Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

A Project On

"PIR Sensor Based Closed Loop System"

[Code No: COEG 304]

Submitted by :

Abiral Adhikari (027923-20)

Samir Wagle (027980-20)

Regal Adhikari (028041-20)

Siris Mainali(026177-19)

Submitted to :

Mr. Ayush Bista

Department of Computer Science and Engineering

Chapter 1 Introduction

A closed-loop system is a control system where the output of the system is fed back to the input, allowing for self-regulation and continuous adjustment. This feedback loop helps maintain the desired output and makes the system more responsive to changes or disturbances.

The main objective of this project is to achieve a closed loop system where we can control the running of the motor and lighting of the led. For this, we used Arduino Uno with PIR sensor to control this system. PIR sensor or Passive Infra-Red Sensor is used to monitor motion detection continuously. When motion is detected, the motor and LED are turned ON for some time whereas there is no motion detected, the motor and led are turned OFF. Since the system is turned ON and OFF automatically using the PIR sensor reading, a closed loop system is formed, which is the main objective for this project.

Chapter 2 Components Needed

- Arduino Uno
- LED
- Passive Infra Red(PIR) Sensor
- DC motor (5V)
- Jumper Wire

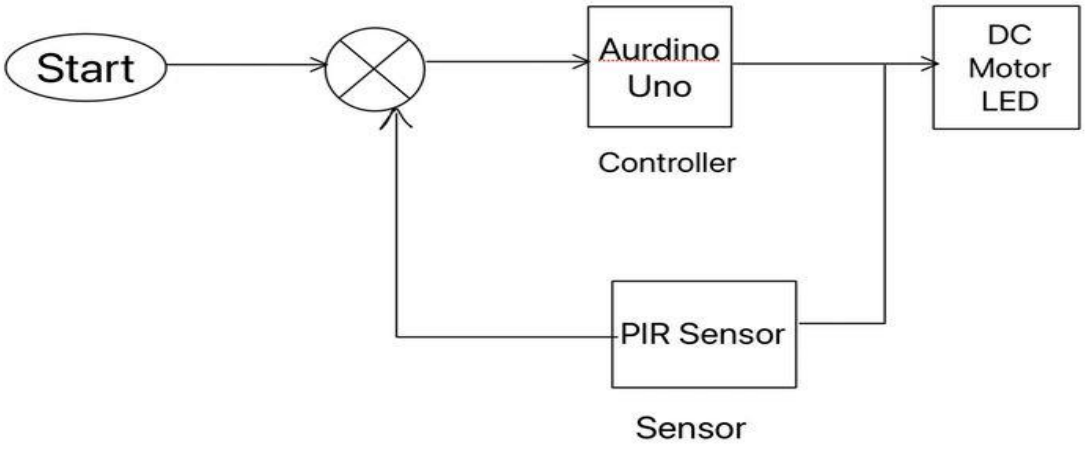
Chapter 3 Procedure and Methods

First, we connect the Arduino Uno to the computer to upload the Arduino Code and supply power. We connect the supply pin of PIR sensor to 5V pin and GND pin to GND of Arduino. The Output pin of the PIR sensor isconnected to Digital Pin 2 of Arduino , from where we get the sensor input. We connect the supply of the DC motor to Digital Pin 4 and GND to the Digital Pin 7. We control the rotation of DC motor by sending HIGH or LOW signal through Pin 4 while Pin 7 is always LOW.Similarly we connect the supply of the LED to Digital Pin 9 and GND to Digital Pin 10 and control the state of LED by sending either HIGH or LOW through Pin 9, while Pin 10 is always set to LOW.

Chapter 4 Outcome

As the components of the PIR sensor were put together, we were able to achieve our expected outcome. Initially, when the PIR sensor read the surrounding terrain, no motion was detected. When it got crossed by human or anything emitting heat came near the PIR sensor's range, the LED lit up and motor rotated for 5 seconds. Same was observed when the object left the range of the sensor. Again, when the sensor took the reading afer 5 second delay and no motion was detected by the PIR sensor this time, the LED and the motor remained off , waiting to be turned on again when motion was detected. The system ran on the loop as long as the power supply was provided to the system.

Chapter 5 Closed Loop System Diagram



Chapter 6 Video Link and Hardware Photo

The video link of our project demonstratino is in following link:
https://drive.google.com/file/d/1Qn5ApQYNlXg6oabJqNIOc5Rdq8HwQqEU/view?usp=sharing

The hardware snap of our project is provided below :

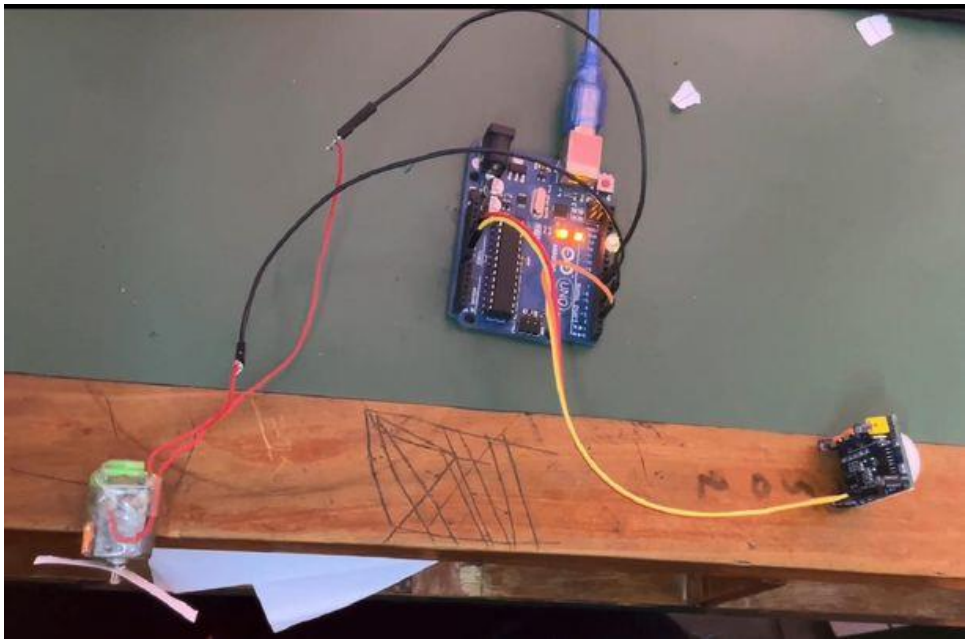


Fig: Hardware View with Labeling

Chapter 7 Circuit Diagram

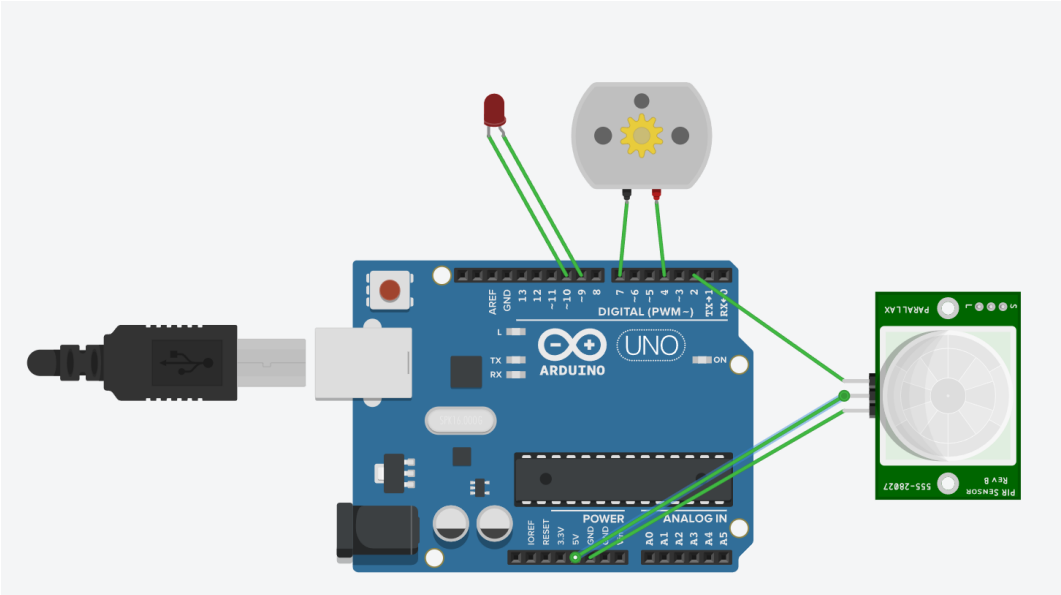


Fig: Circuit Diagram

Chapter 8 Code

```
 // Define the PIR sensor pin
const int pirPin = 2;  // Connected to digital pin 2
const int motor1 = 4;
const int motor2 = 7;
const int led1 = 9;
const int led2 = 10;
int flag = 0;

void setup() {
  // Initialize serial communication at 9600 baud rate
  Serial.begin(9600);

  // Set the PIR pin as input
  pinMode(pirPin, INPUT);
  pinMode(motor1, OUTPUT);
  pinMode(motor2, OUTPUT);
  pinMode(led1, OUTPUT);s
  pinMode(led2, OUTPUT);

  // Wait for the PIR sensor to calibrate (usually 30 seconds)
  Serial.println("PIR Sensor Initializing...");
  delay(30000);  // Wait for 30 seconds
  Serial.println("PIR Sensor Ready!");
}

void loop() {
  // Read the PIR sensor value
  int pirValue = digitalRead(pirPin);
  Serial.println(pirValue);
  if(pirValue == 1){
    Serial.println("Motion detected");
  }else{
    Serial.println("No Motion detected");
  }
  if(pirValue == 1 && flag == 0){

      digitalWrite(led1, HIGH);
      digitalWrite(led2, LOW);
      digitalWrite(motor1, HIGH);
      digitalWrite(motor2, LOW);
      flag = 1;
      delay(5000);
```

```
  }

  if(pirValue != 1 && flag ==1){

      digitalWrite(led1, LOW);
      digitalWrite(led2, LOW);
      digitalWrite(motor1, LOW);
      digitalWrite(motor2, LOW);
      flag = 0;
  }


}
```

Chapter 9 Conclusion

In this way, a closed loop control system containing LED Control by providing a feedback of the motion sensor(PIR sensor) was constructed.

With the design and the implementation, we came to understand practically that the closed loop systems measure the surrounding terrain so that the on-off state of LED is controlled according to the readings of the PIR sensor.