

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

КУРСОВИЙ ПРОЄКТ

на тему:

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

Виконав студент групи ІП-41

Іван Драч

Керівник проєкту:

Остап ЮНАК

Курсовий проєкт перевірений

і допущений до захисту

“ ____ ” _____ 2025 р.

Курсовий проєкт при захисті оцінений

Львів 2025

ЗАВДАННЯ
на курсовий проєкт

Драчу Івану Анатолійовичу

(прізвище, ім'я та по батькові)

з навчальної дисципліни: ПРОГРАМНО-АПАРАТНІ ЗАСОБИ
INTERNET-РЕЧЕЙ

Студент групи: **ІІІ-41**

1. Тема проєкту: Виготовлення охоронного пристрою на платі Arduino з використанням датчиків HC-SR501 I HC-SR04 та виводом інформації на LCD дисплей.

2. Дата видачі завдання: ”22” вересня 2025 р

3. Термін здачі курсового проєкту: ”15” 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 BOM (BillofMaterials) для виконання проєкту (див. ДОДАТОК Б).
- 4.4 Розробити таблицю з'єднань пінів (PinConnectionTable) використаних компонентів для реалізації пристрою (див. ДОДАТОК В).
- 4.5 Реалізувати прошивку на Arduino (коментарі в коді, структура проєкту, використані бібліотеки).
- 4.6. Зробити на монтажній платі безпечного 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)

Іван Драч

(імя та прізвище)

Керівник проекту

(підпис)

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

Мета роботи:

Розробити апаратно-програмний пристрій на базі платформи Arduino відповідно до запропонованої тематики, який виконує визначений набір функцій. Підготувати звіт, що міститиме техніко-конструкторську документацію та детальну інструкцію з експлуатації.

Актуальність:

Актуальність цієї курсової роботи зумовлена значним попитом на поєднання апаратного та програмного забезпечення в сучасних інженерних розробках, а також зростаючим значенням мікроконтролерів у сфері освіти та промисловості.

● Технологічна та Інженерна Актуальність

- **Поширеність IoT (Internet of Things):** Сучасні технології автоматизації, моніторингу та управління значною мірою базуються на використанні компактних, економічних і енергоефективних пристроїв. Платформа Arduino стала фактично стандартним інструментом для швидкого прототипування та створення пристроїв, які взаємодіють із фізичним середовищем, що ідеально узгоджується із загальносвітовою тенденцією розвитку Інтернету речей.
- **Демократизація розробки:** Arduino, як відкрита апаратно-програмна платформа, істотно спрощує доступ для початківців-інженерів та стимулює розвиток інновацій. Вона надає можливість зосередитися на логіці проекту та його функціональності, уникаючи

необхідності занурення в низькорівневе програмування чи складний апаратний дизайн.

- **Міждисциплінарний характер:** Розробка апаратно-програмного пристрою передбачає поєднання знань з електроніки, програмування мікроконтролерів (C/C++), теорії керування та створення інтерфейсів. Це сприяє формуванню універсальних інженерних навичок, які є надзвичайно важливими для роботи в областях робототехніки, автоматизації та розробки вбудованих систем.
- **Прикладна та Практична Актуальність**
 - **Вирішення конкретної прикладної задачі:** Робота включає не лише теоретичний аналіз, а й розробку діючого фізичного прототипу (моніторинг параметрів, автоматизоване управління, збір даних). Це наочно демонструє вміння автора перенести теоретичні знання у практичну площину та реалізувати їх у формі інженерного рішення.
 - **Економічна ефективність:** Використання Arduino дає змогу створювати функціональні пристрої з мінімальними затратами як на компоненти, так і на час розробки. Це особливо важливо для інноваційних стартапів та освітніх проєктів.
 - **Можливість подальшого розвитку:** Створений прототип є міцним фундаментом для подальшого розширення, удосконалення та переходу на промислові мікроконтролери, що відкриває наукові перспективи як для дипломної роботи, так і для можливої комерціалізації.

Отже, створення апаратно-програмного комплексу на основі платформи Arduino є актуальним, економічно виправданим і надзвичайно практичним напрямом досліджень, який водночас сприяє вивченню сучасних апаратних технологій та розвитку базових інженерних навичок.

1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ

1.1 Апаратно-програмна платформа Arduino

Arduino є відкритою платформою для розробки прототипів електронних пристроїв, яка характеризується простотою використання як апаратного, так і програмного забезпечення. Її структура включає фізичну друковану плату з мікроконтролером та інтегроване середовище розробки (IDE), що призначене для написання програмного коду.

Рис. 1.1 Вигляд плати Arduino з мікроконтролером ATmega328P2

Історія створення:

Проект Arduino було започатковано у 2005 році в Інституті дизайну взаємодії (Interaction Design Institute Ivrea), розташованому в місті Івреа, Італія. Ініціаторами стали Массімо Банці, Девід Куартільєс, Том Іго, Джанлука Мартіно та Девід Мелліс. Основною ідеєю створення платформи було забезпечення студентів доступним і зручним інструментом для реалізації інтерактивних проєктів, який не вимагав би глибоких знань у галузі схемотехніки.

Причини вибору платформи:

У рамках цієї курсової роботи було обрано плату Arduino Uno R3, що базується на мікроконтролері ATmega328P. Її популярність пояснюється такими 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 являє собою набір інструментів, бібліотек, шаблонів та правил, що забезпечує структурований підхід до розробки програмного забезпечення. Він значно полегшує та прискорює процес створення проєктів, дозволяючи розробникам уникати написання базового коду з нуля. Це дає можливість сконцентруватися на розробці унікальної бізнес-логіки продукту. Фреймворки широко застосовуються для роботи з різними типами проєктів, включаючи веб, мобільні та десктопні додатки.

C++ є універсальною мовою програмування високого рівня, що забезпечує підтримку об'єктно-орієнтованого підходу. У розробці для Arduino вона дає змогу ефективно управляти пам'яттю мікроконтролера й взаємодіяти з апаратними регістрами, зберігаючи водночас зрозумілий і зручний 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

Датчик руху HC-SR501 (PIR Sensor) працює на основі виявлення змін інфрачервоного (теплого) випромінювання та належить до типу пасивних інфрачервоних датчиків (Passive Infrared Sensor). Усі об'єкти з температурою, що перевищує абсолютний нуль, випромінюють тепло, яке вловлюється датчиком. У середині пристрою піроелектричний елемент фіксує зміни інтенсивності цього випромінювання. Коли теплий об'єкт, наприклад людина, рухається через зони, які формуються лінзою Френеля, елемент генерує електричний сигнал тривоги.

Рис 1.3.1 Вигляд датчику руху HC-SR501

Ультразвуковий датчик відстані HC-SR04 функціонує за принципом ехолокації (SONAR). Його конструкція включає два п'єзoelementи – випромінювач і приймач.

1. Випромінювач генерує пачку ультразвукових імпульсів частотою 40 кГц;
2. Звукова хвиля відбивається від перешкоди та повертається назад до приймача;
3. Мікроконтролер вимірює час проходження сигналу і розраховує відстань.

Рис 1.3.2 Вигляд датчику HC-SR04

Рідкокристалічний дисплей LCD 1602 з використанням інтерфейсу I2C призначений для відображення символьних даних. Основною перевагою цієї реалізації є застосування адаптера I2C (Inter-Integrated Circuit), який дозволяє передавати інформацію через лише дві сигнальні лінії: SDA та SCL. Це значно спрощує структуру пристрою у порівнянні з традиційним паралельним підключенням.

Рис 1.3.2 Вигляд рідкокристалічного дисплею (LCD 1602) з I2C

2 ТЕХНІЧНЕ ЗАВДАННЯ

2.1 Назва роботи

Розробка охоронної системи на основі плати Arduino із застосуванням датчиків HC-SR501 і HC-SR04, а також відображення інформації на LCD-дисплеї.

2.2 Мета роботи

Створити охоронний пристрій на основі плати Arduino із застосуванням датчиків HC-SR501 і HC-SR04, забезпечивши виведення інформації на LCD-дисплей.

2.3 Завдання

- Ознайомитись з інструкцією до практичної роботи;
- Ознайомитись із принциповою електричною схемою охоронного 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;
- HC-SR501;
- HC-SR04;
- Резистори 220 Ом;
- П'єзозумер — для звукового сигналу;
- З'єднувальні провідники типу «тато-тато» та «тато-мама».

2.5 Очікуваний результат

Принцип роботи охоронної системи ґрунтується на постійному спостереженні за зоною відповідальності завдяки використанню пасивного інфрачервоного датчика руху (HC-SR501). У стані чергування система залишається в режимі спокою: зелений світлодіод горить, а на LCD-дисплеї відображається повідомлення про безпеку. У разі виявлення теплового випромінювання від об'єкта, щорухається, мікроконтролер оперативно переходить у режим «Тривога». У цьому стані активуються червоний світлодіод і звуковий сигнал, а ультразвуковий датчик (HC-SR04) визначає точну відстань до об'єкта, миттєво відображаючи отримані дані на екрані. Після припинення фіксації руху система автоматично повертається до режиму очікування.

3 АПАРАТНА ЧАСТИНА

3.1 Підбір BOM (BillofMaterials) — це перелік матеріалів та компонентів, необхідних для складання пристрою або виготовлення виробу. Він може бути оформлений у вигляді таблиці, щомістить усі деталі, потрібні для реалізації проекту.

Таблиця 3.1.1 Перелік необхідних компонентів

№	Назва компонента	Кількість	Позначення на схемі	Характеристики	Примітка
1	ArduinoUno	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 MotionSensor, 5 В	Виявлення руху об'єктів (тепла)
4	HC-SR04	1	DIST1	UltrasonicSensor, 2–400 см	Вимірювання відстані до об'єкта
5	LCD 1602 (I2C)	1	U2	16x2 символів, I2C інтерфейс	Відображення статусу системи
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

№	Назва компонента	Кількість	Позначення на схемі	Характеристики	Примітка
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	DigitalPin 9	Сигнал запуску
		Echo	DigitalPin 10	Прийом сигналу
		GND	GND	Земля
2	HC-SR501 (PIR)	VCC	5V	Живлення
		OUT	DigitalPin 7	Сигнал тривоги
		GND	GND	Земля
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 (+)	DigitalPin 4	Через резистор 220 Ом
		Катод (-)	GND	

№	Компонент	Вивід компонента	З'єднано з піном Arduino	Примітка / Функція
5	Світлодіод (Зелений)	Анод (+)	DigitalPin 5	Через резистор 220 Ом
		Катод (-)	GND	
4	П'єзозумер	Плюс (+)	DigitalPin 6	-
		Мінус (-)	GND	

4 ПРОГРАМНА ЧАСТИНА

Розробка програмного забезпечення для мікроконтролера проводилася в середовищі Arduino IDE. Як мову програмування використовували C++ разом зі стандартними бібліотеками Wiringi спеціалізованими бібліотеками для роботи з периферійними пристроями (Wire.h, LiquidCrystal_I2C.h).

Алгоритм роботи пристрою базується на постійному спостереженні за простором за допомогою інфрачервоного датчика руху, а також вимірюванні відстані до об'єкта у разі спрацювання тривоги. Дані потім відображаються на LCD дисплеї, супроводжуючись звуковим сигналом. Програмне забезпечення поділено на три ключові етапи: підключення необхідних бібліотек та оголошення констант, налаштування та ініціалізація портів, а також основний цикл виконання функцій.

4.1 Опис алгоритму роботи

- Підключення бібліотек і оголошення змінних: Для роботи з дисплеєм через інтерфейс I2C використовується бібліотека LiquidCrystal_I2C. З метою підвищення зручності читання коду та швидкого налаштування схеми, номери пінів задані у вигляді іменованих констант:
 - Датчики: pinTrig (9), pinEcho (10), pinPIR (7).
 - Індикація: pinLedGreen (4), pinLedRed (5), pinBuzzer (6). Також оголошено глобальні змінні для збереження часу відгуку ехо-сигналу (duration), обчисленої відстані (distance) та поточного стану датчика руху (pirState).
- **Ініціалізація:** При подачі живлення на мікроконтролер виконується функція налаштування:
 - Ініціалізується LCD дисплей та вмикається його підсвітка.

- Порти для керування світлодіодами, зумером та тригером ультразвукового датчика (Trig) налаштовуються як виходи (OUTPUT).
- Порти для зчитування сигналів з PIR-датчика та ехо-сигналу (Echo) налаштовуються як входи (INPUT).
- На дисплей виводиться вітальне повідомлення про запуск системи.
- **Основний цикл:** Програма виконує нескінченний цикл, що реалізує логіку охоронної системи:
 - **Зчитування даних:** Мікроконтролер здійснює опитування цифрового піну датчика HC-SR501. Одночасно створюється імпульс тривалістю 10 мкс на піні Trig, після чого за допомогою функції pulseIn вимірюється час, за який сигнал повертається на піні Echo. Отримане значення часу потім перетворюється у показник відстані (в см).
 - **Обробка логіки:**
 - **Режим «Тривога»:** У разі отримання датчиком руху високого сигналу (HIGH), програма активує червоний світлодіод, видає звуковий сигнал за допомогою функції tone, а також відображає на дисплеї повідомлення "ALARM!" разом із поточною відстанню до об'єкта.
 - **Режим «Охорона»:** У разі відсутності виявленого руху (LOW) активується зелений світлодіод, тоді як червоний світлодіод та зумер залишаються вимкненими. На дисплеї при цьому відображається статус "SAFE".
 - Цикл повторюється з мінімальною затримкою, щоб забезпечити стабільність показників.

4.2 Код програми

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

LiquidCrystal_I2C display(0x27, 16, 2);

const int TRIGGER_PIN = 9;
const int ECHO_PIN    = 10;
const int SENSOR_PIR  = 7;
const int LED_OK      = 4;
const int LED_ALARM   = 5;
const int ALARM_BEEP  = 6;

long pulseDuration;
int currentDistance;
bool isMotionDetected = false;

void setup() {
    pinMode(TRIGGER_PIN, OUTPUT);
    pinMode(ECHO_PIN, INPUT);
    pinMode(SENSOR_PIR, INPUT);
    pinMode(LED_OK, OUTPUT);
    pinMode(LED_ALARM, OUTPUT);
    pinMode(ALARM_BEEP, OUTPUT);
}
```

```
Serial.begin(9600);

display.init();
display.backlight();

display.setCursor(2, 0);
display.print("GUARD SYSTEM");
display.setCursor(3, 1);
display.print("INITIALIZING");
delay(2000);
display.clear();
}

void loop() {
  isMotionDetected = digitalRead(SENSOR_PIR);

  digitalWrite(TRIGGER_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIGGER_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIGGER_PIN, LOW);

  pulseDuration = pulseIn(ECHO_PIN, HIGH);
  currentDistance = pulseDuration * 0.034 / 2;

  if (isMotionDetected) {
    activateAlarm(currentDistance);
  }
}
```

```

    } else {
        standbyMode();
    }

    delay(150);
}

void standbyMode() {
    digitalWrite(LED_OK, HIGH);
    digitalWrite(LED_ALARM, LOW);
    noTone(ALARM_BEEP);

    display.setCursor(0, 0);
    display.print("STATE: SECURE  ");
    display.setCursor(0, 1);
    display.print("Scanning...  ");
}

void activateAlarm(int range) {
    digitalWrite(LED_OK, LOW);

    digitalWrite(LED_ALARM, HIGH);
    tone(ALARM_BEEP, 1200);

    display.setCursor(0, 0);
    display.print("STATE: INTRUDER!");
}

```

```
display.setCursor(0, 1);  
display.print("Range: ");  
display.print(range);  
display.print(" cm  ");  
  
delay(150);  
noTone(ALARM_BEEP);  
digitalWrite(LED_ALARM, LOW);  
delay(100);  
}
```

5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

Перевірка роботи створеного пристрою «Автономна охоронна система» здійснювалася у двістадії: оцінка правильності складання апаратної частини та функціональне тестування програмного забезпечення із моделюванням тривожних ситуацій.

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 та HC-SR04 (відповідність пінів VCC та GND, щоб уникнути короткого замикання);
- переконано у коректності підключення I2C-модуля LCD дисплея (лінії SDA та SCL);
- перевірено полярність світлодіодів та наявність струмообмежувальних резисторів у їхніх колах;
- перевірено надійність з'єднань провідників на платі breadboard.

5.2 Процес тестування

Після підключення плати Arduino UNO до персонального комп'ютера через USB-інтерфейс було виконано компіляцію та завантаження програмного коду (скетчу) в пам'ять мікроконтролера. У ході експериментального дослідження проведено симуляцію роботи системи у двох різних режимах:

- **Режим «Очікування» (відсутність руху):**

- Система ініціалізується, на LCD дисплеї з'являється стартове повідомлення.
- Через 2 секунди активується зелений світлодіод.
- На дисплеї відображається напис «Status: SAFE», звуковий сигнал відсутній.

- **Режим «Тривога» (імітація проникнення):**

- При помаху рукою перед лінзою датчика HC-SR501 система миттєво реагує зміною стану.
- Зелений світлодіод гасне, починає блимати червоний світлодіод.
- Активується п'єзозумер, подаючи переривчастий звуковий сигнал.
- На дисплей виводиться повідомлення «Status: ALARM!».
- У нижньому рядку дисплея в реальному часі відображається відстань до об'єкта (руки), виміряна ультразвуковим датчиком HC-SR04 (наприклад, «Dist: 15 cm»).

5.3 Результати

Під час випробувань було встановлено:

- **Чутливість:** PIR-датчик коректно фіксує рух на відстані до 3 метрів (в межах кімнати);
- **Точність вимірювань:** Ультразвуковий датчик надає покази відстані з похибкою не більше 1-2 см, що є достатнім для визначення положення об'єкта;
- **Індикація:** Інформація на LCD дисплеї відображається чітко, контрастність налаштована коректно. Логіка перемикання світлодіодів (Червоний/Зелений) повністю відповідає стану системи;
- **Стабільність:** Збоїв у роботі мікроконтролера при одночасній роботі двох датчиків та дисплея не виявлено.

5.4 Висновок

Результати тестування підтверджують, що розроблена охоронна система працює стабільно і виконує поставлене технічне завдання. Забезпечено коректну взаємодію датчиків руху та відстані з засобами візуального та звукового сповіщення. Пристрій готовий до демонстрації як діючий макет охоронної сигналізації.

ВИСНОВОК

Підчас виконання курсової роботи було створено і впроваджено макет автономної охоронної системи на основі мікроконтролера ArduinoUno, оснащений функцією вимірювання відстані до об'єкта.

- **Апаратна реалізація:** Досліджено ключові принципи розробки систем безпеки із застосуванням сенсорів різного типу. Успішно реалізовано інтегровану роботу пасивного інфрачервоного датчика (HC-SR501), призначеного для виявлення руху, та активного ультразвукового сонара (HC-SR04), який використовується для вимірювання відстані. Впровадження інтерфейсу I2C при підключенні LCD-дисплея дало змогу суттєво оптимізувати апаратну схему і зменшити потребу у використанні цифрових портів мікроконтролера.
- **Програмна реалізація:** У ході дослідження створено адаптивний програмний алгоритм мовою C++, призначений для виконання опитування датчиків у режимі реального часу. Застосування спеціалізованих бібліотек, таких як LiquidCrystal_I2C та Wire, забезпечило точну і коректну обробку та виведення текстової інформації. Додатково, впровадження логічних умов у структуру програми сприяло ефективній та безпомилковій реакції системи на можливі тривожні події.
- **Працездатність системи:** Тестування підтвердило, що пристрій повністю відповідає технічному завданню. Система плавно переходить з режиму очікування в режим тривоги при фіксації руху, забезпечуючи при цьому світлозвукову сигналізацію та відображення точної відстані до порушника на екрані.
- **Навчальна цінність:** Робота сприяла зміцненню знань у галузі основ електроніки та схемотехніки, зосереджуючи увагу на підключенні

навантажень за допомогою транзисторних ключів або резисторів. Крім того, вдалося вдосконалити практичні навички роботи з протоколами передачі даних і налаштування програмного коду;

5.5 Перспективи розвитку:

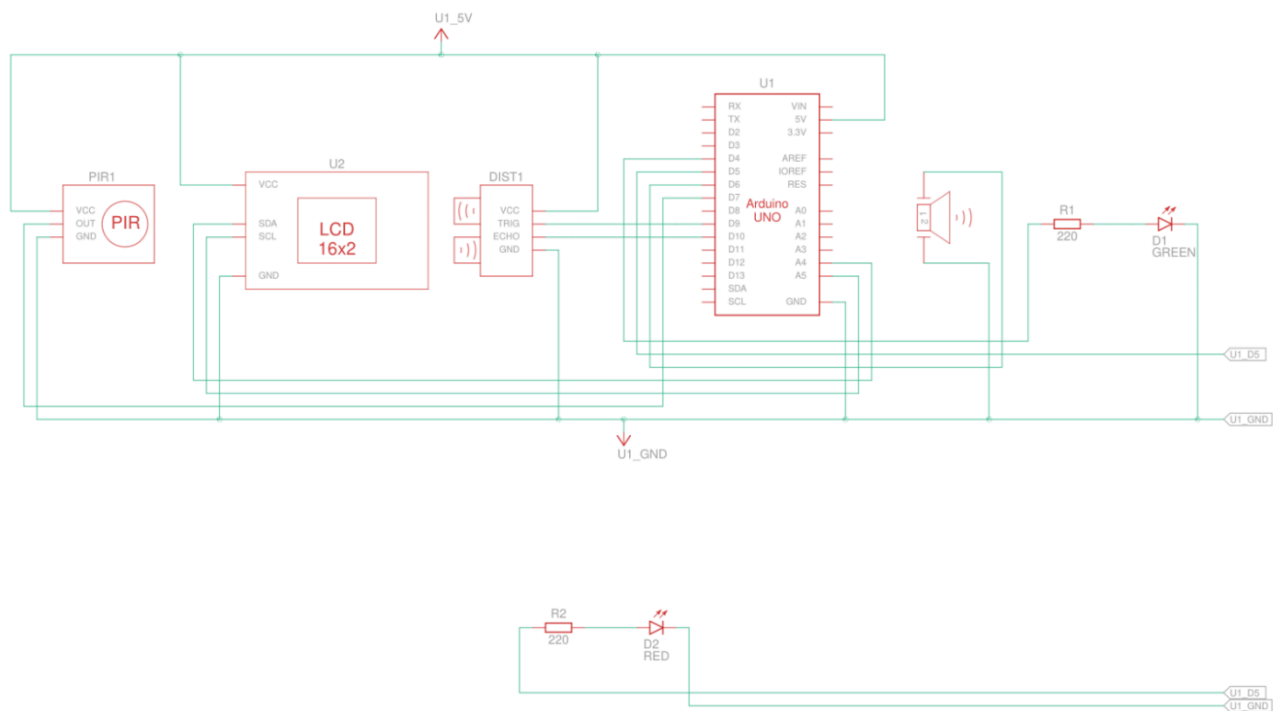
Розроблений пристрій є базовим прототипом, який має значний потенціал для модернізації та перетворення на повноцінну систему «Розумний дім» або комерційну сигналізацію. До основних напрямів подальшого вдосконалення можна віднести:

- **Дистанційне сповіщення:** Інтеграція GSM-модуля (наприклад, SIM800L) або Wi-Fi модуля (ESP8266) для надсилання SMS-повідомлень або сповіщень у месенджер власника при спрацюванні сигналізації;
- **Контроль доступу:** Додавання матричної клавіатури (Keypad 4x4) або RFID-зчитувача для можливості постановки та зняття об'єкта з охорони за допомогою пароля або картки доступу;
- **Енергонезалежність:** Розробка схеми безперебійного живлення з використанням літій-іонних акумуляторів та модуля зарядки, а також програмна оптимізація енергоспоживання (режим сну) для тривалої автономної роботи;
- **Логування подій:** Підключення модуля годинника реального часу (RTC) та SD-карти для запису історії спрацювань (дата, час та відстань до об'єкта) у текстовий файл;
- **Конструктивне виконання:** Перенесення схеми з макетної плати (breadboard) на друковану плату (PCB) та розробка корпусу методом 3D-друку для захисту електроніки від зовнішніх впливів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- **Блум Д. Вивчаємо Arduino:** інструменти та методи технічного чарівництва / Джеремі Блум. — К.: БХВ, 2016. — 336 с.
- **Шилдт Г. С++:** 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.
- **Рюмик С. М.** Мікроконтролери AVR: практикум для початківців / С. М. Рюмик. — Х.: Ранок, 2018. — 112 с.
- **Галкін В. І.** Промислова електроніка та мікросхемотехніка: підручник / В. І. Галкін. — К.: Вища школа, 2017. — 320 с.
- **Alvins, B.** Mastering Arduino Programming (2021). *Призначення: Програмування мікроконтролерів, синтаксис C/C++.*
- **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.*
- **Marr, B.** Internet of Things (IoT) in Business (2022). *Призначення: Концепції IoT, мережеві протоколи та стандарти для пристроїв.*

ДОДАТОК 1



Принципова електрична схема підключення