>

Automated Door System Using Arduino and sensor-Based control

MAHIDUL ISLAM NIHAD
dept. Electrical and Electronic Engineering
American International UniversityBangladesh
Dhaka-1229, Bangladesh
21-45423-3 @student.aiub.edu

MD.SHAFKAT KAMAL
dept. Electrical and Electronic Engineering
American International UniversityBangladesh
Dhaka-1229, Bangladesh
21-44867-2 @student.aiub.edu

MD.LOKMAN HOSSEN JONY
dept. Electrical and Electronic Engineering
American International UniversityBangladesh
Dhaka-1229, Bangladesh
20-42052-1@student.aiub.edu

MD. LABIBUL HAQUE LABIB
dept. Electrical and Electronic Engineering
American International UniversityBangladesh
Dhaka-1229, Bangladesh
21-45252-2 @student.aiub.edu

MD. RAKIBUR RAHMAN NAYEM
dept. Electrical and Electronic Engineering
American International UniversityBangladesh
Dhaka-1229, Bangladesh
21-45233-2 @student.aiub.edu

MUAMMAR TAZWAR
dept. Electrical and Electronic Engineering
American International UniversityBangladesh
Dhaka-1229, Bangladesh
19-40019-1@student.aiub.edu

***Abstract*—** This study presents the design and implementation of a microcontroller-based automated door system utilizing an Arduino Uno, a PIR sensor (HC-SR501), and an ultrasonic sensor (HC-SR04) to enhance convenience and safety. The system integrates motion detection via the PIR sensor, which triggers a 180-degree servo motor to open the door, while the ultrasonic sensor ensures obstacle detection within a 20 cm range to prevent collisions. Testing demonstrated reliable motion detection (3–7 m range) and accurate distance measurement (±3 mm precision), though occasional false triggers occurred in environments with infrared interference or temperature fluctuations. The servo motor operated efficiently but exhibited minor delays due to sensor processing overhead. While real-time monitoring was achieved via the Arduino Serial Monitor, hardware constraints limited the inclusion of an OLED display. Overall, the system proved effective in automating door operations, balancing responsiveness and safety. Future improvements could incorporate noise-filtering algorithms, adjustable sensitivity settings, and a dedicated display module to optimize robustness and user experience. This project highlights the feasibility of low-cost, sensor-driven automation for security and energy-saving applications.

***Index Terms*— Automated door system, Arduino Uno, PIR sensor, Ultrasonic sensor, Motion detection, Obstacle detection, Servo motor, Real-time monitoring, Sensor automation, Energy-saving applications.**

I. INTRODUCTION

Automated door systems have become integral to modern infrastructure, offering enhanced accessibility, security, and energy efficiency in residential, commercial, and industrial settings. Traditional systems often rely on single-sensor inputs, which may lack adaptability to dynamic environments or fail to address safety concerns such as obstacle collisions. This project addresses these limitations by proposing a dual-sensor automated door system that combines motion detection and proximity sensing for improved reliability.

The system employs a PIR sensor (HC-SR501) to detect human movement through infrared radiation changes, triggering a 180-degree servo motor to actuate the door. Concurrently, an ultrasonic sensor (HC-SR04) monitors the door's pathway, preventing activation if obstacles are detected within 20 cm. An Arduino Uno microcontroller processes sensor inputs and coordinates motor control, ensuring seamless integration of safety and automation.

Key innovations include the use of adjustable sensor parameters (e.g., sensitivity, delay time) to tailor performance to specific environments and the incorporation of real-time feedback via serial communication. Testing under varied conditions revealed high motion detection accuracy and reliable obstacle avoidance, though challenges such as false triggers (due to environmental infrared noise) and minor servo delays were noted. The absence of a dedicated OLED displays necessitated reliance on the Arduino Serial Monitor, underscoring opportunities for hardware enhancements.

This work demonstrates the practicality of combining cost-effective sensors and microcontrollers to create adaptable automated systems. By addressing current limitations through software-based noise filtering and modular design, the project lays a foundation for scalable, user-centric automation solutions in smart infrastructure. The following sections detail the system design, component functionalities, experimental results, and proposed optimizations.

II. METHODOLOGY

The automated door system integrates an Arduino Uno, a PIR sensor (HC-SR501) for motion detection, an ultrasonic sensor (HC-SR04) for obstacle detection, and a servo motor to control the door's movement. When the PIR sensor detects motion, the system checks the ultrasonic sensor for obstacles within a 20 cm range. If no obstacle is present, the servo motor unlocks and opens the door, keeping it open for five seconds before closing it. If an object is detected within 20 cm

>

while motion is present, the door remains closed to prevent collisions.
The system is programmed using C++ in the Arduino IDE, processing sensor data in real time to automate door operation. Sensor readings are continuously monitored through the Arduino Serial Monitor to ensure proper functionality.

III. Block diagram/Circuit diagram

A. Block Diagram

For the described system, the block diagram on Fig.1 is showing the work flow:

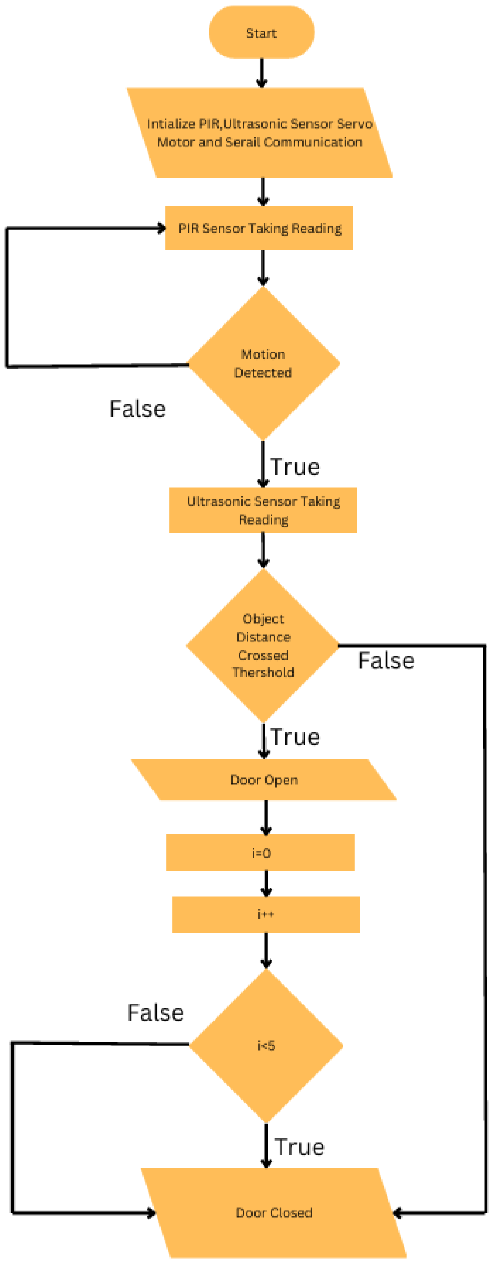

Fig.1: Block diagram of Automated door system

The system initiates with PIR sensor and ultrasonic sensor taking data. When PIR sensors detect any movement, it will then see the readings of ultrasonic sensor installed on the other side if the ultrasonic sensor detects object that is in 20 cm distance it will then open gate lock after 5 seconds it will again close the door. If the Distance of object is less than 20 cm while a motion is detected it will remain the door close.

B. Circuit Diagram

The circuit diagram at fig.2 shows how the sensors are connected to the Arduino:

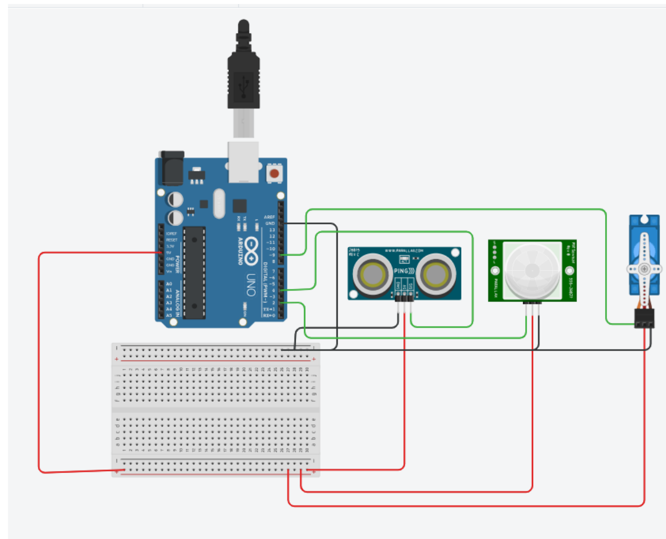

Fig.2- System circuit diagram

IV. List of components and description

A. Simulation

The system uses the components listed on TABLE I.

TABLE I

LIST OF COMPONENTS FOR THE SYSTEM

No.	Name of Equipment	Description
1	PIR sensor (HC-SR501)	To detect motion by sensing changes in infrared radiation emitted from moving objects.
2	Ultrasonic sensor (HC-SR04)	For detecting the distance to an object using sonar.
3	180-degree servo motor	Used on angular or linear position and for specific velocity.
4	Arduino Uno	Programmable open-source microcontroller board which is integrated into a variety of electronic projects.
5	I2C OLED display	To visually display text and graphics on a small screen by sending data to the display controller via the I2C communication protocol.

B. Sensor descriptions

PIR sensor (HC-SR501):

Fig.3: PIR sensor[6] .

The HC-SR501 PIR sensor is a motion detection module that detects infrared radiation changes caused by moving objects such as humans or animals. It operates by sensing variations in infrared energy within its detection range (between 3–7 meters) and outputs a HIGH signal when motion is detected. The module features adjustable sensitivity and delay time using onboard potentiometers. It operates on a 5V–20V power supply. It is widely used in security systems, automated doors and lighting and energy-saving applications due to its low power consumption and reliable motion detection. It was used to detect the movement around the automated door.

Ultrasonic sensor (HC-SR04):

Fig.4: Ultrasonic sensor.[7]

The HC-SR04 ultrasonic sensor is a distance measurement module that uses ultrasonic waves to determine the distance of an object. It works by emitting an ultrasonic pulse (40 kHz) through its trigger pin and measuring the time taken for the echo to return after bouncing off an object. The sensor operates within a range of 2 cm to 400 cm with an accuracy of ±3 mm. It requires a 5V power supply and communicates using trigger and echo pins, making it easy to interface with microcontrollers like Arduino and Raspberry Pi. This was used to measure any obstacle within the door's range and send feedback

V. Hardware Model

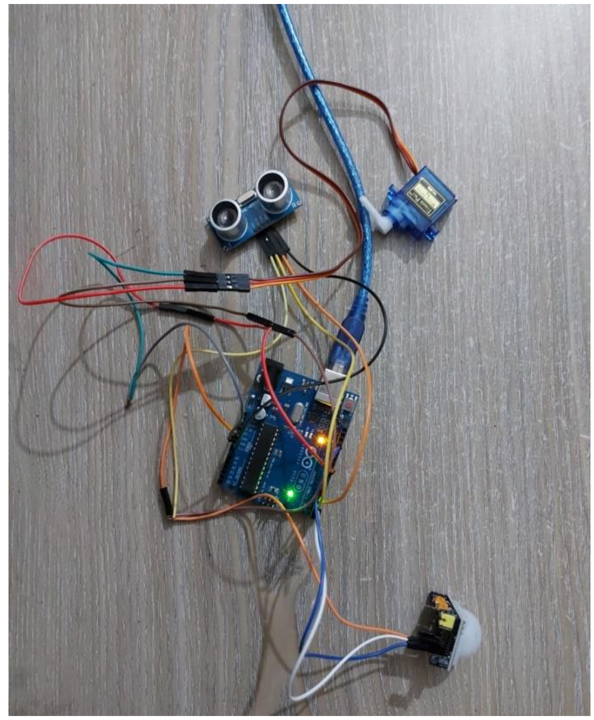

Fig.5- hardware implementation of the system

The shown hardware at fig.5 works as automated system where components work together to detect motion, measure object distance, and control door movement. Here the servo motor indicates the door opening and closing while the sensors detect the motoion and obstacle to make a decision on the servo motor moving and indicating door opening.

VI. Result Analysis

The automated door system was successfully implemented using an Arduino Uno, a PIR sensor (HC-SR501) for motion detection, and an ultrasonic sensor (HC-SR04) for obstacle detection. The system was tested under different conditions to evaluate its performance in terms of responsiveness, accuracy, and reliability.

A. Motion Detection Performance

The PIR sensor effectively detected human movement within its range, ensuring timely activation of the door-opening mechanism. However, due to the sensor's sensitivity, occasional false triggers were observed, especially in environments with rapid temperature fluctuations or infrared interference.

B. Obstacle Detection Accuracy:

The ultrasonic sensor provided precise distance measurements, allowing the system to prevent door activation when an obstacle was detected within 20 cm. Tests showed that the sensor performed reliably in detecting solid objects, but accuracy slightly varied with reflective and irregularly shaped obstacles.

C. Servo Motor Control:

The 180-degree servo motor responded efficiently to the control signals from the Arduino, smoothly opening the door upon valid motion detection and obstacle clearance. However, slight delays were observed due to sensor response time and processing overhead.

D. System Efficiency and Reliability

The system minimized manual effort while ensuring safety through motion-based automation. It functioned reliably under standard conditions, though external factors such as sensor noise and environmental changes slightly affected performance. Future improvements could include:

- Filtering sensor noise using software-based averaging techniques.

- Using a more robust display for real-time feedback.

- Implementing adjustable sensitivity for better customization.

Overall, the system successfully enhanced convenience and security, demonstrating the effectiveness of integrating motion and distance sensing for automated door control.

VII. Limitations

Due to hardware constraints, an OLED display could not be used, requiring real-time system monitoring via the Serial Monitor. This method effectively displayed sensor readings and system status, but a dedicated display would improve user experience and standalone functionality. Moreover, the sensors not necessary work all the time.

References

[1] C. Lasya, N. M. Sai, N. G. Reddy, N. M. Reddy, L. S. and S. S., "Hardware Integration of Sensors for Automating Smart Homes," 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2022, pp. 202-206, doi: 10.1109/ICACRS55517.2022.10029290.

[2] M. Elimarteena and V. S. Nagaraju, "IOT Based Home Security System with PIR and Ultrasonic Detection Using Arduino," 2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2021, pp. 497-502, doi: 10.1109/ICECA52323.2021.9675979.

[3] H. Trivedi, S. Parmar and S. Gajjar, "Smart Door System to Prevent COVID-19 Transmission," 2023 International Conference on Inventive Computation Technologies (ICICT), Lalitpur, Nepal, 2023, pp. 976-981, doi: 10.1109/ICICT57646.2023.10134100.

[4] F. Alsuhaym, T. Al-Hadhrami, F. Saeed and K. Awuson-David, "Toward Home Automation: An IoT Based Home Automation System Control and Security," 2021 International Congress of Advanced Technology and Engineering (ICOTEN), Taiz, Yemen, 2021, pp. 1-11, doi: 10.1109/ICOTEN52080.2021.9493464.

[5] S. Sahu, R. Singh, P. Arya and R. Nirala, "Smart Home Automation Lighting System and Smart Door Lock using Internet of Things," 2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2022, pp. 1320-1325, doi: 10.1109/ICAC3N56670.2022.10074243.

[6] https://dhakarobotics.com/product/3301-hc-sr501-pir-sensor-motion-sensor/

[7] https://store.roboticsbd.com/sensors/22-ultrasonic-sonar-sensor-hc-sr04-robotics-bangladesh.html

Appendix

System code:
Appendix:

```
#include <Servo.h>

// Pin definitions
const int pirPin = 2;     // PIR sensor pin (outside)
const int trigPin = 3;    // Ultrasonic sensor Trig pin (inside)
const int echoPin = 4;    // Ultrasonic sensor Echo pin (inside)
const int servoPin = 9;   // Servo motor control pin

// Threshold distance (in cm)
const int distanceThreshold = 20;

// Servo object
Servo myServo;

void setup() {
  pinMode(pirPin, INPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);

  myServo.attach(servoPin);
  myServo.write(0); // Keep the door closed initially

  Serial.begin(9600); // Start serial communication
  Serial.println("System Initialized");
}

void loop() {
  // Read PIR sensor (outside motion detection)
  bool motionDetected = digitalRead(pirPin) == HIGH;

  // Read distance from ultrasonic sensor (inside obstacle
detection)
  long distance = measureDistance();

  // Print sensor readings to Serial Monitor with a slower rate
  Serial.print("Motion: ");
  Serial.print(motionDetected ? "Detected" : "Not Detected");
  Serial.print(" | Distance: ");
```

```
>

  Serial.print(distance);
  Serial.println(" cm");

  // Act based on sensor data
  if (motionDetected) {
    if (distance > distanceThreshold) {
      Serial.println("Motion detected, and no obstacle inside.
Opening door...");
      myServo.write(90); // Open the door
      delay(5000);       // Keep the door open for 5 seconds
      myServo.write(0);  // Close the door
      Serial.println("Door closed.");
    } else {
      Serial.println("Motion detected, but obstacle inside. Door
can't be opened.");
    }
  } else if (distance <= distanceThreshold) {
    Serial.println("Obstacle detected inside, but no motion
outside. Door remains closed.");
  }

  delay(1000); // Slower delay for better understanding of
serial monitoring
}

long measureDistance() {
  // Send a 10us pulse to the Trig pin
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Read the Echo pin and calculate distance
  long duration = pulseIn(echoPin, HIGH);
  long distance = duration * 0.034 / 2; // Convert to cm
  return distance;
}
```