



AEC6: ARDUINO

PROYECTOS 1

ISMAEL HERNÁNDEZ CLEMENTE

Control de Motor DC y Tira de LEDs con Arduino

Introducción

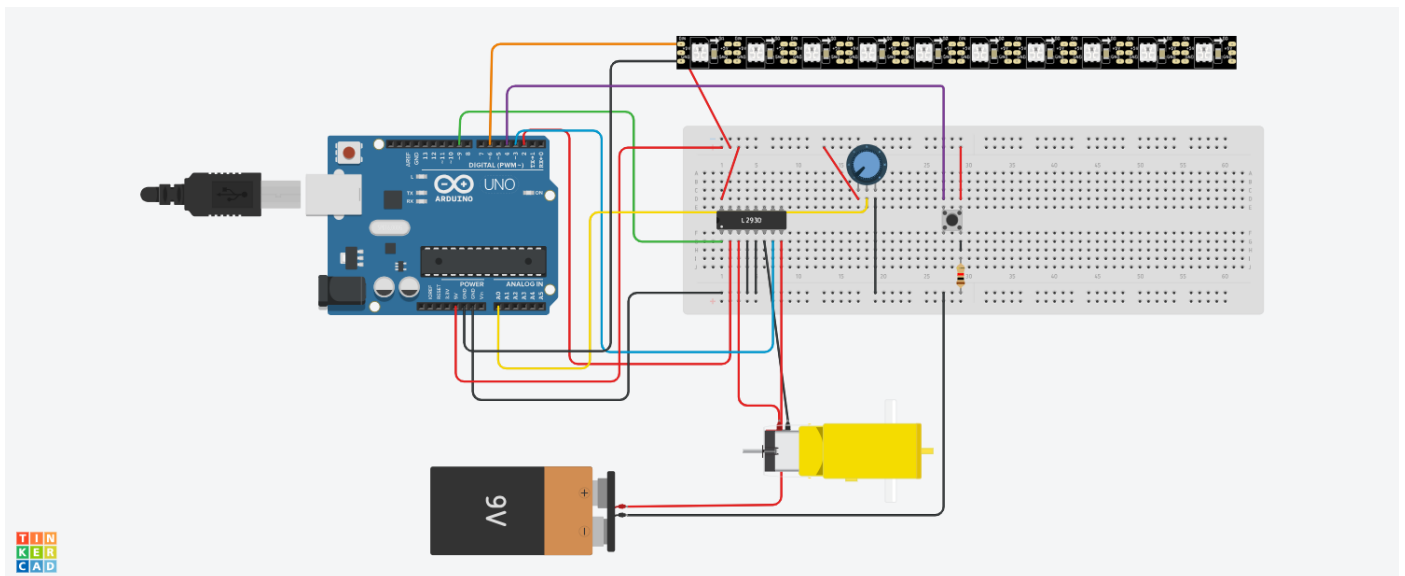
En este proyecto, se busca controlar un motor de corriente continua (DC) utilizando un puente H L293D, un potenciómetro y un pulsador, además de visualizar la velocidad y el sentido de giro del motor mediante una tira de LEDs WS2812S. El objetivo es crear un sistema interactivo en el que el usuario pueda ajustar la velocidad y el sentido de giro del motor con un potenciómetro y encender o apagar el motor con un pulsador. La tira de LEDs proporcionará una indicación visual de la velocidad del motor, variando el número de LEDs encendidos proporcionalmente a la velocidad del motor y mostrando colores diferentes según el sentido de giro.

Objetivos

- Controlar la velocidad y el sentido de giro de un motor DC utilizando un potenciómetro.
- Encender y apagar el motor mediante un pulsador.
- Utilizar una tira de LEDs WS2812S para indicar visualmente la velocidad del motor.
- Cambiar el color de los LEDs según el sentido de giro del motor.
- Documentar todo el proceso y proporcionar un diagrama del circuito, la lista de componentes, imágenes del montaje y el código fuente.

Diagrama del Circuito

Para el diseño del circuito se utilizará Tinkercad, una plataforma en línea que permite crear y simular circuitos electrónicos. El diagrama del circuito se exportará desde Tinkercad y se incluirá en esta sección.



Lista de Componentes

A continuación, se presenta la lista de componentes necesarios para el montaje del circuito:

- 1 Arduino Uno
- 1 L293D (Puente H)

- 1 motor DC
- 1 potenciómetro
- 1 pulsador
- 1 tira de 10 LEDs WS2812S
- 1 pila de 9V
- Resistencias de 10kΩ
- Cables de conexión
- Protoboard

Código de Programación (.ino)

El siguiente código en lenguaje Arduino controla el motor DC y la tira de LEDs según lo descrito en los objetivos del proyecto.

```
#include <Adafruit_NeoPixel.h>

// Definiciones de pines
const int motorEnablePin = 9;
const int motorInputPin1 = 2;
const int motorInputPin2 = 3;
const int potPin = A0;
const int buttonPin = 4;
const int LED_PIN = 6;
const int LED_COUNT = 10;

bool motorState = false;
int buttonState;
int lastButtonState = LOW;
unsigned long lastDebounceTime = 0;
unsigned long debounceDelay = 50;

// Inicialización de la tira de LEDs
Adafruit_NeoPixel strip(LED_COUNT, LED_PIN, NEO_GRB + NEO_KHZ800);

void setup() {
  pinMode(motorEnablePin, OUTPUT);
  pinMode(motorInputPin1, OUTPUT);
  pinMode(motorInputPin2, OUTPUT);
  pinMode(buttonPin, INPUT);

  digitalWrite(motorEnablePin, LOW);
  Serial.begin(9600);

  strip.begin();
  strip.show(); // Inicializa todos los LEDs apagados
}
```

```

void loop() {
    int reading = digitalRead(buttonPin);

    // Manejo del debounce del botón
    if (reading != lastButtonState) {
        lastDebounceTime = millis();
    }

    if ((millis() - lastDebounceTime) > debounceDelay) {
        if (reading != buttonState) {
            buttonState = reading;
            if (buttonState == HIGH) {
                motorState = !motorState; // Alterna el estado del motor
            }
        }
    }

    if (motorState) {
        int potValue = analogRead(potPin);
        int speed = map(potValue, 0, 1023, -255, 255); // Mapea el valor del potenciómetro a -255 a 255
        int numLEDs = map(abs(speed), 0, 255, 0, LED_COUNT); // Mapea la velocidad a la cantidad de LEDs

        // Actualiza los LEDs según la velocidad y el sentido de giro
        for (int i = 0; i < LED_COUNT; i++) {
            if (i < numLEDs) {
                if (speed > 0) {
                    strip.setPixelColor(i, strip.Color(0, 255, 0)); // Verde
                } else {
                    strip.setPixelColor(i, strip.Color(255, 0, 0)); // Rojo
                }
            } else {
                strip.setPixelColor(i, strip.Color(0, 0, 0)); // Apagado
            }
        }
        strip.show();

        // Control del motor
        if (speed > 0) {
            digitalWrite(motorInputPin1, HIGH);
            digitalWrite(motorInputPin2, LOW);
            analogWrite(motorEnablePin, speed);
        } else if (speed < 0) {
            digitalWrite(motorInputPin1, LOW);
            digitalWrite(motorInputPin2, HIGH);
            analogWrite(motorEnablePin, -speed);
        } else {
            analogWrite(motorEnablePin, 0);
        }

        for (int i = 0; i < LED_COUNT; i++) {
            strip.setPixelColor(i, strip.Color(0, 0, 0)); // Apaga todos los LEDs
        }
        strip.show();
    }

    lastButtonState = reading;
}

```

