Smart Home Automation and Security System with Human Detection

Jerin Romijah Tuli
Roll: 2003056
Section: A
Department of CSE
Rajshahi University Of Engineering And Technology

October 7, 2024

1 Introduction

In this project, we develop an integrated home automation and security system utilizing an Arduino Uno microcontroller. The system enhances home security by first detecting human presence using an ultrasonic sensor. Upon detection, it prompts the user to enter a password via a keypad to unlock the door. Once unlocked, the system transitions into an automation mode, controlling lighting based on ambient light and occupancy, while also monitoring gas levels for safety. A Liquid Crystal Display (LCD) provides real-time feedback to the user regarding the system's status and sensor readings.

2 Components Used

The system comprises the following components:

- **Arduino Uno**: Central microcontroller managing inputs from sensors and keypad, and controlling outputs such as the door lock and relay modules.
- **Liquid Crystal Display (LCD)**: Displays user prompts, lock status, and sensor data.
- **4x4 Keypad**: Input device for entering the unlock password.
- **Ultrasonic Sensor (HC-SR04)**: Detects human presence by measuring distance.
- **PIR Sensor**: Detects motion to determine occupancy for home automation.
- **Gas Sensor (MQ-2)**: Monitors gas levels to detect potential leaks.
- **Light Dependent Resistor (LDR)**: Measures ambient light levels to control lighting.
- **Relay Module**: Controls external electrical devices like lights based on sensor inputs.

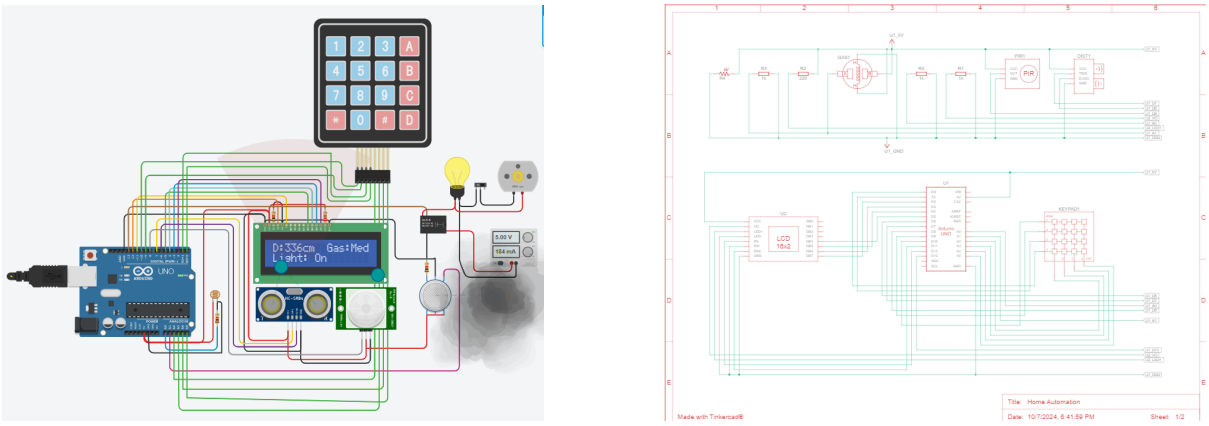
- **EEPROM**: Stores the master password securely for the lock mechanism.

3 System Design

The system is architected to prioritize security by incorporating human detection before granting access through a password mechanism. Upon successful authentication, it seamlessly transitions into an automation mode to manage home utilities efficiently.

3.1 Circuit Overview

The Arduino Uno serves as the core of the system, interfacing with various sensors and output modules. The ultrasonic sensor is connected to digital pins 6 and 7 for triggering and echo, respectively. The LCD is interfaced using digital pins 2, 3, 4, 5, 11, and 12. The keypad is connected to analog pins A2 to A5 and digital pins 0, 1, 9, and 10. Sensors like PIR, LDR, and gas sensor are connected to designated analog and digital pins, while the relay module controls the lighting system based on sensor inputs.



(a) Circuit design (circuit view)

(b) Circuit design (Schematic view)

Figure 1: Circuit Diagram of the Smart Home Automation and Security System with Human Detection

3.2 Operation

1. **Human Detection**: The ultrasonic sensor continuously measures the distance to detect human presence within a predefined threshold (e.g., 20 cm). If a human is detected, the system prompts the user to enter a password.

2. **Password Entry**: Upon human detection, the LCD displays a password entry prompt. The user inputs the password via the keypad. The system verifies the entered password against the stored master password in the EEPROM.

3. **Authentication**:

   - **Correct Password**: The door unlocks, and the system enters automation mode, controlling home utilities based on sensor data.
   - **Incorrect Password**: An error message is displayed, and the system prompts for password entry again.

4. **Home Automation**: In automation mode, the system manages lighting based on ambient light detected by the LDR and occupancy detected by the PIR sensor. It also monitors gas levels using the gas sensor, triggering alerts if dangerous levels are detected.

4 Code Implementation

The following Arduino code governs the operation of the home automation and security system. It integrates human detection, password authentication, and automation features, ensuring a secure and efficient smart home environment.

```
#include <LiquidCrystal.h>
#include <Keypad.h>
#include <EEPROM.h>

// LCD and Keypad for door lock
#define Password_Length 5
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

const byte ROWS = 4;
const byte COLS = 4;
char keys[ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byte rowPins[ROWS] = {10, 9, 1, 0};
byte colPins[COLS] = {A2, A3, A4, A5};
Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

char Data[Password_Length];
char Master[Password_Length];
byte data_count = 0;
char key;
byte mode = 0;  // 0 for locked, 1 for unlocked

// Sensor and relay pins for home automation
int releNO = 13;
int inputPir = 8;
int sensorLDR = A0;
int PINO_SGAS = A1;
int distanceThreshold = 20;  // Threshold set to 20 cm
int cm = 0;
long readUltrasonicDistance(int triggerPin, int echoPin);

// Variables to store previous sensor values
int previousCm = -1;
int previousGasLevel = -1;

// Variable to track human detection
bool humanDetected = false;
bool enterPasswordDisplayed = false;  // New flag to control LCD
    updates for password

```

```
45 // Function for checking password
46 void Check_EEPROM() {
47   EEPROM.get(0, Master);
48   if (Master[0] == 0 && Master[1] == 0 && Master[2] == 0 && Master[3]
       == 0) {
49     char FirstTimePassword[] = {'1', '2', '3', '4'};
50     EEPROM.put(0, FirstTimePassword);
51     EEPROM.get(0, Master);
52   }
53 }
54
55 void setup() {
56   Serial.begin(9600);
57   lcd.begin(16, 2);
58   pinMode(releNO, OUTPUT);
59   pinMode(inputPir, INPUT);
60   pinMode(sensorLDR, INPUT);
61   Check_EEPROM();
62 }
63
64 void loop() {
65   // Read distance from ultrasonic sensor
66   cm = 0.01723 * readUltrasonicDistance(7, 6);
67
68   // Human detected within threshold distance
69   if (!humanDetected && cm <= distanceThreshold) {
70     lcd.clear();
71     lcd.setCursor(0, 0);
72     lcd.print("Human detected!");
73     delay(1000);  // Wait for 1 second
74     humanDetected = true;  // Mark human as detected
75     lcd.clear();
76     enterPasswordDisplayed = false;  // Reset flag for password prompt
77   }
78
79   if (humanDetected && mode == 0) {  // Prompt for password after
       detection
80     if (!enterPasswordDisplayed) {
81       lcd.clear();
82       lcd.setCursor(0, 0);
83       lcd.print("Enter Password:");
84       enterPasswordDisplayed = true;  // Ensure password prompt is
           displayed only once
85     }
86
87     key = keypad.getKey();
88     if (key) {
89       Data[data_count] = key;
90       lcd.setCursor(4 + data_count, 1);
91       lcd.print("*");
92       data_count++;
93     }
94
95     if (data_count == Password_Length - 1) {
96       if (!strcmp(Data, Master)) {  // Password correct
97         lcd.clear();
98         lcd.setCursor(0, 0);
99         lcd.print("Unlocking...");
```

```
100         delay(2000);
101         lcd.clear();
102         lcd.setCursor(0, 0);
103         lcd.print("Door Unlocked");
104         mode = 1;  // Switch to home automation
105         humanDetected = false;  // Reset human detection after
                unlocking
106         delay(2000);
107         lcd.clear();
108         enterPasswordDisplayed = false;  // Reset flag
109       } else {  // Incorrect password
110         lcd.clear();
111         lcd.setCursor(0, 0);
112         lcd.print("Incorrect Pass");
113         delay(2000);
114         lcd.clear();
115         lcd.setCursor(0, 0);
116         lcd.print("Enter Password:");  // Re-display password prompt
117         enterPasswordDisplayed = true;  // Keep password prompt
                displayed
118       }
119       data_count = 0;  // Reset password input
120     }
121   }
122
123   // Home automation activates only after the door is unlocked
124   if (mode == 1) {
125     int val = digitalRead(inputPir);
126     int resuldoSensorLDR = analogRead(sensorLDR);
127     int gasLevel = analogRead(PINO_SGAS);
128
129     // Update LCD for distance reading
130     if (cm != previousCm) {
131       lcd.setCursor(0, 0);
132       lcd.print("D:");
133       lcd.print(cm);
134       lcd.print("cm  ");
135       previousCm = cm;  // Store the current distance
136     }
137
138     // LDR and PIR control
139     if (resuldoSensorLDR < 600 && val == HIGH) {
140       digitalWrite(releNO, HIGH);
141       lcd.setCursor(0, 1);
142       lcd.print("Light: On ");
143       delay(5000);
144     } else {
145       digitalWrite(releNO, LOW);
146       lcd.setCursor(0, 1);
147       lcd.print("Light: Off");
148       delay(300);
149     }
150
151     // Update LCD for gas sensor reading
152     if (gasLevel != previousGasLevel) {
153       lcd.setCursor(8, 0);
154       lcd.print(" Gas:");
155       if (gasLevel <= 85) {
```

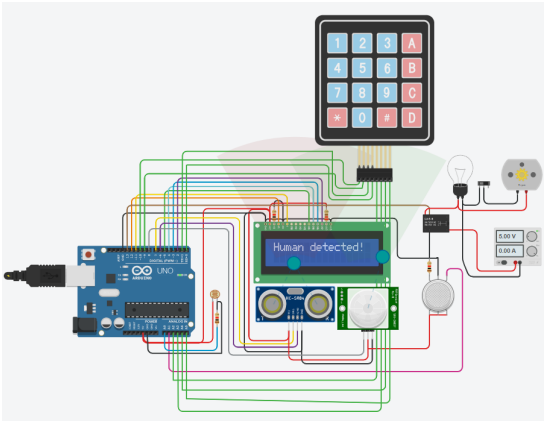
```
        lcd.print("Low  ");
      } else if (gasLevel <= 120) {
        lcd.print("Med  ");
      } else if (gasLevel <= 200) {
        lcd.print("High ");
      } else if (gasLevel <= 300) {
        lcd.print("Ext  ");
      }
      previousGasLevel = gasLevel;  // Store the current gas level
    }

    delay(250);  // Add delay for smooth display
  }
}

// Ultrasonic sensor function
long readUltrasonicDistance(int triggerPin, int echoPin) {
  pinMode(triggerPin, OUTPUT);
  digitalWrite(triggerPin, LOW);
  delayMicroseconds(2);
  digitalWrite(triggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerPin, LOW);
  pinMode(echoPin, INPUT);
  return pulseIn(echoPin, HIGH);
}
```

Listing 1: Arduino Code for Smart Home Automation and Security System

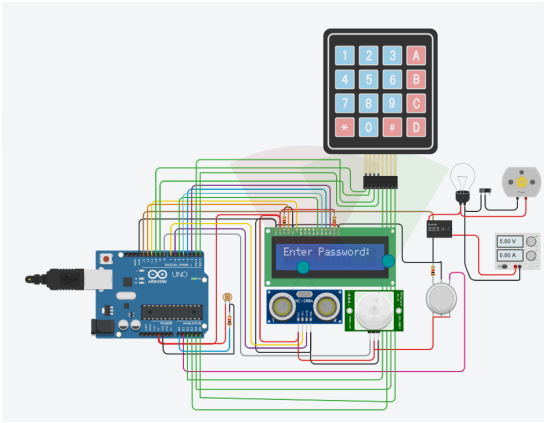
5 Results and Discussion

The implemented system successfully integrates human detection with a secure password mechanism, followed by home automation functionalities. The ultrasonic sensor reliably detects human presence within the set threshold, prompting the LCD to request password entry. Upon entering the correct password, the system unlocks the door and transitions into automation mode. The LCD provides clear feedback throughout the process, enhancing user interaction.

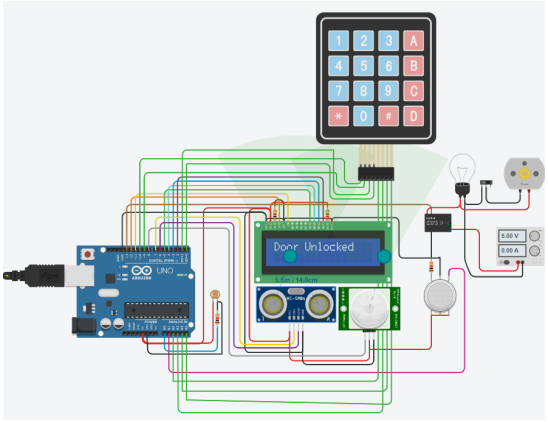
In automation mode, the LDR and PIR sensor effectively control the lighting based on ambient light and occupancy, ensuring energy efficiency. The gas sensor accurately monitors gas levels, displaying appropriate warnings and ensuring safety by alerting users in case of hazardous gas concentrations. The use of EEPROM ensures that the master password is stored securely, preventing unauthorized access.



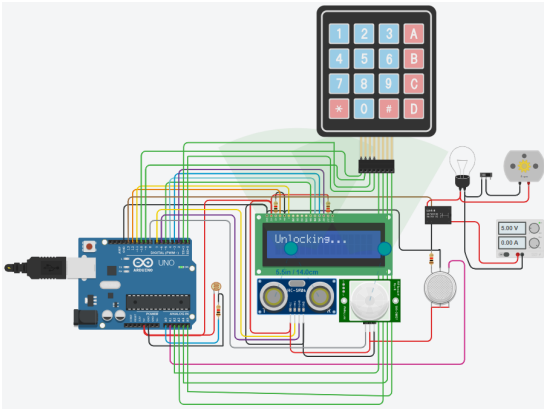
(a) Ultrasonic sensor continuously detects human presence and triggered password prompt



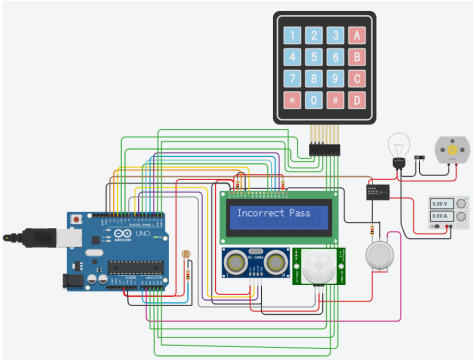
(b) House holder will give 4 digit password to unlock the door



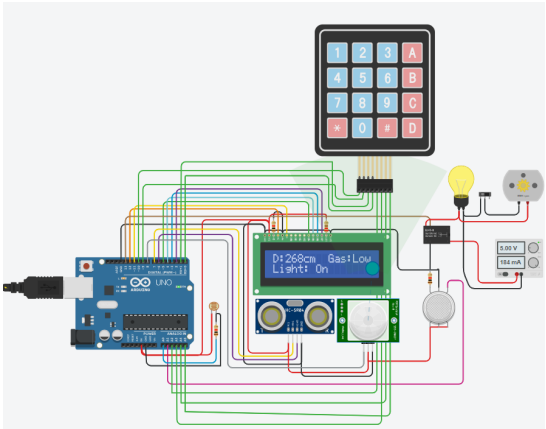
(c) Door Unlocked!



(d) Unlocking

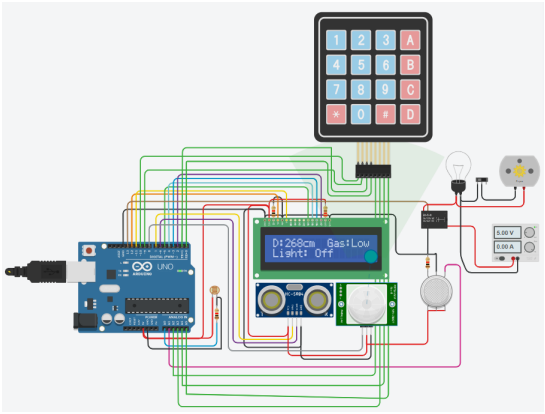


(e) The entered password is incorrect.

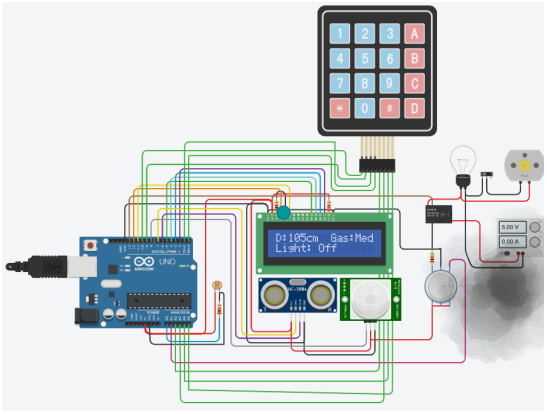


(f) Pir sensor detects human presence and switches on the light and DC fan; a Relay module acts as an external switch.

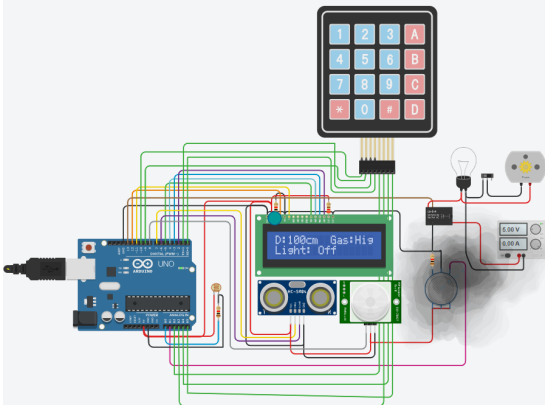
Figure 2: Operational feedback from the Smart Home Automation and Security System



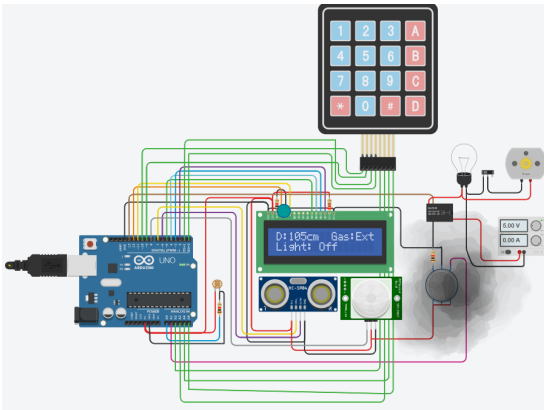
(a) Gas sensor detects the gas level continuously.



(b) Gas sensor detects the gas level and alerts the user. If there is medium level gas leakage, it shows "Med".



(c) If there is high level gas leakage, it shows "High".



(d) If the gas level is higher than the threshold, it shows "Exit".

Figure 3: Operational feedback from the Gas Sensor .

6 Conclusion

This project demonstrates the effective integration of security and automation in a smart home system using Arduino. By prioritizing human detection before authentication, the system enhances security measures. The seamless transition to home automation post-authentication optimizes energy usage and ensures environmental safety. Future enhancements could include wireless connectivity for remote monitoring, integration with mobile applications, and expanding the range of controlled devices for a more comprehensive smart home experience.

7 References

References

[1] Arduino Uno, *Arduino Documentation*. Available at: `https://www.arduino.cc/en/Guide/ArduinoUno`.

[2] LiquidCrystal Library, *Arduino Documentation*. Available at: https://www.arduino.cc/en/Reference/LiquidCrystal.

[3] Keypad Library, *Arduino Playground*. Available at: https://playground.arduino.cc/Code/Keypad/.

[4] PIR Sensor, *SparkFun Guide*. Available at: https://www.sparkfun.com/products/13285.

[5] LDR Working, *Electronics Tutorials*. Available at: https://www.electronics-tutorials.ws/io/ldr.html.

[6] Gas Sensor (MQ-2), *Datasheet*. Available at: https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-2.pdf.

[7] EEPROM Library, *Arduino Documentation*. Available at: https://www.arduino.cc/en/Reference/EEPROM.

[8] Ultrasonic Sensor HC-SR04, *SparkFun Guide*. Available at: https://www.sparkfun.com/products/15569.

[9] Relay with Arduino, *Last Minute Engineers Tutorial*. Available at: https://lastminuteengineers.com/arduino-relay-tutorial/.