

**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»

Виконала студентка групи ІП-41

Яна БРИЙОВСЬКА

Керівник проєкту:

Остап ЮНАК

Курсовий проєкт перевірений і

допущений до захисту

“ ____ ” _____ 2025 р.

Курсовий проєкт при захисті оцінений

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)

Керівник проекту

_____ (підпис)

Яна БРИЙОВСЬКА

_____ (імя та прізвище)

Остап ЮНАК

_____ (імя та прізвище)

ЗМІСТ

ВСТУП

1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ

2 ТЕХНІЧНЕ ЗАВДАННЯ

3 АПАРАТНА ЧАСТИНА

4 ПРОГРАМНА ЧАСТИНА

5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

ВИСНОВКИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

ДОДАТКИ

ВСТУП

Мета роботи:

Створення апаратно-програмного комплексу на базі платформи Arduino, який реалізує визначений функціонал відповідно до теми. Також передбачено підготовку звіту, що включає техніко-конструкторську документацію та інструкцію для користувача.

Актуальність:

Доцільність виконання цієї роботи зумовлена значним попитом на поєднання апаратного забезпечення та програмних рішень у сучасній інженерії. Окрім того, роль мікроконтролерних систем невинно зростає як у промисловій сфері, так і в навчальних процесах.

- **Технологічна та інженерна значущість:**

- **Розвиток IoT (Інтернету речей):** Сучасні системи керування, моніторингу та автоматизації потребують енергоефективних, економічних та компактних пристроїв. Платформа Arduino стала стандартом для оперативного прототипування гаджетів, що взаємодіють із фізичним середовищем, що повністю відповідає світовим тенденціям розвитку IoT;
- **Доступність розробки:** Завдяки своїй відкритості (Open-Source), платформа Arduino суттєво знижує поріг входження для початківців, стимулюючи інноваційну діяльність. Це дає можливість розробнику фокусуватися на функціональності та логіці проєкту, не відволікаючись

на складне проєктування схем чи низькорівневе кодування;

- **Міждисциплінарний підхід:** Створення такого пристрою вимагає поєднання знань із різних галузей: електроніки, проєктування інтерфейсів, теорії управління та програмування на C/C++. Такий комплексний підхід дозволяє здобути навички, необхідні для роботи у сферах вбудованих систем, автоматизації та робототехніки;
- **Практична цінність:**
 - **Вирішення прикладної проблеми:** Робота виходить за рамки теорії, оскільки передбачає конструювання діючого фізичного прототипу (збір даних, моніторинг, автоматичне керування). Це підтверджує вміння автора реалізовувати теоретичні знання на практиці;
 - **Економічна доцільність:** Застосування Arduino дозволяє мінімізувати витрати на деталі та час розробки, отримуючи при цьому функціональний виріб, що є критично важливим для навчальних проєктів та стартапів;
 - **Перспективи масштабування:** Створений прототип може стати основою для подальшої модернізації або переходу на промислові контролери, що відкриває можливості для написання дипломної роботи чи комерційного використання.

Таким чином, створення пристрою на Arduino є актуальним, економічно вигідним та практичним завданням, що сприяє формуванню важливих інженерних компетенцій.

1 ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ

1.1 Апаратно-програмна платформа Arduino

Arduino представляє собою платформу з відкритим вихідним кодом (open-source), призначену для розробки прототипів електроніки. Вона базується на гнучкому та зрозумілому програмному й апаратному забезпеченні. Система включає фізичну друковану плату з мікроконтролером та інтегроване середовище розробки (IDE) для написання програм.



Рис. 1.1 Вигляд плати Arduino з мікроконтролером ATmega328P2

Історична довідка:

Проект зародився у 2005 році в італійському місті Івреа, в Інституті проектування взаємодії. Засновниками платформи виступили Массімо Банці, Девід Квартільєс, Том Іго, Джанлука Мартіно та Девід Мелліс. Головною ідеєю було надати студентам недорогий та простий інструмент для реалізації інтерактивних проєктів без необхідності глибокого вивчення схемотехніки.

Обґрунтування вибору платформи:

У даній курсовій роботі використовується плата Arduino Uno R3 на основі мікроконтролера ATmega328P. Її популярність пояснюється такими перевагами:

- Цінова доступність: Низька вартість елементної бази;
- Універсальність: ПЗ для розробки сумісне з Linux, macOS та Windows;
- Зручність: Просте програмування через USB-порт без потреби в зовнішніх програматорах;
- Відкритість: Схеми плат є публічними, що спрощує підключення різноманітних модулів та сенсорів.

1.2 Програмування: C++ та Arduino IDE

Для програмування контролера використовується мова C++. Середовище Arduino IDE базується на спрощеному фреймворку Wiring. Фреймворк — це набір правил, бібліотек та інструментів, що структурують процес створення ПЗ, прискорюючи розробку. Це дозволяє уникнути написання рутинного коду та зосередитися на бізнес-логіці. C++ — це універсальна високорівнева мова з підтримкою об'єктно-орієнтованого підходу. У контексті Arduino вона забезпечує ефективне управління пам'яттю та регістрами при збереженні зрозумілого синтаксису. Програма (скетч) складається з двох основних функцій:

- `setup()` — запускається одноразово при старті для конфігурації пінів;
- `loop()` — виконується циклічно, містячи основний алгоритм пристрою.

1.3 Принципи дії сенсорів

Датчик руху HC-SR501 (PIR Sensor): Робота пасивного інфрачервоного сенсора базується на фіксації змін теплового випромінювання. Будь-який об'єкт із температурою вище абсолютного нуля випромінює тепло. Коли теплий об'єкт (наприклад, людина) рухається крізь зони лінзи Френеля,

піроелектричний елемент детектує зміну інтенсивності випромінювання та генерує сигнал тривоги.



Рис 1.3.1 Вигляд датчику руху HC-SR501

Ультразвуковий далекомір HC-SR04: Цей сенсор використовує принцип ехолокації (SONAR) і складається з приймача та випромінювача.

1. Випромінювач посилає ультразвуковий сигнал частотою 40 кГц.
2. Звукова хвиля відбивається від перешкоди та повертається до датчика.
3. Мікроконтролер фіксує час польоту сигналу та обчислює дистанцію.



Рис 1.3.2 Вигляд датчику HC-SR04

Дисплей LCD 1602 з модулем I2C: Для відображення інформації обрано символний рідкокристалічний екран. Ключовою особливістю є використання протоколу I2C (Inter-Integrated Circuit). Це дозволяє передавати дані лише двома проводами (SDA та SCL), що суттєво спрощує схему порівняно зі стандартним паралельним підключенням.



Рис 1.3.2 Вигляд рідкокристалічного дисплею (LCD 1602) з I2C

2 ТЕХНІЧНЕ ЗАВДАННЯ

2.1 Назва роботи

Розробка та макетування охоронної системи на базі мікроконтролера Arduino з інтеграцією датчиків руху (HC-SR501), відстані (HC-SR04) та виведенням даних на LCD-дисплей.

2.2 Мета роботи

Створити функціональний прототип пристрою безпеки, використовуючи платформу Arduino, налаштувати взаємодію сенсорів HC-SR501 та HC-SR04, а також забезпечити візуалізацію стану системи на рідкокристалічному екрані.

2.3 Завдання

- Опрацювати методичні вказівки до практичної частини роботи;
- Проаналізувати принципову електричну схему проєктованої охоронної системи на базі Arduino Uno;
- Виконати комутацію компонентів на безпайковій макетній платі відповідно до розробленої схеми;
- Розробити та відлагодити програмний код (скетч) для керування мікроконтролером.

2.4 Вимоги до апаратного забезпечення: Система повинна включати наступні елементи:

- Джерело живлення постійного струму (5 В);

- Мікроконтролерна плата Arduino Uno R3;
- Макетна плата (Breadboard) для прототипування;
- Символьний дисплей LCD 1602 з інтерфейсом I2C;
- Світлодіодна індикація (зелений та червоний LED);
- Пасивний інфрачервоний сенсор руху HC-SR501;
- Ультразвуковий далекомір HC-SR04;
- Резистори номіналом 220 Ом;
- П'єзоелектричний випромінювач (зумер) для акустичної сигналізації;
- Набір з'єднувальних проводів (типи «штекер-штекер» та «штекер-гніздо»).

2.5 Очікуваний сценарій роботи

Логіка функціонування системи ґрунтується на постійному скануванні периметра за допомогою PIR-сенсора HC-SR501.

- **Режим спокою:** Коли рух відсутній, світиться зелений індикатор, а на дисплеї відображається статус безпеки.
- **Режим тривоги:** При детектуванні теплового об'єкта миттєво спрацьовує червоний світлодіод та звукова сирена. Паралельно активується сонар HC-SR04, який визначає дистанцію до порушника і транслює ці дані на екран у реальному часі.
- **Відновлення:** Якщо рух припиняється, пристрій автоматично переходить у черговий режим.

3 АПАРАТНА ЧАСТИНА

3.1 Підбір BOM

Для реалізації проекту було сформовано Bill of Materials (BOM) — перелік необхідних матеріалів. Це ключовий документ, що визначає склад виробу.

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
6	Світлодіод (Червоний)	1	D1_RED	5 мм, 2 В, 20 мА	Індикація тривоги
7	Світлодіод	1	D1_GREEN	5 мм, 2 В, 20 мА	Індикація

	(Зелений)				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
		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

Розробка керуючої програми (прошивки) для мікроконтролера виконувалася в середовищі Arduino IDE. Код написано мовою C++ із залученням стандартного фреймворку Wiring, а також спеціалізованих бібліотек Wire.h та LiquidCrystal_I2C.h для забезпечення коректної взаємодії з периферійними пристроями.

Загальна логіка роботи системи базується на циклічному скануванні зони відповідальності за допомогою інфрачервоного сенсора руху. У разі виявлення загрози активується алгоритм вимірювання відстані до об'єкта, супроводжуваний звуковою сигналізацією та виводом даних на LCD-екран. Структура програми складається з трьох логічних блоків: декларації бібліотек і змінних, блоку налаштувань (setup) та основного виконавчого циклу (loop).

4.1 Опис логіки функціонування

- **Підготовчий етап (декларація):** Для керування дисплеєм через протокол I2C підключено бібліотеку LiquidCrystal_I2C. З метою покращення читабельності коду та спрощення подальшого конфігурування, фізичні порти мікроконтролера закріплено за іменованими константами:
 - *Сенсори:* pinTrig (9), pinEcho (10), pinPIR (7).
 - *Модулі індикації:* pinLedGreen (4), pinLedRed (5), pinBuzzer (6). Також оголошено глобальні змінні для збереження проміжних даних: тривалості імпульсу (duration), розрахованої дистанції (distance) та поточного статусу датчика руху (pirState).
- **Налаштування (Setup):** При старті мікроконтролера одноразово виконується процедура ініціалізації:

- Запускається LCD-дисплей та активується його підсвітка.
- Піни, що відповідають за керування світлодіодами, зумером та тригером сонара (Trig), конфігуруються як виходи (OUTPUT).
- Піни для прийому даних від PIR-сенсора та ехо-сигналу сонара (Echo) налаштовуються як входи (INPUT).
- На екран виводиться стартове повідомлення про активацію системи безпеки.
- **Виконавчий цикл (Loop):** Програма переходить у режим нескінченного циклу, виконуючи наступні операції:
 - **Збір телеметрії:** Контролер зчитує логічний стан датчика HC-SR501. Паралельно ініціюється вимірювання відстані: на пін Trig подається імпульс тривалістю 10 мкс, після чого функція pulseIn фіксує час повернення сигналу на пін Echo. Отримане значення часу автоматично конвертується у сантиметри.
 - **Керуюча логіка:**
 - *Режим «Тривога»:* Якщо датчик руху повертає високий логічний рівень (HIGH), система переходить в активний стан: запалюється червоний світлодіод, зумер генерує звуковий сигнал (функція tone), а на дисплеї відображається напис "**ALARM!**" разом із поточною відстанню до об'єкта.
 - *Режим «Охорона»:* За відсутності руху (сигнал LOW) система перебуває в режимі очікування: активний зелений світлодіод, звукова сигналізація вимкнена, на екрані статус "**SAFE**".
 - Цикл завершується короткою затримкою для забезпечення стабільності зчитування показників.

4.2 Код програми

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

LiquidCrystal_I2C lcd(0x27, 16, 2);

const int pinTrig = 9;
const int pinEcho = 10;
const int pinPIR = 7;
const int pinLedGreen = 4;
const int pinLedRed = 5;
const int pinBuzzer = 6;

long duration;
int distance;
int pirState = LOW;

void setup() {
  pinMode(pinTrig, OUTPUT);
  pinMode(pinEcho, INPUT);
  pinMode(pinPIR, INPUT);
  pinMode(pinLedGreen, OUTPUT);
  pinMode(pinLedRed, OUTPUT);
  pinMode(pinBuzzer, OUTPUT);
}
```

```
Serial.begin(9600);

lcd.init();
lcd.backlight();

lcd.setCursor(0, 0);
lcd.print("Security System");
lcd.setCursor(0, 1);
lcd.print("System Active...");
delay(2000);
lcd.clear();
}

void loop() {
  pirState = digitalRead(pinPIR);

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

  duration = pulseIn(pinEcho, HIGH);
  distance = duration * 0.034 / 2;

  if (pirState == HIGH) {
    alertMode(distance);
  }
}
```

```
    } else {  
        safeMode();  
    }  
  
    delay(200);  
}  
  
void safeMode() {  
    digitalWrite(pinLedGreen, HIGH);  
    digitalWrite(pinLedRed, LOW);  
    digitalWrite(pinBuzzer, LOW);  
  
    lcd.setCursor(0, 0);  
    lcd.print("Status: SAFE    ");  
    lcd.setCursor(0, 1);  
    lcd.print("No Motion    ");  
}  
  
void alertMode(int dist) {  
    digitalWrite(pinLedGreen, LOW);  
  
    digitalWrite(pinLedRed, HIGH);  
    tone(pinBuzzer, 1000);  
  
    lcd.setCursor(0, 0);  
    lcd.print("Status: ALARM!  ");  
  
    lcd.setCursor(0, 1);
```

```
lcd.print("Dist: ");  
lcd.print(dist);  
lcd.print(" cm    ");  
  
delay(100);  
noTone(pinBuzzer);  
digitalWrite(pinLedRed, LOW);  
delay(100);  
}
```

5 ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

Випробування розробленої системи «Автономна охоронна сигналізація» проводилися комплексно і включали два ключові етапи: перевірку фізичного монтажу схеми та функціональне тестування програмного алгоритму з імітацією реальних загроз.

5.1 Попередня діагностика (Hardware Check)

Перед першим запуском було здійснено ретельний візуальний та інструментальний контроль макетної плати для уникнення апаратних пошкоджень:

- **інспекція живлення:** перевірено коректність підведення ліній VCC та GND до сенсорів HC-SR501 та HC-SR04 для запобігання короткому замиканню;
- **перевірка інтерфейсів:** підтверджено правильність комутації шини I2C (лінії SDA/SCL) для LCD-дисплея;

- компоненти: перевірено полярність підключення світлодіодів та наявність відповідних резисторів у ланцюгах;
- контакти: пересвідчилися у надійності фіксації з'єднувальних провідників на платі breadboard.

5.2 Методика проведення тестування

Процес перевірки розпочався з підключення апаратної частини (Arduino UNO) до USB-порту комп'ютера, після чого було здійснено компіляцію програмного коду та його завантаження у мікроконтролер. Для верифікації працездатності було відтворено два основні сценарії роботи пристрою:

- **Штатний режим (Спокій):**
 - Відбувається запуск системи, про що інформує стартова заставка на LCD-екрані.
 - Із затримкою у 2 секунди вмикається зелений індикатор готовності.
 - Дисплей транслює статус «Status: SAFE», при цьому звукове оповіщення деактивовано.
- **Режим реагування (Тривога):**
 - При потраплянні об'єкта (наприклад, руки) у зону чутливості PIR-сенсора HC-SR501 система миттєво фіксує порушення.
 - Відбувається зміна світлової індикації: зелений світлодіод вимикається, активується миготливий червоний сигнал.

- Вмикається звуковий супровід тривоги за допомогою п'єзозумера.
- Екран виводить попередження «Status: ALARM!».
- У нижньому рядку дисплея в режимі реального часу візуалізується дистанція до об'єкта, отримана з ультразвукового далекоміра HC-SR04 (зразок виводу: «Dist: 15 cm»).

5.3 Результати

За підсумками проведених випробувань зафіксовано наступні технічні показники:

- **Радіус дії:** Датчик руху демонструє стабільну реакцію на рухомі об'єкти на дистанції до 3 метрів, що відповідає розмірам стандартного приміщення.
- **Точність вимірів:** Похибка ультразвукового сонара складає не більше 1-2 см, що є задовільним показником для локалізації об'єкта.
- **Інформативність:** Дані на рідкокристалічному дисплеї зчитуються без перешкод, контрастність достатня. Світлова індикація (червоний/зелений спектр) коректно відображає поточний статус охоронної системи.
- **Надійність:** Система працює стійко, конфліктів апаратних ресурсів при одночасному залученні сенсорів та дисплея не виявлено.

5.4 Висновки

Експериментальна перевірка підтвердила, що спроектований охоронний пристрій функціонує стабільно та у повній відповідності до технічного завдання. Досягнуто безпомилкової взаємодії між детекторами руху/відстані та модулями оповіщення. Макет повністю готовий до демонстрації можливостей.

ВИСНОВКИ

У результаті виконання курсового проєкту було успішно розроблено, змонтовано та протестовано діючий прототип автономної охоронної системи на базі платформи Arduino Uno, здатної не лише виявляти рух, але й визначати відстань до об'єкта.

Підсумки реалізації проєкту:

- **Апаратна архітектура:** Вдалося поєднати в одній системі сенсори різного типу дії: пасивний інфрачервоний датчик HC-SR501 (для детекції присутності) та ультразвуковий модуль HC-SR04 (для локалізації об'єкта). Впровадження шини I2C для комутації LCD-дисплея дозволило оптимізувати схемотехніку, зменшивши кількість задіяних портів та дротів.

- **Програмне забезпечення:** Розроблений на C++ алгоритм забезпечує стабільний моніторинг датчиків у реальному часі. Завдяки використанню оптимізованих бібліотек (LiquidCrystal_I2C, Wire) та чіткій логічній структурі коду, система безпомилково ідентифікує загрози та керує периферією.

- **Функціональність:** Результати тестування підтвердили повну

відповідність пристрою технічним вимогам. Система демонструє надійний перехід із режиму «Очікування» в режим «Тривога», супроводжуючи це візуальними та звуковими ефектами, а також виводом точних метричних даних на екран.

- **Практична цінність:** Робота сприяла закріпленню інженерних навичок у сферах схемотехніки (робота з навантаженнями, резистивними дільниками) та програмування мікроконтролерів (налагодження коду, робота з інтерфейсами передачі даних).

5.5 Перспективи розвитку:

Створений макет є надійною базою для подальшого вдосконалення та інтеграції в екосистему «Розумний дім». Потенціал розвитку проєкту охоплює такі напрямки:

1. **Дистанційний моніторинг:** Додавання комунікаційних модулів (GSM SIM800L або Wi-Fi ESP8266) дозволить системі відправляти тривожні SMS або повідомлення в Telegram безпосередньо власнику приміщення.

2. **Система авторизації:** Впровадження інтерфейсів контролю доступу, таких як матрична клавіатура або RFID-сканер, надасть можливість зручної постановки та зняття об'єкта з охорони за допомогою PIN-коду або безконтактної карти.

3. **Енергоефективність та автономність:** Інтеграція блоку безперебійного живлення (на базі Li-Ion акумуляторів) забезпечить роботу при знеструмленні мережі, а програмна реалізація режимів сну (Deep Sleep) суттєво подовжить час роботи від батареї.

4. **Аналітика подій (Чорна скринька):** Підключення модуля

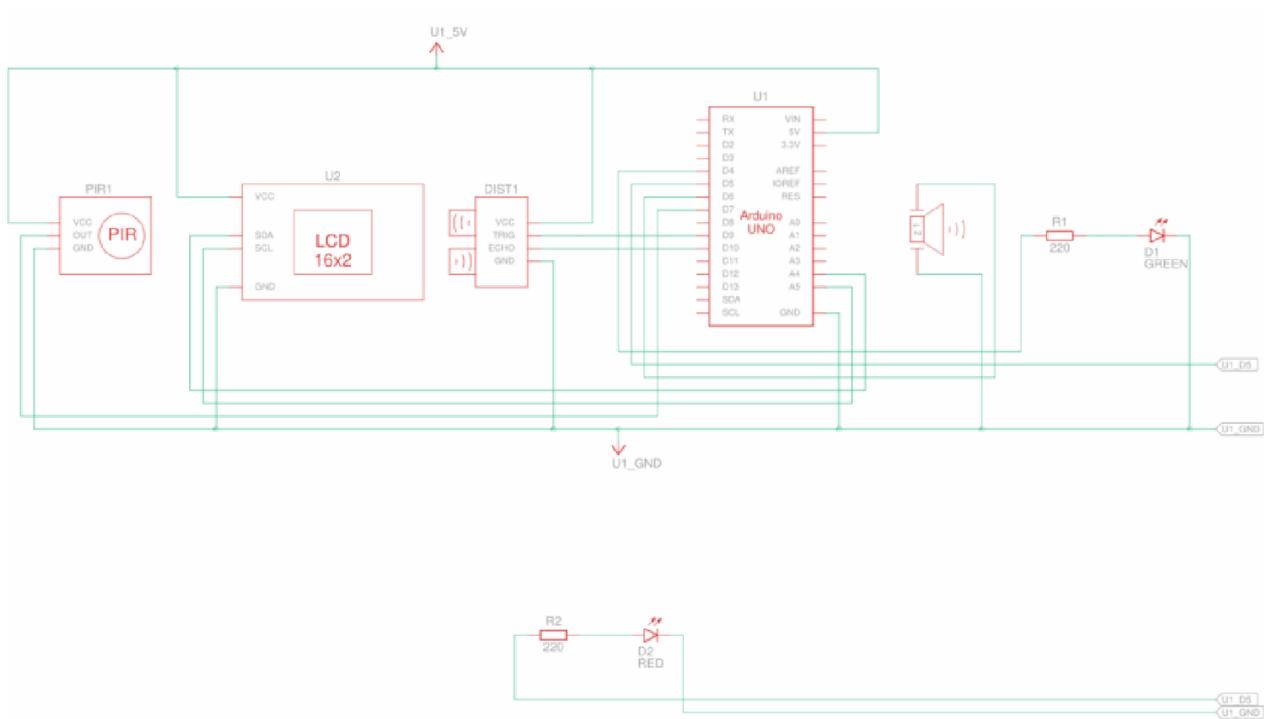
реального часу (RTC) та SD-карти дозволить вести детальний лог усіх спрацювань із фіксацією точної дати, часу та дистанції до порушника для подальшого аналізу.

5. Промисловий дизайн: Логічним завершенням проєкту стане перенесення схеми з макетної плати на друковану плату (PCB) та розробка ергономічного корпусу за допомогою 3D-друку для захисту електроніки.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- Блум Д. Вивчаємо Arduino: інструменти та методи технічного чарівництва / Джеремі Блум. — Київ : БХВ, 2016. — 336 с.
- Галкін В. І. Промислова електроніка та мікросхемотехніка: підручник / В. І. Галкін. — Київ : Вища школа, 2017. — 320 с.
- Петренко А. І. Основи мікропроцесорної техніки та програмування мікроконтролерів: навчальний посібник / А. І. Петренко. — Київ : Каравела, 2020. — 288 с.
- Рюмик С. М. Мікроконтролери AVR: практикум для початківців / С. М. Рюмик. — Харків : Ранок, 2018. — 112 с.
- Шилдт Г. С++: базовий курс / Герберт Шилдт. — 3-тє вид. — Москва : Вільямс, 2019. — 624 с.
- Alvins B. Mastering Arduino Programming. 2021.
- Axelson D. Embedded Systems Design. 2020.
- Marr B. Internet of Things (IoT) in Business. 2022.
- Schwartz M. Arduino Electronics Blueprints. 2019.

ДОДАТОК 1



Принципова електрична схема підключення