

Arduino

DHT11 Sensor Temperatura y Humedad

LunikSoft

Fco. Javier Andrade

<http://www.luniksoft.info>

El autor no se responsabiliza del contenido. No asume ninguna responsabilidad, por cualquier inexactitud en la documentación, en los programas, en el cableado y diagramas, ni de los daños derivados de la utilización del material proporcionado.

INTRODUCCIÓN :

Ahora que ya hemos hecho los primeros contactos con el mundo Arduino y Lazarus, vamos a entrar en el mundo de obtener información de nuestro entorno para poder procesarla y actuar en consecuencia. Para ello comenzamos con el primer artículo de una serie de ellos donde emplearemos sensores para obtener información. En este caso comenzaremos con el DHT11, un sensor de temperatura y humedad, y por su puesto es bastante económico.

DHT11

Este sensor de Temperatura y Humedad es uno de los sensores mas económicos de este tipo. Hay que tener en cuenta que no es excesivamente preciso (%5 en humedad y 2% en temperatura) ni tolerante, pero si deseamos algo mas de precisión podemos ir a al hermano DHT22.

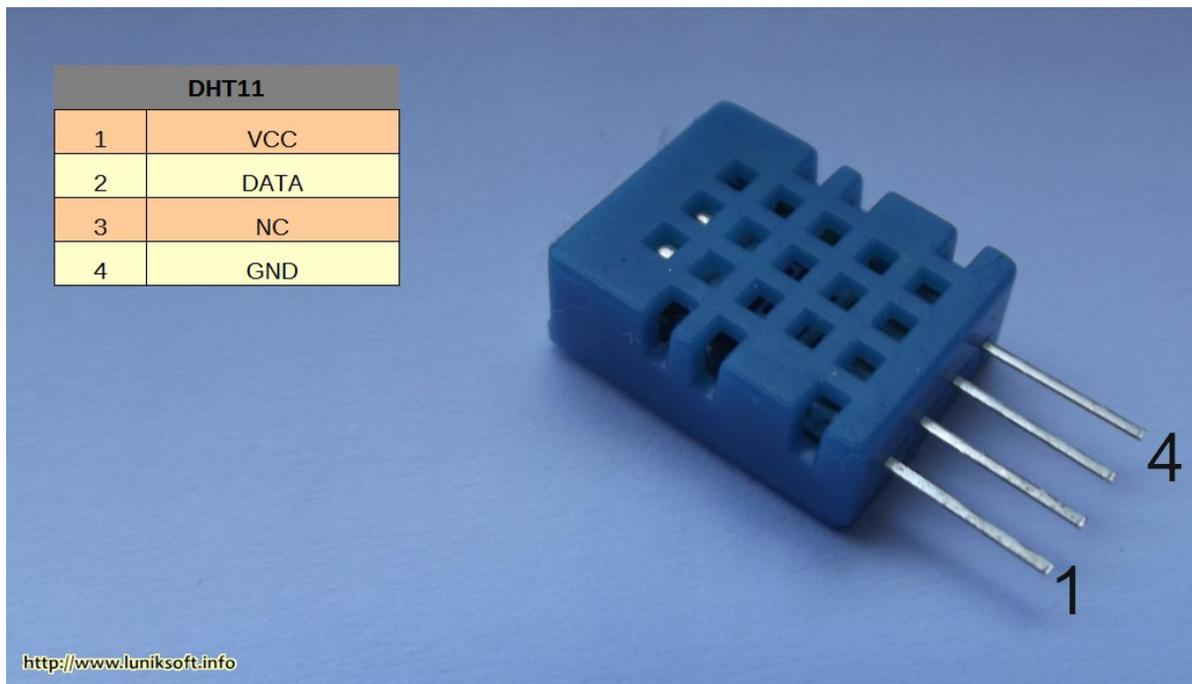
Las características principales del DHT11 son :

- Esta calibrado desde fabrica.
- Tensión 3.5v-5.5v DC
- Humedad Relativa entre 20%-95%. Margen error 5%.
- Temperatura entre 0 y 50 °C. Margen error 2%.
- Resolución de temperatura y humedad en rangos de 1 (no proporciona decimales).

En el mercado se encuentran a la venta en varias formas :

- Sueltas. Como el que aparece en la imagen. En cuyo caso hay que poner una resistencia pull-up (normalmente entre 5k y 10k)
- En una pequeña placa, que ya incorpora la resistencia. Y otra variante de esta que ademas de la resistencia incorpora un condensador para filtrado.

Para comunicarnos con él se emplea 1 hilo (Single-Wire Two-Way). Así este circuito unicamente es necesario emplear 3 conexiones (pines 1, 2 y 4) : dos de alimentación y una para la comunicación.



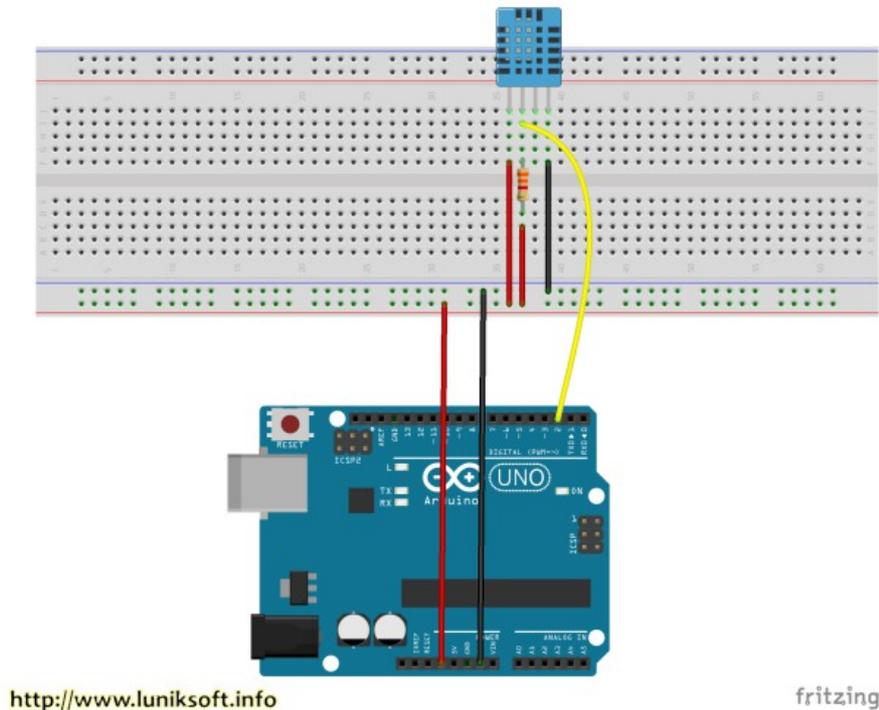
No entra dentro de este artículo entrar en los detalles del protocolo de la comunicación, si se desea profundizar en ella se puede consultar el datasheet o simplemente ver alguno de los muchos artículos que hay en la red sobre el tema.

COMPONENTES

- 1 x Arduino Uno (o compatible)
- 1 x DHT11 (sensor temperatura y humedad)
- 1 x Resistencia 10k o 5K.
- 1 x Placa de pruebas (recomendado)
- 1 x Juego Cables

CIRCUITO

El circuito es sencillo simplemente tenemos que alimentar el componentes y poner el pin de datos una resistencia pull-up. Aconsejan una resistencia entre 5k y 10k. En mi caso el sensor lo alimente con 3v3 le puse una de 3k3 y me funciono perfecto. Si se conecta 5v poner 10k



CODIGO FUENTE (SKETCH)

Para emplear este componente vamos a emplear una las librerías que hay en internet.

Lo primero que vamos a hacer es leer el valor del sensor y mostrarlo al PC mediante la comunicación serie. El código fuente se ha partido del publicado en [TallerArduinio](http://www.tallerardunio.com) (una muy página sobre Arduino)

```
#include <DHT.h>

// Liberia para emplear el sensor
#include "DHT.h"

// Pin donde esta conectado el pin DATA (Entradas digitales)
#define DHTPIN 2

// Descomentar el tipo de sensor empleado
#define DHTTYPE DHT11// DHT 11
// #define DHTTYPE DHT22//DHT22(AM2302)
// #define DHTTYPE DHT21//DHT21(AM2301)

// Inicliza una variable de tipo DHT para poder emplear el sensor
DHT dht(DHTPIN,DHTTYPE); //Indica el pin con el que trabajamos y el
tipo de sensor

int maxh=0, minh=100, maxt=0, mint=100; //Variables para ir
comprobando maximos y minimos
```

```
int value;

void setup()
{
    // Inicializaicón comunicacion serie para ver los datos en el
ordenador
    Serial.begin(9600);
    Serial.println("Comprobacion sensor DHTxx:");

    //Iniciamos el sensor
    dht.begin();
}

void loop()
{

    // Leemos temperatura y humedad. Es un sensor lento.
    int h = dht.readHumidity();
    int t = dht.readTemperature();

    // Comprobamos si lo que nos ha devuelto el sensor son numeros
correctos.
    if (isnan(t) || isnan(h))
    {
        Serial.println("Fallo al leer del sensor DHT"); //Mostramos
mensaje de fallo si no son numeros
    } else {

        //Mostramos mensaje con valores actuales de humedad y temperatura
        Serial.print("Humedad relativa: ");
        Serial.print(h);
        Serial.print(" %\t");
        Serial.print("Temperatura: ");
        Serial.print(t);
        Serial.println(" *C");

        // Comprobacion de maximos y minimos de humedad y temperatura
        if (maxh<h)
            maxh=h;

        if (h<minh)
            minh=h;

        if (maxt<t)
            maxt=t;

        if (t<mint)
            mint=t;

        // Mensajes de maximos y minimos
        Serial.print("Max: ");
        Serial.print(maxh);
        Serial.print(" % ");
        Serial.print("Min: ");
```

```
Serial.print(minh);  
Serial.print(" %\t");  
  
Serial.print("Max: ");  
Serial.print(maxt);  
Serial.print(" *C ");  
Serial.print("Min: ");  
Serial.print(mint);  
Serial.println(" *C\n");  
}  
  
// Pequeña pausa. 1sg  
delay(1000);  
}
```

Verificamos y subimos al Arduino. Una vez hecho esto podemos abrir el Monitor Serie para ver los datos.

