

Pametni sistem detekcije plina s dodatnim funkcionalnostima i tasterom za isključivanje alarma

Radila: Grabić Majra

Sadržaj

| | |
|--------------------------------------|---|
| Opis | 3 |
| Funkcionalnosti | 3 |
| Detekcija plina | 3 |
| Prikaz stanja na LCD display-u | 3 |
| Senzori pokreta | 3 |
| LED indikatori | 4 |
| Taster za isključivanje alarma | 4 |
| Tehnički zahtjevi | 4 |
| Arhitektura sistema | 4 |
| Kod projekta | 5 |

Opis

Cilj ovog projekta je stvoriti pametan sistem koji će detektirati prisutnost gasa u prostoriji te uključivati dodatne funkcionalnosti poput prikazivanja stanja sistema na LCD display-u, upotrebe senzora pokreta, LED dioda i dodavanja tastera za isključivanje alarma.

Kontekst upotrebe pomaže da se bolje razumiju zahtevi i ciljevi koji okružuju primjenu pametnih projekata. To omogućava inženjerima da prilagode dizajn i funkcionalnosti kako bi zadovoljili specifične potrebe korisnika u datom okruženju. U kontekstu upotrebe pametnih projekata s Arduino platformom za detekciju plina, senzora pokreta, alarma i tastera za isključivanje alarma spada:

1. **Specifičnosti okoline:** Da li je sistem namenjen za kućnu upotrebu, industrijsko okruženje ili neko drugo mjesto? Različite okoline zahtijevaju prilagođene parametre i funkcionalnosti.
2. **Potrebe korisnika:** Koje rizike želimo otkloniti detekcijom plina? Kako korisnici žele interagovati sa sistemom? Da li postoji potreba za trenutnim obaveštenjem ili automatizovanim reakcijama?
3. **Ciljevi sistema:** Šta sistem tačno pokušava postići? Da li je fokus na bezbjednosti, energetske efikasnosti ili nečemu drugom?
4. **Interakcija sa drugim sistemima:** Da li sistem komunicira sa drugim pametnim uređajima, aplikacijama ili servisima? Postoji li integracija sa drugim IoT uređajima ili pametnim kućnim sistemima?

U kontekst upotrebe našeg projekta ulazile su sljedeće specifikacije:

1. Sistem je namijenjen za kućnu upotrebu
2. Pri detekciji plina i/ili pokreta mora se oglasiti alarm, na koji korisnik može reagovati pritiskom tastera i isključivanjem alarma.
3. Cilj sistema jeste na bezbjednosti kućnih prostorija. Plin je uzročnik velikog broja smrtnih slučajeva kada dođe do njegovog neopaženog isticanja.
4. Sistem nema komunikaciju sa drugim uređajima, nego postoji kao autonoman sistem.

Funkcionalnosti

Detekcija plina

- Integrirati gasni senzor koji će detektirati prisutnost gasa u prostoriji.
- U slučaju detekcije gasa, aktivirati alarmni signal i/ili prikazati obavijest na LCD display-u.

Prikaz stanja na LCD display-u

- Ugraditi LCD display kako bi prikazao trenutno stanje sistema.
- Prikazivati informacije o detekciji gasa i stanju sistema.

Senzori pokreta

- Dodati senzore pokreta za dodatnu sigurnost.

- Aktivirati alarmni signal ili pokazati obavijest na LCD display-u kada se detektira pokret, osim u situacijama kada je plin detektiran.

LED indikatori

- Ugraditi LED diode koje će svijetliti u različitim bojama kako bi označile trenutno stanje sistema.
- Zelena svjetla za normalno stanje, crvena za detekciju gasa.

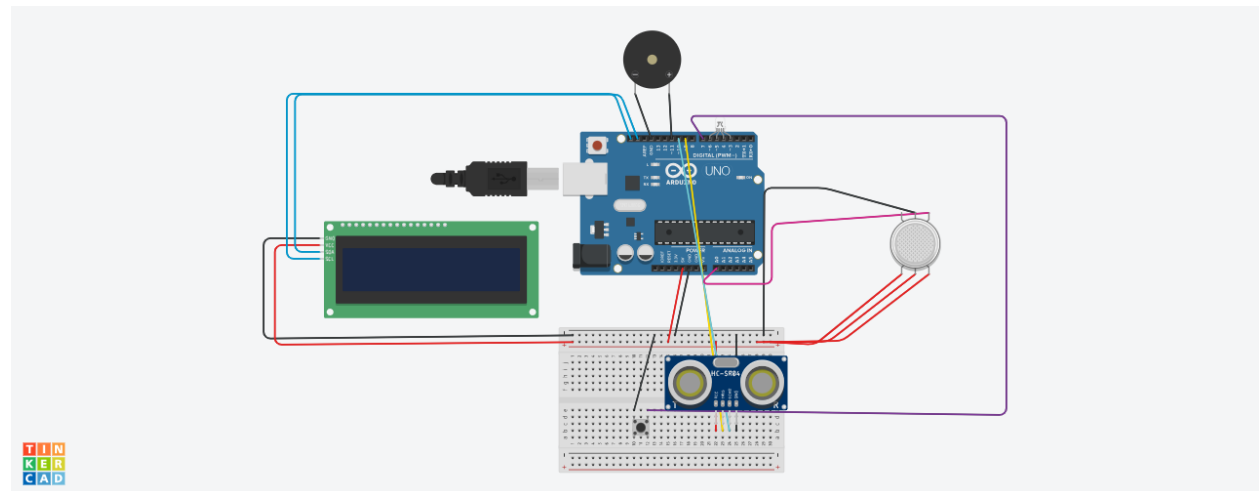
Taster za isključivanje alarma

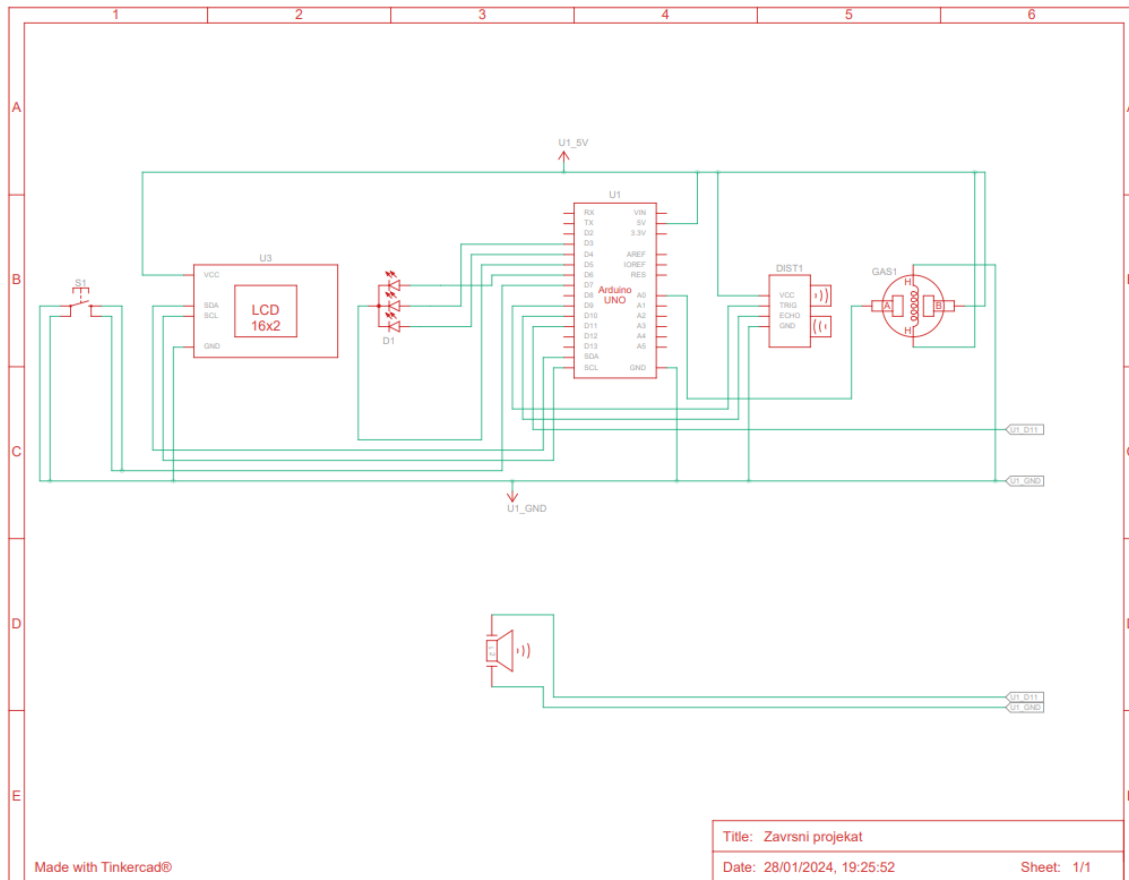
- Dodati taster koji će omogućiti korisniku isključivanje alarma nakon što je gas detektiran.
- Omogućiti dodatnu signalizaciju prilikom isključivanja alarma (ispis na display).

Tehnički zahtjevi

- Korištenje Arduino platforme za kontrolu dimnog senzora, LCD display-a, senzora pokreta, LED dioda i tastera.
- Implementacija programskog koda za obradu podataka s senzora gasa i senzora pokreta.
- Povezivanje s LCD zaslonom kako bi se prikazivala relevantna stanja sistema.
- Upotreba LED dioda za vizualnu signalizaciju.
- Integracija tastera i programiranje za isključivanje alarma.

Arhitektura sistema





Kod projekta

```
/*Uključivanje biblioteka funkcija za display i melodije buzzera*/
#include <LiquidCrystal_I2C.h>
#include "pitches.h"

const int trigPin = 9;
const int echoPin = 10; //pinovi za senzor pokreta
long udaljenost;
long trajanje; //varijable za racunanje udaljenosti i trajanja pokreta

LiquidCrystal_I2C lcd(0x27, 16, 2); //definisanje varijable lcd za korištenje u nastavku
int val = A0; //input za senzor plina
int MAX = 400; //max vrijednost plina do koje ga senzor ignorise

int buzzerPin = 13; //pin na kom se nalazi + buzzera
const int buttonPin = 7; //pin na kom se nalazi taster
```

/*Funkcija sigurno poziva se kada je stanje sistema sigurno, tj. kada nema pokreta ni plina.
Postavlja boju diode na zelenu, isključuje ton buzzera i na displayu se ispisuje da je siguran sistem.*/

```
void sigurno(){  
    noTone(buzzerPin);  
    digitalWrite(3, LOW);  
    digitalWrite(4, HIGH);  
    digitalWrite(5, HIGH);  
    lcd.clear();  
    lcd.println("Sistem siguran");  
}
```

/*Funkcija button poziva se kada je taster pritisnut.
Isključuje buzzer, postavlja boju diode na zelenu i na display ispisuje korisnikov prekid alarma.
Nakon toga čeka 2 sekunde i sistem vraća u sigurno stanje.*/

```
void button(){  
    digitalWrite(buzzerPin, HIGH);  
    noTone(buzzerPin);  
    digitalWrite(3, LOW);  
    digitalWrite(4, HIGH);  
    digitalWrite(5, HIGH);  
    lcd.clear();  
    lcd.println("Korisnicki prekinuto");  
    delay(3000);  
    sigurno();  
}
```

```
void setup()  
{  
    lcd.init();  
    lcd.backlight();  
    Serial.begin(9600);  
  
    pinMode(val, INPUT);  
    pinMode(3, OUTPUT); //pin za zelenu boju  
    pinMode(4, OUTPUT); //pin za plavu boju  
    pinMode(5, OUTPUT); //pin za crvenu boju  
    pinMode(6, OUTPUT); //dovod napajanja na diodu  
    pinMode(trigPin, OUTPUT);  
    pinMode(echoPin, INPUT);  
    pinMode(buzzerPin, OUTPUT);  
    pinMode(buttonPin, INPUT_PULLUP);  
    digitalWrite(6, HIGH);  
}
```

```

void loop()
{
    val = analogRead(A0);
    digitalWrite(trigPin, LOW);
    delay(30);
    digitalWrite(trigPin, HIGH);
    delay(30);
    digitalWrite(trigPin, LOW);
    trajanje = pulseIn(echoPin, HIGH); //račun trajanja i udaljenosti
    udaljenost = trajanje * 0.034 / 2;

    if (val >= MAX) { //ako je senzor očitao veći nivo plina od MAX
        digitalWrite(3,HIGH);
        digitalWrite(4, HIGH);
        digitalWrite(5, LOW); //postavlja crvenu boju diode
        digitalWrite(buzzerPin, LOW);
        tone(buzzerPin, 443, trajanje); //definiše melodiju buzzera i trajanje kao
        trajanje očitavanja

        if(digitalRead(buttonPin)==LOW){ //ako je pritisnut taster poziva funkciju
        button
            button();
        }

        lcd.clear();
        lcd.println("Pozor! Plin");
    }else if(val<MAX && udaljenost < 10){ //ako je senzor očitao manji nivo plina od
    MAX, ali postoji pokret
        digitalWrite(5, LOW);
        digitalWrite(4, HIGH);
        digitalWrite(3, HIGH);
        digitalWrite(buzzerPin, LOW);
        tone(buzzerPin, 443, trajanje);

        if(digitalRead(buttonPin)==LOW){
            button();
        }

        lcd.clear();
        lcd.println("Pokret");
    } else{ //u slučaju da nema ni pokreta ni plina, stanje sistema je sigurno
        sigurno();
    }
}

```