



POLITECHNIKA OPOLSKA

PW VI – Podstawy sztucznej inteligencji

Kierunek studiów:	Informatyka I-go stopnia	Rok studiów:	III
Numer grupy:	4		
Rok akademicki:	2024/2025	Semestr:	V

Temat:

System monitorowania bezpieczeństwa domu

Nr indeksu	Imię i nazwisko	Data oddania I	Data oddania II	OCENA
104459	Yurii Yaremchuk	16.12.2024		
104458	Dmytro Volynski	16.12.2024		
103156	Yurii Vakhmistrov	16.12.2024		

Termin zajęć:		Prowadzący:
dzień:	Poniedziałek	Marcin Kowol
godzina:	9:15 - 10:00	

Opis projektu: System monitorowania bezpieczeństwa domu

Cel projektu

Celem systemu jest monitorowanie bezpieczeństwa domu poprzez wykrywanie niepożądanych zdarzeń takich jak ruch w pomieszczeniu, otwarcie drzwi i okna, oraz monitorowanie temperatury. System sygnalizuje alarm wizualnie (LED), dźwiękowo (buzzer) i tekstowo na wyświetlaczu LCD.

Główne komponenty

1. **Arduino Uno** - centralna jednostka sterująca.
2. **Czujnik ruchu PIR** - wykrywa ruch w pomieszczeniu.
3. **Czujnik otwarcia okna (Slide Switch)** - wykrywa stan otwarcia/zamknięcia okna.
4. **Czujnik otwarcia drzwi (Slide Switch)** - wykrywa stan otwarcia/zamknięcia drzwi.
5. **Czujnik temperatury TMP36** - mierzy temperaturę w pomieszczeniu.
6. **LCD 16x2 (I2C)** - wyświetla informacje o stanie systemu.
7. **Buzzer** - emituje dźwięk w razie alarmu.
8. **LED-y (2 sztuki)**:
 1. Jeden dla ogólnego alarmu.
 2. Drugi dla sygnalizacji otwartego okna.
9. **Przycisk resetu** - umożliwia ręczne wyłączenie alarmu.

Jak działają komponenty

- **Czujnik ruchu PIR:** Wykrywa ruch i aktywuje alarm.
- **Slide Switch:**
 - Jeden monitoruje stan okna (otwarte/zamknięte).
 - Drugi monitoruje stan drzwi (otwarte/zamknięte).
- **Czujnik TMP36:** Konwertuje sygnał analogowy na temperaturę w °C, a alarm jest uruchamiany, gdy temperatura przekracza 50°C.
- **LCD 16x2:** Wyświetla aktualny stan systemu, np. temperaturę, stan drzwi i okna.
- **LED i Buzzer:** Sygnalizują alarm wizualnie i dźwiękowo.
- **Przycisk resetu:** Wyłącza alarm i resetuje stan systemu.

Funkcjonalność systemu

1. **Wykrywanie ruchu:** Gdy czujnik PIR wykryje ruch, system aktywuje alarm i wyświetla na LCD informację: „Ruch wykryty”.
2. **Ochrona okien i drzwi:** Jeśli okno lub drzwi zostaną otwarte (stan LOW na Slide Switch), system uruchamia alarm, zapala odpowiednie LED i wyświetla na LCD: „Window OPEN!” lub „Drzwi otwarte!”.
3. **Monitorowanie temperatury:** System mierzy temperaturę. Gdy temperatura przekroczy 50°C, uruchamia alarm z komunikatem: „Temp High!”.
4. **Reset alarmu:** Przycisk resetu umożliwia wyłączenie alarmu.

Jak działa system

1. System uruchamia się i wyświetla na LCD: „System Ready”.
2. System stale monitoruje stan czujników (PIR, drzwi, okno, temperatura).
3. W razie wykrycia zdarzenia (ruch, otwarte drzwi/okno, wysoka temperatura), uruchamiany jest alarm:
 - Buzzer emituje dźwięk.
 - LED zapala się.
 - LCD informuje o przyczynie alarmu.
4. Po naciśnięciu przycisku reset alarm zostaje wyłączony.

Problemy, które system rozwiązuje

- Zabezpieczenie domu przed intruzami poprzez monitorowanie ruchu.
- Ostrzeżenie o otwartych drzwiach lub oknach.
- Powiadomienie o niebezpiecznej temperaturze w pomieszczeniu.

Zalety systemu

- Wielofunkcyjność (ruch, drzwi, okno, temperatura).
- Wizualna, dźwiękowa i tekstowa sygnalizacja alarmu.
- Niski koszt realizacji na bazie Arduino.
- Łatwość rozbudowy o dodatkowe czujniki.

Możliwe ulepszenia

1. Zastosowanie dodatkowych modułów komunikacji, np. Wi-Fi (ESP8266), do powiadomień na telefon.
2. Dodanie czujnika gasu
3. Dodanie akumulatora do pracy w przypadku braku zasilania.

Podział zadań

Yurii Yaremchuk:

- Projektowanie obwodu elektrycznego w Tinkercad.
- Konfiguracja czujników drzwi i okien (Slide Switch).

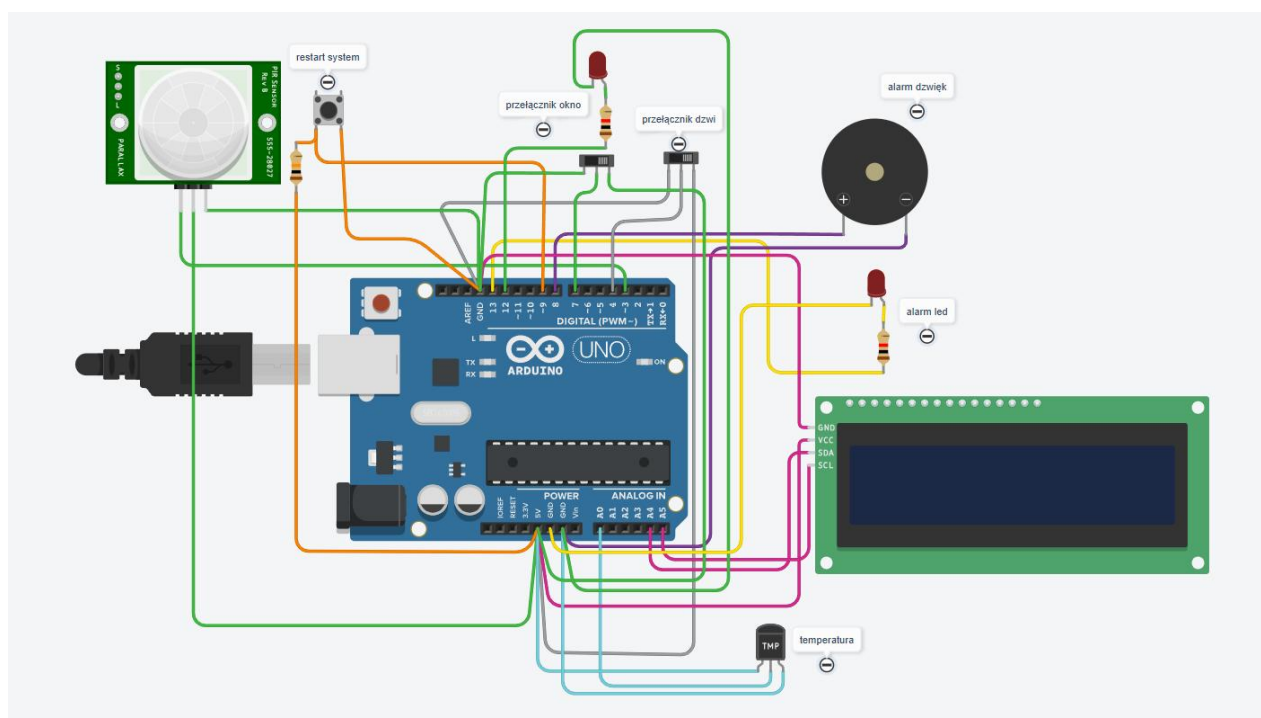
Yurii Vakhmistrov:

- Programowanie Arduino, w tym obsługa LCD i logika resetu alarmu.
- Kalibracja czujnika temperatury TMP36.

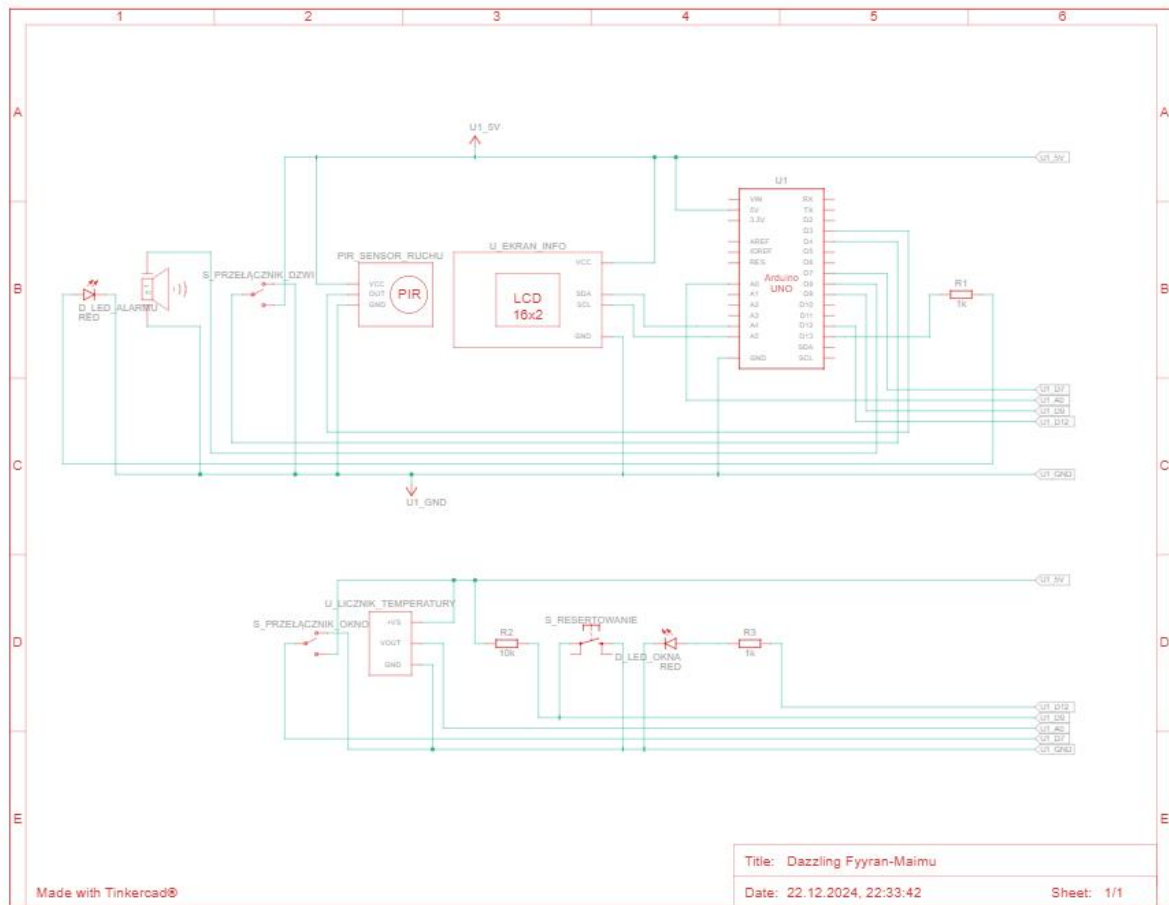
Dmytro Volynski:

- Testowanie systemu i optymalizacja kodu.
- Weryfikacja działania komponentów w symulacji.

➤ Widok układu



➤ **Widok schematyczny**



➤ **Lista komponentów**

Name	Quantity	Component
U1	1	Arduino Uno R3
U_Ekran Info	1	PCF8574-based, 39 (0x27) LCD 16 x 2 (I2C)
PIEZO_Alarm	1	Piezo
PIR_Sensor Ruchu	1	PIR Sensor
S_Przełącznik dzwi S_Przełącznik okno	2	Slideswitch
R1 R3	2	1 kΩ Resistor
D_Led Alarmu D_LED Okna	2	Red LED
U_Licznik temperatury	1	Temperature Sensor [TMP36]
S_Resertowanie	1	Pushbutton
R2	1	10 kΩ Resistor

➤ Kod systemu

```

1 #include <Wire.h>
2 #include <LiquidCrystal_I2C.h>
3
4 // Piny
5 const int pirPin = 3;           // Czujnik PIR
6 const int doorSensorPin = 4;    // Czujnik drzwi
7 const int windowSensorPin = 7;  // Czujnik okna (przełącznik suwakowy)
8 const int buzzerPin = 8;        // Buzzer
9 const int ledPin = 13;          // Dioda LED
10 const int windowLedPin = 12;    // Dioda LED dla wskazania stanu okna
11 const int tempPin = A0;         // Czujnik temperatury
12 const int resetButtonPin = 9;   // Przycisk do resetowania alarmu
13
14 // Wyświetlacz LCD
15 LiquidCrystal_I2C lcd(0x27, 16, 2);
16
17 // Zmienna dla stanu alarmu
18 bool alarmActive = false;
19 bool windowAlarm = false; // Dodatkowa zmienna dla stanu alarmu okna
20
21 void setup() {
22     // Konfiguracja pinów
23     pinMode(pirPin, INPUT);
24     pinMode(doorSensorPin, INPUT_PULLUP);
25     pinMode(windowSensorPin, INPUT_PULLUP); // Czujnik okna
26     pinMode(buzzerPin, OUTPUT);
27     pinMode(ledPin, OUTPUT);
28     pinMode(windowLedPin, OUTPUT); // Dioda LED dla okna
29     pinMode(resetButtonPin, INPUT_PULLUP); // Przycisk z podciągającym rezystorem
30
31     // Inicjalizacja LCD
32     lcd.init();
33     lcd.backlight();
34     lcd.setCursor(0, 0);
35     lcd.print("System Ready");
36     delay(2000);
37     lcd.clear();
38
39     // Inicjalizacja Serial Monitor
40     Serial.begin(9600);
41 }
42
43 void loop() {
44     // Odczyt danych z czujników
45     int pirState = digitalRead(pirPin);
46     int doorState = digitalRead(doorSensorPin);
47     int windowState = digitalRead(windowSensorPin); // Stan okna
48     int tempValue = analogRead(tempPin);
49     float voltage = tempValue * (5.0 / 1023.0);
50     float temperatureC = (voltage - 0.5) * 100.0;
51
52     // Aktualizacja stanu alarmu dla okna
53     if (windowState == LOW) { // Jeśli okno otwarte
54         windowAlarm = true;
55         digitalWrite(windowLedPin, HIGH); // Włącz diodę LED dla okna
56     } else { // Jeśli okno zamknięte
57         windowAlarm = false;
58         digitalWrite(windowLedPin, LOW); // Wyłącz diodę LED dla okna
59     }
60
61     // Sprawdzanie, czy trzeba aktywować alarm ogólny
62     if (temperatureC > 50.0 || pirState == HIGH || doorState == LOW || windowAlarm) {
63         alarmActive = true; // Aktywuj alarm
64     } else {
65         alarmActive = false; // Wyłącz alarm, jeśli żadne warunki nie są spełnione
66     }
67
68     // Sprawdzanie stanu alarmu
69     if (alarmActive) {
70         lcd.setCursor(0, 0);
71         lcd.print("ALARM ACTIVE! ");
72         digitalWrite(ledPin, HIGH); // Włącz ogólną diodę LED
73         digitalWrite(buzzerPin, HIGH); // Włącz buzzer
74

```



```

75 // Wyświetlanie powodu alarmu
76 if (temperatureC > 50.0) {
77     lcd.setCursor(0, 1);
78     lcd.print("Temp High! ");
79 } else if (pirState == HIGH) {
80     lcd.setCursor(0, 1);
81     lcd.print("Ruch wykryty! ");
82 } else if (doorState == LOW) {
83     lcd.setCursor(0, 1);
84     lcd.print("Drzwi otwarte!");
85 } else if (windowState == LOW) { // Jeśli alarm dla okna aktywny
86     lcd.setCursor(0, 1);
87     lcd.print("Window OPEN! ");
88 }
89
90 // Sprawdzanie przycisku do resetowania alarmu
91 if (digitalRead(resetButtonPin) == LOW) { // Jeśli przycisk wciśnięty
92     alarmActive = false; // Zresetuj stan alarmu
93     digitalWrite(ledPin, LOW); // Wyłącz ogólną diodę LED
94     digitalWrite(buzzerPin, LOW); // Wyłącz buzzer
95     lcd.setCursor(0, 0);
96     lcd.print("System Reset ");
97     delay(2000);
98     lcd.clear();
99 }
100 } else {
101     // Jeśli brak alarmu
102     lcd.setCursor(0, 0);
103     lcd.print("Temp: ");
104     lcd.print(temperatureC);
105     lcd.print(" C ");
106     lcd.setCursor(0, 1);
107     if (windowState == HIGH) { // Okno zamknięte
108         lcd.print("System OK ");
109     } else { // Okno otwarte
110         lcd.print("Window OPEN! ");
111     }
112     digitalWrite(ledPin, LOW); // Wyłącz ogólną diodę LED
113     digitalWrite(buzzerPin, LOW); // Wyłącz buzzer
114 }
115
116 // Wysyłanie danych do Serial Monitor
117 Serial.print("Temp: ");
118 Serial.print(temperatureC);
119 Serial.print(" C, PIR: ");
120 Serial.print(pirState);
121 Serial.print(", Door: ");
122 Serial.print(doorState);
123 Serial.print(", Window: ");
124 Serial.println(windowState);
125
126 delay(1000); // Opóźnienie 1 sekunda
127 }

```

Podsumowanie

Stworzony system monitorowania bezpieczeństwa domu to wszechstronny projekt bazujący na Arduino, który skutecznie łączy monitorowanie ruchu, ochronę drzwi i okien oraz kontrolę temperatury. Projekt jest łatwy w realizacji i może być rozbudowany o nowe funkcjonalności. Każdy członek zespołu wniósł swój wkład, co pozwoliło na stworzenie funkcjonalnego i efektywnego rozwiązania.