# SENSORES RELATIVOS AL PROYECTO:

El presente documento presenta algunos de los sensores que se pusieron en discusión para su elección. Se analizaran pros y contra de en sus diferentes categorías para su implementación.

# Sensor de Temperatura y Humedad ambiente

## DHT11/DHT22

### Características:

* + **DHT11**: Rango de humedad del 20% al 80%, precisión ±5% RH.
  + **DHT22**: Rango de humedad del 0% al 100%, precisión ±2% RH.
  + **Rango de temperatura**: DHT11 (-20°C a 60°C), DHT22 (-40°C a 80°C).

### Datos Técnicos:

* + **DHT11**:
    - **Voltaje de operación**: 3.3V a 5V  
      **Consumo de corriente**: 1mA
  + **DHT22**:
    - **Voltaje de operación**: 3.3V a 6V  
      **Consumo de corriente**: 1.5mA

### Implementación:

* + **Conexión**:
    - **VCC** a **3.3V o 5V** del ESP32  
      **GND** a **GND** del ESP32  
      **Data** a **GPIO 4** del ESP32 (usar una resistencia pull-up de 10kΩ entre VCC y Data)

**Programación** (Ejemplo para DHT22):

#include <DHT.h>  
#define DHTPIN 4  
#define DHTTYPE DHT22  
DHT dht(DHTPIN, DHTTYPE);  
void setup() {  
Serial.begin(115200);  
dht.begin();  
}

void loop() {  
float h = dht.readHumidity();  
 float t = dht.readTemperature();  
if (isnan(h) || isnan(t)) {  
 Serial.println("Falla en la lectura del DHT!");  
return; }  
 Serial.print("Humedad: ");  
 Serial.print(h);  
 Serial.print("% Temperatura: ");  
 Serial.print(t);  
 Serial.println("°C");  
 delay(2000);}

**Diagrama de Conexión**:  
ESP32 DHT22  
---------------------  
3.3V --> VCC  
GND --> GND  
GPIO 4 --> Data

## AHT10

### Características:

* + **Rango de temperatura**: -40°C a 85°C, ±0.1.
  + **Rango de humedad**: 0% a 100% RH (±2% RH)

### Datos Técnicos:

* + - **Voltaje de operación**: 2.4V a 3.6V  
      **Consumo de corriente**: 23 µA

### Implementación:

* + **Conexión**:
    - **VCC** a **3.3V** del ESP32 (Nunca a 5V)
    - **GND** a **GND** del ESP32
    - **SDA: Conecta el pin SDA del AHT10 al pin SDA del ESP32 (generalmente GPIO 21 o 22).**
    - **SCL: Conecta el pin SCL del AHT10 al pin SCL del ESP32 (generalmente GPIO 23 o 26).**
    - Ambos SDA como SCL se conectan con resistencias pull-up.

**Programación**

#include <Wire.h>

#include "Adafruit\_AHTX0.h"

Adafruit\_AHTX0 aht;

void setup() {

Serial.begin(9600);

Wire.begin();

if (!aht.begin()) {

Serial.println("Could not find a valid AHTX0 sensor, check wiring!");

while (1);

}

}

void loop() {

sensors\_event\_t humidity, temp;

aht.getEvent(&humidity, &temp);

Serial.print("Humidity: ");

Serial.print(humidity.relative\_humidity);

Serial.println("%");

Serial.print("Temperature: ");

Serial.print(temp.temperature);

Serial.println("°C");

delay(5000);

}

# Sensor de Temperatura, Humedad y Presión Atmosférica

## BME280

* Características:
  + Sensores: Temperatura, humedad, presión atmosférica.  
    Rango de temperatura: -40°C a 85°C.  
    Rango de humedad: 0% a 100%.  
    Precisión: ±1°C, ±3% RH.

### Datos Técnicos:

* + Voltaje de operación: 3.3V.  
    Interfaz: I2C o SPI.

### Implementación:

* + **Conexión** (I2C):
    - VCC a 3.3V, GND a GND, SDA a GPIO 21, SCL a GPIO 22.

**Programación**:

#include <Wire.h>  
#include <Adafruit\_Sensor.h>  
#include <Adafruit\_BME280.h>  
Adafruit\_BME280 bme;  
void setup() {  
Serial.begin(115200);  
if (!bme.begin(0x76)) {  
Serial.println("Could not find a valid BME280 sensor, check wiring!");  
while (1);  
}}

void loop() {  
 Serial.print("Temperature = ");  
 Serial.print(bme.readTemperature());  
 Serial.println(" °C");  
 Serial.print("Humidity = ");  
 Serial.print(bme.readHumidity());  
 Serial.println(" %");  
 Serial.print("Pressure = ");  
 Serial.print(bme.readPressure() / 100.0F);  
Serial.println(" hPa");  
 delay(2000);}  
**Diagrama de Conexión**:

ESP32 BME280  
---------------------  
3.3V --> VCC  
GND --> GND  
GPIO 21 --> SDA  
GPIO 22 --> SCL

## Sensor de Humedad del Suelo Capacitivo (YL-69)

* Características:
  + **Rango de humedad**: 0% a 100%
  + **Tipo**: Capacitivo, más duradero que los sensores resistivos.

### Datos Técnicos:

* + **Voltaje de operación**: 3.3V a 5V
  + **Consumo de corriente**: Bajo

### Implementación:

* + **Conexión**:
    - **VCC** a **3.3V o 5V** del ESP32  
      **GND** a **GND** del ESP32  
      **AO** a un **GPIO analógico** del ESP32 (ej. GPIO 34)

**Programación**:

int sensorPin = 34; // GPIO 34 for analog read  
int sensorValue = 0;  
void setup() {  
Serial.begin(115200);}

void loop() {  
sensorValue = analogRead(sensorPin);  
float moisture = map(sensorValue, 0, 4095, 0, 100);  
Serial.print("Humedad del suelo:");  
Serial.print(humedad);  
Serial.println("%");  
delay(2000);  
}

**Diagrama de Conexión**:

ESP32 YL-69  
---------------------  
3.3V --> VCC  
GND --> GND  
GPIO 34 --> AO

# Sensor de pH

## Sensor de pH Analogico (Analog pH Sensor)

### Características:

* + **Rango de pH**: 0-14 pH  
    **Precisión**: ±0.1 pH

### Datos Técnicos:

* + **Voltaje de operación**: 5V  
    **Salida**: Analógica

### Implementación:

* + **Conexión**:
    - **VCC** a **5V** del ESP32  
      **GND** a **GND** del ESP32  
      **VOUT** a un **GPIO analógico** del ESP32 (ej. GPIO 34)

**Programación**:

int sensorPin = 34; // GPIO 34   
int sensorValue = 0;  
float pHValue = 0;

void setup() {  
Serial.begin(115200);}

void loop() {  
sensorValue = analogRead(sensorPin);  
pHValue = map(sensorValue, 0, 4095, 0, 14);   
Serial.print("Valor Ph: ");  
Serial.println(pHValue);  
 delay(2000);  
}

**Diagrama de Conexión**:

ESP32 pH Sensor  
---------------------  
5V --> VCC  
GND --> GND  
GPIO 34 --> VOUT

## Modelo: DFRobot Gravity pH Sensor

### Características:

* + Rango de pH: 0-14 pH.
  + Precisión: ±0.1 pH.

### Datos Técnicos:

* + Voltaje de operación: 5V.
  + Salida: Analógica.

### Implementación:

* + **Conexión**: VCC a 5V, GND a GND, VOUT a GPIO analógico (ej. GPIO 34).
  + **Programación**:

int sensorPin = 34;  
int sensorValue = 0;  
float pHValue = 0;

void setup() {  
 Serial.begin(115200);  
}

void loop() {

sensorValue = analogRead(sensorPin);  
 pHValue = map(sensorValue, 0, 4095, 0, 14);  
Serial.print("pH Value: ");  
Serial.println(pHValue);  
delay(2000);}

**Diagrama de Conexión**:

ESP32 pH Sensor  
---------------------  
5V --> VCC  
GND --> GND  
GPIO 34 --> VOUT

# Sensores de Lluvia

## Sensor de Lluvia FC-37

### Características:

* + Tipo analógico.  
    Detecta lluvia mediante la conductividad del agua.

### Datos Técnicos:

* + Voltaje de operación: 3.3V a 5V.  
    Salida: Analógica.

### Implementación:

* + **Conexión**: VCC a 3.3V o 5V, GND a GND, AO a GPIO analógico (ej. GPIO 34).

**Programación**:

int sensorPin = 34;  
int sensorValue = 0;  
void setup() {  
 Serial.begin(115200);}

void loop() {  
sensorValue = analogRead(sensorPin);  
int rainDetected = (sensorValue > 1000) ? 1 : 0;  
Serial.print("Lluvia detectectada: ");  
 Serial.println(rainDetected);  
delay(2000);}

**Diagrama de Conexión**:

ESP32 FC-37  
---------------------  
5V --> VCC  
GND --> GND  
GPIO 34 --> AO

# Sensores de Humedad del Suelo

## Sensor de Humedad del Suelo Resistivo (Analogico) Modelo: YL-69

### Características:

* + Tipo resistivo.  
    Rango de humedad: 0% a 100%.  
    Menos duradero, propenso a la corrosión.

### Datos Técnicos:

* + Voltaje de operación: 3.3V a 5V.  
    Salida: Analógica.

### Implementación:

* + **Conexión**: VCC a 3.3V o 5V, GND a GND, AO a GPIO analógico (ej. GPIO 34).

**Programación**:

int sensorPin = 34;  
int sensorValue = 0;  
void setup() {  
Serial.begin(115200);}

void loop() {  
sensorValue = analogRead(sensorPin);  
float moisture = map(sensorValue, 0, 4095, 0, 100);  
Serial.print("Soil Moisture: ");  
Serial.print(moisture);  
Serial.println("%");  
delay(2000);}

**Diagrama de Conexión**:

ESP32 YL-69  
---------------------  
3.3V --> VCC  
GND --> GND  
GPIO 34 --> AO

## Sensor de Humedad del Suelo Capacitivo (Analogico) Modelo: FC-28

### Características:

* + Tipo capacitivo.  
    Rango de humedad: 0% a 100%.  
    Mayor durabilidad comparado con resistivos.

### Datos Técnicos:

* + Voltaje de operación: 3.3V a 5V.  
    Salida: Analógica.

### Implementación:

* + **Conexión**: VCC a 3.3V o 5V, GND a GND, AO a GPIO analógico (ej. GPIO 34).

**Programación**:

int sensorPin = 34;  
int sensorValue = 0;

void setup() {  
Serial.begin(115200);}

void loop() {  
sensorValue = analogRead(sensorPin);  
float moisture = map(sensorValue, 0, 4095, 0, 100);  
 Serial.print("Soil Moisture: ");  
Serial.print(moisture);  
Serial.println("%");  
delay(2000);}

**Diagrama de Conexión**:

ESP32 FC-28  
---------------------  
3.3V --> VCC  
GND --> GND  
GPIO 34 --> AO