

UNIVERSIDAD TECNOLOGICA NACIONAL

TECNICATURA UNIVERSITARIA EN PROGRAMACION

SISTEMA DE PROCESAMIENTO DE DATOS

2023

Docentes:

- Carolina Mihovilcevic
- Sebastian Bruselario

Integrantes:

- Ramiro Trevisan
- Juan Martin Masqueda
- Lucas Ruggiero

INDICE

1-Introduccion

1.1-Problema

1.2-Componentes a utilizar

1.3-Diseño del sistema

2-Desarrollo

2.1-Explicacion del código

2.2-Utilizacion de la placa Arduino

3-Sistema Finalizado

TRABAJO PRÁCTICO

Sistema de Riego Automatizado

1. Introduccion

1.1 Problema

Un cliente acudió a nosotros con la necesidad de automatizar un sistema de riego, ventilación y luminaria led de su jardín.

El cliente quiere que el sistema de riego automatizado se encienda dependiendo de los siguientes factores:

- Cuando la temperatura sea mayor a 30°C se tendrá que activarse el sistema de ventilación
- Cuando la humedad sea menor a 80% se tendrá que activar el sistema de riego.
- Cuando el jardín detecte menor intensidad de luz, se tendrá que encender la luminaria de su jardín.

Además, se tendrá que mostrar en una pantalla led el estado del suelo, Suelo Seco si la humedad es menor a 80% y Suelo Húmedo si la humedad es mayor al 80%.

También se mostrará en pantalla la temperatura del ambiente.

1.2 Componentes a utilizar:

- 1 Placa Arduino
- Cables
- 1 Sensor de Temperatura
- 1 Sensor de Humedad
- 1 Fotorresistencia (LDR)
- 1 LED RGB
- 4 Resistencias x 220 Ohm
- 2 Motores DC (uno para ventilación, otro para abrir canilla de riego)
- 1 Pantalla LED 16x2
- 1 Potenciómetro
- 1 Protoboard

1.3-Diseño del sistema

Los componentes claves del sistema serán:

Sensores:

- Sensor de temperatura: Se utilizara para medir la temperatura del ambiente
- Sensor de humedad: Se utilizara para medir la humedad del suelo
- Sensor de luz: Se utilizara para detectar la intensidad de la luz en el jardín

Salidas:

- Motor de riego: Se encenderá simulando un regador para mantener el suelo húmedo cuando la humedad sea menor al 80%
- Motor de ventilación: Se activara cuando la temperatura sea mayor a 30°C para proporcionar ventilación al jardín
- Luminaria LED: Se utilizan luces LED para iluminar el jardín cuando se detecte menor intensidad de luz. En nuestro caso utilizaremos un LED RGB que cambiara de color evidenciando una disminución de la luz del ambiente.

Pantalla LED:

- Se utilizara una pantalla LED que mostrara la información en tiempo real, el estado del suelo(SECO /HUMEDO) y la temperatura del ambiente

2. Desarrollo

El código en la placa Arduino será en lenguaje de programación C, dándole funcionalidad a cada uno de sus componentes dándole instrucciones exactas para su correcto funcionamiento.

2.1-Explicación del código:

Además del código base se utilizara la librería LiquidCrystal, la cual será necesaria para que la placa Arduino pueda controlar la pantalla LCD. Para incluir esta librería lo haremos con la línea de código **#include <LiquidCrystal.h>**.

Luego declaramos de forma global las variables necesarias, como los pines que utilizara la luz RGB, los pines asignados para el motor de ventilación y el motor de riego, el pin analógico que usara el sensor de fotorresistencia para almacenar el valor de luz y también declararemos variables que usaremos más adelante para almacenar los valores de temperatura y humedad.

Dentro de la función loop() daremos la directiva a los sensores de que realicen la medición de sus respectivos valores. Al incluirlo dentro de esta función nos aseguramos que las mediciones se renueven constantemente lo que brindara una mayor precisión y una medición constante.

Con la medición de temperatura tuvimos que utilizar un cálculo para poder convertir la medición en grados Celsius: **tmp = (((5 * valorTmp * 100) / 1024) - 50);** La variable valorTmp es la medición en vivo que arroja el sensor de temperatura, a la cual se le aplican una serie de operaciones matemáticas que la convertirán en su equivalente en grados Celsius, este valor será asignado a la variable tmp.

Con la medición del sensor de humedad nos encontramos con un problema, ya que los valores que arroja el mismo van desde 0 hasta 876, mientras que el cliente nos solicitó la validación en porcentaje. Para resolver esto usamos la función map: ***porcentaje = map(humedad,0,876,0,100)***. Esta función toma los valores que envía el sensor de humedad y realiza una conversión de los valores, en este caso entre 0 y 100 representando un porcentaje.

Una vez que tenemos todos los valores de la forma que queremos realizamos las validaciones necesarias con la estructura de control ***if***, donde si la medición es menor a lo solicitado por el cliente procedemos a ejecutar las respuestas necesarias del sistema para restaurar los valores a la cantidad necesaria.

Por último se mostrara en la pantalla LCD el estado de humedad del suelo y de temperatura del ambiente.

2.2-Utilización de la placa Arduino:

Hablando sobre el uso que le vamos a dar a la placa Arduino, utilizaremos tanto pines digitales como pines analógicos.

En el caso de la pantalla LCD, la luminaria LED y los motores utilizaremos pines digitales, ya que no será necesaria una medición continua sino que se maneja por encendido o apagado, esto nos permitirá un control más preciso. Por ejemplo, los motores se encenderán según un umbral fijo (valor mínimo de humedad o valor máximo de temperatura).

Estos pines también nos facilitaran el desarrollo del código ya que una vez declarados, se activaran o desactivaran con la función ***digitalWrite()***.

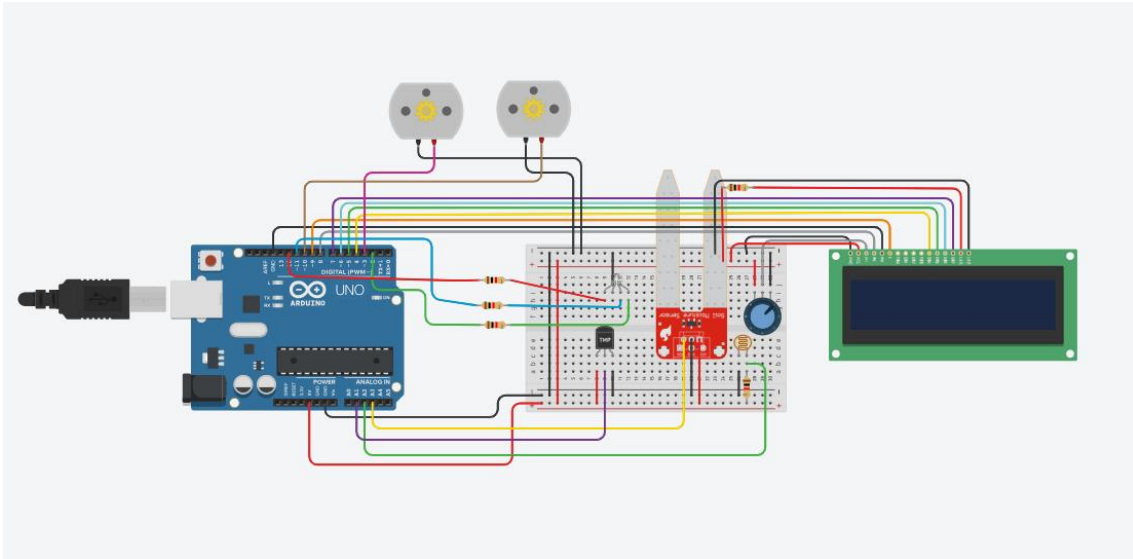
Usaremos pines analógicos en el caso de los sensores de temperatura, humedad y luz, ya que estos proporcionan datos en forma variable. Los pines analógicos nos permitirán una lectura precisa de estos sensores.

Al darse una medición continua y constante, el uso de pines analógicos nos permitirá identificar cualquier cambio en las condiciones ambientales del jardín por más mínimo que sea. Esto asegurara una respuesta optima del sistema.

Según los requerimientos del cliente los umbrales definidos serán:

- Cuando la temperatura supere los 30°C se activara el motor del ventilador
- Si la humedad en el suelo es inferior a 80%, se activara el motor de riego
- Cuando el sensor de luz detecte una intensidad insuficiente, encenderá las luces del jardín

3. Sistema finalizado



Codigo:

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

int rojo = 12;
int azul = 11;
int verde = 2;
int m_vent = 3;
int m_riego = 10;
int luz = A2;
float valorTmp;
float humedad;
float tmp;

void setup() {
  lcd.begin(16, 2);
  pinMode(m_vent, OUTPUT);
  pinMode(m_riego, OUTPUT);
  pinMode(rojo, OUTPUT);
  pinMode(azul, OUTPUT);
  pinMode(verde, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  valorTmp = analogRead(A1);
  humedad = analogRead(A3);
```

```

int foto = analogRead(luz);

tmp = (((5 * valorTmp * 100) / 1024) - 50);
lcd.setCursor(0, 0);
lcd.print("TEMP: ");
lcd.print(tmp);
lcd.print(" C");

if (tmp > 30) {
    digitalWrite(m_vent, HIGH);
} else {
    digitalWrite(m_vent, LOW);
}

if (foto < 680) {
    digitalWrite(rojo, 0);
    digitalWrite(verde, 255);
    digitalWrite(azul, 0);
} else {
    digitalWrite(rojo, 255);
    digitalWrite(verde, 0);
    digitalWrite(azul, 0);
}

float porcentaje = map(humedad, 0, 876, 0, 100);

lcd.setCursor(0, 1);
if (porcentaje < 80) {
    lcd.print("SUELO SECO");
    digitalWrite(m_riego, HIGH);
} else {
    lcd.print("SUELO HUMEDO");
    digitalWrite(m_riego, LOW);
}

delay(1000);
lcd.clear();
}

```