

**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.»

Виконала студентка групи ІІІ-41

Христина ОМЕЛЮХ

Керівник проєкту:

Остап ЮНАК

Курсовий проєкт перевірений

і допущений до захисту

“ ____ ” _____ 2025 р.

Курсовий проєкт при захисті оцінений

Львів 2025

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)

Керівник проекту

_____ (підпис)

Христина ОМЕЛЮХ

_____ (імя та прізвище)

Остап ЮНАК

_____ (імя та прізвище)

ЗМІСТ

ВСТУП	3
1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ	5
2 ТЕХНІЧНЕ ЗАВДАННЯ	9
3 АПАРАТНА ЧАСТИНА	11
4 ПРОГРАМНА ЧАСТИНА	14
5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ	20
ВИСНОВКИ	23
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	25
ДОДАТКИ	26

ВСТУП

Метою роботи є створення апаратно-програмного пристрою на основі платформи Arduino, здатного виконувати запланований набір функцій. Проект передбачає розробку технічної та конструкторської документації, а також написання детальної інструкції з експлуатації, що дозволить ефективно використовувати розроблений пристрій. Актуальність теми обумовлена швидким зростанням попиту на інтеграцію апаратного та програмного забезпечення в сучасних технологічних рішеннях. Зокрема, мікроконтролери, такі як Arduino, стають усе популярнішими не лише в промисловості, а й у сфері освіти. Це підтверджує важливість досліджень у цьому напрямку. Технологічна та інженерна значущість проекту ґрунтується на кількох ключових аспектах:

1. Розвиток Інтернету речей: Сучасні системи автоматизації, моніторингу й управління орієнтовані на використання компактних, економічних і енергоощадних пристроїв. Arduino, ставши фактичним стандартом для швидкого створення прототипів, ідеально відповідає світовим тенденціям у розвитку Інтернету речей.

2. Демократизація інженерної розробки: Завдяки відкритому характеру платформи Arduino, процес входження в інженерну діяльність спрощується для початківців, даючи можливість реалізувати інноваційні ідеї доступно й ефективно.

3. Міждисциплінарний підхід: Створення апаратно-програмного пристрою вимагає об'єднання знань у галузях електроніки, програмування (C/C++), теорії управління та розробки інтерфейсів. Така інтеграція сприяє формуванню широкої технічної підготовки, необхідної в робототехніці, автоматизації та вбудованих системах.

Прикладна і практична цінність проекту полягає у наступному:

1. Виконання конкретного завдання: Основний акцент зроблений на створення повнофункціонального прототипу для вирішення практичних задач у моніторингу, автоматизації чи збору даних. Це демонструє здатність застосовувати отримані теоретичні знання на практиці.

2. Економічна ефективність: Використання платформи Arduino дозволяє з мінімальними витратами часу та ресурсів створювати функціональні пристрої. Такий підхід є цінним для освітніх програм і стартапів, орієнтованих на інновації.

3. Перспективи розвитку: Створений прототип може стати основою для подальшого масштабування проекту, його модернізації або адаптації до промислових стандартів. Це значно підвищує потенціал розробки як для наукових досліджень, так і для майбутньої комерціалізації.

Таким чином, розробка апаратно-програмного пристрою на основі платформи Arduino є актуальним, економічно виправданим і практично значущим дослідженням. Воно сприяє вивченню сучасних апаратно-програмних технологій і формуванню важливих інженерних компетенцій, які мають вирішальне значення в сучасній технічній сфері.

1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ

1.1 Апаратно-програмна платформа Arduino UNO

Arduino є багатофункціональною платформою з відкритим кодом, створеною для проєктування та тестування електронних пристроїв. Вона об'єднує в собі просте у використанні апаратне забезпечення та інтуїтивно зрозуміле програмне середовище.

Основу платформи становлять фізична плата з мікроконтролером та інтегроване середовище розробки (IDE), яке забезпечує написання й завантаження програмного коду.

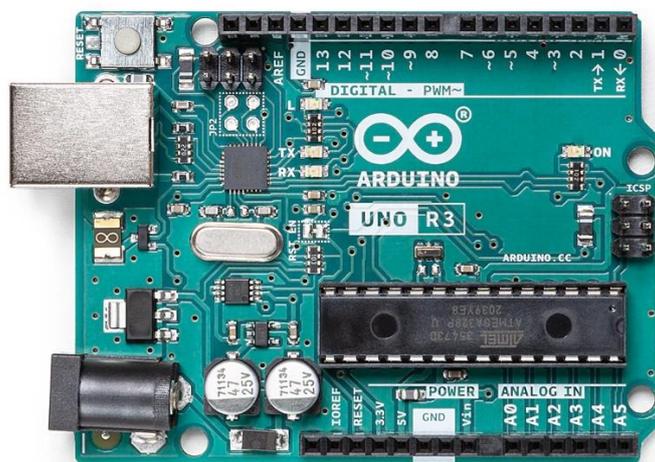


Рис. 1.1 Вигляд плати Arduino з мікроконтролером ATmega328P2

Історія створення: проєкт Arduino з'явився у 2005 році в Інституті проєктування взаємодії, що знаходиться в Івреа, Італія. Його засновниками стали Массімо Банці, Девід Куартільєс, Том Іго, Джанлука Мартіно та Девід Мелліс.

Основною метою створення цієї платформи було забезпечити студентів доступним і простим у використанні інструментом для реалізації інтерактивних проєктів, який не потребував би поглиблених знань у галузі схемотехніки.

Причини вибору платформи:

Моя курсова робота зосереджена на використанні плати Arduino Uno R3, створеної на базі мікроконтролера ATmega328P2. Її широке застосування пояснюється кількома важливими перевагами: - Економічність: Невисока вартість компонентів робить її доступною для багатьох користувачів.

- Кросплатформність: Інтегроване середовище розробки підтримує Windows, macOS і Linux, забезпечуючи гнучкість у виборі операційної системи.

- Зручність використання: Інтуїтивно зрозумілий інтерфейс дозволяє програмувати плату через USB, обходячи необхідність зовнішнього програматора.

- Відкрита архітектура: Схеми пристрою знаходяться у відкритому доступі. Це сприяє легкій інтеграції різноманітних модулів і сенсорів, розширюючи функціональні можливості плати.

1.2 Мова програмування C++ та середовище Arduino IDE

Програмування мікроконтролера здійснюється за допомогою мови C++ у середовищі розробки Arduino IDE, яке базується на спрощеному фреймворку під назвою Wiring.

Фреймворк являє собою набір попередньо підготовлених інструментів, бібліотек, шаблонів і правил, які упорядковують процес створення програмного забезпечення. Його головна мета — спростити та прискорити розробку, виключаючи необхідність писати стандартний код вручну. Це дозволяє розробникам зосередитися на створенні унікальної логіки продукту, значно підвищуючи продуктивність.

Загалом, фреймворки активно застосовуються у розробці програмного забезпечення для веб-додатків, мобільних платформ і

настільних систем. C++ є потужною мовою високого рівня з підтримкою об'єктно-орієнтованого підходу. У середовищі Arduino вона надає можливість ефективно управляти пам'яттю мікроконтролера, взаємодіяти з апаратними регістрами і при цьому забезпечує зрозумілий та зручний синтаксис для програмування.

У середовищі Arduino створена програма називається скетчем (sketch) і повинна містити дві обов'язкові функції:

- `setup()` — виконується один раз після запуску для налаштування конфігурації пінів;
- `loop()` — викликається циклічно і реалізує основний алгоритм роботи пристрою.

1.3 Принципи роботи використаних сенсорів

Датчик руху HC-SR501 (PIR Sensor) функціонує на основі принципу виявлення змін у тепловому випромінюванні об'єктів навколишнього середовища. Пасивний інфрачервоний сенсор, або *Passive Infrared Sensor*, здатен розпізнавати коливання в інфрачервоному випромінюванні, яке властиве всім тілам із температурою вище абсолютного нуля. Усередині пристрою розташовується піроелектричний елемент, що чутливо реагує на переміщення джерел тепла, наприклад, людей. Коли теплий об'єкт проходить через сектори лінзи Френеля, рівень випромінювання в полі зору змінюється, що спричиняє створення електричного імпульсу та активацію тривожного сигналу.



Рис 1.3.1 Вигляд датчику руху HC-SR501

Ультразвуковий датчик відстані HC-SR04.

Робота датчика ґрунтується на технології ехолокації (SONAR). Пристрій включає два п'єзоелементи: випромінювач і приймач.

1. Випромінювач створює серію ультразвукових імпульсів із частотою 40 кГц.
2. Звукова хвиля відбивається від об'єкта і повертається до приймача.
3. Мікроконтролер визначає час, протягом якого сигнал пройшов шлях, і обчислює відстань до об'єкта.



Рис 1.3.2 Вигляд датчику HC-SR04

Рідкокристалічний дисплей (LCD 1602), оснащений інтерфейсом I2C, призначений для виведення символної інформації. Основною перевагою цього рішення є використання адаптера I2C (Inter-Integrated Circuit), який дозволяє здійснювати передачу даних всього через дві сигнальні лінії: SDA та SCL.



Рис 1.3.2 Вигляд рідкокристалічного дисплею (LCD 1602) з I2C

2 ТЕХНІЧНЕ ЗАВДАННЯ

2.1 Назва роботи

Виготовлення охоронного пристрою на платі Arduino з використанням датчиків HC-SR501 і HC-SR04 та виводом інформації на LCD дисплей.

2.2 Мета роботи

Виготовити охоронний пристрій на платі Arduino з використанням датчиків HC-SR501 і HC-SR04 та виводом інформації на LCD дисплей.

2.3 Завдання

- Ознайомитись з інструкцією до практичної роботи;
- Ознайомитись із принциповою електричною схемою охоронний пристрою на Arduino Uno;
- Зібрати на монтажній платі цифрову схему охоронного пристрою згідно електричної принципової схеми
- Написати скетч роботи пристрою на Arduino Uno;

2.4 Технічні вимоги

- 5 В постійного струму;
- Arduino Uno R3;
- Breadboard (безпайкова макетна плата);
- LCD дисплей 1602 з I2C модулем;
- Світлодіоди: Зелений, Червоний;
- Датчик HC-SR501;
- Датчик HC-SR04;
- Резистори 220 Ом;
- П'єзовипромінювач — для звукового сигналу;
- З'єднувальні провідники типу «тато-тато» та «тато-мама»

2.5 Очікуваний результат

Охоронна система працює за принципом постійного контролю зони спостереження за допомогою пасивного інфрачервоного датчика руху HC-SR501. У звичайному стані вона діє в режимі очікування: зелений індикатор залишається увімкненим, а на LCD-дисплеї виводиться повідомлення про те, що ситуація безпечна. Якщо пристрій виявляє теплове випромінювання від об'єкта, який рухається, мікроконтролер миттєво переводить систему в режим тривоги. Увімкнюється червоний індикатор, пролунає звуковий сигнал, а ультразвуковий сенсор HC-SR04 визначає точну дистанцію до об'єкта і відображає ці дані на екрані у режимі реального часу. Коли рух припиняється, система автоматично повертається в початковий стан очікування.

3 АПАРАТНА ЧАСТИНА

3.1 Підбір BOM

BOM (Bill of Materials) є детальним списком усіх матеріалів, комплектуючих і компонентів, необхідних для складання пристрою або виготовлення конкретного виробу. Зазвичай цей список представлено у форматі таблиці, де впорядковано всі елементи, необхідні для реалізації заданого проєкту.

Такий перелік зазвичай містить назви деталей, їх кількість, технічні характеристики та іншу важливу інформацію, яка спрощує процес закупівлі та виробництва. BOM виступає як базовий інструмент для планування ресурсів, забезпечення точності виконання проєкту і ефективної організації всіх виробничих процесів.

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	HC-SR501	1	PIR1	PIR Motion Sensor, 5 В	Виявлення руху об'єктів (тепла)
4	HC-SR04	1	DIST1	Ultrasonic Sensor, 2–400 см	Вимірювання відстані до об'єкта
5	LCD 1602 (I2C)	1	U2	16x2 символів, I2C інтерфейс	Відображення статусу системи

№	Назва компонента	Кількість	Позначення на схемі	Характеристики	Примітка
6	Світлодіод (Червоний)	1	D1_RED	5 мм, 2 В, 20 мА	Індикація тривоги
7	Світлодіод (Зелений)	1	D1_GREEN	5 мм, 2 В, 20 мА	Індикація режиму "Охорона"
8	П'єзозумер	1	SP1	Активний, 5 В	Звукове оповіщення
9	Резистори	2	R1, R2	220 Ом, 0.25 Вт	Обмеження струму LED
10	Провідники	20	-	Dupont (Male-Male, Male-Female)	Електричні з'єднання

Таблиця 3.1.2 Таблиця з'єднань пінів

№	Компонент	Вивід компонента	З'єднано з піном Arduino	Примітка / Функція
1	HC-SR04 (Ultrasonic)	VCC	5V	Живлення
		Trig	Digital Pin 9	Сигнал запуску
		Echo	Digital Pin 10	Прийом сигналу
		GND	GND	Земля
2	HC-SR501 (PIR)	VCC	5V	Живлення
		OUT	Digital Pin 7	Сигнал тривоги
		GND	GND	Земля
3	LCD 1602 (I2C)	GND	GND	Земля
		VCC	5V	Живлення

№	Компонент	Вивід компонента	З'єднано з піном Arduino	Примітка / Функція
		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	
4	П'єзозумер	Плюс (+)	Digital Pin 6	-
		Мінус (-)	GND	

4 ПРОГРАМНА ЧАСТИНА

Програмне забезпечення для мікроконтролера було створено в середовищі Arduino IDE з використанням мови програмування C++. При його розробці застосовували стандартні бібліотеки Wiring та спеціалізовані бібліотеки, зокрема Wire.h і LiquidCrystal_I2C.h, які забезпечують взаємодію з різноманітними периферійними пристроями.

Функціональна основа пристрою полягає в постійному моніторингу простору за допомогою інфрачервоного датчика руху. У разі спрацювання тривоги здійснюється вимірювання відстані до виявленого об'єкта. Отримані дані виводяться на LCD-екран і супроводжуються звуковим сигналом, що привертає увагу до ситуації.

Програмний код структурований у три головні етапи: підключення та оголошення бібліотек і констант, налаштування портів і виконання ініціалізації, а також основний цикл обробки операцій.

4.1 Опис алгоритму роботи

Для забезпечення функціонування дисплея через інтерфейс I2C було інтегровано бібліотеку LiquidCrystal_I2C. Для підвищення зручності в читанні коду і спрощення налаштування схеми, номери використовуваних пінів були призначені за допомогою іменованих констант:

- Датчики: - pinTrig (9) - pinEcho (10) - pinPIR (7)
- Індикація: - pinLedGreen (4) - pinLedRed (5) - pinBuzzer (6)

Також створено глобальні змінні для зберігання наступних параметрів: тривалість ехо-сигналу (duration), обчислена відстань (distance) та поточний стан датчика руху (pirState). Ініціалізація системи починається автоматично після подачі живлення на мікроконтролер. Цей процес відповідає за налаштування пристрою та забезпечення його коректної роботи.

На початковому етапі здійснюється ініціалізація LCD-дисплея, включаючи активацію його підсвітки. Одразу після цього проводиться налаштування портів мікроконтролера: вихідні порти конфігуруються для управління світлодіодами, звуковим сигналом (зумером) та тригером ультразвукового датчика (Trig), а вхідні порти призначаються для отримання сигналів від PIR-датчика та ехо-відгуку (Echo). Після успішного виконання цих операцій на дисплей виводиться привітання, яке сигналізує про коректний запуск системи.

Робота програми здійснюється в рамках основного циклу, який виконує безперервну логіку функціонування охоронної системи:

- **Зчитування даних:** Мікроконтролер опитує цифровий пін датчика HC-SR501 для визначення стану. Одночасно формується імпульс тривалістю 10 мікросекунд на пині Trig, після чого за допомогою функції pulseIn реєструється час, за який сигнал повертається до пину Echo. Це значення конвертується у відстань до об'єкта, виражену в сантиметрах.

- **Обробка логіки:** У разі виявлення руху, коли датчик руху фіксує високий сигнал (HIGH), активується режим "Тривога". У цьому режимі загоряється червоний світлодіод, подається звуковий сигнал за допомогою функції tone, а на дисплеї з'являється повідомлення "ALARM!" із зазначенням виміряної відстані до об'єкта. Якщо ж рух не виявлено і датчик перебуває в стані LOW, активується режим "Охорона". У цьому випадку вмикається зелений світлодіод, а червоний світлодіод і зумер вимикаються. На екран виводиться повідомлення "SAFE", яке свідчить про безпечний стан системи. Цикл повторюється з невеликою затримкою, що забезпечує стабільність зчитування даних та точність вимірювань.

4.2 Код програми

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

LiquidCrystal_I2C lcdDisplay(0x27, 16, 2);

const uint8_t trigPin = 9;
const uint8_t echoPin = 10;
const uint8_t pirPin = 7;
const uint8_t ledOk = 4;
const uint8_t ledBad = 5;
const uint8_t buzzPin = 6;

unsigned long echoTime;
int currentDistance;
bool motionDetected;

int measure();
void idleState();
void dangerState(int value);

void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(pirPin, INPUT);
  pinMode(ledOk, OUTPUT);
  pinMode(ledBad, OUTPUT);
  pinMode(buzzPin, OUTPUT);

  Serial.begin(9600);

  lcdDisplay.init();
  lcdDisplay.backlight();
  lcdDisplay.setCursor(0, 0);
  lcdDisplay.print("Init complete");
  lcdDisplay.setCursor(0, 1);
  lcdDisplay.print("System online");
  delay(2000);
  lcdDisplay.clear();
}

void loop() {
```

```

motionDetected = digitalRead(pirPin);
currentDistance = measure();

if (motionDetected) {
  dangerState(currentDistance);
} else {
  idleState();
}

delay(200);
}

int measure() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(3);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(12);
  digitalWrite(trigPin, LOW);

  echoTime = pulseIn(echoPin, HIGH, 30000);
  if (echoTime == 0) return -1;

  return (int)(echoTime * 0.017);
}

void idleState() {
  digitalWrite(ledOk, HIGH);
  digitalWrite(ledBad, LOW);
  noTone(buzzPin);

  lcdDisplay.setCursor(0, 0);
  lcdDisplay.print("Mode: SAFE  ");
  lcdDisplay.setCursor(0, 1);
  lcdDisplay.print("No motion  ");
}

void dangerState(int value) {
  digitalWrite(ledOk, LOW);
  digitalWrite(ledBad, HIGH);
  tone(buzzPin, 1100);

  lcdDisplay.setCursor(0, 0);
  lcdDisplay.print("ALERT ACTIVE  ");
}

```

```
lcdDisplay.setCursor(0, 1);  
lcdDisplay.print("Dist: ");  
  
if (value >= 0) {  
  lcdDisplay.print(value);  
  lcdDisplay.print(" cm ");  
} else {  
  lcdDisplay.print("Unknown ");  
}  
  
delay(120);  
noTone(buzzPin);  
digitalWrite(ledBad, LOW);  
delay(120);  
}
```

5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

5.1 Підготовка до тестування

Функціональне тестування створеного пристрою, який називається "Автономна охоронна система", проходило у два ключові етапи. Спершу було проведено перевірку правильності встановлення апаратної частини, після чого здійснено тестування програмного забезпечення з використанням моделювання надзвичайних ситуацій.

Перед включенням живлення була проведена детальна перевірка макетної плати, з метою запобігання можливим помилкам у збірці:

Переглянуто правильність підключення датчиків HC-SR501 та HC-SR04, особливо звернуто увагу на пін VCC і GND, щоб уникнути короткого замикання.

Контроль відповідності під'єднання I2C-модуля LCD-дисплея щодо ліній SDA і SCL забезпечив коректне функціонування передачі даних.

Виконано перевірку полярності світлодіодів і наявність струмообмежувальних резисторів у відповідних електричних схемах. Додатково проведено тестування фіксації провідників на breadboard для забезпечення надійності контактів та стабільності роботи.

5.2 Процес тестування

Після підключення плати Arduino UNO до персонального комп'ютера через USB-інтерфейс успішно виконана компіляція та завантаження скетчу в пам'ять мікроконтролера. У процесі тестування було змодельовано функціонування системи в двох операційних режимах:

- Режим очікування (немає активного руху): - Система запускається, і на LCD-дисплеї з'являється початкове повідомлення. - Через 2 секунди активується зелений світлодіод, що сигналізує про готовність системи до

роботи. - На екрані зображається текст «Status: SAFE», причому звуковий сигнал залишається вимкненим.

- Режим тривоги (виявлення руху, імітація проникнення):

- Після фіксації руху, наприклад, помаху рукою перед сенсором HC-SR501, система миттєво переходить у стан тривоги.

- Зелений світлодіод вимикається, натомість починає блимати червоний.

- Активується п'єзозумер, який видає переривчастий звуковий сигнал.

- LCD-дисплей повідомляє «Status: ALARM!».

- У нижній частині дисплея в режимі реального часу показується відстань до об'єкта (наприклад, руки), визначена за допомогою ультразвукового датчика HC-SR04, із відображенням значення: «Dist: 20 cm».

5.3 Результати

Під час проведення тестів були зафіксовані такі результати: ● Чутливість: PIR-датчик демонструє надійну реакцію на рух на відстані до 3 метрів у межах приміщення;

- Точність: Ультразвуковий датчик забезпечує вимірювання відстані з похибкою в межах 0.5-2 см, що повністю відповідає вимогам для коректного визначення положення об'єкта;

- Індикація: Виведення інформації на LCD-дисплеї відзначається високою чіткістю, а рівень контрастності налаштований оптимально. Світлодіоди (червоний/зелений) чітко вказують на актуальний стан системи;

- Стабільність: Мікроконтролер працює безперебійно, навіть у випадках одночасного використання обох датчиків та дисплея.

5.4 Висновки

Результати проведених тестувань підтверджують стабільну функціональність розробленої системи безпеки, яка повністю відповідає прописаним вимогам технічного завдання. Забезпечено ефективну взаємодію між датчиками руху й відстані та системами звукового і візуального оповіщення. Пристрій повністю готовий для демонстрації як робочий прототип охоронної сигналізації.

ВИСНОВКИ

Під час розробки курсової роботи вдалося спроектувати і реалізувати макет автономної системи охорони, створений на основі платформи мікроконтролера Arduino Uno та оснащений функцією визначення відстані до об'єкта. Апаратна частина.

Проведено ґрунтовний аналіз принципів створення систем безпеки з використанням різних типів сенсорів. У макеті забезпечено ефективну взаємодію пасивного інфрачервоного датчика руху HC-SR501 з активним ультразвуковим сенсором HC-SR04, який застосовується для вимірювання дистанції. Використання протоколу I2C при підключенні LCD-дисплея сприяло оптимізації електронної схеми, зокрема шляхом звільнення цифрових портів мікроконтролера.

Програмна складова. Розроблений адаптивний алгоритм на C++ дозволяє забезпечувати безперервне зчитування даних із сенсорів у реальному часі. Завдяки використанню спеціалізованих бібліотек (LiquidCrystal_I2C, Wire) реалізовано виведення текстових даних, а чітко сформовані умови логічної структури гарантують коректну реакцію системи на потенційні тривожні події. Перевірка працездатності. У процесі тестування було підтверджено відповідність роботи пристрою заявленим технічним характеристикам.

Система надійно переходить із режиму очікування в активний стан в разі виявлення руху, супроводжуючи це світловою і звуковою сигналізацією. Одночасно на дисплеї відображається точне значення відстані до виявленого об'єкта.

5.5 Перспективи розвитку:

Розроблений пристрій являє собою базовий прототип із великим потенціалом для подальшого вдосконалення та трансформації в повноцінну систему "Розумний дім" або функціональну сигналізацію для комерційного використання. Основними напрямками покращення та розширення можливостей можна визначити такі:

- Дистанційне оповіщення. Запровадження GSM-модуля (на кшталт SIM800L) або Wi-Fi модуля (ESP8266) дозволить реалізувати надсилання SMS-повідомлень або сповіщень у месенджери власникові у разі активації сигналізації.

- Система контролю доступу. З цією метою можливе додавання функціональних компонентів, наприклад, матричної клавіатури (Keypad 4x4) чи RFID-зчитувача, для забезпечення активації або деактивації охоронної системи за допомогою пароля чи RFID-картки.

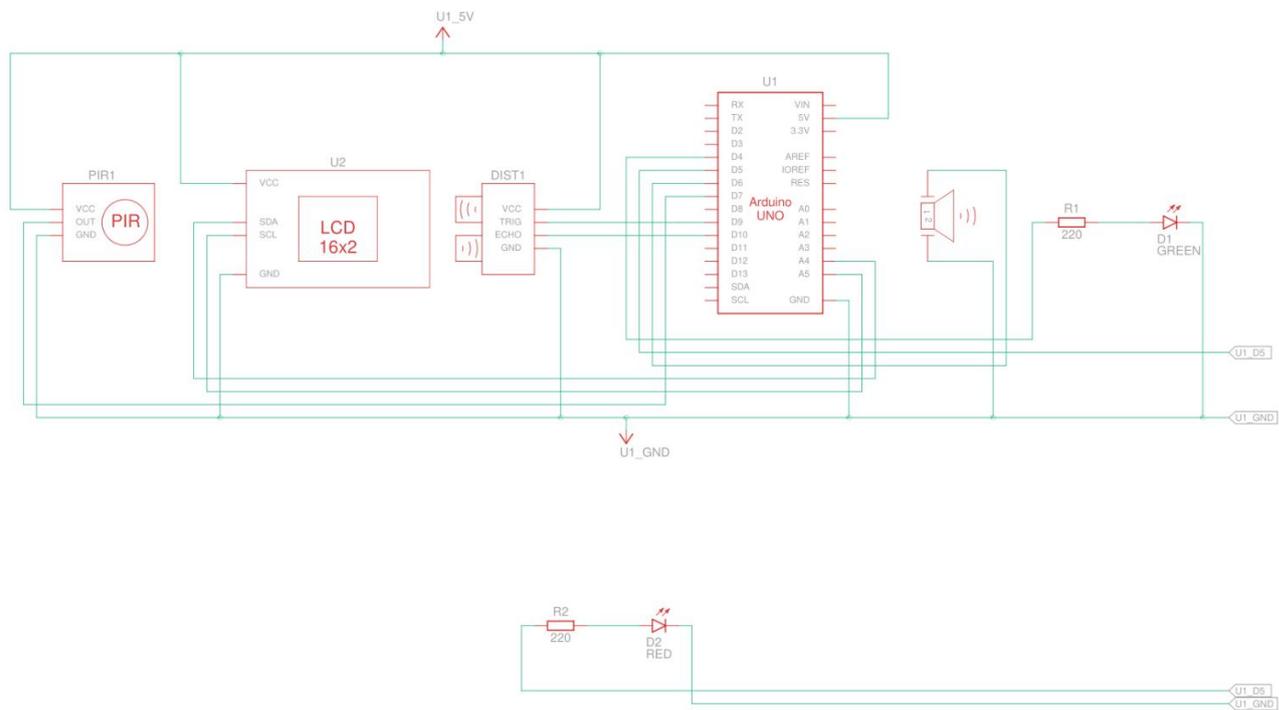
- Енергонезалежність. Заплановано створення автономної системи живлення на основі використання літій-іонних акумуляторів і зарядного модуля, а також оптимізацію програмного забезпечення для зниження енергоспоживання. Це включає впровадження режиму енергозбереження (режим сну) для тривалої автономної роботи.

- Логування подій. Передбачається підключення модуля реального часу (RTC) та SD-карти для запису даних про всі спрацювання у форматі текстових файлів, включаючи інформацію про час, дату і навіть відстань до об'єкта на момент спрацювання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- **Alvins, B.** *Mastering Arduino Programming (2021). Призначення: Програмування мікроконтролерів, синтаксис C/C++.*
- **Шилдт Г.** *C++: базовий курс / Герберт Шилдт. — 3-тє вид. — М.: Вільямс, 2019. — 624 с.*
- **Петренко А. І.** *Основи мікропроцесорної техніки та програмування мікроконтролерів: навчальний посібник / А. І. Петренко. — К.: Каравела, 2020. — 288 с.*
- **Schwartz, M.** *Arduino Electronics Blueprints (2019). Призначення: Інтеграція компонентів, використання бібліотек, створення скетчів.*
- **Axelson, D.** *Embedded Systems Design (2020). Призначення: Фундаментальні основи проєктування вбудованих систем, архітектура.*
- **Блум Д.** *Вивчаємо Arduino: інструменти та методи технічного чарівництва / Джеремі Блум. — К.: БХВ, 2016. — 336 с.*

ДОДАТОК 1



Принципова електрична схема підключення