

Proyecto Final

Sistema de comunicación para monitoreo de temperatura y humedad

Luis Fernando Granados Zambrano
lf.granadoszambrano@ugto.mx, NUA: 280509

Prof. Mtro David Alejandro Balcazar Torres

Informática Industrial

*División de Ingenierías Campus Irapuato-Salamanca
Universidad de Guanajuato*



Índice

1. Resumen	3
2. Objetivo	3
3. Introducción	3
4. Desarrollo	4
4.1. Materiales utilizados	4
4.2. Procedimiento	4
4.3. Transmisor	5
4.3.1. Código de programación	5
4.3.2. Implementación hardware	6
4.4. Receptor	7
4.4.1. Código de programación	7
4.4.2. Implementación hardware	8
5. Resultados	9
5.1. Resultado interfaz PCB Transmisor	10
5.2. Resultado interfaz PCB Receptor	10
5.3. Evidencias	10
6. Conclusión	11
7. Referencias	11

1. Resumen

En el presente informe se describe el procedimiento para construir un sistema de comunicación en donde se implementan los elementos de hardware y software para realizar el sensado de temperatura y humedad mediante dos placas de arduino UNO como transmisor y receptor. El funcionamiento de este sistema esta dado por un código programable donde se establecen las acciones de cada componente electrónico y el tipo de modulo a utilizar en arduino.

2. Objetivo

Implementar un sistema de comunicación por radiofrecuencia 2.4 GHz para monitoreo de temperatura y humedad de un lugar específico utilizando un transmisor que mediante un sensor DHT11 tome las mediciones y las mande al receptor a una distancia máxima de 1 kilómetro para mostrar en una pantalla LCD los valores medidos.

3. Introducción

El sistema de comunicación es un modelo de sistema que describe los intercambios de comunicación entre dos estaciones, transmisor y receptor. Las señales o información pasan de la fuente a la distensión a través de lo que se llama canal, que representa una forma en que la señal la usa para moverse desde la fuente hacia el destino. Para transmitir señales en el sistema de comunicación, primero debe ser procesado por varias etapas, comenzando desde la representación de la señal, hasta la conformación de la señal hasta la codificación y modulación. Después de preparar la señal transmitida, pasó a la línea de transmisión del canal y debido a que la señal que atraviesa este medio se enfrenta a muchas deficiencias tales como ruido, atenuación y distorsión. [5]

4. Desarrollo

4.1. Materiales utilizados

- 2 Arduino UNO
- 2 Modulos arduino nRF24L01
- Pantalla LCD 16x2
- Sensor DHT11
- 2 Protoboard
- Cables Jumper
- Software Arduino IDE

4.2. Procedimiento

Se construyó un sistema de comunicación utilizando Arduino UNO como el que se muestra en la figura 1, que puede monitorear la temperatura y humedad de un medio específico y enviar los datos medidos a través de un módulo de antena de 2.4 Ghz a una distancia máxima de 1 kilómetro para mostrar los datos en un receptor mediante una pantalla LCD.



Figura 1: Esquema del proyecto

Para llevar a cabo la realización del sistema se construyó de manera independiente un circuito transmisor y receptor, en donde cada uno tiene un código programable específico y en conjunto establecen una comunicación funcionando finalmente como un sistema.

4.3. Transmisor

4.3.1. Código de programación

```
#include <SPI.h> //Librería de protocolo de comunicación serial
#include <nRF24L01.h> //Librería correspondiente al módulo de antena
#include <RF24.h> //Librería correspondiente al módulo de antena
#include <DHT11.h> //Librería de módulo de sensor de temperatura y humedad

int pin = A0; //Declaración de pin analógico A0
DHT11 dht11(pin); //Definimos el pin digital donde se conecta el sensor
float temperature[2]; //Declaración de variable flotante para temperatura

double Fahrenheit(double celsius) { //En este comando convertimos los grados Fahrenheit a Celsius
    return ((double)(9 / 5) * celsius) + 32;
}

double Kelvin(double celsius) { //Mostrar por default grados Celsius
    return celsius + 273.15;
}

RF24 radio(8, 9); //Pin del Arduino conectado al pin Chip Enable (CE) y Chip Select (CS) del módulo
const uint64_t pipe = 0xE8E8F0F0E1LL; //Dirección de canal de radio única para que los 2 nodos se comuniquen
float temp, humi; //Declaración de constantes flotantes para temperatura y humedad

void setup() {
    Serial.begin(9600); //Inicializamos el puerto serie
    radio.begin(); //Inicializamos el NRF24L01
    radio.openWritingPipe(pipe); //Abrir un canal de escritura
}

void loop(void)
{
    dht11.read(humi, temp); //Leer las variables de humedad y temperatura del sensor
    temperature[0] = temp; //Asignar valor a temp
    temperature[1] = humi; //Asignar valor a humi
    radio.write(temperature, sizeof(temperature)); //Envío de datos
    delay (1000); //Retardo para cada medición
}
```

Figura 2: Código del transmisor documentado

4.3.2. Implementación hardware

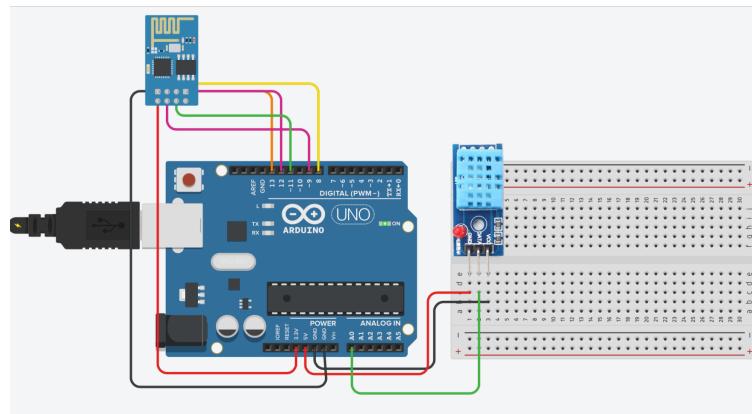


Figura 3: Esquemático circuito Transmisor

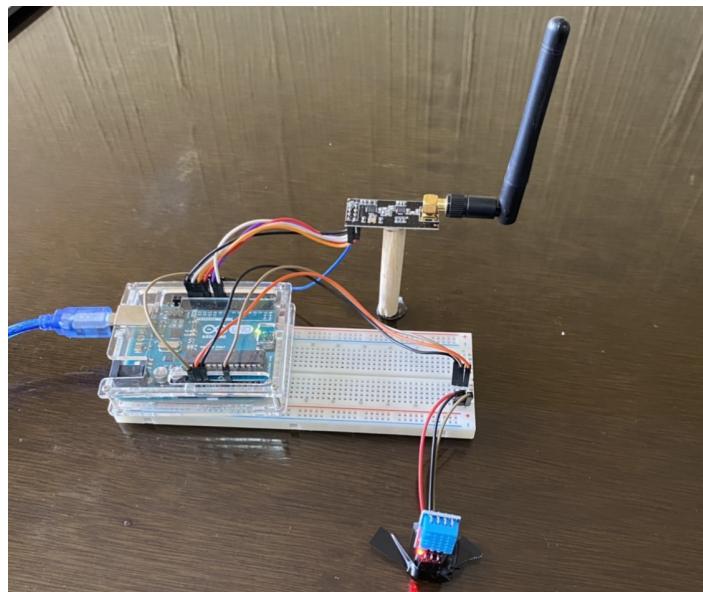


Figura 4: Transmisor

4.4. Receptor

4.4.1. Código de programación

```
#include <SPI.h> //Librería de protocolo de comunicación serial
#include <RF24.h> //Librería correspondiente al módulo nRF24L01
#include <DHT11.h> //Librería de módulo de sensor de temperatura y humedad
#include <LiquidCrystal.h> //Librería correspondiente a pantalla LCD
#include <Wire.h> //Librería establece protocolo I2C/TWI

//Inicializar la librería indicando los pins de la interfaz
LiquidCrystal lcd(7, 6, 5, 4, 3, 2); //RS, RW, E, D4,D5, D6, D7

float temperature[2];
/*Crear un objeto RF24 al que llamaremos "radio". Los dos argumentos que se utilizan
en la función son los numeros de pin que vamos a usar para CS y CE */

RF24 radio(8, 9); //Pin del Arduino conectado al pin Chip Enable (CE) y Chip Select (CS) del módulo
const uint64_t pipe = 0xE8E8F0F0E1LL; // Dirección de canal de radio única para que los 2 nodos se comuniquen

void setup()
{
    Serial.begin(9600); //inicializamos el puerto serie
    lcd.begin(16, 2); // Inicia LCD 16x02 (columnas, filas)

    radio.begin(); //inicializamos el NRF24L01
    radio.openReadingPipe(1, pipe); //Abrimos el canal de Lectura
    radio.startListening(); // Este comando establece este modulo como el receptor

    lcd.print("Proyecto Final"); //Muestra mensaje
    delay(3000); //Retardo 3 s
    lcd.clear(); //Limpia pantalla
    lcd.setCursor(0, 0); //Cursor en posición 0,0
    lcd.print("Informática");
    lcd.setCursor(0, 1);
    lcd.print("Industrial");
    delay(3000);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Temp & Humedad");
    delay(2500);

    lcd.clear();
    lcd.print("Iniciando...");
    delay(2000);
}

void loop()
{
    if (radio.available()) //Si la señal transmisora esta disponible
    {
        bool done = false;
        while (!done)
        {
            done = radio.read(temperature, sizeof(temperature)); //Leemos los datos y guardamos en la variable temperature[]
            lcd.clear(); //limpia LCD
            delay(500);

            lcd.setCursor(0, 0); //Posición
            lcd.print("Temp"); //Imprime mensaje
            lcd.setCursor(0, 1); //Posición
            lcd.print("Humedad"); //Imprime mensaje
            lcd.setCursor(9, 0); //Posición
            lcd.print(temperature[0]); //Imprime valor de variable
            lcd.print(" C"); //Imprime símbolo 'C'
            lcd.setCursor(9, 1); //Posición
            lcd.print(temperature[1]); //Imprime valor de variable
            lcd.print(" %"); //Imprime símbolo '%'
            delay(1000);
        }
    }
    else
    {
        lcd.setCursor(0, 0); //Posición
        lcd.print("Sin señal"); //Imprime mensaje que no detecta señal
    }
}
```

Figura 5: Código del receptor documentado

4.4.2. Implementación hardware

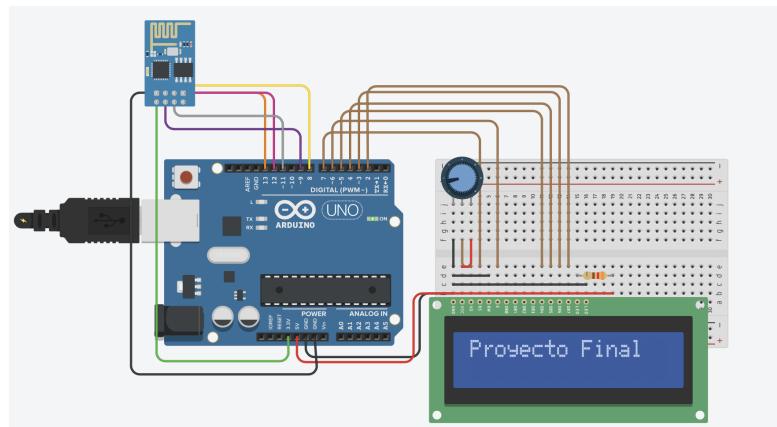


Figura 6: Esquemático circuito receptor



Figura 7: Receptor

5. Resultados

Se obtuvo un sistema de comunicación como el que se muestra en la Figura 8 con un transmisor y receptor independientes el uno del otro.



Figura 8: Sistema de comunicación para monitoreo de temperatura y humedad



Figura 9: Visualización de datos en pantalla LCD

5.1. Resultado interfaz PCB Transmisor

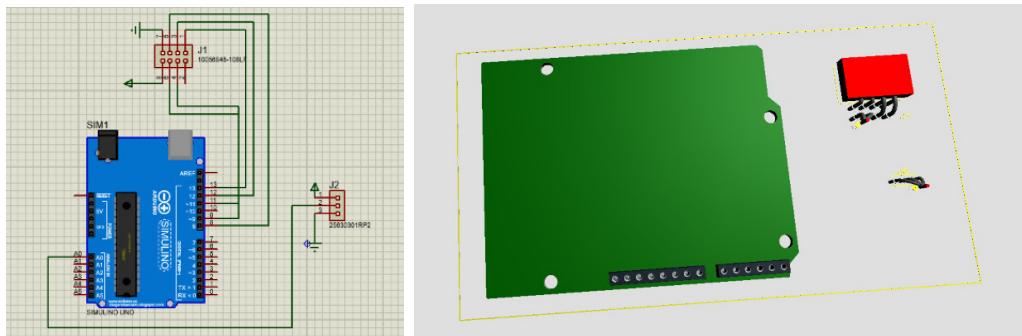


Figura 10: Vista pcb transmisor

5.2. Resultado interfaz PCB Receptor

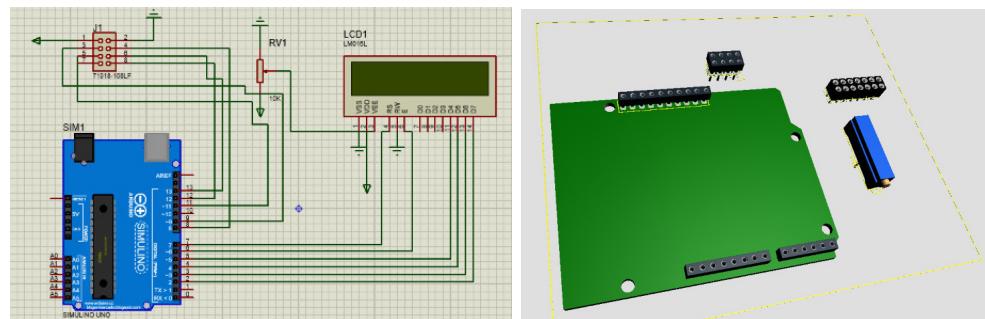


Figura 11: Vista pcb receptor

5.3. Evidencias

Link video de funcionamiento:

https://drive.google.com/file/d/1gscqF9vBqYtLdgLAsRWcyWD1CA8j_8v4/view?usp=sharing

Link repositorio GitHub:

https://github.com/LuisGrZam/Proyecto_blink

Link presentación: <https://view.genial.ly/60cac0e61056980d61137e02/presentation-informatica-industrial>

6. Conclusión

Durante la realización de este proyecto, se pusieron a prueba los conocimientos adquiridos en el curso de informática industrial debido a que se construyó e implementó un sistema de comunicación serial en el rango de microondas para el monitoreo de temperatura y humedad a partir de un código de programación que por medio de este se pueda implementar un hardware específico. Como resultado final, se obtuvo una comunicación exitosa entre el transmisor y el receptor, logrando así el objetivo de este proyecto.

7. Referencias

- [1] <https://www.arduino.cc/en/Reference/LiquidCrystal>
- [2] <https://www.youtube.com/watch?v=UZrZh9k0MsE&t=1403s>
- [3] <https://aprendiendoarduino.wordpress.com/2016/07/03/display-lcd/>
- [4] <https://www.youtube.com/watch?v=QU8tBCHjZy4&t=146s>
- [5] Haykin, S. S. (2000). Communication Systems (4 Sub ed.). John Wiley & Sons Inc.