor(0, 0);
    lcd.print("T:");
    lcd.setCursor(2, 0);
    lcd.print(temp);
    lcd.setCursor(6, 0);
    lcd.write(byte(0));
    lcd.setCursor(7, 0);

```

```

    lcd.print("C ");
} else if (temp < 0 && temp >= (-41)) {
    lcd.setCursor(0, 0);
    lcd.print("T:");
    lcd.setCursor(2, 0);
    lcd.print(temp);
    lcd.setCursor(8, 0);
    lcd.write(byte(0));
    lcd.setCursor(9, 0);
    lcd.print("C ");
} else {
    lcd.setCursor(0, 0);
    lcd.print("T:");
    lcd.setCursor(2, 0);
    lcd.print("ERROR!!! ");
}
////////////////////////////////////

////////////////////////////////////Lightning////////////////////////////////////
if (light == 100) {
    lcd.setCursor(9, 1);
    lcd.print("L:");
    lcd.setCursor(11, 1);
    lcd.print(light);
    lcd.setCursor(14, 1);
    lcd.print("%");
    lcd.setCursor(15, 1);
    lcd.write(byte(1));
} else if (light < 100 && light >= 10) {
    lcd.setCursor(9, 1);

```

```

lcd.print("L:");
lcd.setCursor(11, 1);
lcd.print(light);
lcd.setCursor(13, 1);
lcd.print("% ");
lcd.setCursor(15, 1);
lcd.write(byte(1));
} else if (light < 10 && light >= 0) {
  lcd.setCursor(9, 1);
  lcd.print("L:");
  lcd.setCursor(11, 1);
  lcd.print(light);
  lcd.setCursor(12, 1);
  lcd.print("% ");
  lcd.setCursor(15, 1);
  lcd.write(byte(1));
} else {
  lcd.setCursor(9, 1);
  lcd.print("L:");
  lcd.setCursor(11, 1);
  lcd.print("ERROR");
}

//////////Setting the lighting to trigger the relay/////
if (flagMenu == 4) {
  lcd.setCursor(0, 0);
  lcd.print("LIGHT ON/OFF  ");
  lcd.setCursor(0, 1);
  lcd.write(byte(2));
  lcd.setCursor(1, 1);

```

```

lcd.print("VALUE:");
lcd.setCursor(7, 1);
lcd.print(lightMin);
lcd.setCursor(10, 1);
lcd.print("%");
lcd.setCursor(15, 1);
lcd.write(byte(1));

if (butLeft.click()) {
  lightMin--;
  if (lightMin <= 0) lightMin = 0;
}
if (butRight.click()) {
  lightMin++;
  if (lightMin >= 100) lightMin = 100;
}

if (Serial.available() > 0) {
  int a = Serial.read();

  if(a == 'a') {irsend.sendNEC(0xFFB97F, 32); delay(40);} //Brightness+
  if(a == 'b') {irsend.sendNEC(0xFF11CF, 32); delay(40);} //Brightness-
  if(a == 'c') {irsend.sendNEC(0xFF718F, 32); delay(40);} //On
  if(a == 'd') {irsend.sendNEC(0xFFD92F, 32); delay(40);} //Off

  if(a == 'e') {irsend.sendNEC(0xFF817F, 32); delay(40);} //Red
  if(a == 'f') {irsend.sendNEC(0xFF09DF, 32); delay(40);} //Green
  if(a == 'g') {irsend.sendNEC(0xFF61BF, 32); delay(40);} //Blue
  if(a == 'h') {irsend.sendNEC(0xFFE91F, 32); delay(40);} //White
}

```

```
if(a == 'i') {irsend.sendNEC(0xFFA64F, 32); delay(40);}
if(a == 'j') {irsend.sendNEC(0xFF36DF, 32); delay(40);}
if(a == 'k') {irsend.sendNEC(0xFF732F, 32); delay(40);}
if(a == 'l') {irsend.sendNEC(0xFFE37F, 32); delay(40);}

if(a == 'q') {irsend.sendNEC(0xFF3477, 32); delay(40);}
if(a == 'r') {irsend.sendNEC(0xFF47A7, 32); delay(40);}
if(a == 's') {irsend.sendNEC(0xFF93B7, 32); delay(40);}
if(a == 't') {irsend.sendNEC(0xFFD347, 32); delay(40);}

if(a == 'u') {irsend.sendNEC(0xFF3577, 32); delay(40);} //Cold colour
if(a == 'v') {irsend.sendNEC(0xFF34D7, 32); delay(40);} //Neutral colour
if(a == 'w') {irsend.sendNEC(0xFF64A7, 32); delay(40);} //Warm colour
}
}
```