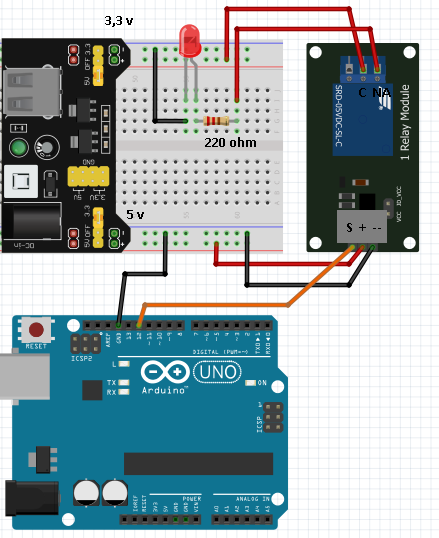
Activar Led con Relé electromecánico

**Circuito de conexión**

****

**Programación en Arduino**

**El led se activa ingresando la tecla A mediante el teclado del PC en el minitor serie de arduino y se desactiva ingresando la tecla D.**

int ledPin = 12; // Pin donde está conectado el LED

char dato; // Variable para recibir datos del puerto serie

void setup() {

pinMode(ledPin, OUTPUT); // Configura el pin 12 como salida

digitalWrite(ledPin, LOW); // Asegura que el LED inicie apagado

Serial.begin(115200); // Inicia la comunicación serie

Serial.println("Presione A para encender el LED, B para apagarlo");

}

void loop() {

// Verifica si hay datos disponibles en el puerto serie

if (Serial.available() > 0) {

dato = Serial.read(); // Lee el carácter enviado desde la PC

if (dato == 'A' || dato == 'a') { // Si presionan A o a

digitalWrite(ledPin, HIGH);

Serial.println("LED ENCENDIDO");

}

else if (dato == 'D' || dato == 'd') { // Si presionan B o b

digitalWrite(ledPin, LOW);

Serial.println("LED APAGADO");

}

}

}

Área ejercicio

Hacer un programa en arduino que active y desactive automáticamente un led sin usar la instrucción delay. Usar el terminal 12.

**Resolución**

**Programa en Arduino**

int ledPin = 12; // Pin del LED

unsigned long intervalo = 1000; // Intervalo de tiempo en milisegundos

unsigned long tiempoPrevio = 0; // Guarda el último tiempo registrado

bool estadoLed = LOW; // Estado actual del LED

void setup() {

pinMode(ledPin, OUTPUT); // Configura el pin como salida

digitalWrite(ledPin, LOW); // Inicia apagado

}

void loop() {

unsigned long tiempoActual = millis(); // Tiempo desde que se inició Arduino

// Verifica si ya pasó el intervalo

if (tiempoActual - tiempoPrevio >= intervalo) {

tiempoPrevio = tiempoActual; // Actualiza el tiempo previo

estadoLed = !estadoLed; // Invierte el estado del LED

digitalWrite(ledPin, estadoLed);

}

}