

UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E  
Ingenierías

## **Proyecto final**

Nombres:

Kevin eduardo lopez martinez  
Zahara Nathalia Ibarra Casillas  
Brian Michel Rubio Martínez

Carrera:

Ingeniería Fotónica

Actividad:

Proyecto final

20 de Mayo de 2025

# Índice

<b>1. Objetivo General</b>	<b>2</b>
<b>2. Descripción General</b>	<b>2</b>
<b>3. Requerimientos del Sistema</b>	<b>2</b>
<b>4. Funcionamiento</b>	<b>2</b>
<b>5. Diagrama Esquemático</b>	<b>3</b>
<b>6. Descripción de Puertos</b>	<b>3</b>
<b>7. Diagrama de Flujo</b>	<b>4</b>
<b>8. Conclusiones</b>	<b>7</b>

# 1. Objetivo General

### Objetivo General

Desarrollar un sistema que pueda localizar y seguir una luz de forma precisa dependiendo de un chasis de auto para Arduino.

# 2. Descripción General

### Descripción General

Este sistema tiene que localizar y seguir de forma autónoma luz de un foco que se localizará en el suelo.

# 3. Requerimientos del Sistema

### Requerimientos del Sistema

1. El sistema debe de contar con 4 motore de salida los cuales el movimiento dependerá de dos fotorresistencias.
2. Dos de los motores se moverán de forma dependiente de los otros motores.
3. Una vez que las fotorresistencias encuentren una luz tienen que guiar a los motores hasta la luz(objetivo) una vez que llegue a la luz hay que reventar un globo.

# 4. Funcionamiento

La conexión del circuito consta de cuatro motores (TT Motor) los cuales van conectados a un L298NMotorDriverBoard la cual se encarga de distribuir energía a los motores al igual que recibir la señal de activación la cual es controlada por una tarjeta de desarrollo ESP32 la cual nos funciona para poder administrar los datos que reciben las dos fotorresistencias y así poder dirigir los motores hacia una fuente de luz estable.

## 5. Diagrama Esquemático

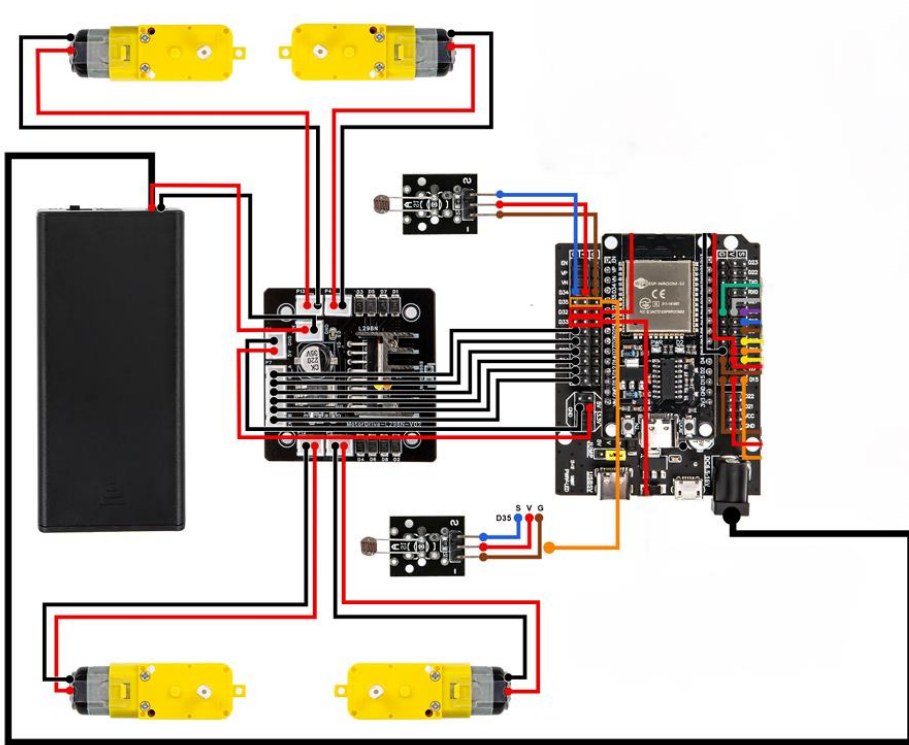


Figura 1: Diagrama Esquemático

## 6. Descripción de Puertos

El sistema utiliza los siguientes pines para la conexión:

- Fotorresistencia L: Pin 35
- Fotorresistencia R: Pin 34
- Control dirección motor A: PIN 27
- Control dirección motor A: PIN 26
- Control dirección motor B: PIN 12
- Control dirección motor B: PIN 14
- Físico para ENA: PIN 25
- Físico para ENB: PIN 13
- Canal PWM para motor B: PIN Digital 4
- Canal PWM para motor A: PIN Digital 3

## 7. Diagrama de Flujo

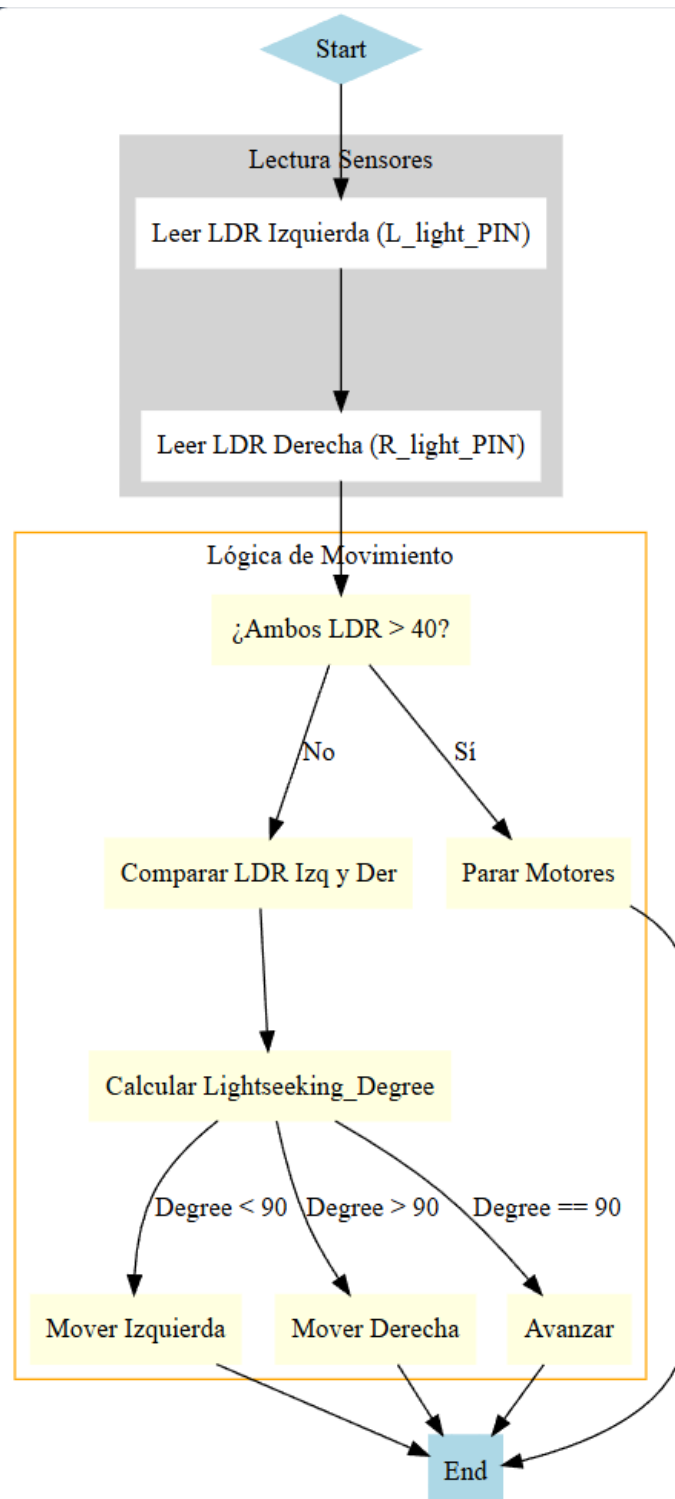


Figura 2: Diagrama de flujo del sistema.

## 8 CONCLUSIONES

---

```
1  volatile float Left_photosensitive;
2  volatile float Right_photosensitive;
3  volatile int Lightseeking_Degree;
4  volatile float f;
5  volatile int speed_value;
6  #define L_light_PIN 35
7  #define R_light_PIN 34
8  #define ENA 3
9  #define ENA_PIN 25
10 #define IN1 27
11 #define IN2 26
12 #define IN3 12
13 #define IN4 14
14 #define ENB 4
15 #define ENB_PIN 13
16 #define carSpeed 250
17
18
19 void Forward() {
20     ledcWrite(ENA, carSpeed); //enable L298n A channel
21     ledcWrite(ENB, carSpeed); //enable L298n B channel
22     digitalWrite(IN1, LOW);
23     digitalWrite(IN2, HIGH);
24     digitalWrite(IN3, LOW);
25     digitalWrite(IN4, HIGH);
26     //Serial.println("Forward");
27 }
28
29 void Back() {
30     ledcWrite(ENA, carSpeed); //enable L298n A channel
31     ledcWrite(ENB, carSpeed); //enable L298n B channel
32     digitalWrite(IN1, HIGH);
33     digitalWrite(IN2, LOW);
34     digitalWrite(IN3, HIGH);
35     digitalWrite(IN4, LOW);
36     //Serial.println("Back");
37 }
38
39 void Left() {
40     ledcWrite(ENA, carSpeed); //enable L298n A channel
41     ledcWrite(ENB, carSpeed); //enable L298n B channel
42     digitalWrite(IN1, LOW);
43     digitalWrite(IN2, HIGH);
44     digitalWrite(IN3, HIGH);
45     digitalWrite(IN4, LOW);
46     //Serial.println("Left");
47 }
48
49 void Right() {
50     ledcWrite(ENA, carSpeed); //enable L298n A channel
51     ledcWrite(ENB, carSpeed); //enable L298n B channel
52     digitalWrite(IN1, HIGH);
53     digitalWrite(IN2, LOW);
54     digitalWrite(IN3, LOW);
55     digitalWrite(IN4, HIGH);
56     //Serial.println("Right");
57 }
58
59 void Stop() {
60     ledcWrite(ENA, 0); //enable L298n A channel
```

## 8 CONCLUSIONES

---

```
61   ledcWrite(ENB, 0); //enable L298n B channel
62   //Serial.println("Stop!");
63 }
64
65 void Light_Seeking() {
66   Left_photosensitive = analogRead(L_light_PIN) / 35;
67   Right_photosensitive = analogRead(R_light_PIN) / 35;
68   Serial.print("Left_photosensitive:");
69   Serial.println(Left_photosensitive);
70   Serial.print("Right_photosensitive:");
71   Serial.println(Right_photosensitive);
72   Serial.println("");
73   if (Left_photosensitive > 40 && Right_photosensitive > 40) {
74     Stop();
75   } else {
76     if (Left_photosensitive > Right_photosensitive) {
77       Lightseeking_Degree = ((float)(Right_photosensitive /
Left_photosensitive)) * 90;
78     } else if (Left_photosensitive <= Right_photosensitive) {
79       Lightseeking_Degree = 180 - ((float)(Left_photosensitive /
Right_photosensitive)) * 90;
80     }
81     if (Lightseeking_Degree < 90) {
82       f = ((float)(Lightseeking_Degree)) / 90;
83       ledcWrite(ENA, carSpeed* 0.5*f); //enable L298n A channel
84       ledcWrite(ENB, carSpeed); //enable L298n B channel
85       digitalWrite(IN1, LOW);
86       digitalWrite(IN2, HIGH);
87       digitalWrite(IN3, LOW);
88       digitalWrite(IN4, HIGH);
89     }
90     if (Lightseeking_Degree > 90) {
91       f = ((float)(180 - Lightseeking_Degree)) / 90;
92       ledcWrite(ENA, carSpeed); //enable L298n A channel
93       ledcWrite(ENB, carSpeed* 0.5*f); //enable L298n B channel
94       digitalWrite(IN1, LOW);
95       digitalWrite(IN2, HIGH);
96       digitalWrite(IN3, LOW);
97       digitalWrite(IN4, HIGH);
98     }
99     if (Lightseeking_Degree == 90) {
100       f = ((float)(Lightseeking_Degree)) / 90;
101       Forward();
102     }
103   }
104 }
105
106 void setup() {
107   Left_photosensitive = 0;
108   Right_photosensitive = 0;
109   Lightseeking_Degree = 0;
110   f = 0;
111   Serial.begin(9600);
112   pinMode(L_light_PIN, INPUT);
113   pinMode(R_light_PIN, INPUT);
114   pinMode(IN1, OUTPUT);
115   pinMode(IN2, OUTPUT);
116   pinMode(IN3, OUTPUT);
117   pinMode(IN4, OUTPUT);
118   pinMode(ENA_PIN, OUTPUT);
119   pinMode(ENB_PIN, OUTPUT);
```

## 8 CONCLUSIONES

---

```
120    ledcSetup(ENA,5000,8);
121    ledcSetup(ENB,5000,8);
122    ledcAttachPin(ENA_PIN,ENA);
123    ledcAttachPin(ENB_PIN,ENB);
124 }
125
126 void loop() {
127     Light_Seeking();
128
129 }
```

Código: seguidor de luz

## 8. Conclusiones

Se cumplieron todos los requerimientos establecidos para el sistema: Los motores fueron controlados de manera eficiente al momento de detectar un nivel de luz adecuado gracias a las fotorresistencias guiando al carro de manera autónoma.