



Tecnicatura Superior en Telecomunicaciones

Modulo: Sensores y actuadores

Alumnos:

- Durigutti, Vittorio | GitHub: <https://github.com/vittoriiodurigutti>
- Zalazar, Joaquín | GitHub: <https://github.com/breaakerr>
- Marquez, José | Github: <https://github.com/marquezjose>
- Lujan, Luciano | Github: <https://github.com/lucianoilujan>
- Velez, Nahuel | Github: <https://github.com/Lucasmurua19>
- Juncos, Lisandro | Github: <https://github.com/Lisandro-05>
- Garzón, Joaquín | Github: <https://github.com/Joacogarzonn>

Profesor: Jorge Morales
Fecha: 18/10/2024

Documento Técnico: Sistema de Riego Automatizado con ESP32

1. Introducción

El presente documento describe el diseño y la implementación de un sistema de riego automatizado utilizando un microcontrolador ESP32. Este sistema utiliza sensores simulados para medir la humedad del suelo, el nivel del tanque y la lluvia, controlando una bomba de agua según los parámetros establecidos.

2. Descripción del Hardware

2.1 Componentes Utilizados

- **Microcontrolador:** ESP32
- **Pantalla LCD:** 16x2 con interfaz I2C
- **Sensores Simulados:**
 - Potenciómetro para simular sensor de lluvia (Pin 34)
 - Potenciómetro para simular sensor resistivo de humedad del suelo (Pin 35)
 - Potenciómetro para simular sensor capacitivo de humedad del suelo (Pin 32)
 - Potenciómetro para simular el nivel del tanque (Pin 33)
- **Relé:** Para controlar la bomba (Pin 25)

2.2 Conexiones

1. Pantalla LCD

Para conectar la pantalla LCD al ESP32, se utilizan los pines de comunicación I2C. La pantalla LCD es una herramienta útil para mostrar información en tiempo real.

- **Conexiones:**
 - SDA (Data Line): Conectar el pin SDA de la pantalla LCD al pin 21 del ESP32.
 - SCL (Clock Line): Conectar el pin SCL de la pantalla LCD al pin 22 del ESP32.

- VCC: Conectar el pin VCC de la pantalla LCD a 3.3V del ESP32.
- GND: Conectar el pin GND de la pantalla LCD a GND del ESP32.

2. Sensores

Los sensores se simulan utilizando potenciómetros conectados a los pines analógicos del ESP32. Esto permite medir diferentes variables como la humedad del suelo y el nivel de lluvia.

- Conexiones de Potenciómetros:
 - Sensor de Lluvia (Potenciómetro):
 - Conectar un extremo del potenciómetro a 3.3V.
 - Conectar el otro extremo a GND.
 - Conectar el pin central (señal) al pin 34 del ESP32.
 - Sensor Resistivo de Humedad del Suelo (Potenciómetro):
 - Conectar un extremo a 3.3V.
 - Conectar el otro extremo a GND.
 - Conectar el pin central al pin 35 del ESP32.
 - Sensor Capacitivo de Humedad del Suelo (Potenciómetro):
 - Conectar un extremo a 3.3V.
 - Conectar el otro extremo a GND.
 - Conectar el pin central al pin 32 del ESP32.
 - Nivel del Tanque (Potenciómetro):
 - Conectar un extremo a 3.3V.
 - Conectar el otro extremo a GND.
 - Conectar el pin central al pin 33 del ESP32.

3. Relé

El relé se utiliza para controlar la bomba de agua, permitiendo encenderla o apagarla según las condiciones establecidas por los sensores.

Conexiones:

- Conectar la entrada del relé al pin 25 del ESP32.
- Asegúrate de que el relé esté alimentado correctamente según sus especificaciones (generalmente requiere conexión a GND y VCC).
- La salida del relé debe estar conectada a la bomba, asegurando que se activa o desactiva según la señal enviada desde el ESP32.

3. Código Fuente

El siguiente código implementa la lógica del sistema de riego:

```
cpp
#include <Arduino.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Configuramos la pantalla LCD (dirección y tamaño 16x2)
LiquidCrystal_I2C pantalla(0x27, 16, 2);

// Pines de los potenciómetros (simuladores de sensores)
int lluvia = 34;
int sueloResistivo = 35;
int sueloCapacitivo = 32;
int tanque = 33;

// Salida del relé para controlar la bomba
int releBomba = 25;

// Umbrales para activar el riego
int umbralLluvia = 2000;
int umbralSueloResistivo = 1500;
int umbralSueloCapacitivo = 1500;
int umbralTanqueBajo = 1000;
int umbralTanqueMedio = 2000;
int umbralTanqueLleno = 3000;

void setup() {
    pantalla.init();
    pantalla.backlight();

    pinMode(releBomba, OUTPUT);
    digitalWrite(releBomba, LOW);

    Serial.begin(9600);

    pantalla.setCursor(0, 0);
    pantalla.print("Sistema Riego");
    pantalla.setCursor(0, 1);
    pantalla.print("Iniciando...");
    delay(2000);
}
```

```

void loop() {

    int valorLluviaRaw = analogRead(lluvia);
    int valorSueloResistivoRaw = analogRead(sueloResistivo);
    int valorSueloCapacitivoRaw = analogRead(sueloCapacitivo);
    int valorTanque = analogRead(tanque);

    int valorLluvia = map(valorLluviaRaw, 0, 4095, 0, 100);
    int valorSueloResistivo = map(valorSueloResistivoRaw, 0, 4095, 0, 100);
    int valorSueloCapacitivo = map(valorSueloCapacitivoRaw, 0, 4095, 0, 100);

    String nivelTanque;

    if (valorTanque >= umbralTanqueLleno) {
        nivelTanque = "Lleno";
    } else if (valorTanque >= umbralTanqueMedio) {
        nivelTanque = "Medio";
    } else {
        nivelTanque = "Bajo";
    }

    pantalla.clear();
    pantalla.setCursor(0, 0);
    pantalla.print("Lluvia: ");
    pantalla.print(valorLluvia);

    pantalla.setCursor(0, 1);
    pantalla.print("SueloRes: ");
    pantalla.print(valorSueloResistivo);

    delay(2000);

    pantalla.clear();
    pantalla.setCursor(0, 0);
    pantalla.print("SueloCap: ");
    pantalla.print(valorSueloCapacitivo);

    pantalla.setCursor(0, 1);
    pantalla.print("Tanque: ");
    pantalla.print(nivelTanque);

    if (valorLluvia < umbralLluvia &&
        (valorSueloResistivo < umbralSueloResistivo || valorSueloCapacitivo < umbralSueloCapacitivo) &&
        nivelTanque != "Bajo") {

        digitalWrite(releBomba, HIGH);
    }
}

```

```
pantalla.setCursor(0,1);  
pantalla.print("Riego: ON ");  
  
Serial.println("Riego activado");  
} else {  
  digitalWrite(releBomba, LOW);  
  pantalla.setCursor(0,1);  
  pantalla.print("Riego: OFF ");  
  Serial.println("Riego desactivado");  
}  
  
Serial.println("-----");  
delay(2000);  
}
```

4. Funcionamiento del Sistema

Lógica General

- El sistema lee los valores de los sensores simulados a través de potenciómetros.
- Los valores se mapean a un rango específico para facilitar su interpretación.
- Se muestran los datos en la pantalla LCD y se envían al monitor serial.
- Se activa o desactiva la bomba según las condiciones de humedad del suelo y el nivel del tanque.

Condiciones para Activar el Riego

- Si no llueve y el suelo está seco (cualquiera de los dos sensores indica baja humedad) y el tanque no está bajo, se activa la bomba.
- Si llueve o el suelo está húmedo o el tanque está bajo, se desactiva la bomba.

Conclusiones:

Este proyecto permite automatizar el riego en función de las condiciones ambientales y del suelo. La implementación es flexible y puede adaptarse a diferentes tipos de sensores y configuraciones.