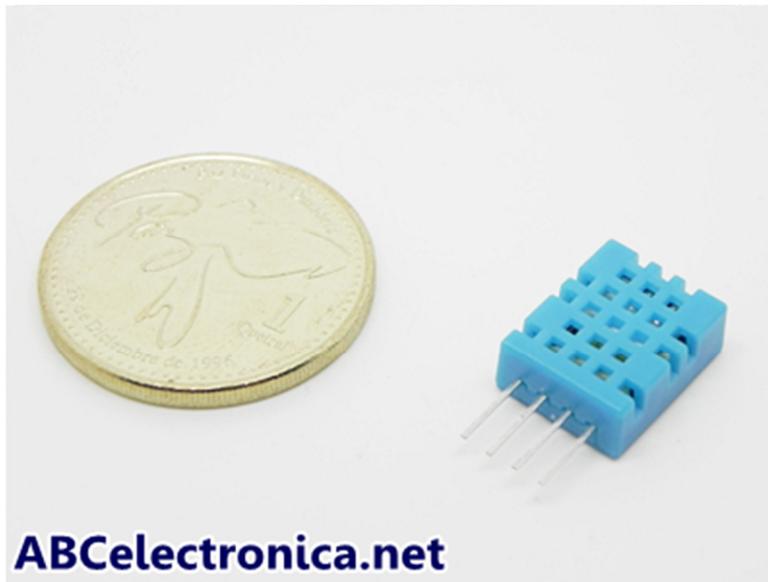


Sensor de humedad y temperatura

DHT 11 y DHT22

Los sensores DHT11 y DHT22 son los más básicos y los más utilizados para implementarlos con Arduino, estos sensores están compuestos en dos partes, un sensor de humedad capacitivo y un termistor, también constan de un circuito integrado básico en el interior que hace la conversión de analógico a digital y este envía una señal digital con la temperatura y la humedad.



DHT11



DHT22

DHT11 vs DHT22

Aunque lucen físicamente similares y tiene la misma identificación de pines, estos poseen características diferentes:

DHT11

- Alimentación de 3.3V a 5VDC
- Corriente máxima 2.5mA durante la conversión
- Lectura de humedad con un +/- 5% de precisión
- Lectura de temperatura con un +/- 2°C de precisión
- Capaz de medir humedad de 20% a 80%
- Capaz de medir temperatura de 0 a 50°C
- No más de 1 Hz en velocidad de muestreo (una vez cada segundo)
- Dimensiones: 15.5mm x 12mm x 5.5mm

DHT 22

- Alimentación de 3.3V a 5VDC
- Corriente máxima 2.5mA durante la conversión
- Lectura de humedad con un +/- 2% a 5% de precisión
- Lectura de temperatura con un +/- 0.5°C de precisión
- Capaz de medir humedad de 0% a 100%
- Capaz de medir temperatura de -40°C a 125°C
- No más de 0.5Hz en velocidad de muestreo (una vez cada dos segundos)
- Dimensiones: 15.1mm x 25mm x 7.7mm

De acuerdo a las características es un sensor ligeramente más preciso y cuenta con un rango un poco mayor. Ambos utilizan un único pin digital de salida y son lentos en cierto aspecto, ya que no se puede consultar más de una vez cada uno o dos segundos.

Conectando un sensor DHT

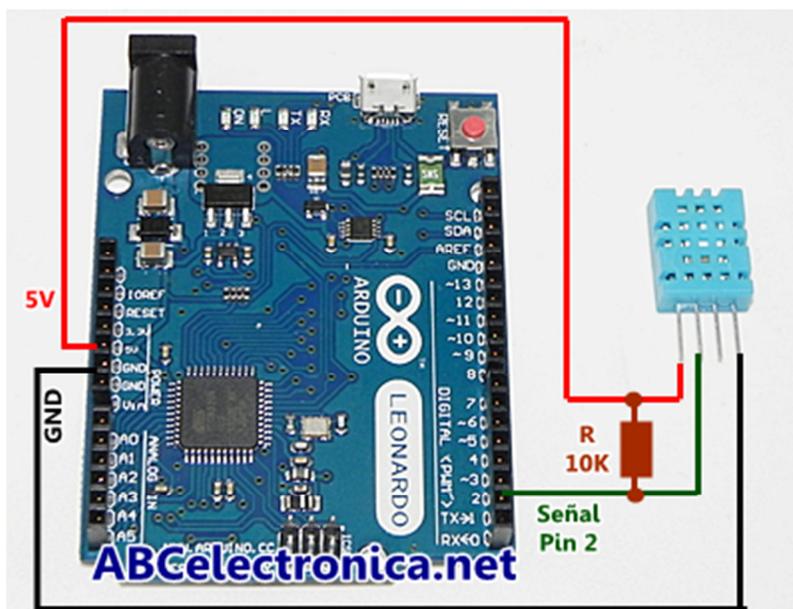
Los sensores DHTxxx cuentan con cuatros pines:

- VCC (3.3VDC A 5VDC) (pin 1)
- Salida de datos
- No se conecta (pin 3)
- GND



El pin 3 no se utiliza, y se debe de colocar una resistencia de 10K entre VCC y pin salida de datos, de esta manera se coloca en *pull up* y la lectura se hace sin problemas.

La siguiente imagen muestra la manera de conectar un DHT11 o un DHT22 a un Arduino UNO o LEONARDO (también un MEGA o NANO)



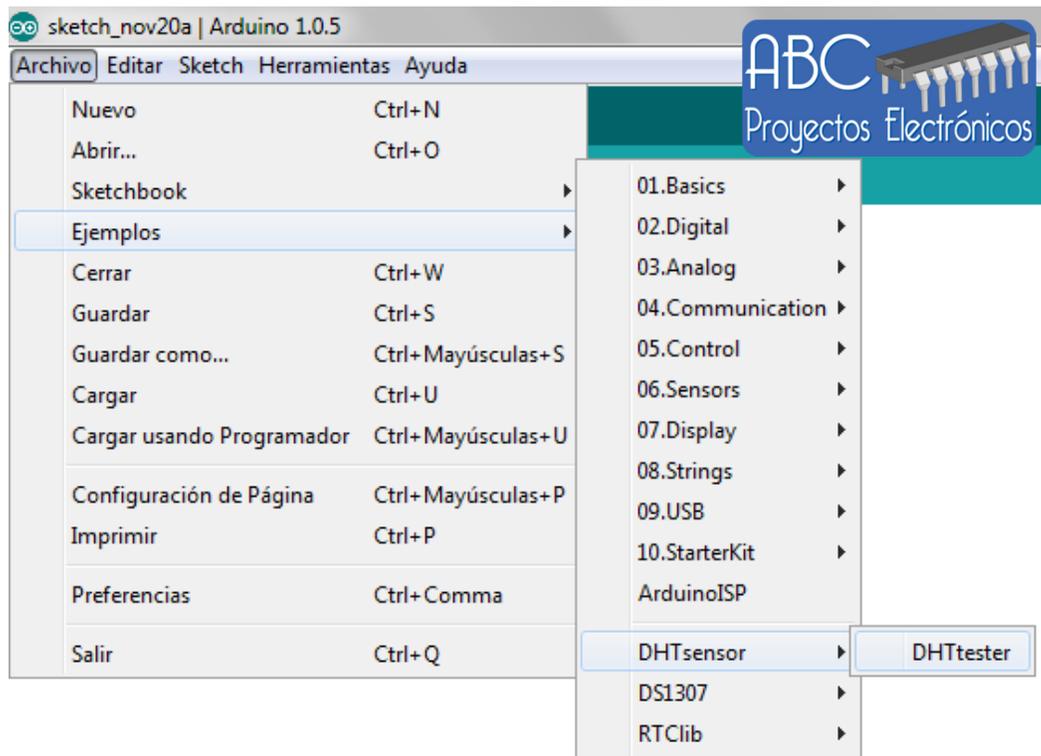
Programando

Primero que nada se deberá de instalar la librería para este sensor

<https://github.com/adafruit/DHT-sensor-library>

Para instalar librería, ver siguiente [tutorial](#)

Una vez instalada la librería, clic en **Archivo** > **Ejemplos** > **DHTsensor** > **DHTtester**



El código de programación será el siguiente:

```
// Ejemplo para lectura de humedad y temperatura
// de los sensores DHTxxx
// Escrito por ladyada,

#include "DHT.h"

#define DHTPIN 2      // pin de salida

// quitar barra de comentario dependiendo del
// tipo de sensor que usara
// -- en este caso se utilizo un DHT11--
#define DHTTYPE DHT11 // DHT 11
// #define DHTTYPE DHT22 // DHT 22 (AM2302)
// #define DHTTYPE DHT21 // DHT 21 (AM2301)

// Conectar pin1 del sensor a +5V
// Conectar pin 2 al pin de arduino
// Conectar pin 4 GND
// Conectar una resistencia de 10K del pin 2 al pin 1 del sensor

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println("DHTxx test!");

  dht.begin();
}

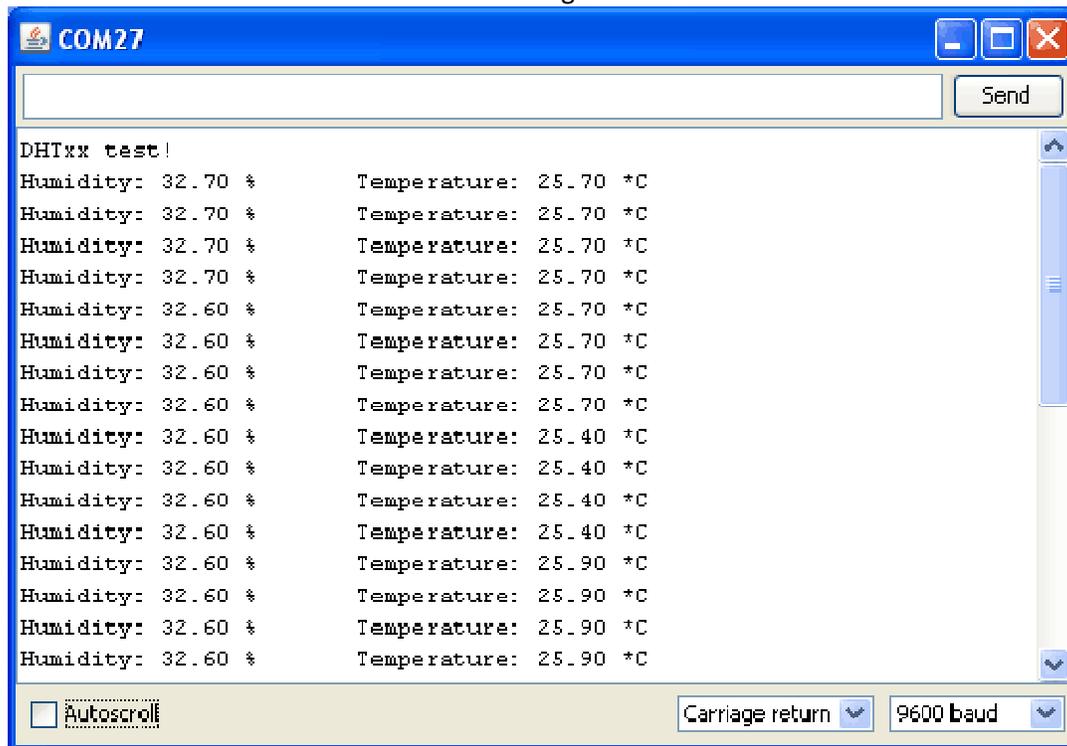
void loop() {
  // Leyendo temperatura y humedad que toma 250ms
  // el sensor puede leer hasta dos segundos despues de la lectura
  // lo que se considera "lento"
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  // revisa si retorna un valor valido, de lo contrario hay un error
  if (isnan(t) || isnan(h)) {
    Serial.println("Failed to read from DHT");
  } else {
    Serial.print("Humidity: ");
    Serial.print(h);
    Serial.print(" %\t");
    Serial.print("Temperature: ");
    Serial.print(t);
    Serial.println(" *C");
  }
}
```

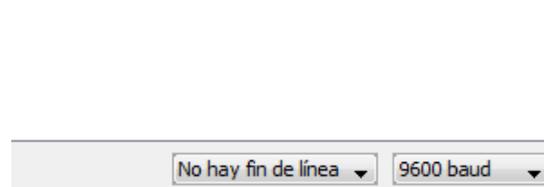
Una vez cargado el programa al Arduino el monitor serial (Herramientas > monitor serial) deberá de mostrar la humedad y temperatura:

```
DHTxx test!  
Humidity: 73.00 %      Temperature: 19.00 *C  
Humidity: 73.00 %      Temperature: 19.00 *C
```

o algo como:



Tomar en cuenta que los baudios del monitor serial deben ser iguales al de código programación ya que de lo contrario aparecerán símbolos de modo aleatorio.



Sobre referencia de la información y para conocer más sobre el código: [Adafruit](#)