



## Plan lekcji - Informatyka

**Przedmiot:** Alternatywna instrukcja IF - zastosowanie w programowaniu ultradźwiękowego systemu bezpieczeństwa

**Grupa docelowa:** Uczniowie 9 klasy (początkujący)

### **Cele / Umiejętności**

Cel 1. Objaśnienie działania instrukcji alternatywnej i zagnieżdżonych

Cel 2. Budowa urządzenia Arduino

Cel 3. Implementacja kodu urządzenia za pomocą instrukcji IF

Cel 4. Badanie funkcjonalności urządzenia

**Metody dydaktyczne:** rozmowa, wyjaśnianie, rozwiązywanie problemów, projektowanie algorytmów, demonstracja

### **Środki / narzędzia / technologie edukacyjne**

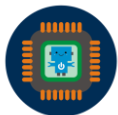
Kalkulator, Internet, edytor Arduino online,

Elementy projektu (Arduino Uno x 1, płyta robocza x 1, czujnik ultradźwiękowy x 1, czerwone x 1 diody LED, żółte x 1, zielone 1 x 1, rezystor x 1, kable x 10)

### **Planowanie działań**

Uczniowie zostaną podzieleni na 3 zespoły, które równolegle wykonają urządzenie i zaprezentują je. W każdym zespole składającym się z 10 uczniów znajdą się uczniowie, którzy zbudują urządzenie, oraz uczniowie, którzy je zaprogramują.

<b>Czas trwania</b>	<b>Działalność</b>	<b>Metody/środki</b>
5 min	Wprowadzenie instrukcji alternatywy IF. Wyjaśnienie składni C++ i zasady wykonywania. Zagnieżdżone, reguły asocjacyjne ELSE.	Wyjaśnienie, rozmowa
5 min	Prezentacja ultradźwiękowego urządzenia zabezpieczającego (krok 1 - załącznik)	Rozwiązywanie problemów, eksplikacja, rozmowa
20 min	Budowa urządzenia (załącznik Krok 2-6)	Rozwiązywanie problemów, eksplikacja, rozmowa
15 min	Programowanie urządzeń	Projektowanie algorytmów



# ROBOSTEM Project

Agreement no: 2019-1-RO01-KA202-063965



Erasmus+

5 min	Testowanie funkcjonalności urządzenia	Demonstracja
-------	---------------------------------------	--------------

## Ocena/informacje zwrotne:

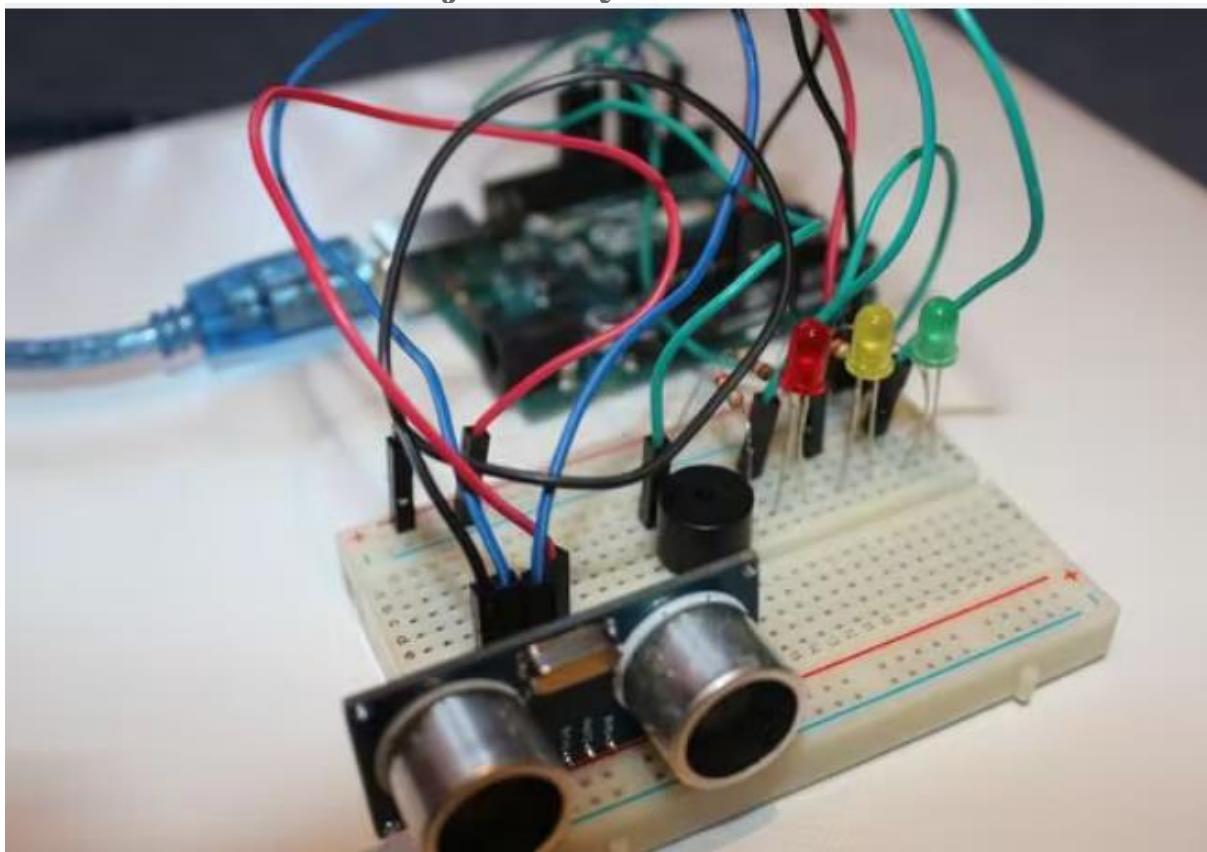
Przetestuj działanie urządzenia w każdej sytuacji przewidzianej w instrukcji IF (ustawianie w różnych odległościach oraz sprawdzanie sygnałów świetlnych i dźwiękowych).

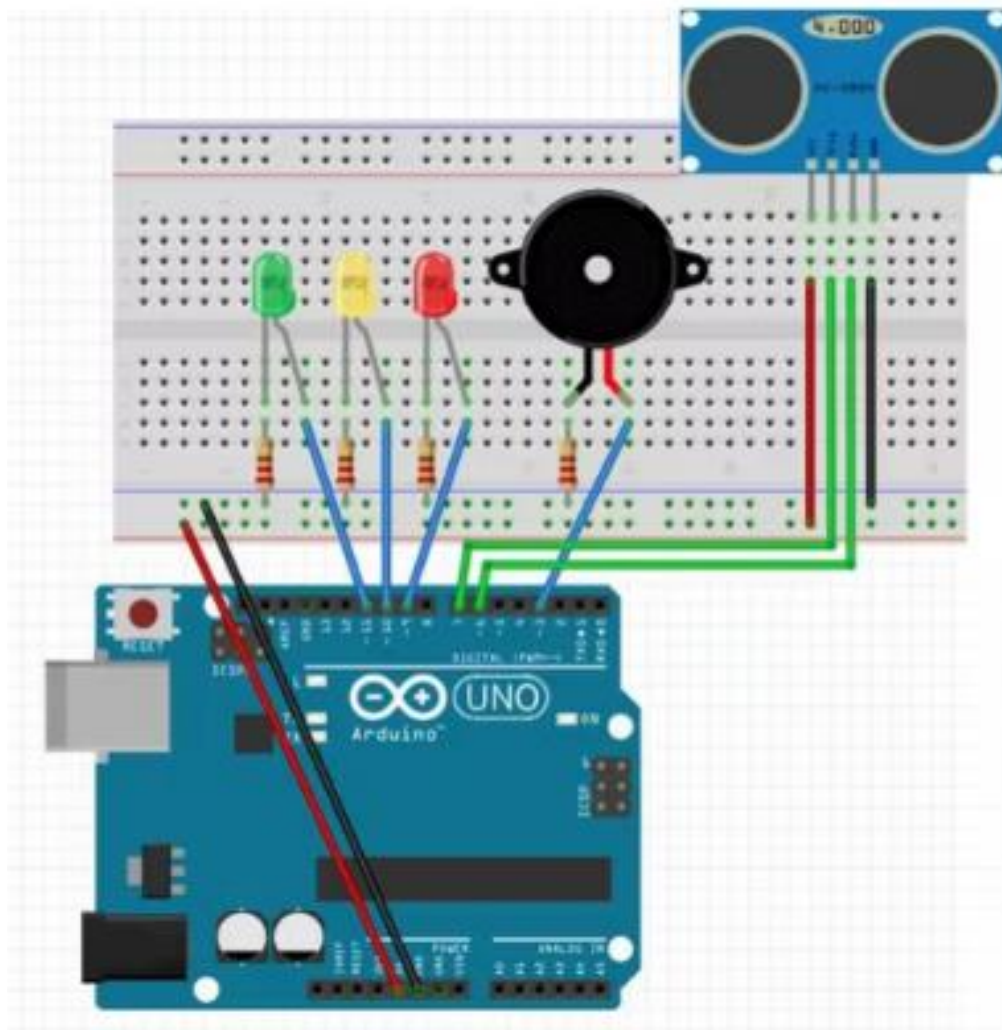
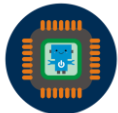
## Bibliografia/Webografia:

[https://create.arduino.cc/projecthub/Krepak/ultrasonic-security-system-3afe13?ref=tag&ref\\_id=kids&offset=3](https://create.arduino.cc/projecthub/Krepak/ultrasonic-security-system-3afe13?ref=tag&ref_id=kids&offset=3)

## Załącznik 1

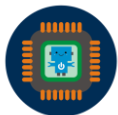
### Krok 1: Prezentacja urządzenia





## Krok 2: Podłącz:

- Czerwony przewód od pinu 5V do kanału dodatniego na płycie drukowanej
- Czarny przewód od styku GND w Arduino do kanału ujemnego na płycie drukowanej
- Buzzer = styk 7
- Czujnik ultradźwiękowy:
  - Echo = styk 3
  - Trig = styk 2
- Diody LED:
  - RedLED = styk 4
  - YellowLED = styk 5
  - GreenLED = styk 6



# ROBOSTEM Project

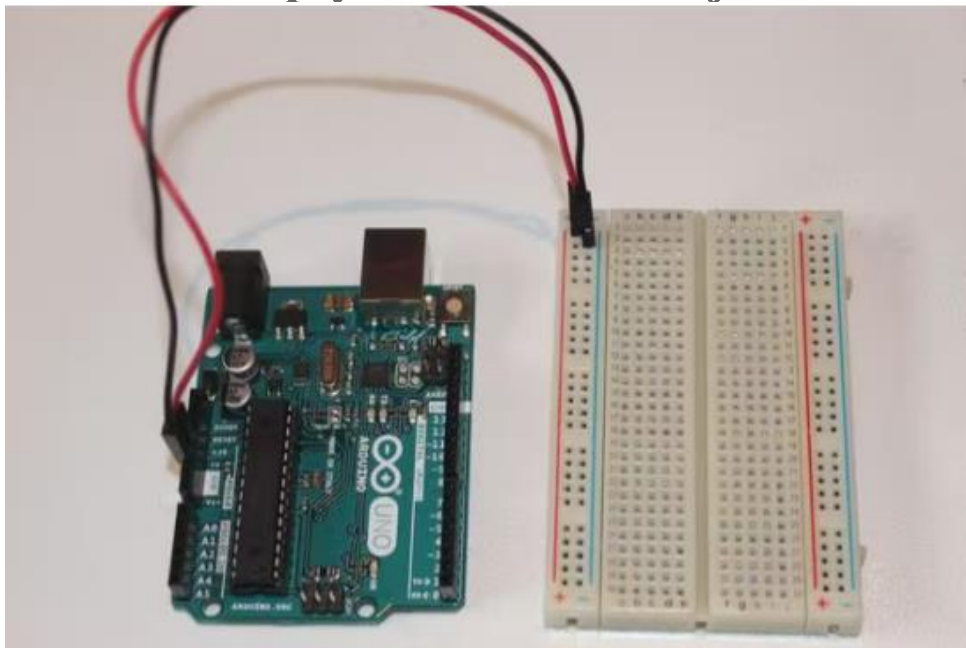
Agreement no: 2019-1-RO01-KA202-063965



Erasmus+

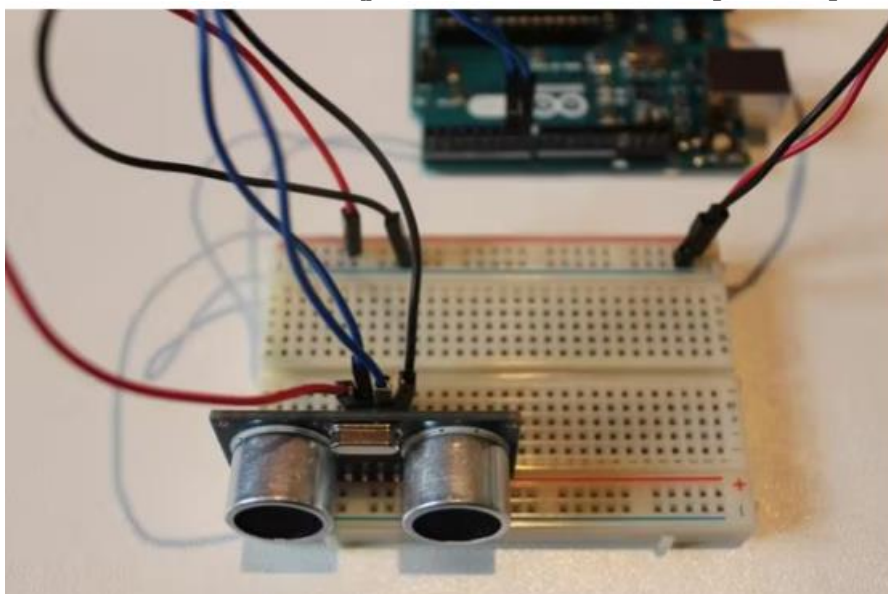
Zielone przewody łączą diody LED w następujący sposób: dioda dodatnia z diodą ujemną do kanału ujemnego na płytce drukowanej, za pomocą opornika 220  $\Omega$

## Krok 3: Montaż płytki drukowanej



Pierwszy pin 5V i GND z Arduino do płytki drukowanej.

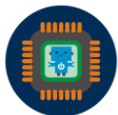
## Krok 4: Montaż - czujnik ultradźwiękowy



Czujnik ultradźwiękowy HC-SR04! Umieść czujnik ultradźwiękowy stroną do góry, jak najdalej po prawej stronie.

Podłącz:





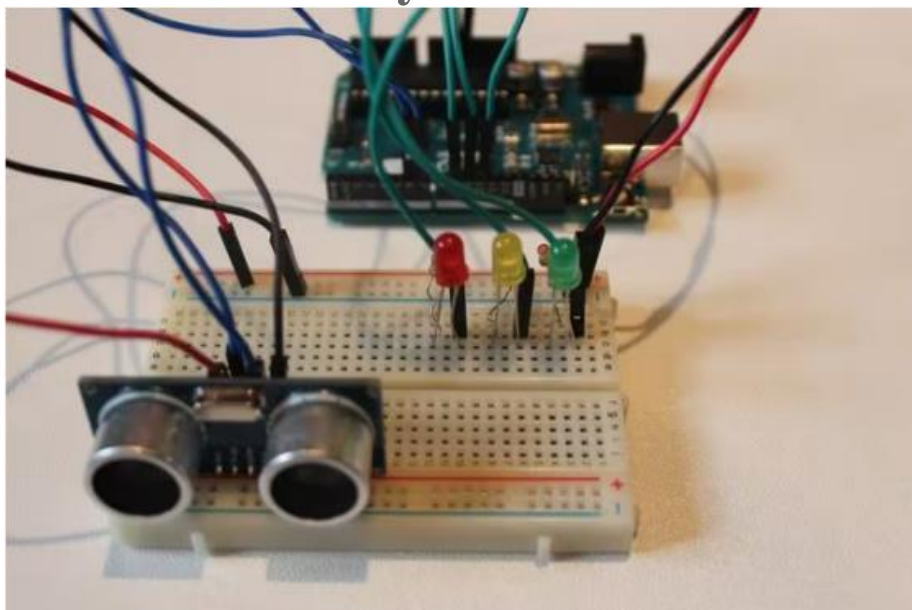
# ROBOSTEM Project

Agreement no: 2019-1-RO01-KA202-063965

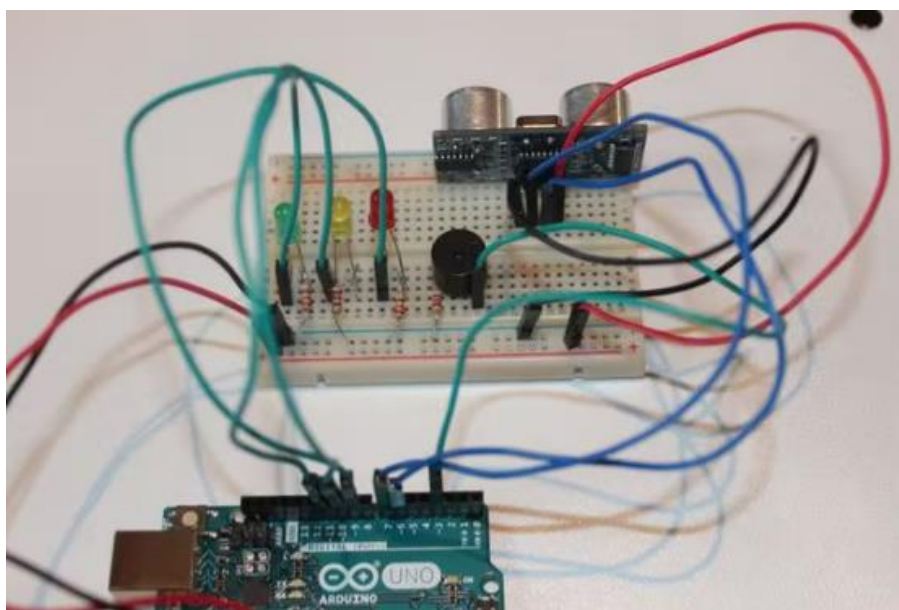


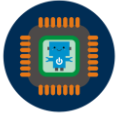
- pin GND z czujnika ultradźwiękowego do kanału ujemnego na płytce drukowanej.
- Połącz pin Trig czujnika z pinem 2 w Arduino
- Połącz pin Echo na czujniku z pinem 3 w Arduino.
- Połącz pin VCC czujnika ultradźwiękowego z kanałem dodatnim na płytce drukowanej.

## Krok 5: Montaż - diody LED



## Krok 6: Montaż - brzęczyk



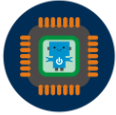


# ROBOSTEM Project

Agreement no: 2019-1-RO01-KA202-063965



```
const int trigPin = 2;
const int echoPin = 3;
const int LEDlampRed = 4;
const int LEDlampYellow = 5;
const int LEDlampGreen = 6;
const int buzzer = 7;
int dźwięk = 500;
void setup() {
  Serial.begin (9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(LEDlampRed, OUTPUT);
  pinMode(LEDlampYellow, OUTPUT);
  pinMode(LEDlampGreen, OUTPUT);
  pinMode(buzzer, OUTPUT);
}
void loop() {
  long durationindigit, distanceincm;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
```



# ROBOSTEM Project

Agreement no: 2019-1-RO01-KA202-063965



```
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
durationindigit = pulseIn(echoPin, HIGH);
distanceincm = (durationindigit * 0,034) / 2;
if (distanceincm > 50) {
    digitalWrite(LEDlampGreen, LOW);
    digitalWrite(LEDlampYellow, LOW);
    digitalWrite(LEDlampRed, LOW);
    noTone (brzęczyk);
}
else if (distanceincm <= 50 && distanceincm > 20) {
    digitalWrite(LEDlampGreen, HIGH);
    digitalWrite(LEDlampYellow, LOW);
    digitalWrite(LEDlampRed, LOW);
    noTone (brzęczyk);
}
else if (distanceincm <= 20 && distanceincm > 5) {
    digitalWrite(LEDlampYellow, HIGH);
    digitalWrite(LEDlampGreen, HIGH);
    digitalWrite(LEDlampRed, LOW);
    ton(buzzer, 500);
}
else if (distanceincm <= 0) {
    digitalWrite(LEDlampGreen, LOW);
    digitalWrite(LEDlampYellow, HIGH);
    digitalWrite(LEDlampRed, LOW);
    noTone (brzęczyk);
}
else {
    digitalWrite(LEDlampGreen, HIGH);
    digitalWrite(LEDlampYellow, HIGH);
    ton(buzzer, 1000);
    digitalWrite(LEDlampRed, HIGH);
    delay(300);
    digitalWrite(LEDlampRed, LOW);
}
Serial.print(distanceincm);
Serial.println(" cm");
delay(300);
}
```