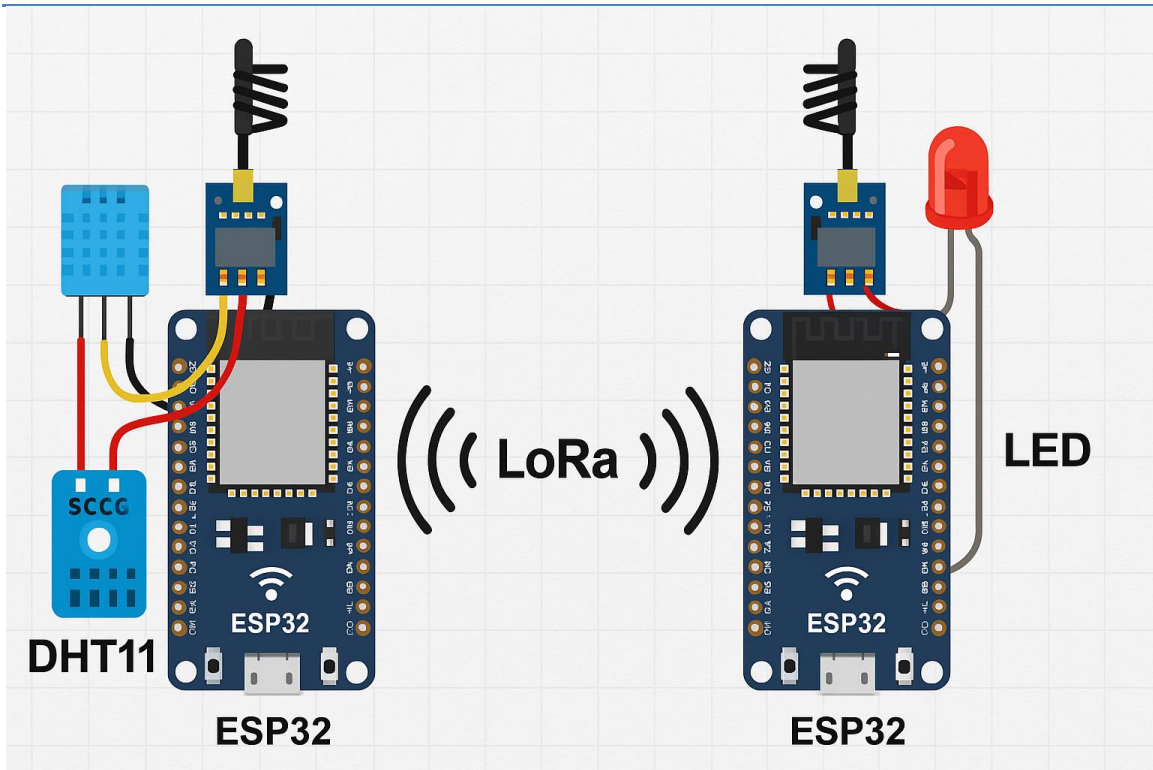


Sistema de monitoreo ambiental por LoRa con activación remota



Integrantes: Fernando Gimenez Coria, Nicolás Barrionuevo, Macarena Aylen Carballo, Raul Jara, Diego Ezequiel Ares, Juan Diego González Antoniazzi.

Materia: Arquitectura y Conectividad

Profesor/a: Jorge Elias Morales

Objetivos del Proyecto

- Desarrollar un sistema de transmisión de datos por radiofrecuencia utilizando tecnología LoRa y placas ESP32.
- Monitorear temperatura y humedad desde un nodo transmisor.
- Activar un actuador (relay) en el nodo receptor si la temperatura supera los 30 °C.

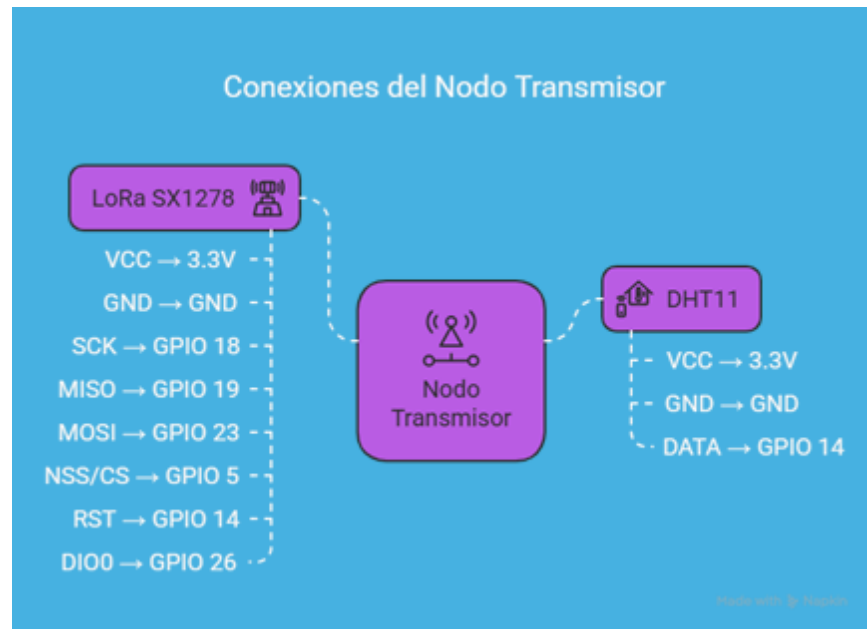
Componentes Utilizados

- 2x ESP32 Dev Module
- 2x Módulo LoRa SX1278
- Sensor DHT11
- Relay o LED para simular el relay
- Protoboard y cables

Conexiones Nodo Transmisor

Conexiones:

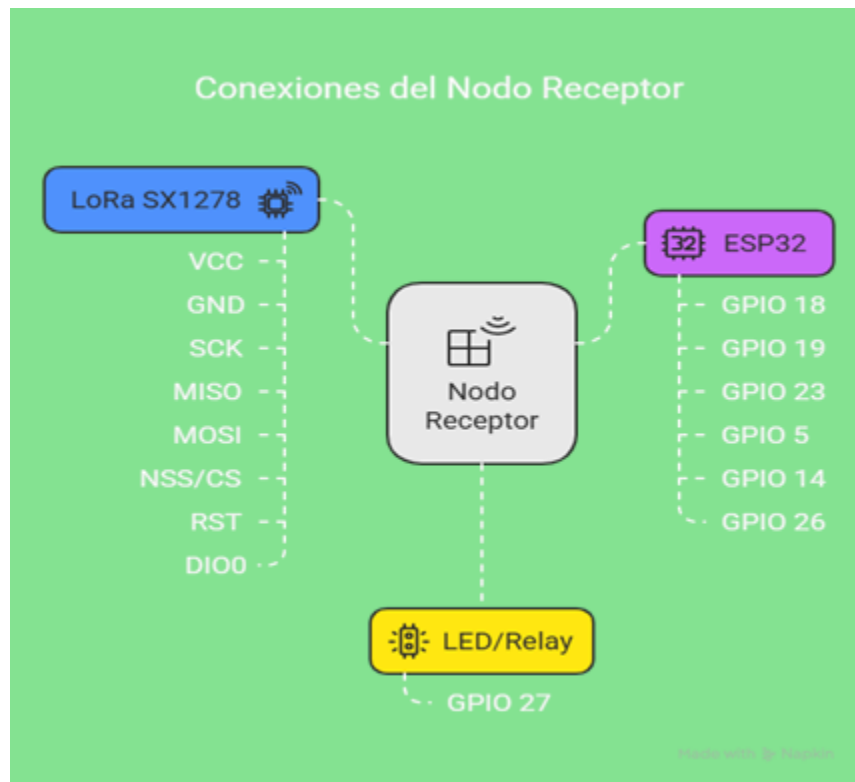
- Sensor DHT11:
- VCC → 3.3V ESP32
- GND → GND ESP32
- DATA → GPIO 4
- Módulo LoRa SX1278:
- NSS (SS) → GPIO 5
- RESET → GPIO 14
- DIO0 → GPIO 26
- MOSI → GPIO 23
- MISO → GPIO 19
- SCK → GPIO 18



Conexiones Nodo Receptor

Conexiones:

- Módulo LoRa SX1278:
- Igual al transmisor
- LED/Relay (simulado):
- IN → GPIO 27
- VCC y GND al ESP32



Lógica del Nodo Transmisor

- Se utiliza el sensor DHT11 para leer datos de temperatura y humedad.
- Estos valores se envían a través del módulo LoRa.

```

#include <Arduino.h>
#include <DHT.h>
#include <SPI.h>
#include <LoRa.h>

#define DHTPIN 4
#define DHTTYPE DHT11
#define SS 5
#define RST 14
#define DIO0 26

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  dht.begin();
  LoRa.setPins(SS, RST, DIO0);
}

```

```

    LoRa.begin(433E6);
}

void loop() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();

    if (!isnan(h) && !isnan(t)) {
        Serial.printf("Temp: %.1f°C | Hum: %.1f%%\n", t, h);
        LoRa.beginPacket();
        LoRa.printf("Temp: %.1f Hum: %.1f", t, h);
        LoRa.endPacket();
    } else {
        Serial.println("Fallo al leer DHT11.");
    }

    delay(5000);
}

```

Lógica del Nodo Receptor

- El nodo receptor escucha mensajes LoRa.
- Si la temperatura recibida supera los 30 °C, activa un relay (o LED).

Código:

```

#include <SPI.h>
#include <LoRa.h>

#define LORA_SCK 18
#define LORA_MISO 19
#define LORA_MOSI 23
#define LORA_SS 5
#define LORA_RST 14
#define LORA_DIO0 26
#define RELAY_PIN 27

void setup() {
    Serial.begin(115200);
    pinMode(RELAY_PIN, OUTPUT);
    digitalWrite(RELAY_PIN, LOW);

    LoRa.setPins(LORA_SS, LORA_RST, LORA_DIO0);
    LoRa.begin(433E6);
}

```

```

void loop() {
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    String incoming = "";
    while (LoRa.available()) {
      incoming += (char)LoRa.read();
    }

    Serial.println("Mensaje recibido: " + incoming);

    if (incoming.indexOf("Temp:") != -1) {
      int tempIndex = incoming.indexOf("Temp:") + 5;
      int spaceIndex = incoming.indexOf(' ', tempIndex);
      String tempStr = incoming.substring(tempIndex, spaceIndex);
      float temp = tempStr.toFloat();

      if (temp > 30) {
        digitalWrite(RELAY_PIN, HIGH);
        Serial.println(">>> Relay ON (Temp alta)");
      } else {
        digitalWrite(RELAY_PIN, LOW);
        Serial.println(">>> Relay OFF (Temp normal)");
      }
    }
  }
}

```

Pruebas y Resultados

Evidencias:

- Captura de mensajes seriales: temperatura y humedad enviadas correctamente.
- Activación visual del LED cuando la temperatura superó los 30 °C.

Dificultades encontradas:

- Lectura nula del DHT11 (resuelto revisando alimentación).
- Error de conexión del módulo LoRa (resuelto verificando pines).

Conclusiones

- Se logró una comunicación efectiva entre dos nodos ESP32 mediante LoRa.
- Se comprobó el envío correcto de datos ambientales.
- Se implementó un control remoto básico de actuadores.

- El sistema puede escalarse para monitoreo agrícola, domótica o control ambiental.