

Informe trabajo practico integrador: *“Sistema de riego automático”*

Integrante(s): Francisco Madrid

Año: 2023

Curso: 6to Electrónica - Grupo 2

Instituto Leonardo Murialdo

Microcontrolador a utilizar: Arduino UNO

- Í N D I C E -

INTRODUCCIÓN

DESARROLLO

Explicación del funcionamiento del proyecto

Diagrama de bloques

Descripción del hardware

Descripción del software/código

ANEXOS

FUENTES

Introducción

Para este proyecto, pensé en que a veces nos podemos olvidar de regar las plantas, con tantas cosas para hacer en un día. Entonces se me ocurrió hacer un proyecto que solucione este problema, es decir, una planta que se riega sola.

El concepto final funciona de la siguiente manera:

"Una planta que está conectada a un sensor de humedad, el cual lee la humedad cada 12 horas, haciendo que la planta tenga un máximo de 2 regados por día. Además, irá conectada una pantalla LCD el cual mostrará el porcentaje de humedad y la última hora de regado".

Desarrollo

Explicación del funcionamiento del proyecto

El proyecto funciona de la siguiente manera:

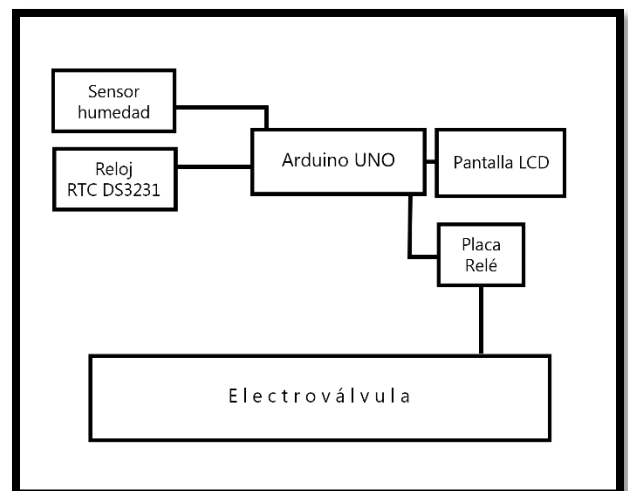
Primero que todo, el arduino se inicializa, y luego lee el valor de humedad, con ayuda del sensor. El arduino sensara desde el inicio, cada 12 horas, la humedad. Si el arduino detecta que la tierra está seca, entonces enviará una señal a una placa que activará una electroválvula, regando así la planta. en caso contrario se mantiene como está hasta el siguiente registro.

Descripción del hardware

En el hardware, tenemos diferentes componentes:

- Sensor de humedad de suelo YL69.
- Reloj de tiempo real DS3231.
- Electroválvula
- Placa relé (varios componentes)
- Arduino UNO
- LCD I2C.

Diagrama de bloques



Sensor de humedad de suelo YL69.

Este sensor tiene la capacidad de medir la humedad del suelo. Aplicando una pequeña tensión entre los terminales del módulo YL-69 hace pasar una corriente que depende básicamente de la resistencia que se genera en el suelo y ésta depende mucho de la humedad. Por lo tanto, al aumentar la humedad la corriente crece y al bajar la corriente disminuye.



El dispositivo cuenta con una lectura digital y una analógica, la analógica siendo la más precisa. [Fuente](#)

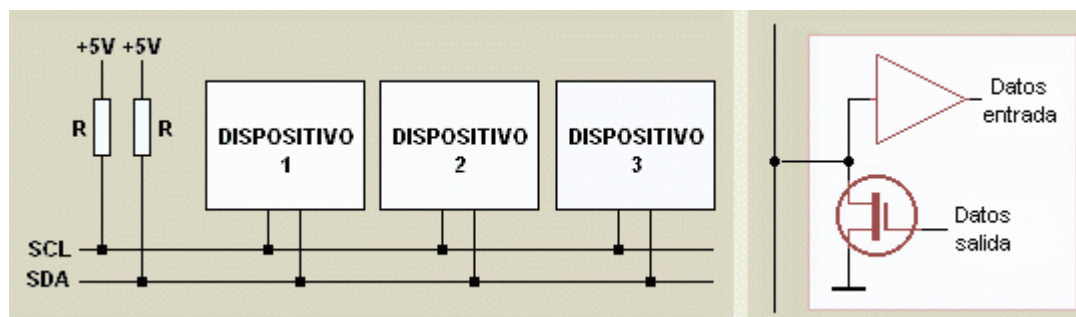
Como se puede deducir, este dispositivo medirá la humedad cuando el RTC se lo indique.



¿Qué es un bus I2C?

Abreviatura de Inter-IC (inter integrated circuits), un tipo de bus diseñado por Philips Semiconductors a principios de los 80s, que se utiliza para conectar circuitos integrados (ICs). El I2C es un bus con múltiples maestros, lo que significa que se pueden conectar varios chips al mismo bus y que todos ellos pueden actuar como maestro, sólo con iniciar la transferencia de datos. Este bus se utiliza dentro de una misma placa de un dispositivo.

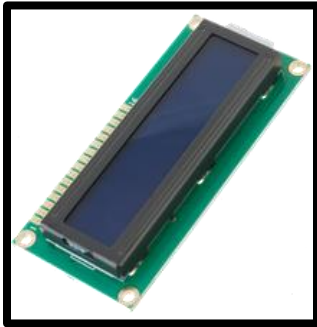
[Fuente](#)



Reloj de tiempo real DS3231 (RTC)



Como su nombre lo indica, el DS3231 es un reloj de tiempo real que es capaz de almacenar la fecha y hora sin necesidad de una fuente de poder externa, viene con un zócalo de pila de 3,3V. En mi caso, se comunica por el Bus I2C. Es el encargado de, cada 12 horas, indicarle al sensor de humedad que mida la humedad y que este mismo active la electroválvula en caso de que la planta esté seca.



Pantalla LCD I2C

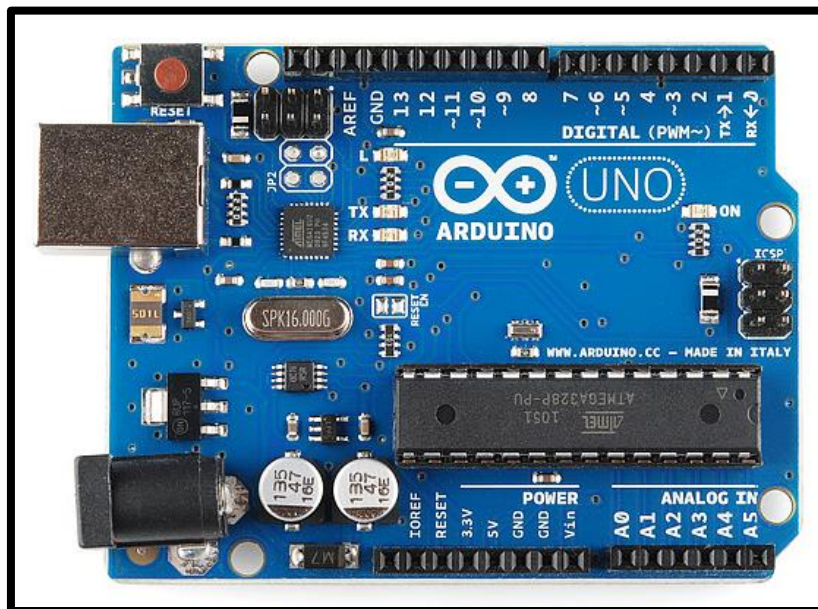
LCD o pantalla de cristal líquido, está conformada por un número de píxeles en color o monocromos colocados delante de una fuente de luz o reflectora, es delgada y plana.

A menudo se utiliza en dispositivos electrónicos de baterías, ya que maneja cantidades muy pequeñas de energía eléctrica.

Es un dispositivo empleado para la **visualización de contenidos o información** de una forma gráfica, mediante caracteres, símbolos o pequeños dibujos, los cuales pueden variar todo esto en función del modelo.

Arduino UNO

El Arduino es una placa basada en un microcontrolador ATMEL. Los microcontroladores son circuitos integrados en los que se pueden grabar instrucciones, las cuales las escribes con el lenguaje de programación que puedes utilizar en el entorno Arduino IDE. Estas instrucciones permiten crear programas que interactúan con los circuitos de la placa. [Fuente](#). En el proyecto es el “Cerebro” encargado de definir entradas y salidas de información.



Interfaz Relé (Placa de relé)

La interfaz relé ayuda a poder transformar una señal digital a una analógica, y poder activar, no transformar, una conexión 220VCA con 5VCC

Electroválvula

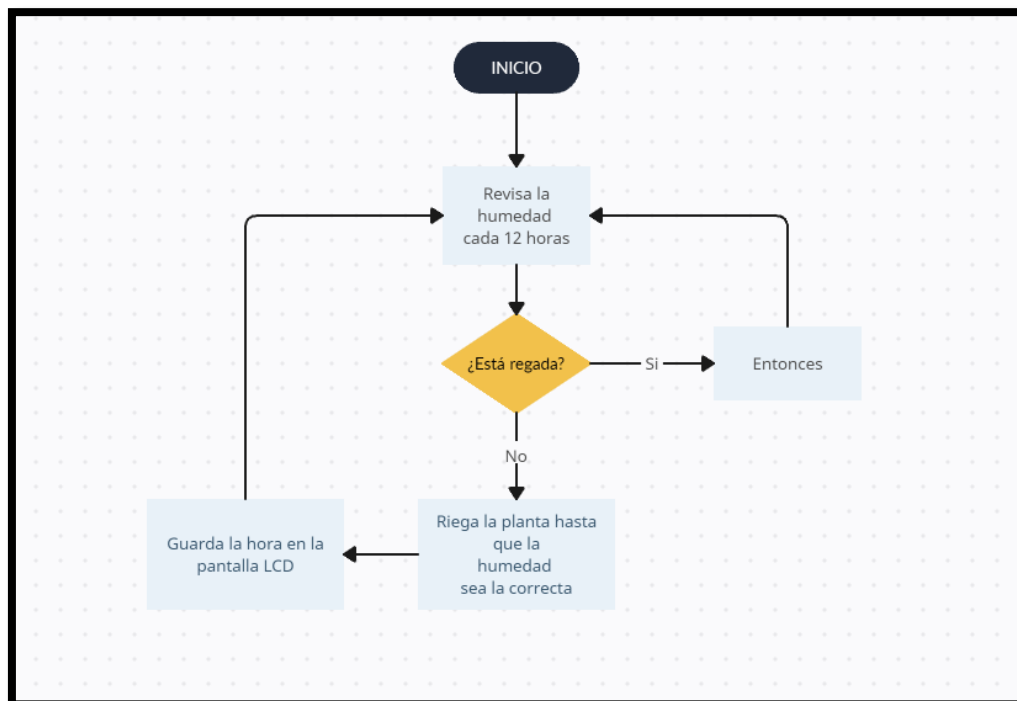
Este dispositivo sirve para regar la planta y va conectada a la interfaz relé, para que el Arduino sea capaz de enviarle la señal más allá de la diferencia de corriente/tensión.

Explicación del funcionamiento del software.

En código, lo primero que hace es inicializar cada dispositivo por separado y calibrar el ds3231. Luego, Revisa inicialmente el numero que devuelve el sensor de humedad (0 a 1023) y como fue planteado antes, si el Arduino detecta un nivel menor a ~700 en este sensor entonces activa la electroválvula, enviando un 1 (high) a través de la salida digital 2 (D2). El Arduino espera a que el sensor de humedad supere el nivel de humedad requerido para apagar la electroválvula, e inicia un contador de 12 horas para censar de nuevo el regado.

Por otro lado, Cuando la planta termina de regarse extrae la hora actual del momento en el que la planta figuró como “Regada” y lo imprime en el LCD. En el LCD también se encuentra un Porcentaje de humedad, hecho con la función `map`.

Diagrama de flujo.



Codigo

A continuación, dejo el [código con comentarios](#) para que puedan leerlo. Recomendando ingresar al link.

```
// INCLUIR LIBRERIAS
#include <Wire.h>
#include <RTCLib.h>
#include <LiquidCrystal_I2C.h>
// DEFINICION DE VARIABLES
RTC_DS3231 rtc;
#define EV 2
unsigned int HUMEDAD=0;
unsigned int inicia=0;
int tiempo = 12;
int e=1;
unsigned int LECTURAP, LECTURAQ;
LiquidCrystal_I2C lcd(0x27,16,2);

// INICIALIZACION DE DISPOSITIVOS
void setup(){
  Serial.begin(9600);

  // Inicializacion de RTC. Revisa si está conectado o no.
  if (! rtc.begin()) {
    Serial.println("Modulo RTC3231 no conectado");
    while (1);}
  rtc.begin();

  lcd.init();          // Inicializacion LCD
  lcd.backlight();
  lcd.clear();

  pinMode(EV, OUTPUT); // Pin 2 como salida

  // Calibrado de RTC. Extrae la hora del dispositivo al que esté conectado el
  arduino.
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
}

// CODIGO PRINCIPAL/BUCLE
void loop() {
  DateTime now = rtc.now(); // Ponemos una variable que significa el tiempo
  HUMEDAD=analogRead(A0);  // Leemos la humedad de 0 a 1023 en el
  sensor de humedad

  // Contador.
```

```
if(now.hour()-inicia>=tiempo ){ // Cada 12 horas,
// SI LA PLANTA ESTÁ SECA:
    if(HUMEDAD>670 && e==1){ // Revisa si la planta está seca.
        digitalWrite (EV, HIGH); // Enciende el la regadora,
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("REGANDO");
        e=2; // Y nos lleva al estado 2
    }
    inicia=now.hour();
}

//CUANDO LA PLANTA YA ESTÁ REGADA:
if(HUMEDAD<669 && e==2){ // Comprobamos si se rego la planta
fijandonos en la humedad, y si estamos en el estado 2 (estado de apagado)
    digitalWrite(EV, LOW); // Apagamos la electrovalvula
    lcd.setCursor(0,0);
    lcd.print("Regado-"); // Guardamos la hora a la que se termino de regar
el LCD.

    lcd.print(now.hour()); // Le pide la hora al rtc
    lcd.print(":"); // : Para separar
    lcd.print(now.minute()); // Ahora pide los minutos
    lcd.print(":"); // : Para separar
    lcd.print(now.second()); // Y por ultimo los segundos
    e=1;
}

// IMPRESION DEL PORCENTAJE DE HUMEDAD
LECTURAP=map (HUMEDAD, 1023, 205, 0, 100); // pasamos los valores de 0 a
1023 a porcentaje (0 a 100)
    lcd.setCursor(0,1); // ponemos el cursor en el principio del LCD
    lcd.print("Humedad= "); // en las siguientes 3 lineas imprimimos el
porcentaje de humedad
    lcd.print(LECTURAP);

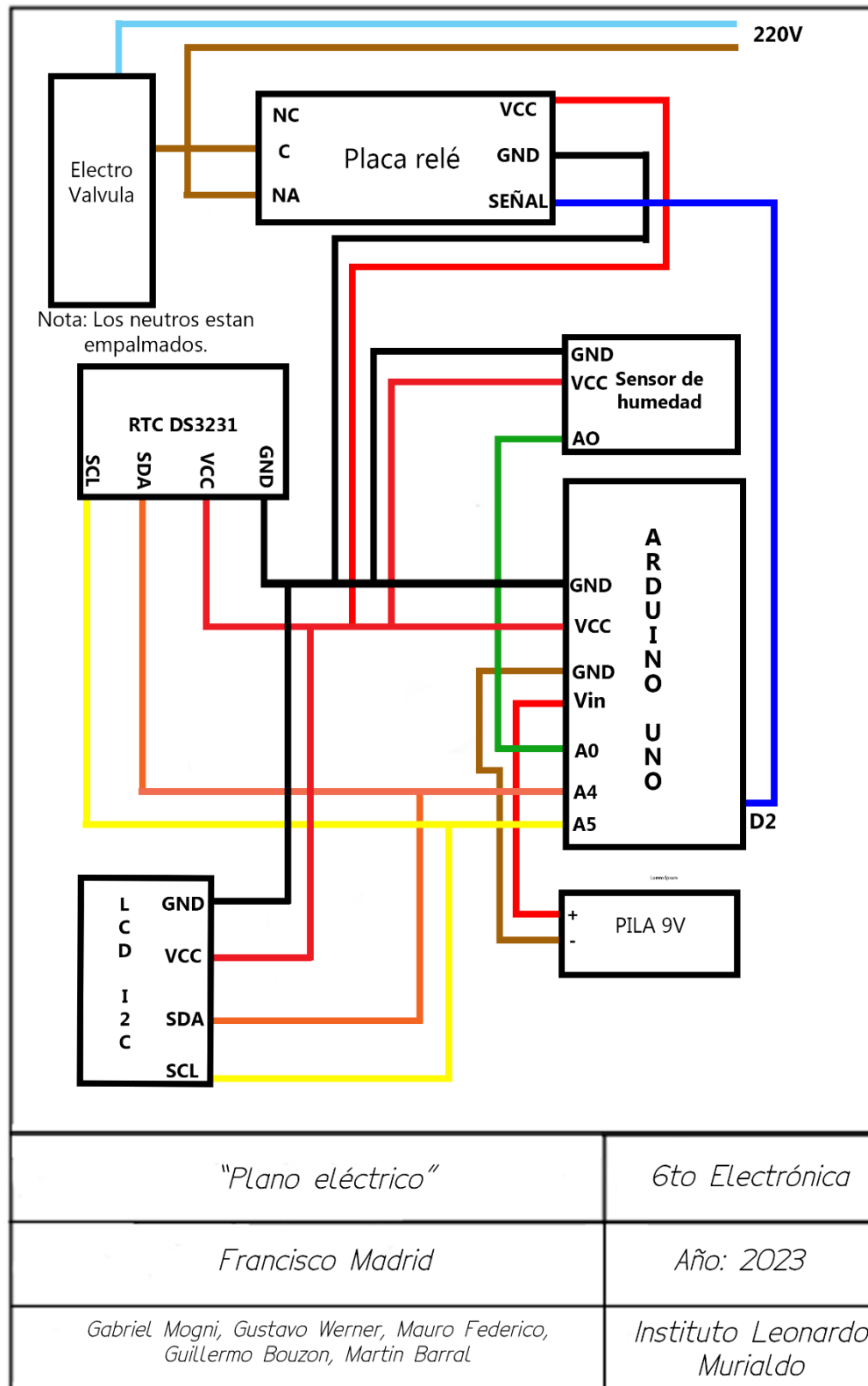
// EXTRAS
// Arreglo de falla que hacia que el porcentaje quede
impreso 2 veces en el LCD.
if(LECTURAP!=LECTURAQ+1 && LECTURAP!=LECTURAQ-1){
    if(LECTURAP<10){
        lcd.setCursor(10,1);
        lcd.print("% ");
    }
}
```



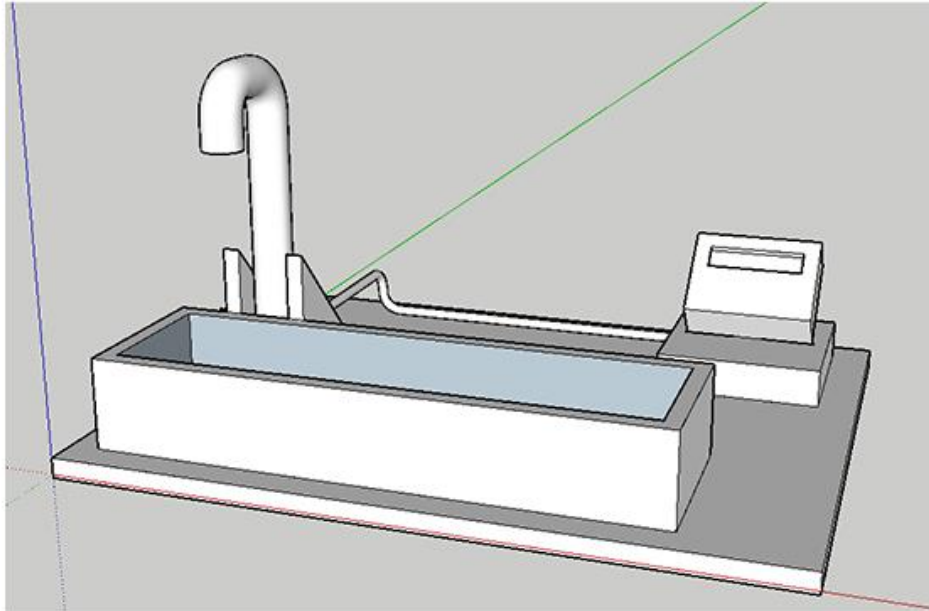
```
    else{  
    lcd.setCursor(11,1);  
    lcd.print("%");  
    }  
  
}  
                                     // parte de codigo adicional para hacer revisiones.  
LECTURAQ=LECTURAP;  
Serial.println(LECTURAP);  
Serial.println(HUMEDAD);  
delay(100);  
}
```

Anexos

Plano electrico



Plano de maqueta 3D



<i>"Diseño de maqueta"</i>	<i>6to Electrónica</i>
<i>Francisco Madrid</i>	<i>Año: 2023</i>
<i>Gabriel Mogni, Gustavo Werner, Mauro Federico, Guillermo Bouzon, Martin Barral</i>	<i>Instituto Leonardo Murialdo</i>