



# UNIVERSIDAD NACIONAL DEL ALTIPLANO

FACULTAD DE INGENIERÍA ELECTRICA, ELECTRONICA  
Y DE SISTEMAS

ESCUELA PROFESIONAL DE INGENIERIA DE SISTEMAS

"Sistema de Riego Automatizado"

CURSO  
NACIONAL DEL  
SISTEMAS ELECTRONICOS Y DIGITALES

PRESENTA

Anco Coaquira Jorge Luis

Delgado Jimenez Mario Saul

Machaca Cahuana Gianmarco Alain Bruno

DOCENTE

GONZALES PACO MAGALI GIANINA

14/07/2025



## **Resumen**

El presente proyecto presenta el diseño e implementación de un sistema de riego automático utilizando una placa Arduino Uno, sensores DHT11 para monitoreo de temperatura y humedad, una bomba de agua controlada por un módulo relé y una pantalla LCD para visualización en tiempo real. El sistema permite activar el riego automáticamente cuando la temperatura es igual o superior a 25 °C y la humedad relativa del suelo desciende por debajo del 20 %, o manualmente mediante un botón. Se busca mejorar la eficiencia en el uso del agua, automatizando el proceso de riego de cultivos o jardines. Este sistema representa una solución educativa y económica, ideal para estudiantes que deseen integrar conocimientos de sensores, programación y control automático en aplicaciones de la vida real.

## **Introducción**

La automatización de procesos agrícolas es una necesidad creciente en la actualidad, especialmente en zonas donde el recurso hídrico debe ser administrado de manera eficiente. El riego automático con Arduino surge como una alternativa accesible y educativa para automatizar el suministro de agua a las plantas, basándose en condiciones ambientales captadas mediante sensores. Mediante el uso de la placa Arduino Uno, sensores DHT11, una bomba de agua y un módulo relé, se construyó un sistema capaz de tomar decisiones automáticas para activar el riego o permitir la activación manual. La pantalla LCD integrada facilita la visualización en tiempo real de los parámetros ambientales.

## **Objetivos**

## **Objetivo General**

Diseñar y construir un sistema de riego automático controlado por Arduino que active el riego según condiciones de temperatura y humedad del ambiente.

## **Objetivos Específicos**

1. Monitorear temperatura y humedad utilizando el sensor DHT11.
2. Controlar el encendido de una bomba de agua mediante un módulo relé.
3. Implementar una visualización en tiempo real mediante una pantalla LCD.
4. Integrar una función de riego manual mediante un botón.
5. Evaluar el funcionamiento del sistema en condiciones simuladas de temperatura y humedad.

## **Definición del Problema**

En zonas donde el riego constante y eficiente es crucial para el crecimiento de las plantas, la gestión manual del agua se convierte en una tarea ineficiente y propensa a errores. Muchas veces, el riego se realiza sin considerar las condiciones reales del ambiente, lo cual puede resultar en desperdicio de agua o en daños a las plantas. Este proyecto busca resolver esta problemática mediante un sistema automatizado capaz de tomar decisiones de riego según condiciones ambientales.

## Justificación

La necesidad de automatizar procesos en la agricultura y jardinería impulsa el desarrollo de soluciones accesibles y replicables. Arduino ofrece una plataforma ideal para prototipos educativos que permiten a los estudiantes explorar conceptos de programación, electrónica y control de sistemas en tiempo real. Este proyecto no solo promueve el aprendizaje, sino que también ofrece una aplicación práctica con impacto directo en el ahorro de agua y energía.

## Marco Teórico

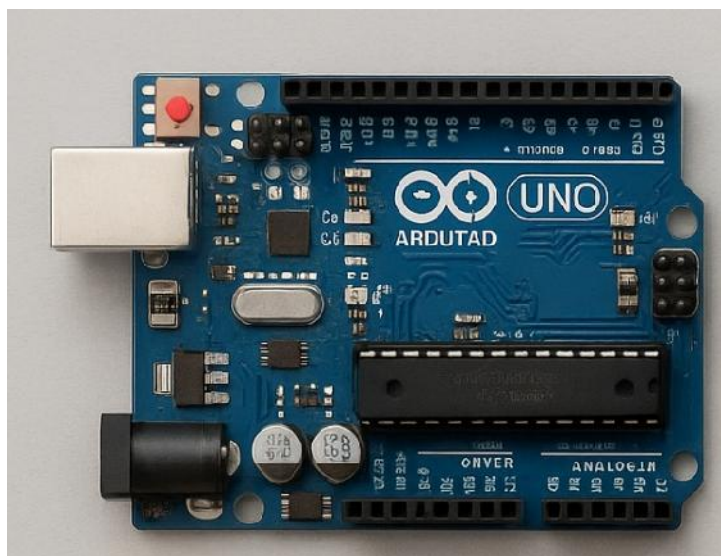
### Modos de Operación

- ) **Arduino Uno:** Placa de desarrollo basada en el microcontrolador ATmega328P. Es ampliamente utilizada en aplicaciones educativas y de prototipado rápido.
- ) **Sensor DHT11:** Sensor digital que mide temperatura y humedad relativa. Proporciona datos en forma digital al Arduino.
- ) **Relé:** Dispositivo electromecánico que permite activar cargas de mayor potencia (como bombas) desde una señal digital de bajo voltaje.
- ) **Pantalla LCD 16x2:** Permite la visualización de mensajes y datos en dos filas de 16 caracteres.
- ) **Programación en Arduino:** Uso del entorno Arduino IDE para programar la lógica de lectura de sensores, control de actuadores y despliegue de información en pantalla.

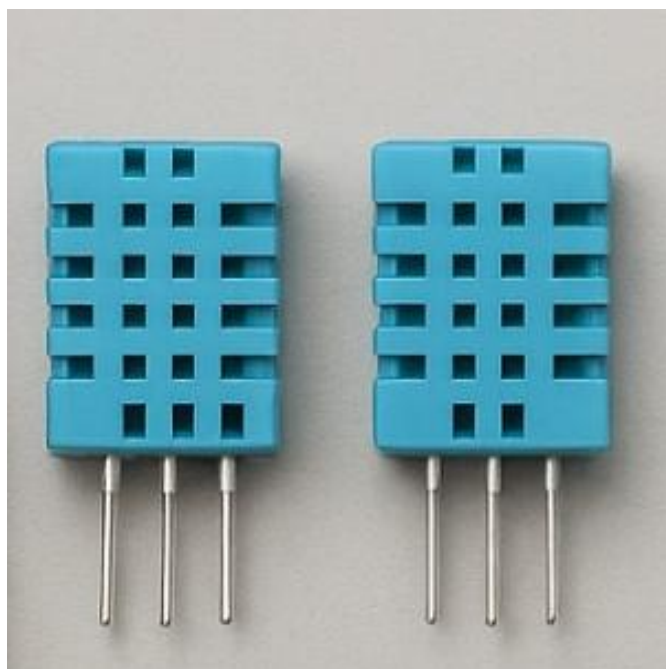
## Desarrollo

### Materiales Utilizados Componentes Electrónicos

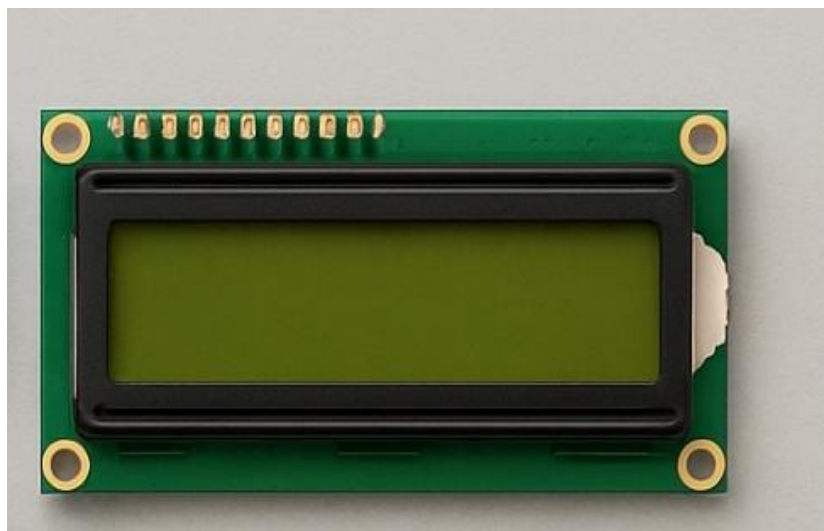
) Placa Arduino Uno



) Sensor DHT11 (x2)



) Pantalla LCD 16x2



) Bomba de agua



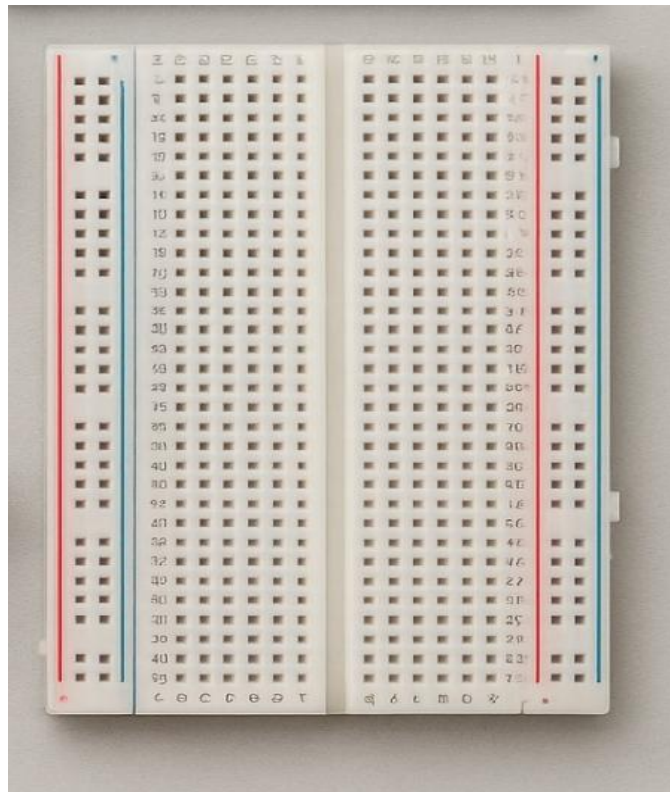
) Relé de 5V



) Botón pulsador



) Protoboard



) Fuente de alimentación de 5V, cables y resistencias





## Construcción del Circuito

### Paso 1: Montaje

1. Conectar el sensor DHT11 al pin digital 9.
2. Conectar el módulo relé al pin digital 8 para controlar la bomba.
3. Conectar la pantalla LCD a los pines 2 al 7 mediante la librería LiquidCrystal.
4. Conectar el botón al pin digital 10.
5. Programar el Arduino para leer datos del sensor DHT11.
6. Establecer la condición: si temperatura  $\geq 25$  °C y humedad  $\leq 20$  %, activar la bomba por 5 segundos.
7. Si se presiona el botón, activar la bomba por 1 segundo.
8. Mostrar constantemente los valores de temperatura y humedad en la pantalla LCD.
9. Mostrar mensajes de error si el sensor no responde.

### Análisis e interpretación de resultados

- ) El sistema responde correctamente cuando las condiciones ambientales cumplen los criterios programados.

- ) La bomba se activa por el tiempo establecido, mostrando un mensaje de "Regando" en la pantalla.
- ) En ausencia de datos del sensor (por desconexión), el sistema muestra mensajes de "Error".
- ) El botón funciona como activador manual del riego, complementando el modo automático.

### **Resultados**

- ) Se logró una automatización efectiva del riego.
- ) La visualización en pantalla mejora la experiencia del usuario.
- ) El sistema funciona de forma estable y con bajo consumo energético.
- ) Se identificó la importancia de utilizar sensores calibrados y componentes de calidad.

### **Conclusiones**

- ) El proyecto cumplió con los objetivos planteados, demostrando la viabilidad de un sistema de riego automático educativo.
- ) La integración de sensores, actuadores y una interfaz visual facilita el monitoreo ambiental y el control de riego.

- ) Se fomentó el aprendizaje práctico de programación, electrónica digital y control automático.

### Bibliografía y Referencias

Banzi, M., & Shiloh, M. (2015). *Getting started with Arduino* (3rd ed.). Maker Media.

Adafruit Industries. (s.f.). *DHT11 Basic temperature-humidity sensor + extras*. Recuperado de <https://learn.adafruit.com/dht>

Arduino. (s.f.). *Arduino Documentation*. Recuperado el 7 de julio de 2025, de <https://www.arduino.cc>

Arduino. (s.f.). *LiquidCrystal Library*. Recuperado de <https://www.arduino.cc/en/Reference/LiquidCrystal>

### Anexos

- ) Fotografías del montaje en protoboard y pruebas.



) Código fuente completo.

### CÓDIGO COMPLETO:

```
#include <LiquidCrystal.h>
#include <DHT.h>
#include <DHT_U.h>

//Pines
int bombaAgua = 8;
int sensorTemp = 9;
int boton = 10;

//variables
int temp;
int hum;
int estadoBoton;

//Objetos
LiquidCrystal lcd(7,6,5,4,3,2);
DHT dht(sensorTemp, DHT11);

void setup() {
  //setup Pantalla
  lcd.begin(16, 2);

  //setup rele
  pinMode(bombaAgua,OUTPUT);

  //setup Sensor temp
  dht.begin();

  //setup boton
  pinMode(boton, INPUT);
}

void loop() {
  //Inicializacion de Variables sensor
  temp = dht.readTemperature();
  hum = dht.readHumidity();
  estadoBoton=digitalRead(boton);

  //Pantalla
  lcd.setCursor(0,0);
  lcd.print("Temperatura: " + String(temp) + " °C");
  lcd.setCursor(0,1);
```

```

lcd.print("Humedad: "+ String(hum) +" %    " );

if(temp>=25&&hum<=20){ //Condicion para regar automaticamente
    digitalWrite(bombaAgua, LOW); //Prender bomba de agua

    lcd.setCursor(0,0);
    lcd.print("*****Regando*****"); // Mostrar en la pantalla LCD
    lcd.setCursor(0,1);
    lcd.print("*****");
    delay(5000); //Durante 5 segundos

} else if(estadoBoton==HIGH){ //Si se presiona el botón
    digitalWrite(bombaAgua, LOW); //Prender bomba de agua

    lcd.setCursor(0,0);
    lcd.print("*****Regando*****"); // Mostrar en la pantalla LCD
    lcd.setCursor(0,1);
    lcd.print("*****");
    delay(1000); //Durante un segundo

} else { //Si ninguna de las condiciones anteriores...
    digitalWrite(bombaAgua, HIGH); //Bomba apagada
}

//Si no recibe informacion del sensor
if(temp==0&&hum==0){
    lcd.setCursor(0,0);
    lcd.print("*****ERROR*****"); // Mostrar en la pantalla LCD
    lcd.setCursor(0,1);
    lcd.print("****NO SENSOR****");
    delay(1000); //Actualizacion cada segundo
}
}
}

```

) Diagramas de conexión.

