

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ФАХОВИЙ КОЛЕДЖ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КУРСОВИЙ ПРОЕКТ

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 МТ-41

Дмитро ЛЯЩЕНКО

Керівник проекту:

Остап ЮНАК

Курсовий проект перевірений

і допущений до захисту

“___” _____20__р.

Курсовий проект при захисті оцінений

**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

Лащенко Дмитру Мирославовичу

(прізвище, ім'я та по батькові)

**навчальної дисципліни: ПРОГРАМНО-АПАРАТНІ ЗАСОБИ
INTERNET-РЕЧЕЙ**

Студент групи: МТ-41

1. Тема проєкту: Виготовлення охоронного пристрою на платі Arduino з використання датчиків HC-SR501 і 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 ВОР (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 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			Дмитро Лященко
	(підпис)		(імя та прізвище)
Керівник проекту			Остап ЮНАК
	(підпис)		(імя та прізвище)

ЗМІСТ

ЗМІСТ	2
ВСТУП	3
1. ОГЛЯД ЛІТЕРАТУРИ/АНАЛОГІВ	5
2. ТЕХНІЧНЕ ЗАВДАННЯ	9
3. АПАРАТНА ЧАСТИНА	10
4. ПРОГРАМНА ЧАСТИНА	14
5. ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ	17
6. ВИСНОВКИ	18
7. СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	19
ДОДАТКИ	20

ВСТУП

Мета роботи: Розробити згідно запропонованої теми апаратно-програмний пристрій на базі Arduino, що реалізує заданий набір функцій та підготувати звіт з техніко-конструкторською документацією та інструкцією з експлуатації.

Актуальність теми дослідження

Актуальність даної курсової роботи зумовлена стрімким зростанням попиту на комплексні інженерні рішення, що передбачають глибоку інтеграцію апаратного та програмного забезпечення. У сучасному технічному середовищі мікроконтролерні платформи стають ключовим елементом як виробничих процесів, так і освітніх програм підготовки кваліфікованих кадрів.

1. Технологічний та інженерний аспект

- **1.1. Розвиток екосистеми IoT (Internet of Things).** Ефективність сучасних систем автоматизації, дистанційного моніторингу та управління безпосередньо залежить від наявності компактних та енергоефективних обчислювальних вузлів. Платформа Arduino зарекомендувала себе як стандарт де-факто для оперативного прототипування та створення інтерфейсів взаємодії з фізичним світом, що повністю узгоджується з глобальними трендами розвитку Інтернету речей.
- **1.2. Демократизація процесу розробки.** Використання відкритої архітектури (Open-Source) в екосистемі Arduino суттєво знижує поріг входження в індустрію для початківців та стимулює інноваційну діяльність. Такий підхід дозволяє розробникам зосередити увагу на архітектурі системи та її функціональних

можливостях, мінімізуючи витрати часу на вирішення низькорівневих апаратних проблем.

- **1.3. Міждисциплінарна інтеграція.** Проєктування апаратно-програмних комплексів вимагає синергії знань із суміжних галузей: цифрової електроніки, програмування мікроконтролерів (зокрема мовами C/C++), теорії автоматичного керування та розробки людино-машинних інтерфейсів. Робота над такими проєктами забезпечує формування комплексних інженерних компетенцій, критично необхідних для фахівців у сферах робототехніки, вбудованих систем та автоматизації.

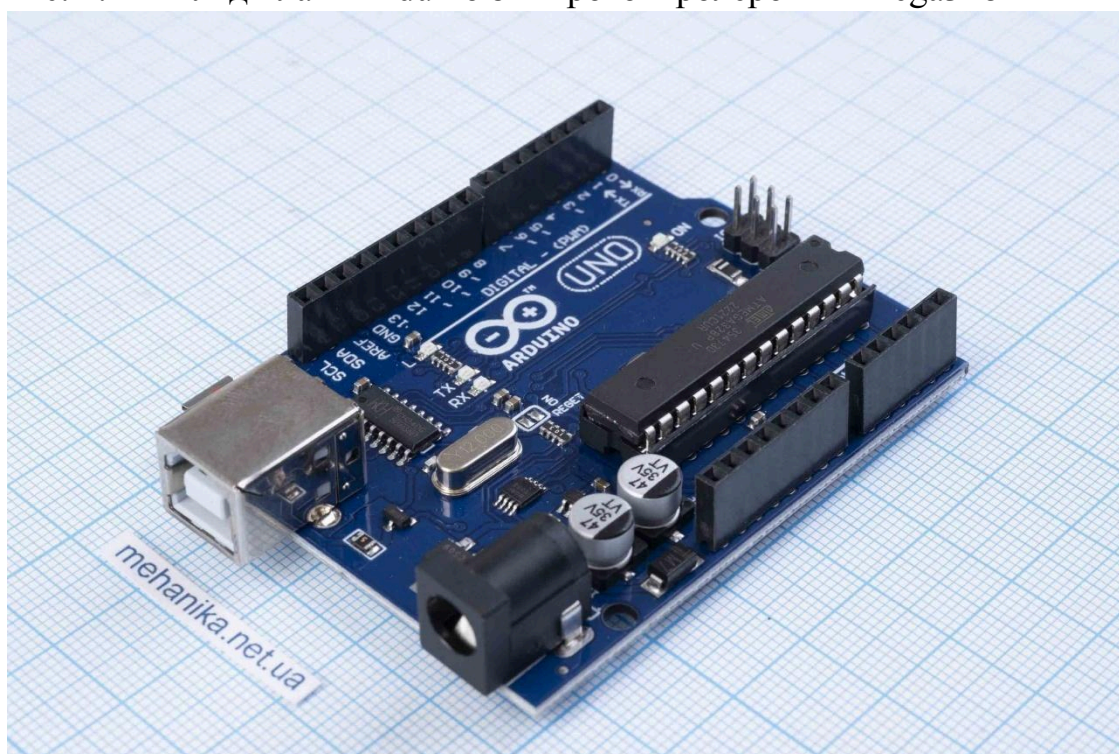
2. Прикладний та практичний вимір

- **2.1. Реалізація прикладних рішень.** Дана робота виходить за межі теоретичного дослідження і передбачає розробку діючого фізичного прототипу, здатного виконувати функції збору даних, моніторингу параметрів або автоматизованого управління. Це демонструє здатність імплементувати теоретичні знання у практичні інженерні конструкції.
- **2.2. Економічна ефективність.** Застосування платформи Arduino дозволяє досягти необхідного функціонала пристрою при мінімальних часових та матеріальних витратах. Це робить подібні рішення оптимальними для реалізації стартапів, освітніх проєктів та систем "розумного дому".
- **2.3. Потенціал масштабування.** Створений прототип виступає надійною базою для подальшої модернізації: від розширення функціонала до перенесення коду на промислові мікроконтролери. Це відкриває перспективи для подальших наукових досліджень у рамках дипломного проєктування або комерціалізації розробки.

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 ATmega328P



Історія створення: 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 Куартільес, 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 та програмуванні низького рівня.

Причини вибору платформи: Для даної курсової роботи обрано плату Arduino Uno R3, яка побудована на базі мікроконтролера ATmega328P. Її використання для створення сонячного трекера зумовлене наступними факторами:

Доступність: Низька вартість плати та компонентів;

Кросплатформеність: Середовище розробки працює на Windows, macOS та Linux;

Наявність АЦП: Вбудований аналого-цифровий перетворювач (АЦП) дозволяє легко зчитувати дані з фоторезисторів та вимірювати напругу;

Бібліотеки: Наявність готових бібліотек (Servo.h, LiquidCrystal_I2C.h) значно спрощує роботу з периферією.

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. У контексті даної роботи він дозволяє керувати сервоприводом та дисплеєм за допомогою високорівневих команд, не заглиблюючись у налаштування таймерів процесора. C++ — це універсальна мова програмування високого рівня. В контексті розробки під Arduino, C++ дозволяє ефективно обробляти математичні обчислення

(необхідні для вольтметра та логіки порівняння освітленості), зберігаючи при цьому швидкодію системи. Програма для Arduino називається «скетчем» (sketch) і має дві обов'язкові функції:

`setup()` — виконується один раз при запуску для ініціалізації сервоприводу та дисплея;

`loop()` — виконується циклічно, реалізуючи алгоритм стеження за світлом та вимірювання напруги.

1.3 Принципи роботи використаних компонентів

Піроелектричний датчик руху (HC-SR501) Основним елементом виявлення порушника є модуль HC-SR501. Принцип його дії базується на реєстрації зміни інфрачервоного випромінювання (теплового фону), яке випромінюють живі об'єкти. Датчик має лінзу Френеля, яка фокусує ІЧ-випромінювання на піроелектричний елемент.

Рис 1.3.1 Вигляд датчику руху (HC-SR501)



Технологія ультразвукової ехолокації (HC-SR04) Для вимірювання точної відстані до об'єкта, що порушив периметр, використовується ультразвуковий метод. Модуль HC-SR04 випромінює імпульс 40 кГц і фіксує час повернення відлуння. Це дозволяє системі не тільки зафіксувати факт руху, але й визначити, як близько знаходиться об'єкт.

Рис 1.3.2 Вигляд ехолокації (HC-SR04)



LCD 1602 з інтерфейсом I2C Для виведення статусу системи ("Охорона", "Тривога") та дистанції до порушника використовується дисплей LCD 1602. Використання шини I2C дозволяє зекономити піни мікроконтролера, використовуючи лише два сигнальні провідники (SDA та SCL).

Рис 1.3.3 Вигляд LCD 1602 з інтерфейсом I2C





2. ТЕХНІЧНЕ ЗАВДАННЯ

1. Назва роботи

Виготовлення на платі Arduino автоматизованого охоронного пристрою з використанням датчика руху HC-SR501, далекоміра HC-SR04 та системи візуально-звукового оповіщення.

2. Мета роботи

Розробити та виготовити прототип охоронної системи, яка здатна виявляти рух у зоні дії, вимірювати відстань до об'єкта вторгнення та інформувати користувача через дисплей та звукову сигналізацію.

3. Завдання

1. Ознайомитись з принципом спільної роботи PIR-сенсора та ультразвукового далекоміра.
2. Розробити схему підключення компонентів у середовищі моделювання Autodesk Tinkercad.
3. Написати програмний код, який при спрацюванні датчика руху активує вимірювання відстані та вмикає тривогу.
4. Налаштувати вивід відповідних повідомлень на LCD дисплей.

4. Технічні вимоги

- **Живлення:** 5 В.
- **Датчик виявлення:** HC-SR501 (реагування на тепловий рух).
- **Датчик уточнення:** HC-SR04 (вимірювання відстані при тривозі).
- **Індикація:** LCD дисплей 1602 (I2C), Червоний світлодіод.
- **Звук:** Активний зумер.

5. Очікуваний результат

Працюючий пристрій, який у стані спокою показує напис "Scanning...", а при виявленні руху вмикає сирену, світлодіод і показує відстань до об'єкта в сантиметрах.

3. АПАРАТНА ЧАСТИНА

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	-	Breadboard Small	Для монтажу схеми
3	HC-SR501	1	PIR1	Ультразвуковий	Датчик руху (інфрачервоний)
4	HC-SR04	1	U2	16x2 символів	Ультразвуковий датчик



№	Назва компонента	Кількість	Позначення на схемі	Характеристики	Примітка
5	LCD 1602 (I2C)	1	DIS1	Активний, 5В	Екран 16x2 символів
6	Зумер (Piezo)	1	BZ1	Активний, 5В	Звуковий сигнал
7	Світлодіод	1	HL1	Червоний, 5мм	Світловий сигнал
8	Резистор	1	R1	220 Ом	Обмеження струму
9	Провідники	15	-	Male-Male, Male-Female	Комутація



Таблиця 3.1.2 Таблиця з'єднань пінів

№	Компонент	Вивід компонента	З'єднано з піном Arduino	Примітка / Функція
1	HC-SR04	TRIG	Digital 7	Вихід імпульсу
		VCC	5V	Живлення датчика
		GND	GND	Земля
		ECHO	Digital 6	Вхід ехо-сигналу
2	HC-SR501	Signal (OUT)	Analog Pin A4	Дані I2C
		GND	GND	Земля
		VCC	5V	Живлення датчика



№	Компонент	Вивід компонента	З'єднано з піном Arduino	Примітка / Функція
3	Зумер і Світлодіод	Зумер +	Digital 4	Керування звуком
		Анод LED (+ R 220 Ом)	Digital 3	Керування світлом
4	LCD 1602	GND	GND	Земля
		VCC	5V	Живлення екрану
		SDA	Analog Pin A4	Лінія даних шини I2C
		SCL	Analog Pin A5	Лінія тактування шини I2C



4. ПРОГРАМНА ЧАСТИНА

Програмне забезпечення розроблено мовою C++ у середовищі Tinkercad. Код базується на циклічному опитуванні датчика руху.

Алгоритм роботи:

1. **Ініціалізація:** Налаштування пінів, запуск LCD, "прогрів" PIR-датчика (калібрування).
2. **Режим очікування:** Якщо на піні 2 (PIR) низький рівень (LOW), система виводить "Scanning area..." і чекає.
3. **Режим тривоги:**
 - Якщо PIR видає HIGH (рух виявлено):
 - Вмикається світлодіод.
 - Запускається функція вимірювання відстані через HC-SR04
 - На дисплей виводиться "!! INTRUDER !!" та дистанція.
 - Зумер подає переривчастий звуковий сигнал.



Код програми

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

LiquidCrystal_I2C lcd(0x20, 16, 2);

const int pirPin = 2;
const int ledPin = 3;
const int buzzerPin = 4;
const int trigPin = 7;
const int echoPin = 6;

int pirState = LOW;
long duration;
int distance;

void setup() {
    lcd.init();
    lcd.backlight();

    pinMode(pirPin, INPUT);
    pinMode(ledPin, OUTPUT);
    pinMode(buzzerPin, OUTPUT);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);

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

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

    if (pirState == HIGH) {

        digitalWrite(ledPin, HIGH);

        distance = getDistance();

        lcd.clear();
```



```
    lcd.setCursor(0, 0);
    lcd.print("!! INTRUDER !!");

    lcd.setCursor(0, 1);
    lcd.print("Dist: ");
    lcd.print(distance);
    lcd.print(" cm");

    tone(buzzerPin, 1000);
    delay(100);
    noTone(buzzerPin);
    delay(100);

} else {

    digitalWrite(ledPin, LOW);
    noTone(buzzerPin);

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("System Armed");
    lcd.setCursor(0, 1);
    lcd.print("Scanning area...");
    delay(500);
}
}

int getDistance() {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    duration = pulseIn(echoPin, HIGH);

    int dist = duration * 0.034 / 2;
    return dist;
}
```

5. ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

Перевірка працездатності розробленого охоронного пристрою проводилася у середовищі симуляції Tinkercad.

Процес тестування:

1. **Запуск симуляції:** При старті на екрані з'являється повідомлення про ініціалізацію, після чого система переходить у режим "System Armed" (Система на охороні).
2. **Імітація руху:** У Tinkercad натискання на датчик HC-SR501 дозволяє симулювати рух об'єкта. При активації "руху" система миттєво зреагувала:
 - Загорівся червоний світлодіод.
 - Увімкнувся звуковий сигнал.
 - На дисплеї змінився напис на "!! INTRUDER !!".
3. **Перевірка далекоміра:** Під час активної тривоги змінювалися показники на датчику HC-SR04. Дисплей коректно відображав відстань (наприклад, "Dist: 125 cm"), оновлюючи дані в реальному часі.
- 4.
5. **Завершення події:** Після припинення симуляції руху система автоматично повернулася у стан спостереження "Scanning area...", вимкнувши звукову та світлову сигналізацію.

Висновки: Тестування підтвердило коректність взаємодії двох типів датчиків: PIR-сенсор успішно виконує роль тригера, а ультразвуковий сенсор надає додаткову інформацію про відстань до порушника.

6. ВИСНОВКИ

У ході виконання курсової роботи було успішно розроблено та реалізовано макет охоронного пристрою з функцією визначення відстані до об'єкта на базі мікроконтролерної платформи Arduino Uno.

Висновки:

1. **Апаратна реалізація:** Опрацьовано принципи побудови комбінованих систем безпеки. Виконано інтеграцію різнотипних датчиків: піроелектричного (HC-SR501) для фіксації руху та ультразвукового (HC-SR04) для вимірювання дистанції. Використання шини I2C для дисплея дозволило спростити схемотехніку та зекономити порти контролера.
2. **Програмна реалізація:** Розроблено скетч мовою C++, який реалізує алгоритм подвійної перевірки. Програма забезпечує перехід системи в активний режим вимірювання лише при наявності сигналу тривоги, що оптимізує роботу пристрою. Використано бібліотеку LiquidCrystal_I2C для зручного відображення статусу охорони.
3. **Працездатність системи:** Тестування підтвердило повну відповідність роботи пристрою технічному завданню. Система миттєво реагує на імітацію руху, вмикає світло-звукову сигналізацію та коректно відображає відстань до "порушника" в сантиметрах на LCD екрані.
4. **Навчальна цінність:** Робота дозволила закріпити навички проектування електронних схем, розібратися з особливостями роботи інфрачервоних та ультразвукових сенсорів, а також поглибити знання з програмування логічних умов та обробки сигналів на Arduino.

Перспективи розвитку:

- **GSM-модуль:** Додавання модуля зв'язку (наприклад, SIM800L) для надсилання SMS-сповіщень власнику про проникнення.
- **Автономність:** Інтеграція літій-іонного акумулятора та модуля контролю заряду для роботи при відключенні електроенергії.
- **Відеофіксація:** Підключення камери (наприклад, ESP32-CAM) для



створення фотографії об'єкта в момент спрацювання датчика руху.



7. СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

Вивчаємо Arduino: інструменти та методи технічного чарівництва / Джеремі Блум. — К.: БХВ, 2016.

Документація до бібліотеки LiquidCrystal I2C [Електронний ресурс].

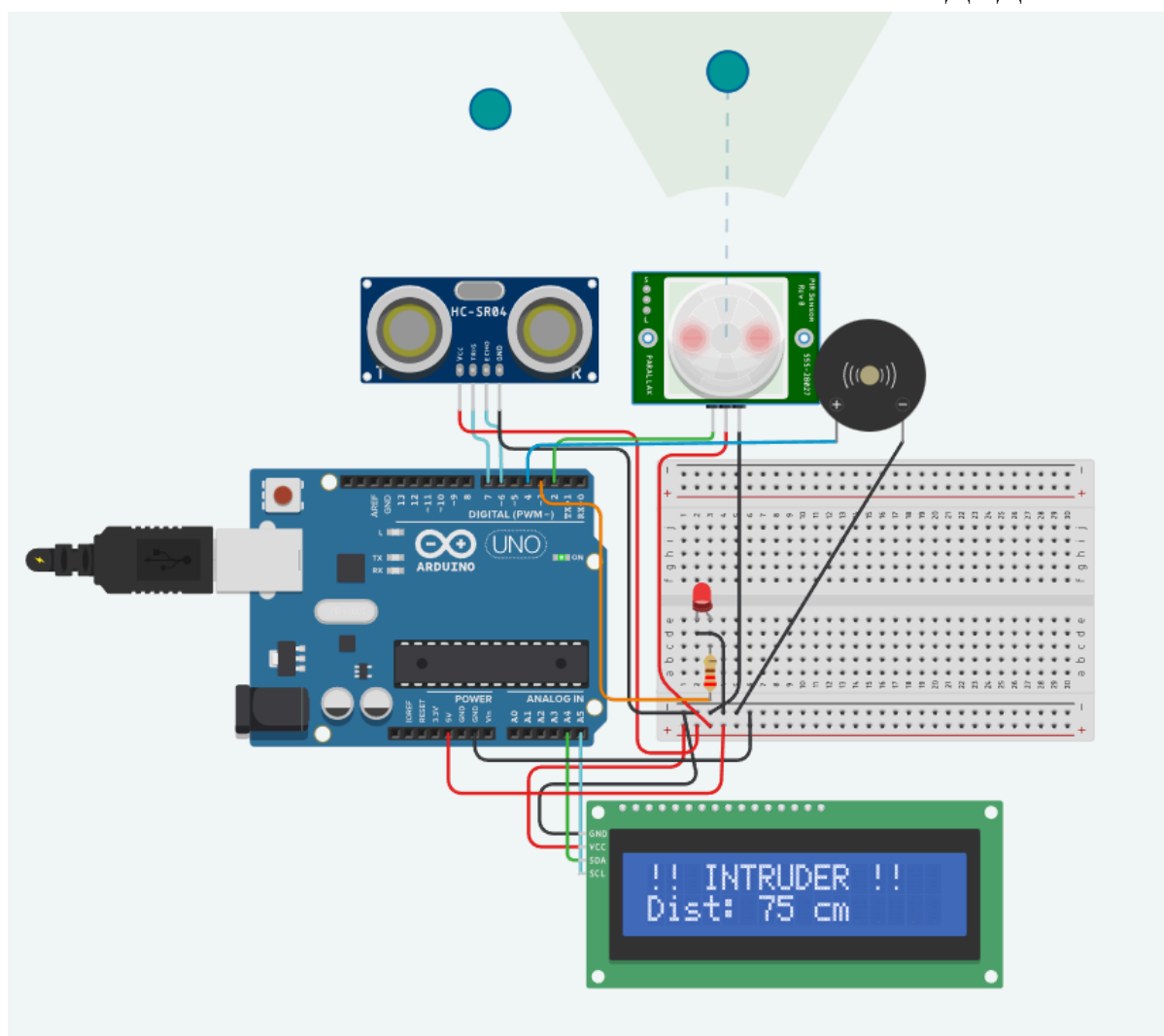
Datasheet HC-SR501 PIR Motion Detector.

Datasheet HC-SR04 Ultrasonic Ranging Module.

Офіційна документація Arduino [Електронний ресурс]. – Режим доступу: <https://www.arduino.cc>

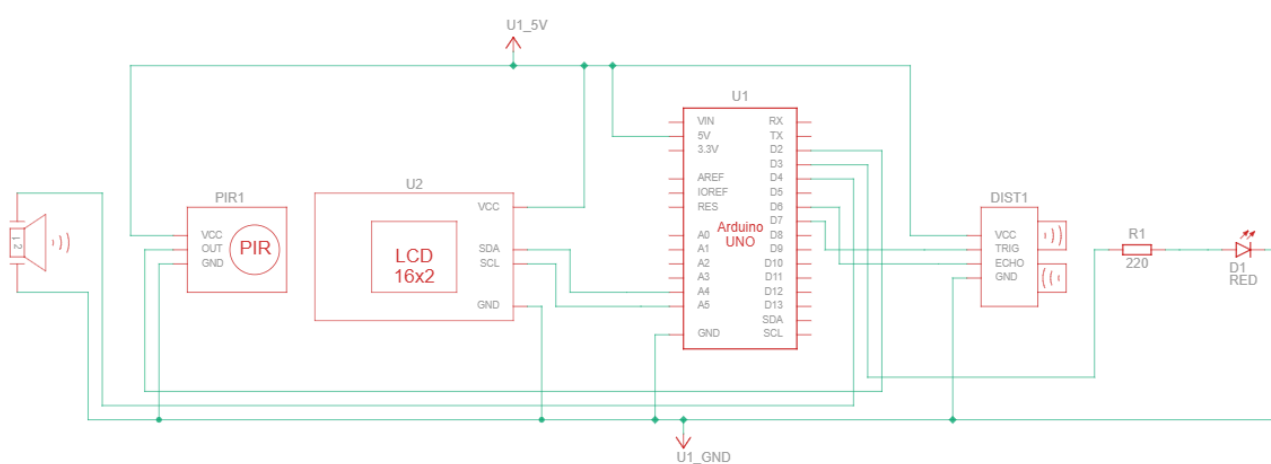


ДОДАТОК 1



Структурна схема реалізації моделі

ДОДАТОК 2



Принципова електрична схема підключення

