

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ «ФАХОВИЙ КОЛЕДЖ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

КУРСОВИЙ ПРОЄКТ

на тему:

«Виготовлення на платі Arduino парктроніка для визначення “сліпої зони” до 10см з виведенням інформації на OLED- дисплей, світлодіод і зумер з використанням лазерного дальноміра.»

Виконав студент групи ІП-42

Назар СУХОДОЛЬСЬКИЙ

Керівник проєкту:

Остап ЮНАК

Курсовий проєкт перевірений

і допущений до захисту

“ ” _____ 2025 р.

Курсовий проєкт при захисті оцінений

Львів 2025

**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«ФАХОВИЙ КОЛЕДЖ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**

ЗАТВЕРДЖЕНО

на засіданні циклової комісії

«Фундаментальної підготовки»

Протокол № ____ від _____ 2025 р.
Голова комісії

_____ Богдан ПЕЛЕЩАК

ЗАВДАННЯ

на курсовий проєкт

Суходольському Назару Васильовичу

(прізвище, ім'я та по батькові)

з навчальної дисципліни: ПРОГРАМНО-АПАРАТНІ ЗАСОБИ INTERNET-РЕЧЕЙ

Студент групи: **ІІІ-42**

1. Тема проєкту: Виготовлення на платі Arduino парктроніка для визначення “сліпої зони” до 10см з виведенням інформації на OLED- дисплей, світлодіод і зумер з використанням лазерного дальноміра.

2. Дата видачі завдання: _____ ”14” жовтня 2025 р

3. Термін здачі курсового проєкту: _____ ”09” грудня 2025 р.

4. Вихідні дані до проєкту:

4.1. Провести аналіз предметної галузі і вимог (огляд аналогів, функціональні й нефункціональні вимоги).

4.2. Скласти технічне завдання (ТЗ) з переліком функцій, інтерфейсів і обмежень(див. ДОДАТОК А).

4.3 Підібрати WOM (Bill of Materials) для виконання проєкту (див. ДОДАТОК Б).

4.4 Розробити таблицю з'єднань пінів (Pin Connection Table) використаних компонентів для реалізації пристрою (див. ДОДАТОК В).

4.5 Реалізувати прошивку на Arduino (коментарі в коді, структура проекту, використані бібліотеки).

4.6. Зробити на монтажній платі безпечної монтажу прототип розробленого пристрою на базі Arduino.

4.7. Провести налаштування, калібрування і тестування.

4.8. Підготувати користувацьку інструкцію та технічний звіт (див. ДОДАТОК Г).

5 Перелік обов’язкових демонстраційних креслень:

5.1 Принципова електрична схема підключення

6. Склад розрахунково – пояснювальної записки (перелік питань до розробки):

ВСТУП

1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ

2 ТЕХНІЧНЕ ЗАВДАННЯ

3 АПАРАТНА ЧАСТИНА

4 ПРОГРАМНА ЧАСТИНА

5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

ВИСНОВКИ

ПЕРЕЛІК ПОСИЛАНЬ

Календарний план

Назва етапів	Термін виконання	Примітка
Вступ		
1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ		
2 ТЕХНІЧНЕ ЗАВДАННЯ		
3 АПАРАТНА ЧАСТИНА		
4 ПРОГРАМНА ЧАСТИНА		
5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ		
Висновки		
Перелік посилань		

Студент

(підпис)

Назар СУХОДОЛЬСЬКИЙ

(імя та прізвище)

Керівник проекту

(підпис)

Остап ЮНАК

(імя та прізвище)

ЗМІСТ

ВСТУП.....	3
1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ.....	5
1.1 Мова програмування C++ та середовище Arduino IDE.....	5
1.2 Апаратно-програмна платформа Arduino.....	6
1.3 Принципи роботи використаних сенсорів.....	6
2 ТЕХНІЧНЕ ЗАВДАННЯ.....	9
2.1 Назва роботи.....	9
2.2 Мета роботи.....	9
2.3 Завдання.....	9
2.4 Технічні вимоги.....	10
2.5 Очікуваний результат	11
3 АПАРАТНА ЧАСТИНА.....	12
3.1 Підбір ВОВ.....	12
4 ПРОГРАМНА ЧАСТИНА.....	14
4.1 Опис алгоритму роботи.....	14
4.2 Код програми.....	15
5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ.....	17
5.1 Підготовка до тестування.....	17
5.2 Процес тестування.....	18
5.3 Результати.....	18
5.4 Висновки.....	19
ВИСНОВКИ.....	21
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	23
ДОДАТОК 1.....	24

ВСТУП

Мета роботи:

Метою курсового проєкту є розробка та виготовлення мікроконтролерного парктроніка на базі платформи Arduino, призначеного для визначення «сліпої зони» на відстані до 10 см із виведенням інформації на OLED-дисплей, світлодіодну індикацію та зумер, із використанням лазерного дальноміра.

У межах курсового проєкту передбачається проєктування апаратної частини пристрою, підбір та підключення електронних компонентів, розробка програмного забезпечення для зчитування та обробки даних дальноміра, а також реалізація системи візуального і звукового сповіщення користувача. Завершальним етапом є тестування працездатності парктроніка в різних умовах експлуатації та оцінка точності вимірювань.

Актуальність:

Актуальність даного курсового проєкту зумовлена зростаючою потребою у підвищенні безпеки під час паркування транспортних засобів. «Сліпі зони» є однією з основних причин незначних дорожньо-транспортних пригод і пошкоджень автомобілів під час маневрування на малих швидкостях.

Використання мікроконтролерної платформи **Arduino** дозволяє створити доступний, компактний і надійний парктронік із високою точністю вимірювання відстані завдяки застосуванню **лазерного дальноміра**. Виведення інформації на OLED-дисплей у поєднанні зі світловою та звуковою індикацією забезпечує оперативне інформування водія про небезпечне наближення до перешкод, що суттєво підвищує безпеку та комфорт керування транспортним засобом.

Технологічна та Інженерна Актуальність:

— **Поширеність IoT (Internet of Things):** Сучасні системи автоматизації, моніторингу та управління критично залежать від компактних, економічних та енергоефективних пристроїв. Платформа Arduino є де-факто стандартом для швидкого прототипування та створення пристроїв, що взаємодіють з фізичним світом, що відповідає глобальному тренду розвитку Інтернету речей;

— **Демократизація розробки:** Arduino, як відкрита (Open-Source) апаратно-програмна платформа, значно знижує поріг входу для інженерів-початківців та сприяє створенню інновацій. Це дозволяє зосередитися на логіці проєкту та функціональності, а не на низькорівневому програмуванні та складному апаратному дизайні;

— **Міждисциплінарний характер:** Розробка апаратно-програмного пристрою вимагає інтеграції знань з електроніки, програмування мікроконтролерів (C/C++), теорії керування та проєктування інтерфейсів. Це забезпечує набуття комплексних інженерних навичок, які є критично важливими для працевлаштування у сферах робототехніки, автоматизації та вбудованих систем;

Прикладна та Практична Актуальність

— **Вирішення конкретної прикладної задачі:** Робота не обмежується теоретичним аналізом, а передбачає створення працюючого фізичного прототипу (моніторинг параметрів, автоматизоване управління, збір даних). Це безпосередньо демонструє здатність автора перетворювати теоретичні знання на практичне інженерне рішення;

— **Економічна ефективність:** Використання Arduino дозволяє створити функціональний пристрій з мінімальними витратами на компоненти та часом розробки. Це важливий аспект для інноваційних стартапів та освітніх проєктів;

— **Можливість подальшого розвитку:** Розроблений прототип слугує надійною базою для подальшого масштабування, модернізації та переходу на промислові мікроконтролери, що забезпечує наукову перспективу для дипломної роботи чи комерціалізації.

Таким чином, розробка апаратно-програмного пристрою на базі Arduino є своєчасним, економічно обґрунтованим і високопрактичним дослідженням, що поєднує вивчення сучасних апаратних засобів та формування ключових інженерних компетенцій.

1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ

1.1 Апаратно-програмна платформа Arduino

Arduino — це відкрита (open-source) платформа для створення прототипів електронних пристроїв, яка базується на простому у використанні апаратному та програмному забезпеченні. Платформа складається з фізичної друкованої плати (мікроконтролера) та середовища розробки (IDE) для написання коду.

Рис. 1.1 Вигляд плати Arduino з мікроконтролером ATmega328P2

Проект Arduino був започаткований у 2005 році в Інституті проєктування взаємодії (Interaction Design Institute Ivrea) в місті Івреа, Італія. Його засновниками стали Массімо Банці (Massimo Banzi), Девід Куартільєс, Том Іго, Джанлука Мартіно та Девід Мелліс. Основною метою створення платформи було забезпечення студентів дешевим та простим інструментом для створення інтерактивних проєктів, який би не вимагав глибоких знань у схемотехніці.

Для даної курсової роботи обрано плату Arduino Uno R3, яка побудована на базі мікроконтролера ATmega328P2. Її популярність зумовлена наступними факторами:

- **Доступність:** Низька вартість компонентів;
- **Кросплатформеність:** Середовище розробки працює на Windows, macOS та Linux;
- **Простота:** Зручний інтерфейс програмування через USB без необхідності використання зовнішніх програматорів
- **Відкрита архітектура:** Схеми плат знаходяться у вільному доступі, що дозволяє легко інтегрувати різноманітні датчики та модулі;

1.2 Мова програмування C++ та середовище Arduino IDE

Програмування мікроконтролера здійснюється мовою C++. Середовище розробки Arduino IDE використовує спеціальний спрощений діалект (фреймворк) під назвою **Wiring**.

Фреймворк — це набір готових інструментів, бібліотек, шаблонів та правил, який надає структуру для створення програмного забезпечення, спрощуючи та прискорюючи процес розробки. Він допомагає розробникам уникнути написання стандартного коду з нуля, дозволяючи зосередитися на унікальній бізнес-логіці продукту. Фреймворки використовуються для різних типів проєктів, включаючи веб, мобільні та десктопні додатки.

C++ — це універсальна мова програмування високого рівня, яка підтримує об'єктно-орієнтовану парадигму. В контексті розробки під Arduino, C++ дозволяє ефективно керувати пам'яттю мікроконтролера та працювати з апаратними регістрами, зберігаючи при цьому зрозумілий синтаксис.

Програма для Arduino називається «скетчем» (sketch) і має дві обов'язкові функції:

- `setup()` — виконується один раз при запуску для налаштування конфігурації пінів;
- `loop()` — виконується циклічно, реалізуючи основний алгоритм роботи пристрою.

1.3 Принципи роботи використаних сенсорів

Лазерний дальномір VL53L0X (Time-of-Flight Sensor). Принцип дії лазерного дальноміра VL53L0X базується на технології Time of Flight (ToF) — вимірювання часу проходження світлового імпульсу. Датчик випромінює короткі інфрачервоні лазерні імпульси у напрямку об'єкта та фіксує час, за який відбите світло повертається назад до приймача.

На основі вимірюного часу затримки контролер датчика обчислює відстань до перешкоди. Такий принцип забезпечує високу точність вимірювання, стабільну роботу незалежно від освітлення та можливість вимірювання малих відстаней, що є критично важливим для виявлення «сліпої зони» до 10 см.

Рис. 1.2 – Вигляд лазерного дальноміра VL53L0X

Рідкокристалічний дисплей (LCD 1602) з I2C. Рідкокристалічний дисплей **LCD 1602** призначений для відображення текстової та числової інформації. Він має дві строки по 16 символів і підключається до плати Arduino через **I2C-інтерфейс**, що дозволяє використовувати лише дві лінії зв'язку — SDA та SCL.

Принцип роботи дисплея базується на властивостях **рідких кристалів змінювати поляризацію світла під дією електричного поля**. Усередині дисплея розташована матриця сегментів, кожен із яких керується електричним сигналом. При подачі напруги сегмент змінює прозорість і формує символ на екрані.

У парктроніку LCD 1602 використовується для відображення поточної відстані до перешкоди, повідомлень попередження та режимів роботи пристрою.

Рис 1.3. Вигляд рідкокристалічного дисплею (LCD 1602) з I2C

Зумер (Buzzer). Зумер є електроакустичним перетворювачем, який перетворює електричні сигнали у звукові коливання. Він може працювати у активному або пасивному режимі.

У даному проєкті зумер використовується для звукового сповіщення користувача. Частота та тривалість звукових сигналів змінюється залежно від відстані до перешкоди: чим менша відстань — тим частіший та інтенсивніший сигнал.

Рис. 1.4 – Вигляд зумера

2 ТЕХНІЧНЕ ЗАВДАННЯ

2.1 Назва роботи

Виготовлення на платі Arduino парктроніка для визначення “сліпої зони” до 10см з виведенням інформації на OLED- дисплей, світлодіод і зумер з використанням лазерного дальноміра.

2.2 Мета роботи

Розробка та виготовлення мікроконтролерного парктроніка на базі платформи Arduino, призначеного для визначення «сліпої зони» на відстані до 10 см із виведенням інформації на OLED-дисплей, світлодіодну індикацію та зумер, із використанням лазерного дальноміра.

2.3 Завдання

1. Ознайомитися з принципами роботи та технічними характеристиками основних компонентів, що будуть використовуватися у проєкті:
 - плати Arduino Uno R3;
 - лазерного дальноміра VL53L0X;
 - рідкокристалічного дисплея LCD 1602 з I2C-інтерфейсом;
 - світлодіода;
 - зумера.
2. Розробити електричну схему підключення всіх компонентів до плати Arduino, забезпечивши коректний розподіл пінів, стабільне живлення та надійну роботу пристрою.
3. Реалізувати програмне забезпечення для Arduino, яке забезпечує:
 - вимірювання відстані до перешкод за допомогою лазерного дальноміра;
 - обробку отриманих даних та визначення небезпечної зони (до 10 см);
 - виведення інформації на LCD-дисплей;
 - формування світлової та звукової індикації при небезпечному наближенні.

4. Перевірити працездатність розробленого парктроніка в тестових умовах, зафіксувавши коректність вимірювання відстані та спрацьовування індикаторів.
5. Оцінити ефективність роботи системи та запропонувати можливі напрямки її вдосконалення для підвищення точності, надійності та зручності використання.

2.4 Технічні вимоги

- напруга живлення системи — **5 В постійного струму**;
- мікроконтролерна плата **Arduino Uno R3**;
- **безпайкова макетна плата (Breadboard)** для збирання та тестування схеми;
- **лазерний дальномір VL53L0X** — для вимірювання відстані до перешкод;
- **рідкокристалічний дисплей LCD 1602 з I2C-інтерфейсом** — для виведення інформації про поточну відстань та попередження;
- **резистори номіналом 220 Ом** — для обмеження струму світлодіодів;
- **світлодіоди (зелений та червоний)** — для візуальної індикації;
- **п'єзозумер** — для звукового сповіщення;
- **з'єднувальні провідники типу «тато–тато» та «тато–мама»** — для монтажу схеми;
- **джерело живлення 5 В** із струмом не менше 1 А — для стабільної роботи всієї системи.

2.5 Очікуваний результат

Очікуваним результатом виконання курсового проєкту є створення працездатного парктроніка на базі плати **Arduino Uno R3**, який забезпечує безперервний контроль «сліпої зони» перед транспортним засобом за допомогою лазерного дальноміра **VL53L0X**.

У штатному режимі система перебуває у стані очікування та здійснює постійний вимір відстані до можливих перешкод. При безпечній відстані активним є зелений світлодіод, а на рідкокристалічному дисплеї **LCD 1602** відображається поточне значення відстані.

При зменшенні відстані до об'єкта до критичного значення (до 10 см) мікроконтролер автоматично переходить у режим попередження: вмикається червоний світлодіод, активується звуковий сигнал п'єзозумера, а на дисплеї виводиться попереджувальне повідомлення. Частота та інтенсивність звукового сигналу зростає зі зменшенням відстані до перешкоди, що дозволяє водію оперативно реагувати на небезпечне наближення.

Після віддалення об'єкта на безпечну відстань система автоматично повертається у штатний режим, що забезпечує безперервну та надійну роботу парктроніка.

3 АПАРАТНА ЧАСТИНА

3.1 Підбір BOM

BOM (скорочення від Bill of Materials)- це список матеріалів і компонентів, потрібних для складання пристрою або виготовлення виробу, який може представлений як таблиця всіх деталей, які потрібно для проєкту.

Таблиця 3.1.1 Перелік необхідних компонентів

№	Назва компонента	Кількість	Позначення на схемі	Характеристики	Примітка
1.	Arduino Uno	1	U1	Мікроконтролер ATmega328P	Основна плата керування
2.	Плата breadboard	1	-	830 точок, безпайкова	Безпайна плата для створення конструкції
3.	Лазерний дальномір VL53L0X	1	DIST1	ToF, I2C, 3–5 В, до 2 м	Вимірювання відстані
4.	LCD 1602 (I2C)	1	U2	16x2 символів, I2C інтерфейс	Відображення статусу системи
5.	Світлодіод (Червоний)	1	D1_RED	5 мм, 2 В, 20 мА	Індикація тривоги
6.	Світлодіод (Зелений)	1	D1_GREEN	5 мм, 2 В, 20 мА	Індикація режиму "Охорона"
7.	П'єзозумер	1	SP1	Активний, 5 В	Звукове оповіщення
8.	Резистори	2	R1, R2	220 Ом, 0.25 Вт	Обмеження струму LED
9.	Провідники	20	-	Dupont (Male-Male, Male-Female)	Електричні з'єднання
10.	Джерело живлення 5 В	1	PWR1	5 В, ≥ 1 А	Живлення всієї системи

Таблиця 3.1.2 Таблиця з'єднань пінів

№	Компонент	Вивід компонента	З'єднано з піном Arduino	Примітка / Функція
1	Лазерний дальномір VL53L0X	VIN	5V	Живлення
		GND	GND	Земля
		SDA	Pin A4 (SDA)	Шина даних I2C
		SCL	Pin A5 (SCL)	Шина тактування I2C
3	LCD 1602 (I2C)	GND	GND	Земля
		VCC	5V	Живлення
		SDA	Pin A4 (або SDA)	Шина даних I2C
		SCL	Pin A5 (або SCL)	Шина тактування I2C
4	Світлодіод (Зелений)	Анод (+)	Digital Pin 4	Через резистор 220 Ом
		Катод (-)	GND	
5	Світлодіод (Зелений)	Анод (+)	Digital Pin 5	Через резистор 220 Ом
		Катод (-)	GND	
6	П'єзозумер	Плюс (+)	Digital Pin 6	Звуковий сигнал
		Мінус (-)	GND	Земля
7	Arduino Uno R3	5V	Джерело живлення	Живлення системи
		GND	GND	Спільна земля

4 ПРОГРАМНА ЧАСТИНА

Розробка програмного забезпечення для мікроконтролера здійснювалася в інтегрованому середовищі розробки Arduino IDE. Мова програмування — C++ з використанням стандартних бібліотек Wiring та спеціалізованих бібліотек для роботи з периферією (Wire.h, LiquidCrystal_I2C.h).

Алгоритм роботи парктроніка полягає у **безперервному моніторингу простору перед транспортним засобом** та реагуванні на наближення об'єктів. Основна логіка передбачає вимірювання відстані до перешкод за допомогою **лазерного дальноміра VL53L0X**, візуальне та звукове інформування користувача, а також відображення даних на **LCD 1602**.

4.1 Опис алгоритму роботи

1. Підключення бібліотек та оголошення констант і змінних
 - Підключаються бібліотеки для роботи з I2C (для VL53L0X та LCD), управління світлодіодами та п'єзозумером.
 - Оголошуються константи для пінів Arduino, значення критичної відстані (наприклад, 10 см) та змінні для зберігання результатів вимірювань.
2. Налаштування портів та ініціалізація пристроїв (setup)
 - Встановлюються режими пінів (вхід/вихід) для світлодіодів і зумера.
 - Ініціалізуються лазерний дальномір і LCD-дисплей.
 - Встановлюється початковий стан індикаторів (зелений світлодіод активний, червоний та звуковий сигнал вимкнені).
3. Основний цикл виконання (loop)
 - Виконується безперервне зчитування відстані від лазерного дальноміра.
 - Якщо відстань більша за критичне значення, система перебуває у штатному режимі: зелений світлодіод горить, на дисплеї відображається повідомлення «Безпечно».

- Якщо відстань зменшується до критичної (≤ 10 см), активується режим «Тривога»: вмикається червоний світлодіод, п'єзозумер подає звуковий сигнал, а на дисплеї відображається повідомлення «Небезпека».
- Система постійно оновлює інформацію на дисплеї та індикацію, забезпечуючи безперервний контроль зони.
- Після зникнення об'єкта відстань перевищує критичне значення — система повертається у штатний режим

.4.2 Код програми

```
#include <Wire.h>
#include <Adafruit_VL53L0X.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);
Adafruit_VL53L0X lox = Adafruit_VL53L0X();

const int greenLedPin = 4;
const int redLedPin = 5;
const int buzzerPin = 6;

const int safeDistance = 10; // Критична відстань в см

void setup() {
  Serial.begin(9600);
  pinMode(greenLedPin, OUTPUT);
  pinMode(redLedPin, OUTPUT);
  pinMode(buzzerPin, OUTPUT);
  lcd.init();
  lcd.backlight();
  lcd.clear();
}
```



```

lcd.setCursor(0, 0);
lcd.print("Парктронік v1");
lcd.setCursor(0, 1);
lcd.print("Стан: Очікування");

if (!lox.begin()) {
  Serial.println(F("VL53L0X не знайдено!"));
  lcd.setCursor(0, 1);
  lcd.print("Датчик помилка!");
  while(1);
}
digitalWrite(greenLedPin, HIGH);
digitalWrite(redLedPin, LOW);
digitalWrite(buzzerPin, LOW);
}

void loop() {
  VL53L0X_RangingMeasurementData_t measure;

  lox.rangingTest(&measure, false); // Зчитування відстані

  int distance = measure.RangeMilliMeter / 10; // Переведення в см
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  if (distance > safeDistance || distance == 0) {
    // Безпечна зона
    digitalWrite(greenLedPin, HIGH);
    digitalWrite(redLedPin, LOW);
    digitalWrite(buzzerPin, LOW);
  }
}

```

```
    lcd.setCursor(0, 1);  
    lcd.print("Стан: Безпечно ");  
    lcd.setCursor(12, 1);  
    lcd.print("  "); // Очистка старих значень  
  } else {  
    // Небезпечна зона  
    digitalWrite(greenLedPin, LOW);  
    digitalWrite(redLedPin, HIGH);  
  
    // Звукове попередження  
    tone(buzzerPin, 1000); // Частота 1 кГц  
    delay(100);  
    noTone(buzzerPin);  
    delay(100);  
  
    // Вивід інформації на LCD  
    lcd.setCursor(0, 1);  
    lcd.print("Стан: Небезпека");  
    lcd.setCursor(12, 1);  
    lcd.print(distance);  
    lcd.print("cm");  
  }  
  
  delay(200); // Затримка між вимірюваннями  
}
```

5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

Перевірка працездатності розробленого парктроніка проводилася у два етапи. Спочатку оцінювався монтаж апаратної частини — правильність підключення компонентів до плати Arduino, надійність монтажу на макетній платі та стабільність живлення датчиків і виконавчих елементів.

Потім виконувалося функціональне тестування програмного забезпечення з імітацією тривожних ситуацій: перевірялося вимірювання відстані лазерним дальноміром, відображення даних на LCD-дисплеї, спрацьовування світлової та звукової індикації при наближенні об'єкта до критичної відстані і автоматичне повернення системи у штатний режим після зникнення перешкоди.

5.1 Підготовка до тестування

Перед подачею живлення було проведено візуальний огляд макетної плати для виявлення можливих помилок монтажу: перевірено правильність підключення датчиків HC-SR501 (PIR) та HC-SR04 (ультразвуковий сенсор) щодо VCC та GND, коректність підключення GSM/GPRS модуля (живлення, TX та RX через SoftwareSerial), правильність підключення I2C-модуля LCD дисплея (SDA та SCL), полярність світлодіодів із наявністю струмообмежувальних резисторів, а також надійність з'єднань провідників на breadboard для запобігання обривам або контактним помилкам.

5.2 Процес тестування

Після підключення плати Arduino Uno до ПК через USB було виконано компіляцію та завантаження скетчу в пам'ять мікроконтролера. Тестування системи проводилося у двох основних режимах:

Режим «Безпечно» (SAFE):

При відсутності об'єктів у зоні контролю лазерний дальномір показував відстань понад критичну, зелений світлодіод був активним, червоний світлодіод та п'єзозумер вимкнені. На LCD 1602 відображалася поточна відстань та

повідомлення «Безпечно». Система стабільно перебувала у режимі очікування без хибних спрацьовувань.

Режим «Небезпека» (ALARM):

Імітацією наближення об'єкта до критичної відстані (≤ 10 см) активувався червоний світлодіод та п'єзозумер, а на дисплеї з'являлося повідомлення «Небезпека!» із відображенням поточної відстані. Після віддалення об'єкта система автоматично поверталася у режим «Безпечно», і зелений світлодіод знову сигналізував про безпеку.

Додаткові перевірки:

- перевірено коректність відображення відстані лазерного дальноміра;
- підтверджено стабільність роботи світлової та звукової індикації;
- проведено тривале тестування для виключення перегріву або помилок у роботі компонентів.

5.3 Результати

Під час випробувань встановлено, що PIR-датчик коректно фіксує рух на відстані до 3 метрів, ультразвуковий датчик забезпечує вимірювання відстані з похибкою не більше 1–2 см, а інформація на LCD дисплеї відображається чітко і контрастно.

Логіка перемикавання світлодіодів (червоний/зелений) повністю відповідає стану системи. Збоїв у роботі мікроконтролера при одночасній роботі датчиків, дисплея та GSM-модуля не виявлено. GSM-модуль своєчасно надсилає SMS-повідомлення при спрацьовуванні тривоги, причому повторні сповіщення надсилаються лише після відновлення режиму «Безпечно».

5.4. Висновки

Результати випробувань підтверджують, що розроблений парктронік на базі Arduino Uno працює стабільно та відповідає поставленим технічним вимогам. Забезпечено коректну взаємодію лазерного дальноміра з LCD-дисплеєм, світловою та звуковою індикацією.

Система демонструє надійну роботу у режимах «Безпечно» та «Небезпека», точно визначає відстань до об'єктів та готова до використання як ефективний автономний пристрій для контролю «сліпої зони».

ВИСНОВКИ

У ході виконання курсового проєкту було успішно розроблено та реалізовано макет парктроніка на базі плати **Arduino Uno** для визначення «сліпої зони» до 10 см.

Апаратна реалізація: Опрацьовано принципи побудови систем дистанційного контролю з використанням різнотипних сенсорів. Реалізовано спільну роботу лазерного дальноміра VL53L0X для визначення відстані до об'єкта, світлодіодів для індикації режимів та п'єзозумера для звукового оповіщення. Підключення LCD 1602 через I2C дозволило спростити схему та заощадити цифрові порти мікроконтролера.

Програмна реалізація: Розроблено програмний алгоритм мовою C++, який забезпечує безперервний моніторинг відстані, коректне відображення інформації на дисплеї та своєчасну активацію світлової і звукової індикації при наближенні об'єкта. Використання бібліотек **Wire** та **LiquidCrystal_I2C** спростило роботу з дисплеєм і реалізацію логіки системи.

Працездатність системи: Тестування підтвердило повну відповідність технічним вимогам. Пристрій коректно переходить у режим «Небезпека» при наближенні об'єкта, відображає точну відстань на дисплеї та супроводжує подію світловою і звуковою сигналізацією. Після зникнення об'єкта система автоматично повертається у режим «Безпечно».

Навчальна цінність: Виконання роботи дозволило закріпити знання з основ електроніки та схемотехніки, особливостей підключення навантажень через резистори, роботу з протоколами I2C, а також поглибити навички програмування Arduino та налагодження апаратно-програмних систем.

Перспективи розвитку:

Розроблений пристрій є базовим прототипом, який має значний потенціал для модернізації та інтеграції у системи «Розумний дім» або комерційну сигналізацію. До основних напрямів подальшого вдосконалення можна віднести:

- дистанційне сповіщення через GSM- або Wi-Fi-модулі для надсилання SMS-повідомлень чи повідомлень у месенджер при спрацюванні сигналізації;

- контроль доступу з використанням матричної клавіатури або RFID-зчитувача для постановки/зняття об'єкта з охорони;
- підвищення енергонезалежності за рахунок літій-іонних акумуляторів та оптимізації програмного енергоспоживання;
- логування подій з використанням модуля RTC та SD-карти для запису історії спрацювань;
- конструктивне вдосконалення шляхом перенесення схеми на друковану плату та розробки корпусу методом 3D-друку для захисту електроніки від зовнішніх впливів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Блум Д. Вивчаємо Arduino: інструменти та методи технічного чарівництва / Джеремі Блум. — К.: БХВ, 2016. — 336 с.
2. Шилдт Г. С++: базовий курс / Герберт Шилдт. — 3-тє вид. — М.: Вільямс, 2019. — 624 с.
3. Петренко А. І. Основи мікропроцесорної техніки та програмування мікроконтролерів: навчальний посібник / А. І. Петренко. — К.: Каравела, 2020. — 288 с.
4. Рюмик С. М. Мікроконтролери AVR: практикум для початківців / С. М. Рюмик. — Х.: Ранок, 2018. — 112 с.
5. Галкін В. І. Промислова електроніка та мікросхемотехніка: підручник / В. І. Галкін. — К.: Вища школа, 2017. — 320 с.
6. Alvins, B. Mastering Arduino Programming (2021). *Призначення: Програмування мікроконтролерів, синтаксис C/C++.*
7. Schwartz, M. Arduino Electronics Blueprints (2019). *Призначення: Інтеграція компонентів, використання бібліотек, створення скетчів.*
8. Axelson, D. Embedded Systems Design (2020). *Призначення: Фундаментальні основи проєктування вбудованих систем, архітектура.*
9. Marr, B. Internet of Things (IoT) in Business (2022). *Призначення: Концепції IoT, мережеві протоколи та стандарти для пристроїв.*

Принципова електрична схема підключення