Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

A Mini-Project Report
on
"Auto Curtains"

[COEG 304]

(For partial fulfillment of 3rd Year/1st Semester in Computer Engineering)

Submitted by:

Sudip Bhattarai (11)
Jenisha Khulal Magar (28)
Prabesh Timilsina (58)

Submitted to:

Aayush Bista

Department of Electrical and Electronics Engineering

Submission Date:
2024/01/04

Table of Content

1. **Introduction……………………………………………………………..** 1
2. **Components required………………………………………………..** 1
3. **Design and Implementation…………………………………………** 4
4. **Images………….…………………………………………………….** **7**

Reference……………………………………………………………... 8

1. Introduction

A closed-loop system or feedback system is a control loop that incorporates feedback, in contrast to an open-loop controller or non-feedback controller. A closed-loop controller uses feedback to control the states or outputs of a dynamical system.

Auto Curtains is a straightforward project in which curtains automatically open or close depending on the time of day. In this project, we measure the external light intensity through the window, and based on that measurement, we either open the curtains if it's bright or close them if it's dark. To achieve this, we utilize an LDR and an ESP32. The proper implementation of this project can contribute to the creation of smart curtains, as such solutions have become increasingly integral to our daily routines.

2. Components required

2.1 L298N Motor Driver

The L298N Motor Driver Module serves as a robust driver for high-power DC and Stepper Motors. It comprises the L298 motor driver IC and a 78M05 5V regulator. With the capability to manage either 4 DC motors or 2 DC motors with both directional and speed control, this module offers versatile motor control.

The ENA and ENB pins serve as speed control inputs for Motor A and Motor B, respectively. On the other hand, the IN1 & IN2 and IN3 & IN4 pins function as direction control inputs for Motor A and Motor B, respectively.

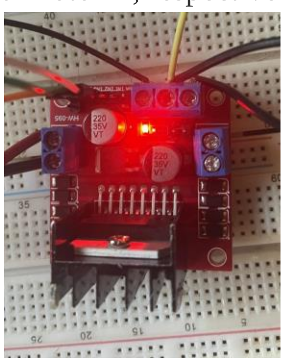

Figure 2.1.1 The L298N Motor Driver

2.2 12V DC-motor 56:1

This DC motor features a full metal gearbox and a 6mm D-shaped output shaft, providing durability and versatility. The motor is designed to operate within a voltage range of 3 to 12V, and its polarity can be easily reversed to change the direction of rotation. The current of 45mA at 12V under no load conditions. It has a gear ratio of 56:1.

Figure 2.2.1 12V DC motor

2.3 Light Dependent Resistor (LDR)

The Light Dependent Resistor (LDR) operates based on the principle of photoconductivity, reflecting its name. This unique resistor undergoes a change in resistance corresponding to the intensity of light. Specifically, its resistance diminishes as the ambient light intensity increases. This property makes the LDR a valuable component in applications where variations in light conditions need to be translated into electrical signals or control mechanisms, allowing for adaptive responses based on changes in ambient light levels.

Figure 2.3.1 LDR

2.4 ESP32

The ESP32 is a single-chip solution encompassing both Wi-Fi and Bluetooth capabilities, operating at a frequency of 2.4 GHz. Engineered with TSMC's advanced 40 nm low-power technology, this chip is optimized for superior power efficiency and RF performance. It demonstrates resilience, adaptability, and reliability across a broad spectrum of applications and power scenarios.

Key Features:

Wi-Fi

- Supports 802.11b/g/n standards
- 2.4 GHz operation for 802.11n, providing speeds up to 150 Mbps
- WMM (Wi-Fi Multimedia) support for improved wireless multimedia applications.

Bluetooth

- Complies with Bluetooth v4.2 BR/EDR (Basic Rate/Enhanced Data Rate) and Bluetooth LE (Low Energy) specifications.
- Offers class-1, class-2, and class-3 transmitter capabilities without the need for an external power amplifier.
- Enhanced Power Control for optimized energy management.

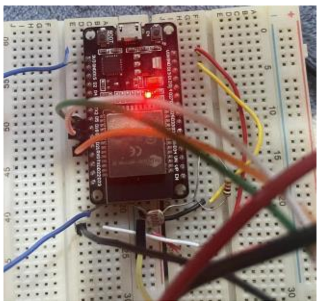

Figure 2.4.1 ESP32

3. Design and Implementation

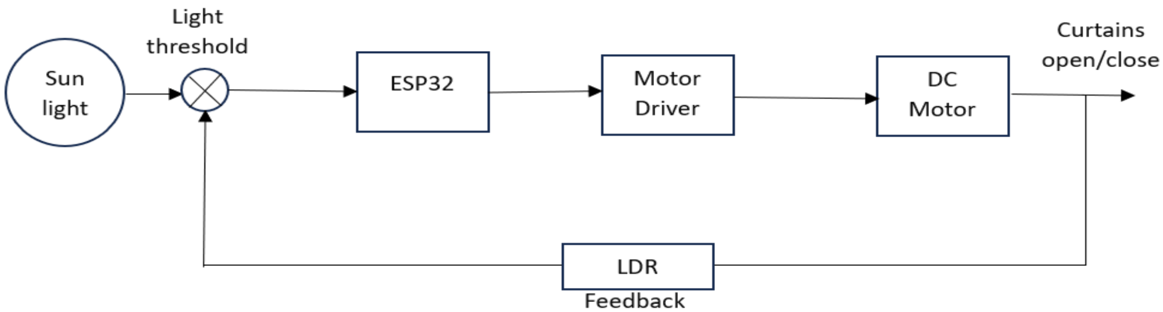

Figure 3.1 Block diagram of our closed-loop system

3.1 Procedure

1. First of all, we checked the functionality of the motor driver by connecting it to the motor and manually supplying the power.
2. For driving the motor, an L298N motor driver was used.
3. ESP32 was connected to the Breadboard.
4. For sensing the light, a light-dependent resistor (LDR) was used. The threshold for the LDR was kept at 50 ohm.
5. The input from LDR was connected to ESP32.
6. The ESP was programmed to rotate the motor clockwise and anticlockwise based on the input from the LDR.
7. For power supply, 12V DC was used was used. The supply was initially given to the motor driver which in turn supplied ESP32 and LDR.
8. Finally, to demonstrate the working of the entire setup, a curtain was drawn on and off with the help of the motor.

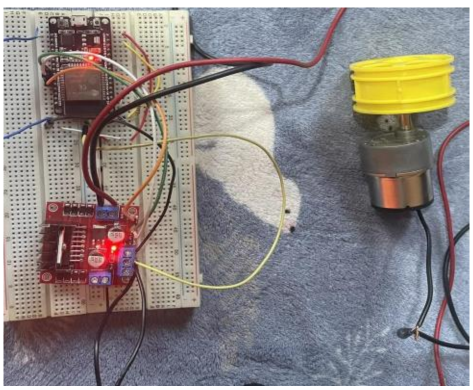

Figure 3.2 Entire setup

3.3 Code implementation

```
#define ldrPin 34
#define motor1 18
#define motor2 19
#define motorPWM 21
const int curtainThreshold = 50;
const int pwm = 255;
```

These lines create constant values for the pin the Light Dependent Resistor (LDR) sensor `ldrPin`, and the two pins for controlling the curtain motor `motor1` and `motor2`, and the Pulse Width Modulation for regulating the motor speed `motorPWM`. There are two constants as well `curtainThreshold`, for the threshold level for the LDR sensor, and `pwm`, the value for motor speed.

```
void setup() {
  Serial.begin(9600);
  pinMode(ldrPin, OUTPUT);
  pinMode(motor1, OUTPUT);
  pinMode(motor2, OUTPUT);
  pinMode(motorPWM, OUTPUT);
}
bool curtainFlag = false;
```

This function is executed once in the start, it initializes serial communication, sets the LDR pin as input and set motor control pins as output.
The constant curtainFlag saves the state of flag(curtains).

```
bool curtainFlag = false;


void loop() {
  int ldr = analogRead(ldrPin);
  stop();
  Serial.println(ldr);
  if(curtainFlag && ldr<curtainThreshold){
    Serial.println("Flag1");
    forward();
    curtainFlag = false;
    delay(10000);
    return;
  }
  if(!curtainFlag && ldr>curtainThreshold){
    Serial.println("Flag2");

    backward();
    curtainFlag = true;
    delay(10000);
    return;
  }
}
```

This is the heart of the code here the analog value from the LDR sensor is read and stored in ‘ldr’. The stop() function stops the motor. Then we print the LDR value in the monitor. Now we check if the curtainFlag is open and ldr values is less than curtainthreshold, if true we print a message “Flag1” and turn the motor forward opening the curtains for 10 seconds. If the curtainFlag is false and LDR value is higher than threshold, then we print “Flag2” the motor turns backward opening the curtains.

```
void forward(){
  digitalWrite(motor1, HIGH);
  digitalWrite(motor2,LOW);
  analogWrite(motorPWM,pwm);
}

void  backward(){
  digitalWrite(motor1, LOW);
  digitalWrite(motor2,HIGH);
  analogWrite(motorPWM,pwm);
}

void stop(){
  digitalWrite(motor1, LOW);
  digitalWrite(motor2,LOW);
  analogWrite(motorPWM,0);
}
```

This snippet basically shows the forward, backward, and stop function, which controls the motors and motorPWM pins. forward() turns the motor forward, backward() turns the motor backward and stop() stops the motor.

4 Images

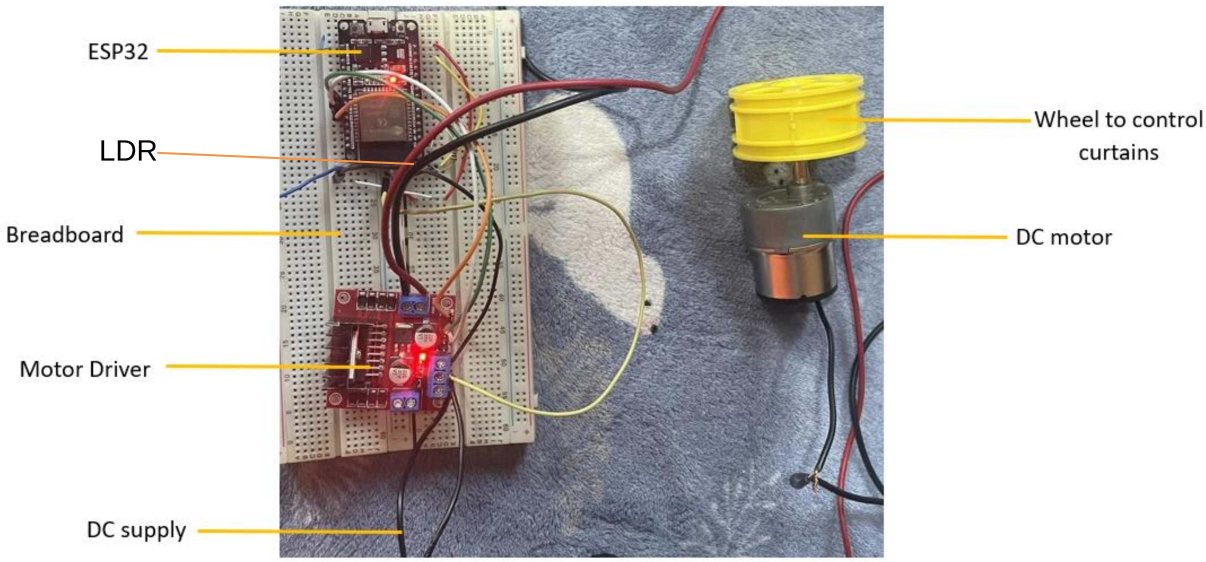

Figure 4.1 Labeled Figure of Auto Curtains

Video Link

https://drive.google.com/file/d/124dfAfofHHu8AFYMHHn2UUfzl7xXmth-/view

Reference

L298N Motor Driver Module. (n.d.). Components101. https://components101.com/modules/l293n-motor-driver-module

Buy DC-motor with gearbox 3-12VDC 56:1 51rpm at the right price @ Electrokit. (n.d.). Buy DC-motor With Gearbox 3-12VDC 56:1 51rpm at the Right Price @ Electrokit. https://www.electrokit.com/en/product/dc-motor-with-gearbox-3-12vdc-561-51rpm/

LDR - Light Dependent Resistor | Working, Types, Application. (2023, August 15). Electronics for You. https://www.electronicsforu.com/technology-trends/learn-electronics/ldr-light-dependent-resistors-basics

Esp32_datasheet_en.pdf. https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf