

Universidad del Valle de Guatemala  
Facultad de Ingeniería  
Departamento de Electrónica y Mecatrónica  
Electrónica Digital 02  
Segundo Semestre 2022

# **PROYECTO 1**

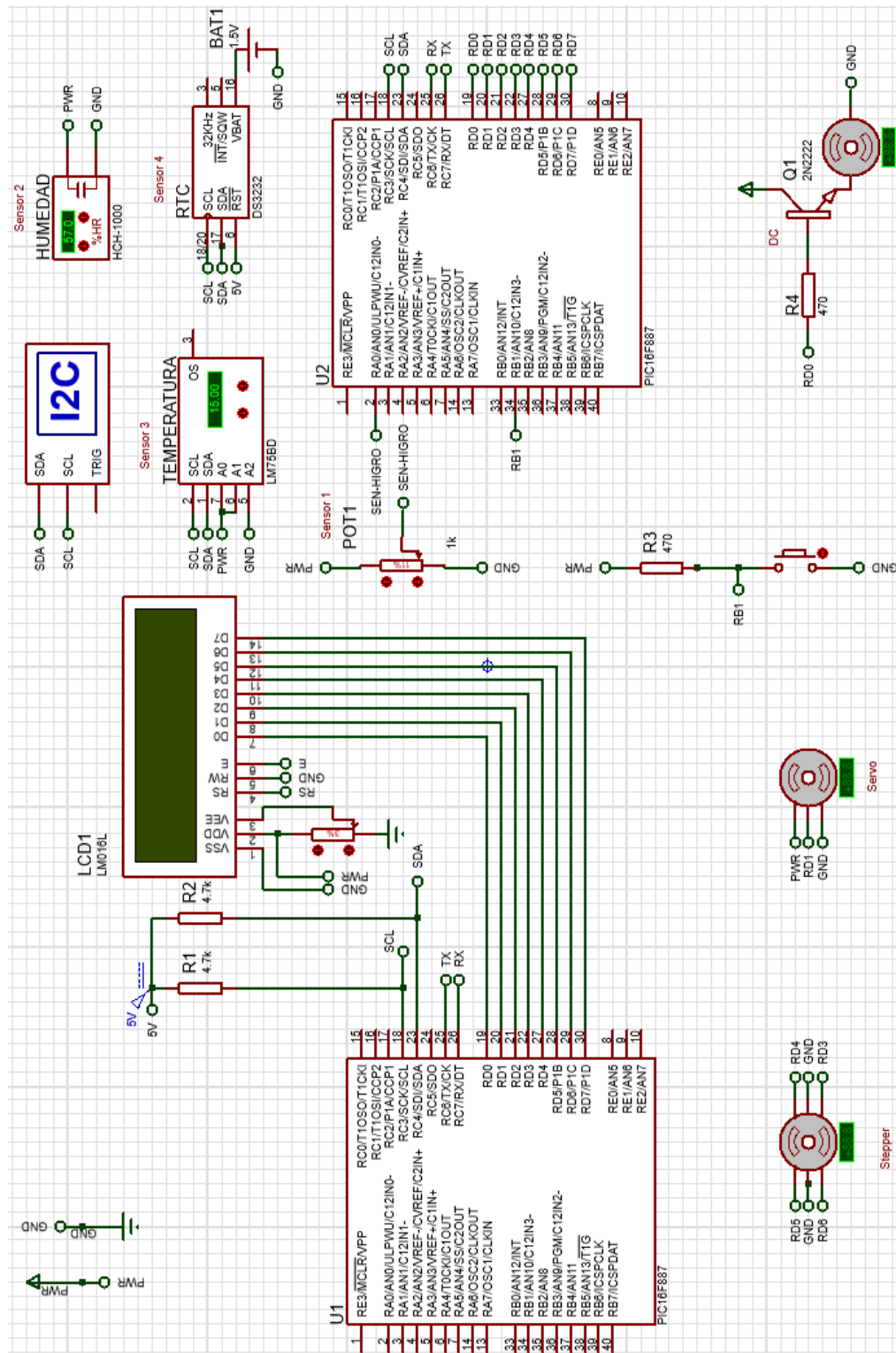
## **RED DE SENSORES**

Integrantes  
Sección 22 - IE3054  
Andrea Rodriguez Zea; 19429  
Paola Andrea Ayala Pineda; 19091

<b>Diagramas de conexión</b>	<b>4</b>
<b>Análisis y explicación de módulos</b>	<b>5</b>
Módulo de Humedad	5
Módulo FC-28	5
Sensor Electrodo YL-69	5
Tarjeta de acondicionamiento YL-38	5
PIC16F887	6
Slave	6
Máster	6
LCD	6
Módulo ULN-2003	6
Motor Stepper 28BYJ-48	7
Spray Cap	7
Módulo de Temperatura	7
Sensor de Temperatura LM75	7
PIC16F887	8
Slave	8
Máster	8
LCD	8
Módulo de transistorización	8
Transistor 2N3904	8
Motor DC	8
Botón	8
Hélice	8
Módulo RTC	9
Módulo RTC DS3231	9
PIC16F887	9

Slave	9
Máster	9
LCD	9
LED	9
Módulo de posicionamiento	10
Potenciómetro	10
ESP32	10
GUI	10
Motor Servo	10
Maceta	10
Módulo de lectura de información	10
<b>Explicación breve de cada sensor/módulo</b>	<b>10</b>
<b>Configuración de los microcontroladores y sensores - explicación</b>	<b>10</b>
<b>Partes importantes del código - copiar una parte del código y explicarla</b>	<b>10</b>
<b>Datasheets</b>	<b>10</b>

## I. Diagramas de conexión



## II. Análisis y explicación de módulos

### A. Módulo de Humedad

#### 1. Módulo FC-28

El módulo se encuentra compuesto de 2 partes, una para la medición de la humedad (YL-69) y una tarjeta de procesamiento de la información recibida por el sensor, que convierte los datos en manejables, las funciones de los mismos se detallan a continuación.

##### *a) Sensor Electrodo YL-69*

Especializado para poder medir la humedad del suelo al aplicar tensión entre sus terminales, de ésta forma hace pasar una corriente que depende de la resistencia que se genera en la tierra, dependiente de la humedad en la misma. Por lo tanto, al aumentar la humedad, la corriente crece y al bajar la corriente disminuye.

Debido a que la humedad presente en la tierra y los minerales en la misma podrían crear una alta cantidad de corrosión sobre los tracks del sensor, se decidió engrosar los mismos con estaño para aumentar la durabilidad de los mismos.

##### *b) Tarjeta de acondicionamiento YL-38*

La tarjeta YL-38 cuenta con 2 pines para conexión al módulo YL-69, 2 para alimentación (GND y VCC) y 2 pines para salidas de datos. El módulo puede ser utilizado de forma analógica para la medición de humedad (A0) o digital como salida indicadora de la superación de un umbral (D0) por medio de comunicación serial de tipo I2C.

Cuenta con un comparador LM393, un OpAmp que calcula los voltajes en dos terminales diferentes y contrasta la diferencia en la cantidad de voltaje para generar un valor.

## 2. PIC16F887

### *a) Slave*

Al momento de recibir a través de sus pines 18 y 23, correspondientes a la comunicación SCL y SDA respectivamente, si encuentra una señal debajo del rango deseado para la humedad a través del módulo FC-28, enviará una señal al módulo ULN-2003 para que éste active el motor stepper y se atomice con agua para ayudar a aumentar los niveles de humedad en la planta.

La lectura de la información del sensor será cada cierto tiempo debido a que la alimentación continua del sensor electrodo podría corroerlo, el intervalo de lectura seteado para el mismo es cada 30 minutos, es decir, con una lectura cada 60000 milisegundos.

Además, se tendrá la opción de movilizar al motor independientemente del temporizador de riego. Los movimientos son hacia adelante, atrás, derecha e izquierda.

### *b) Máster*

La información de humedad recopilada por el PIC esclavo será enviada como variable al maestro para que éste la pueda enviar al LCD conectado en su puerto D para obtener información.

## 3. LCD

Desplegará la hora, la fecha más el valor de la temperatura y humedad. Previo a estos datos se muestra una breve introducción del proyecto y los datos a desplegar serán datos tipo prueba.

## 4. Módulo ULN-2003

Debido a que el motor stepper consume mucha corriente, un microcontrolador no puede controlar el motor directamente. Requiere un controlador IC como para

controlar el motor, éste módulo ofrece una mayor ganancia de corriente y permite que la salida de bajo voltaje y baja corriente de un microcontrolador impulse un motor stepper de alta corriente.

El módulo cuenta con 4 LEDs que nos permiten visualizar los pulsos que envía al motor stepper, de esta forma nos deja saber la velocidad y dirección de giro del mismo. Se encuentra alimentado a través de los pines RD3 a RD6 del PIC esclavo.

#### 5. Motor Stepper 28BYJ-48

Los motores stepper utilizan un engranaje y cuatro electroimanes que forman un anillo alrededor del engranaje. Cada pulso alto enviado por el módulo ULN-2003 energiza la bobina, atrayendo los dientes más cercanos a la rueda dentada y girando el motor en incrementos de ángulo fijo y preciso conocidos como pasos. La secuencia de pulsos determina la dirección de giro del motor, mientras que la frecuencia modula la velocidad y la cantidad de pulsos el ángulo de giro del mismo.

#### 6. Spray Cap

El motor servo proporcionará un medio giro para hacer presión sobre el actuador de un spray y de esta forma poder dispensar agua a la planta.

### B. Módulo de Temperatura

#### 1. Sensor de Temperatura LM75

Éste sensor de temperatura trabaja a alta velocidad por medio del protocolo I2C, y puede convertir la temperatura en una señal digital directamente en el rango de temperatura de  $-55^{\circ}\text{C}$  a  $+125^{\circ}\text{C}$ . El ADC de 11 bits proporciona una resolución de temperatura de  $0,125^{\circ}\text{C}$ .

En su parte inferior cuenta con tres pines de dirección seleccionables, lo que permite conectar ocho dispositivos al mismo tiempo sin conflicto de direcciones en el mismo bus, sin embargo únicamente se seleccionó una.

Se localizó en la parte exterior de la carcasa del proyecto debido a que se buscaba que las lecturas que diera no fuesen sujetas directamente al entorno debajo de la planta, sino el exterior al que la misma estaba expuesta, de esta forma el calentamiento que los sensores, pantalla, humedad y demás pudiesen crear dentro de la carcasa no afectaría el valor de las lecturas del mismo.

## 2. PIC16F887

### *a) Slave*

Recibe a través de sus pines 18 y 23, correspondientes a la comunicación SCL y SDA respectivamente, si detecta una temperatura mayor a los 32°C enviará una señal automática al pic maestro para que éste comience a enviar señal al motor DC y éste comience a ventilar la planta junto a su hélice.

### *b) Máster*

Leerá continuamente el valor de temperatura almacenado en el PIC esclavo recopilado por el sensor para poder desplegarlo en el LCD.

## 3. LCD

Desplegará el valor de la temperatura continuamente, únicamente despliega datos de los números enteros, debido a variaciones de temperatura, el valor desplegado se aproximará según valores.



#### 4. Módulo de transistorización

##### a) *Transistor 2N3904*

Mediante el uso de un transistor se regula la velocidad a la que el motor DC gira agregando así también un botón con un tipo de configuración pull down para que el mismo gire únicamente en el momento que el usuario desee o en caso el valor de humedad y temperatura sean máximos.

##### b) *Motor DC*

El motor opta por una pequeña hélice que permite la generación de aire hacia la planta con el fin de brindarle frescura.

#### 5. Botón

Al ser presionado, iniciará o detendrá la señal proporcionada por el pin el el PIC esclavo para detener o activar el motor DC y consecuentemente el flujo de aire a la planta.

#### 6. Hélice

Con una conexión física al motor DC, cada vez que éste gire, las aspas de la misma generarán un flujo de aire directo a la planta para poder disminuir la temperatura a su alrededor.

### C. **Módulo RTC**

#### 1. Módulo RTC DS3231

Utilizado como reloj y alarma, busca mostrar la hora y la fecha. Por medio de la configuración de una de sus alarmas, comienza un conteo del tiempo de regado hasta que dicha alarma sea apagada.

#### 2. PIC16F887

##### a) *Slave*

Recopila la información en todo momento, el retraso de lectura del mismo es mínimo debido a que se necesita el tiempo real. Se encuentra conectado a los buses SDA y SCL.

El momento en el que el botón para detener el conteo de la alarma será enviado al máster.

*b) Máster*

Recupera la información de lectura del tiempo del slave para mostrarla en el LCD conectado a su puerto D.

Por medio de comunicación SPI con el ESP32, enviará el dato de la hora y fecha a Adafruit.

3. LCD

Despliega las horas y minutos leídos en el RTC.

4. LED

Con varios LEDs se mostrará el transcurso del tiempo desde que se debía regar, cada LED constituirá un plazo de 15 minutos desde que se activó la alarma de riego, se desactiva con un botón que resetea el contador de alarma.

5. Botón

Generará una interrupción el conteo del RTC para que comience el contador de alarma nuevamente.

**D. Módulo de posicionamiento**

1. Potenciómetro

Por medio de la lectura de ángulo de un potenciómetro, se envía la señal al PIC esclavo sobre la angulación a la que se encuentra, y por ende, será el ángulo que la posición del motor servo deberá replicar.

2. Motor Servo

Se colocará en la posición deseada analógicamente por el potenciómetro, o bien, la posición del mismo podrá ser controlada desde una interfaz gráfica remota.

3. Maceta

Conectada al servomotor, se podrá cambiar la posición en la que se encuentra para que a lo largo del día la planta pueda ser rotada y de esta forma poder conseguir luz del sol pareja.

## E. Módulo de lectura de información

### 1. ESP32

A través del PIC maestro y el ESP32 se enviarán los datos de hora, alarma y posicionamiento para visualizarlas en la interfaz de Adafruit por medio de comunicación SPI.

### 2. Adafruit

Despliega la hora guardada en el LCD y muestra los botones de alarma y posicionamiento siendo así que los cambios que se generen dentro de la interfaz serán reflejados en el circuito físico con por medio de leds.

## III. Detalles relevantes del código

- a) El siguiente código describe detalladamente cómo funciona la comunicación I2C que existe entre el microcontrolador maestro y esclavo.

```
I2C_Start(); // Llamamos a la función Start.
```

```
I2C_Write(0xD0); // Escribimos en SSPBUF la dirección de DS1307 1101000 +  
0 de escritura. I2C_Write(0); // Dirección de segundos.
```

```
I2C_ReStart(); // Llamamos a la función ReStart.
```

```
I2C_Write(0xD1); // Escribimos en SSPBUF la dirección de DS1307 1101000 +1  
de lectura. seconds=I2C_Read(); // Cargamos la variable "seconds" con el valor  
de SSPBUF.
```

```
I2C_Ack(); // ACK.
```

```
minutes=I2C_Read(); // Cargamos la variable "minutes" con el valor de SSPBUF.
```

```
I2C_Ack(); // ACK.
```

```
hours=I2C_Read(); // Cargamos la variable "hours" con el valor de SSPBUF.
```

```
I2C_NO_Ack(); // NO ACK.
```

```
I2C_Stop(); // Llamamos a la función Stop.
```

```
__delay_ms(50); // Retardo de 50 ms. //
```

```
I2C_Start(); // Llamamos a la función Start.
```

```
I2C_Write(0x90); // Escribimos en SSPBUF la dirección del sensor de
temperatura + 0 de escritura.
```

```
I2C_Write(0); // Dirección de temperatura.
```

```
I2C_ReStart(); // Llamamos a la función ReStart.
```

```
I2C_Write(0x91); // Escribimos en SSPBUF la dirección del sensor de
temperatura +1 de lectura. temp=I2C_Read(); // Cargamos la variable
"temperatura" con el valor de SSPBUF. I2C_NO_Ack(); // NO ACK.
```

```
I2C_Stop(); // Llamamos a la función Stop.
```

```
__delay_ms(50);
```

- b) El código de abajo muestra la función de conversión de BCD a decimal y viceversa necesaria para los datos a desplegar en el LCD.

```
uint8_t BCD_a_Decimal (uint8_t numero) // Función que convierte un número
BCD a decimal.
```

```
{
```

```
return ((numero >> 4) * 10 + (numero & 0x0F)); // Retornamos la variable
"numero" desplazado 4 posiciones a la izquierda, multiplicado por 10 más
"numero" && 1111.
```

```
}
```

```
uint8_t Decimal_a_BCD (uint8_t numero) // Función que convierte un número
decimal a BCD.
```

```
{
```

```
return (((numero / 10) << 4) + (numero % 10)); // Retornamos la decena de la
variable "numero" desplazado 4 bits a la derecha y las unidades de la variable
"numero" en el nibble menos significativo
```

```
}
```

#### IV. Datasheets

A. ESP32 - <https://bit.ly/3LYhOrO>

B. LCD 16x2 - <https://bit.ly/3dZvdU1>

- C. LED de 2 patas - <https://bit.ly/3fCwwso>
- D. Motor DC Mini - <https://bit.ly/2lZjgT4>
- E. Módulo ULN2003 - <https://bit.ly/3e248j8>
- F. Transistor LN3904 - <https://bit.ly/3ycn3yG>
- G. Motor Servo SG90 - <https://bit.ly/3C1fhsy>
- H. Motor Servo SG90 - <https://bit.ly/3C1fhsy>
- I. Trimpot (Trimmer) - <https://bit.ly/3dXluh0>
- J. Módulo RTC DS3231 - <https://bit.ly/3dZwt9H>
- K. Potenciómetro rotatorio - <https://bit.ly/3rqHdBe>
- L. PIC16F887 - 2 utilizados - <https://bit.ly/3EbUVj0>
- M. Motor Stepper 28BYJ-48 - <https://bit.ly/3fEwpNj>
- N. Sensor de Humedad FC-28 - <https://bit.ly/3RujPNJ>
- O. Sensor de Temperatura LM75 - <https://bit.ly/3C2rqNV>