

Continuando con la modernización de nuestra explotación, se han instalado en una nave industrial diferentes sensores de temperatura, humedad e iluminación para el control de la misma.

Se pide realizar el esquema de conexionado a través de una plaza Arduino de esta instalación, de forma que se active el ventilador de la nave en caso de subida de temperatura; se active un extractor de aire en caso de subida de la humedad; se encienda la iluminación cuando baje la luminosidad del exterior.

Componentes necesarios:

- Microcontrolador:
 - Arduino UNO (o similar).
- Sensores:
 - Temperatura y humedad: sensor **DHT22** o **DHT11**
 - Iluminación: fotorresistencia (LDR)
- Actuadores: módulo de relés de 4 canales (para aislar la corriente de Arduino de la potencia de los ventiladores y luces).
- Alimentación:
 - Fuente de 5 V para Arduino.
 - Línea de potencia para los ventiladores/luces (220 V o 12 V según los equipos que se conecten)

Esquema de conexionado

El diseño se basa en la lectura de señales analógicas/digitales y la activación de relés por nivel bajo o alto.

1. Sensores (entradas)

Sensor	Pin Arduino	Notas
DHT22 (Pin datos)	D2	Requiere una resistencia pull-up de 10kΩ entre V _{cc} y datos
LDR (Luz)	A0	Conectada en divisor de tensión con una resistencia de 10kΩ

2. Actuadores (salidas a relés)

Actuador	Pin Arduino	Acción
Ventilador	D3	Se activa al superar el umbral de temperatura
Extractor	D4	Se activa al superar el umbral de humedad
Iluminación	D5	Se activa cuando el valor de luz en A0 cae

El diseño se basa en leer datos del entorno y enviar señales a los actuadores (ventilador, extractor, luces) a través de relés, ya que estos dispositivos suelen funcionar con voltajes altos (110V/220V) que el Arduino no puede manejar directamente.

Paso a Paso de las Conexiones:1. **Sensor de Temperatura/Humedad (DHT):**

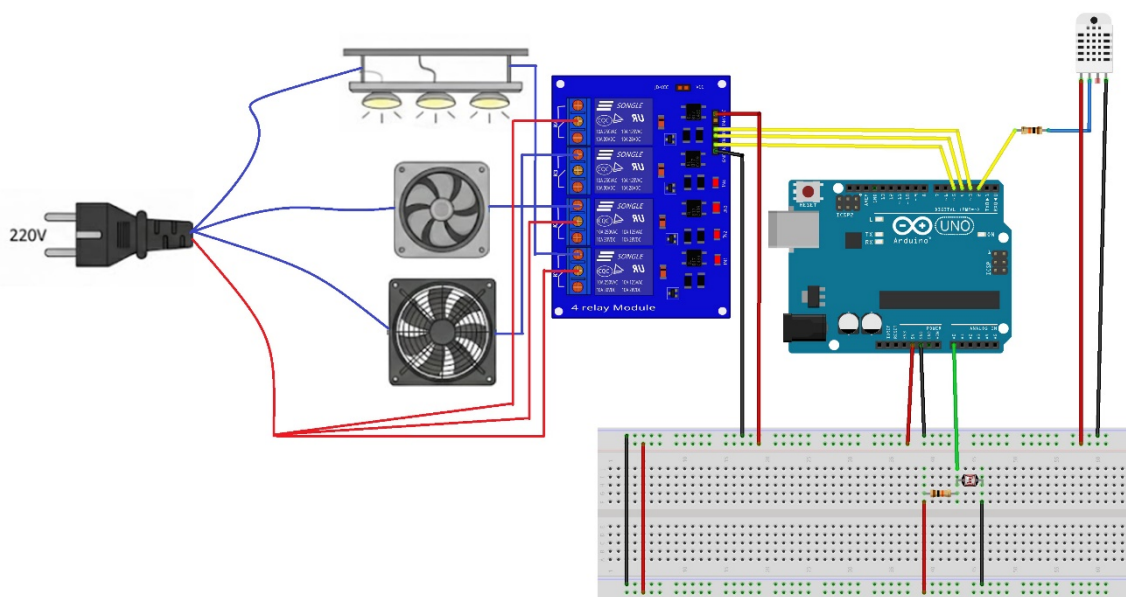
- **VCC** a 5V.
- **GND** a Tierra.
- **DATA** al Pin Digital 2 (con una resistencia de pull-up de 10k Ω si el módulo no la trae).

2. **Sensor de Luz (LDR):**

- Se conecta en serie con una resistencia de 10k Ω para crear un divisor de tensión.
- El punto medio entre la LDR y la resistencia va al **Pin Analógico A0**.

3. Módulo de Relés (Actuadores):

- **Relé 1 (Ventilador):** Entrada de control al Pin 3.
- **Relé 2 (Extractor):** Entrada de control al Pin 4.
- **Relé 3 (Iluminación):** Entrada de control al Pin 5.
- *Nota:* Los dispositivos de potencia se conectan al lado de "Salida" del relé (contacto normalmente abierto).



El código para el control de la nave industrial será:

```
#include "DHT.h"

// --- Configuración de Pines ---
#define DHTPIN 2           // Pin donde conectamos el sensor DHT
#define DHTTYPE DHT22      // Cambiar a DHT11 si usas ese modelo
#define LDRPIN A0          // Pin analógico para la fotoresistencia
#define PIN_VENTILADOR 3   // Relé para el ventilador
#define PIN_EXTRACTOR 4    // Relé para el extractor
#define PIN_LUCES 5        // Relé para la iluminación

// --- Umbrales Personalizables ---
const float UMBRAL_TEMP = 28.0; // Activa ventilador si > 28°C
const float UMBRAL_HUM = 70.0;  // Activa extractor si > 70%
const int UMBRAL_LUZ = 400;     // Activa luces si el valor baja de 400 (0-1023)

DHT dht(DHTPIN, DHTTYPE);
```

```
void setup() {
  Serial.begin(9600);
  dht.begin();

  // Configurar pines de salida
  pinMode(PIN_VENTILADOR, OUTPUT);
  pinMode(PIN_EXTRACTOR, OUTPUT);
  pinMode(PIN_LUCES, OUTPUT);

  // Iniciar todo apagado (Dependiendo del relé, LOW o HIGH puede ser
  apagado)
  digitalWrite(PIN_VENTILADOR, LOW);
  digitalWrite(PIN_EXTRACTOR, LOW);
  digitalWrite(PIN_LUCES, LOW);
}

void loop() {
  // Esperar un par de segundos entre lecturas
  delay(2000);

  // 1. Lectura de Sensores
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  int nivelLuz = analogRead(LDRPIN);

  // Comprobar errores de lectura
  if (isnan(h) || isnan(t)) {
    Serial.println("Error al leer el sensor DHT!");
    return;
  }

  // --- Lógica de Control ---

  // Control Ventilador (Temperatura)
  if (t > UMBRAL_TEMP) {
    digitalWrite(PIN_VENTILADOR, HIGH);
  } else if (t < (UMBRAL_TEMP - 1.0)) { // Margen de seguridad de 1
grado
    digitalWrite(PIN_VENTILADOR, LOW);
  }

  // Control Extractor (Humedad)
  if (h > UMBRAL_HUM) {
    digitalWrite(PIN_EXTRACTOR, HIGH);
  } else if (h < (UMBRAL_HUM - 5.0)) { // Margen de seguridad de 5%
    digitalWrite(PIN_EXTRACTOR, LOW);
  }

  // Control Iluminación (Luz exterior)
  // Nota: A menos luz, el valor de analogRead suele ser menor
  if (nivelLuz < UMBRAL_LUZ) {
    digitalWrite(PIN_LUCES, HIGH);
  } else if (nivelLuz > (UMBRAL_LUZ + 50)) {
    digitalWrite(PIN_LUCES, LOW);
  }

  // Monitoreo por Monitor Serie
  Serial.print("Temp: "); Serial.print(t); Serial.print("°C | ");
  Serial.print("Hum: "); Serial.print(h); Serial.print("% | ");
```



```
Serial.print("Luz: "); Serial.println(nivelLuz);  
}
```
