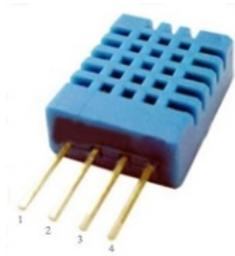


El sensor DHT11, mide la temperatura (t) y humedad relativa (hr). Los rangos de medida que maneja el sensor son; de 20-90% para el hr, con una precisión de  $\pm 5\%$  y 0-50 oC para la t.

El tiempo de respuesta puede variar en relación con la temperatura y humedad ambiental, sin embargo se puede establecer en un rango de 6 a 30 segundos. En relación con la calidad del producto destaca que cada uno de los sensores está calibrado, así también, la hoja de datos técnicos, lo refiere como estable, preciso y con una salida de señal digital.

Es recomendable usar un resistor de 5k a 10k entre los pines 1 y 2.



| Pin | Nombre | Descripción                         |
|-----|--------|-------------------------------------|
| 1   | VDD    | Alimentación de energía 3 – 5.5V DC |
| 2   | DATA   | Datos de salida                     |
| 3   | NC     | Sin conexión                        |
| 4   | GND    | Conexión de tierra                  |

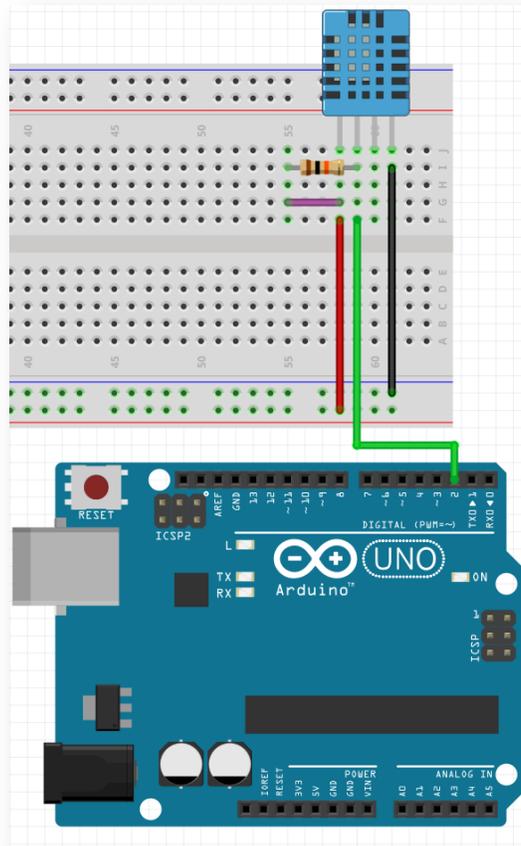
El sensor tiene la capacidad de tomar muestras en forma instantánea (dependiendo las condiciones ambientales), la temperatura se puede manejar en diferentes escalas (Celsius, Fahrenheit, Kelvin, Rankine, Newton, Réaumur y Romer), la humedad relativa la mide en porcentaje.

### *Indicaciones para el correcto funcionamiento del sensor*

- Los vapores de materiales químicos puede interferir con la sensibilidad del dispositivo DHL hasta el punto de dañarlo en forma permanente.
- Si el sensor presenta respuestas irregulares o estuvo expuesto a vapores, se recomienda, guardar el sensor en un lugar donde la temperatura este 50-60 Celsius y una humedad relativa de menos 10% por 2 horas, si el problema persiste se puede intentar nuevamente en estas condiciones; temperatura 20-30 Celsius y humedad mayor a 70 % for 5 hrs.
- Debido a la alta sensibilidad del dispositivo es recomendable tomar la lectura en un espacio en el que no interfiera calor generado por otros factores, incluso el mantener en la mano en la superficie de lectura del sensor puede alterar la lectura.
- La exposición prolongada a los rayos solares y ultravioletas puede arrojar lecturas erróneas.
- La calidad del cable que se utilice para conectar el sensor así como la distancia a la fuente de energía pueden afectar la lectura.
- Para soldar este dispositivo se recomienda que sea con una exposición a temperatura por debajo de los 260 oC y el contacto sea menor a 10 segundos.

- Evitar usar el sensor bajo condiciones de rocío.
- Se recomienda almacenar el dispositivo en condiciones de temperatura de 10-40 oC y humedad menor a 60%.

### Conexión Arduino



El sensor tiene 4 pines, el pin 2 es a través del cual vamos a obtener los datos que registre el dispositivo. Es conveniente utilizar la **librería** del sensor que se obtiene de manera gratuita (<https://github.com/adafruit/DHT-sensor-library>).

Los archivos de librería en Arduino, condensan una colección de tareas relacionadas al dispositivo y generalmente tienen la extensión .h como ocurre en este sensor donde reconocemos el archivo "DHT.h", a través del cual podemos utilizar los comandos:

- dht.readTemperature
- dht.convertCtoF
- dht.readHumidity

Inicialmente es recomendable ejecutar con los archivos que tiene de ejemplo para comprobar que el sensor funciona correctamente.

*Código Tester.ino*

```
// Example testing sketch for various DHT humidity/temperature
sensors
// Written by ladyada, public domain

#include "DHT.h"

#define DHTPIN 2      // what pin we're connected to

// Uncomment whatever type you're using!
//#define DHTTYPE DHT11   // DHT 11
#define DHTTYPE DHT22   // DHT 22  (AM2302)
//#define DHTTYPE DHT21   // DHT 21  (AM2301)

// Connect pin 1 (on the left) of the sensor to +5V
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 4 (on the right) of the sensor to GROUND
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of
the sensor

DHT dht(DHTPIN, DHTTYPE);

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

  dht.begin();
}

void loop() {
  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very
slow sensor)
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  // check if returns are valid, if they are NaN (not a number)
then something went wrong!
  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");
  }
}
```

Es importante leer en forma detallada el código, porque de esta manera es posible identificar algunas recomendaciones del desarrollador de la rutina y a la vez modificarlo en relación a las necesidades personales. Los datos que muestra el monitor serial son de humedad relativa y temperatura. Durante esta etapa se debe modificar la rutina y corroborar que los datos que registra son los que se requieren.

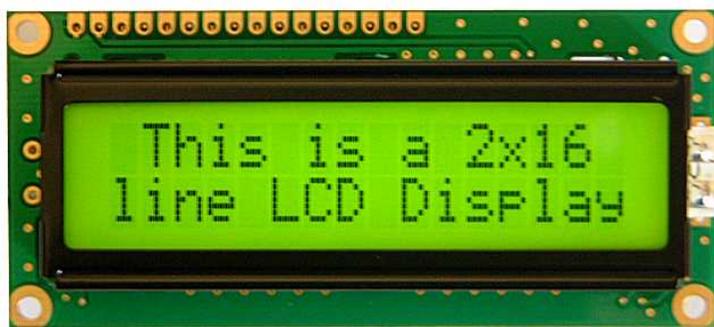
Idealmente, el sistema debe estar independiente al uso de la computadora, en este caso se deben considerar dispositivos adicionales, uno para visualizar los datos del sensor y la alimentación de energía eléctrica a la placa.

Para visualizar datos se propone un display LCD 16 x 2 y utilizar una alimentación de energía a través de una batería de 9V. La placa Arduino Uno funciona de forma óptima con una alimentación de energía de 7 a 12 V.

**Monitor Serial.** El entorno de trabajo del software Arduino permite enviar o mostrar datos al circuito conectado a la placa a través del conector USB, a través de esta opción se configura la velocidad de transmisión de los datos que se asigna en la rutina a través del Serial.begin.

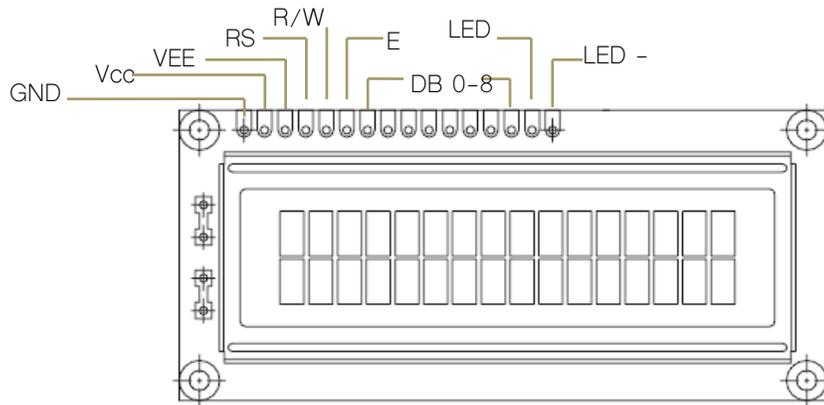
---

### *Pantalla de cristal líquida (16 x 2)*



Es un dispositivo electrónico que tiene la función de una pantalla. 16 x 2 indica que tiene la capacidad para desplegar 16 caracteres ASCII en 2 líneas, cada unidad de caracteres es una matriz de 8 x 5 píxeles.

El dispositivo LCD tiene dos registros "Command" y "Data" El registro de comando almacena las instrucciones para la pantalla, como inicializar, limpiar, ajuste de la posición del cursor, control de pantalla, etc. El dato o registro de datos, almacena la información que mostrará la pantalla, los caracteres deben tener propiedades ASCII.



| Nombre | Función   |
|--------|---|
| GND    | Conector a tierra 0V  |
| VCC    | Alimentación de tensión (4.7V – 5.3V)                           |
| VEE    | Ajuste de contraste a través de un resistor variable            |
| RS     | Selecciona comando de registro “low” y registro de datos “high” |
| R/W    | “low” cuando escribe registro, “high” cuando lee el registro    |
| E      | Envía datos cuando su estado es “high”                          |
| DB0    | Pines de 8 bits de datos  |
| DB1    |   |
| DB2    |   |
| DB3    |   |
| DB4    |   |
| DB5    |   |
| DB6    |   |
| DB7    |   |
| DB8    |   |
| LED+   | Luz de pantalla Vcc 5V  |
| LED-   | luz de pantalla 0V  |

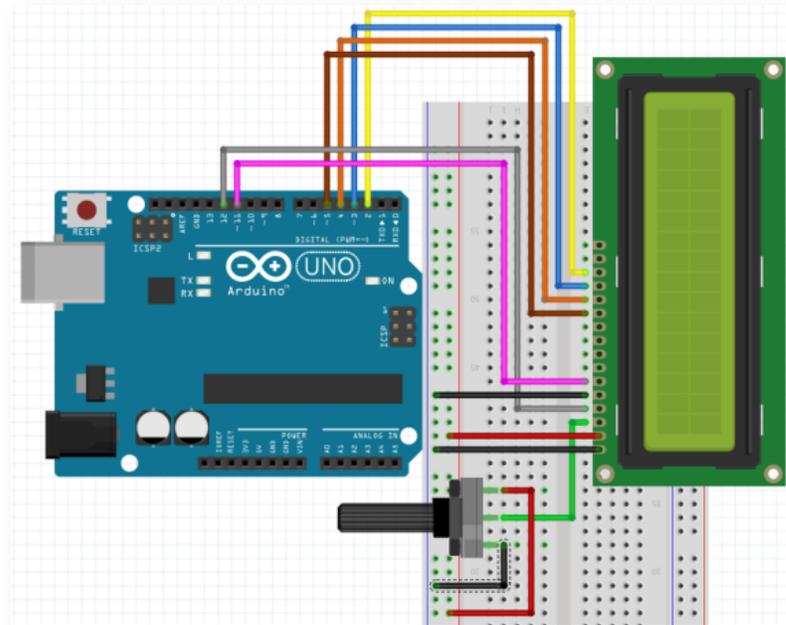
Arduino integra la librería de este componente. Desde el menú de ejemplos y submenú “LiquidCrystal” podemos probar algunas funciones del dispositivo.

## Conexión Arduino

Algunos aspectos a considerar son:

- El pin de la pantalla LCD VEE, se reserva su uso para un dispositivo que regule el contraste (potenciómetro), si no se utiliza entonces emitirá su brillo al máximo.
- El pin LED- se debe proteger con un resistor de 10K y controla el contraste del fondo, mientras que el pin RS controla el contraste del texto.

El diagrama que se muestra utiliza solamente un potenciómetro.



### Código

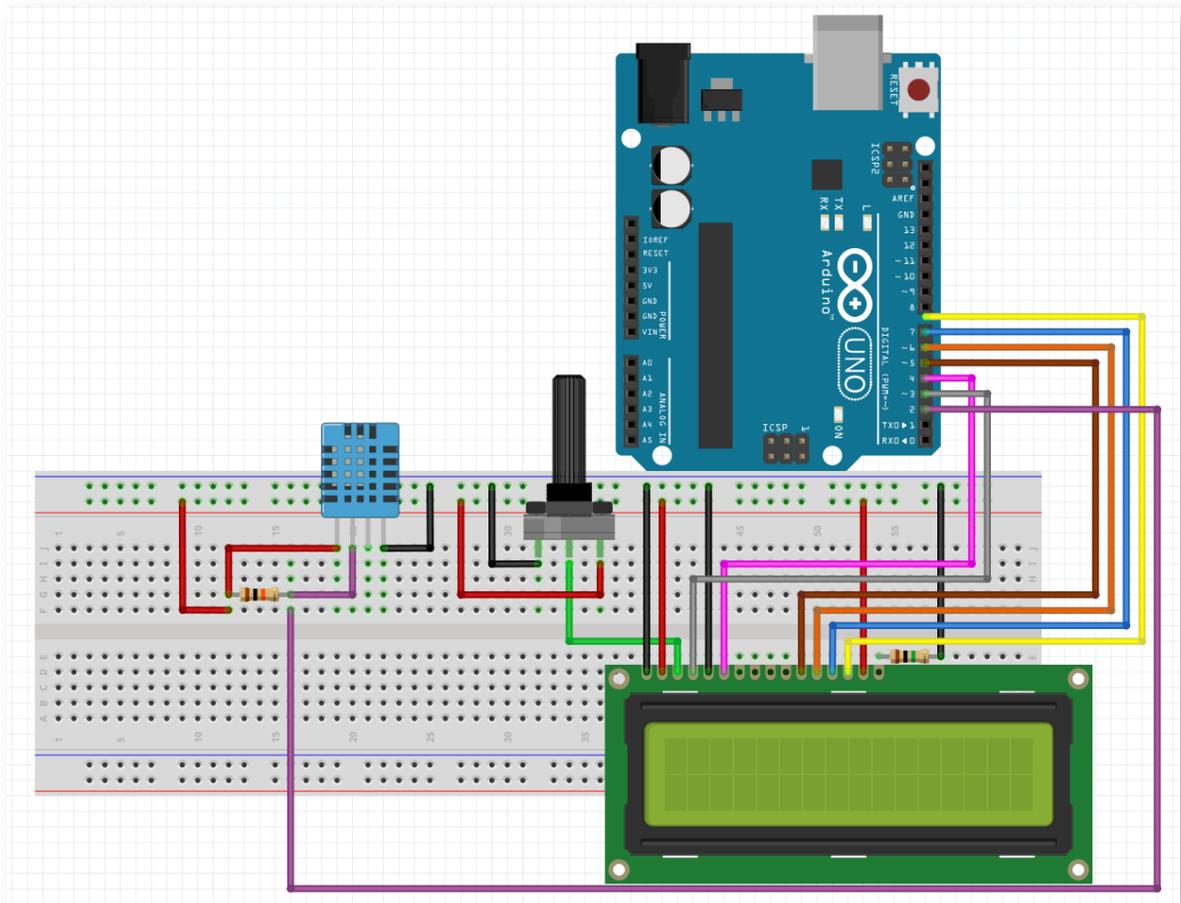
Con la conexión que se muestra en la gráfica se recomienda probar las rutinas que tiene Arduino en sus librerías internas, las cuales podemos probar desde el menú Ejemplos > LiquidCrystal.

Cada uno de los ejemplos, Autoscroll, Blink, Cursor, CustomCharacter, Display, HelloWorld, Scroll, SerialDisplay, setCursor, TextDirection, muestran los recursos de lenguaje que se pueden aplicar en los desarrollos que se realicen.

### Conexión Arduino + sensor + LCD

Las tareas realizadas con anterioridad proporcionan un ejercicio para comprender la utilidad de probar cada uno de los elementos por separado, para entender de qué manera se comunican entre sí, el paso siguiente, es reunir todos los elementos y hacer las adecuaciones pertinentes de acuerdo con nuestro objetivo.

Diagrama de conexión de todo el sistema. La resistencia del sensor es de  $10\text{ K}\Omega$  mientras que la del display es de  $1\text{ M}\Omega$ .



El código a utilizar es el del sensor al cual se le aplicarán algunas líneas para que muestre la información.

Al activar el potenciómetro observaremos que regula la intensidad de contraste del texto, sin embargo para su función no es indispensable manipular este elemento debido a que siempre lo estaremos utilizando en el contraste máximo.

### Referencias

[http://issuu.com/rduinostar/docs/dht11\\_datasheet](http://issuu.com/rduinostar/docs/dht11_datasheet)

<https://learn.adafruit.com/dht>

<http://www.oddwires.com/using-a-dht11-to-measure-temperature-and-humidity/>

<http://www.engineersgarage.com/electronic-components/16x2-lcd-module-datasheet>