

Medindo temperatura e umidade usando o sensor DHT11

O DHT11 é um sensor capaz de medir a temperatura e a umidade do ambiente. Este componente é constituído de duas partes principais: um sensor de umidade capacitivo e um sensor de temperatura termistor NTC, isto é um resistor sensível à variação de temperatura.

A **Figura 1** demonstra a estrutura do sensor DHT11, que possui 4 terminais. O primeiro (VCC) é o terminal de alimentação, que conforme datasheet do fabricante pode ser entre 3 V e 5,5 V. O segundo (DATA) é o pino de comunicação de dados, será através dele que o valor da temperatura e umidade serão comunicados a placa microcontroladora. O terceiro (NC) é um pino vazio. Por fim, o último (GND) é o terminal de terra do sensor.

PINOS DO SENSOR DHT11

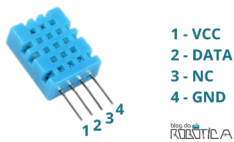


Figura 1: Estrutura do sensor DHT11.

Vamos aproveitar este tutorial para aprender um pouco sobre a utilização de bibliotecas no Arduino IDE. As bibliotecas são um conjunto de instruções desenvolvidas para executar tarefas específicas relacionadas a um determinado dispositivo. No nosso caso utilizaremos as bibliotecas DHT Sensor Library e Adafruit Unified Sensor Lib.

MATERIAIS NECESSÁRIOS

1 x Placa UNO SMD R3 Atmega328 compatível com Arduino UNO;

1 x Cabo USB;

1 x Sensor de umidade e temperatura DHT11;

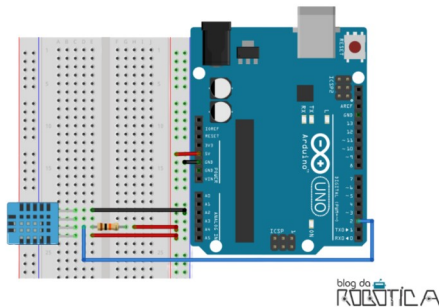
1 x Resistor de 10 kΩ;

1 x Protoboard;

Fios de jumper macho-macho.

ESQUEMÁTICO DE LIGAÇÃO DOS COMPONENTES

Antes de iniciar a montagem elétrica do circuito, certifique-se que a sua placa Arduino esteja desligada. Em seguida, monte o circuito utilizando a protoboard, o resistor, o sensor DHT11 e os fios.



- O pino 1 (VCC) do sensor DHT11 será ligado a alimentação de 5 V da placa Arduino;
- O pino 2 (DATA) deve ser conectado à porta digital 2;
- O pino 3 (NC) não será utilizado;
- E o pino 4 (GND) deve ser conectado ao GND (terra);
- Um resistor pull-up de 10 kΩ deve ser inserido no pino 2 do DHT11.

INSTALANDO AS BIBLIOTECAS

Antes de iniciar a elaboração do código, vamos proceder a instalação das bibliotecas para utilização do sensor DHT11. Estas bibliotecas encontram-se disponíveis para download nos seguintes links:

[DHT Sensor Library](https://www.blogdarobotica.com/biblioteca-dht11): <https://www.blogdarobotica.com/biblioteca-dht11>

[Adafruit Unified Sensor Lib](https://www.blogdarobotica.com/biblioteca-adafruit) <https://www.blogdarobotica.com/biblioteca-adafruit>

Após realizar o download da biblioteca, vamos instalá-la por meio do seguinte caminho: Sketch > Incluir biblioteca > Adicionar biblioteca ZIP, conforme ilustra a Figura 2.

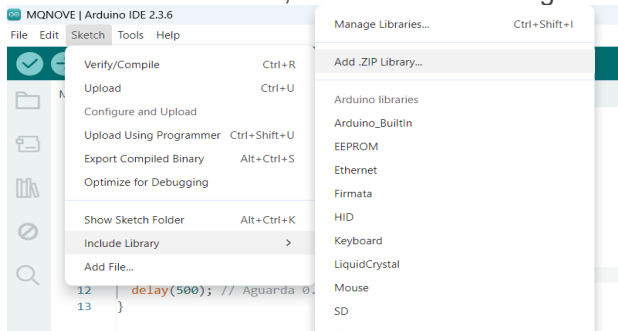


Figura 2: Caminho para incluir biblioteca no Arduino IDE.

Com a biblioteca instalada, vamos a programação do nosso Sketch. Este projeto propõe a utilização do sensor DHT11 para leitura da umidade e temperatura do ambiente e imprimir estes valores na serial do Arduino IDE.

ELABORANDO O CÓDIGO

Com o circuito montado e biblioteca instalada, vamos a programação do Sketch. Para melhor compreensão o código será explicado passo-a-passo a seguir. Neste momento, observe o código abaixo e aproveite para analisar sua estrutura.

```
//Projeto medir a temperatura e a umidade com DHT11
#include <DHT.h>//Inclui a biblioteca DHT Sensor Library
#define DHTPIN 2//Pino digital 2 conectado ao DHT11
#define DHTTYPE DHT11//DHT 11
DHT dht(DHTPIN, DHTTYPE);//Inicializando o objeto dht do tipo DHT passando como parâmetro o pino (DHTPIN) e o
tipo do sensor (DHTTYPE)
void setup() {
  Serial.begin(9600);//Inicializa a comunicação serial
  dht.begin();//Inicializa o sensor DHT11
}
void loop() {
  delay(2000);//Intervalo de dois segundos entre as medições
  float h = dht.readHumidity();// Lê o valor da umidade e armazena na variável h do tipo float (aceita números com casas
decimais)
  float t = dht.readTemperature();// Lê o valor da temperatura e armazena na variável t do tipo float (aceita números com
casas decimais)
  if (isnan(h) || isnan(t)) { //Verifica se a umidade ou temperatura são ou não um número
    return; //Caso não seja um número retorna
  }
  Serial.print("Umidade: "); //Imprime no monitor serial a mensagem "Umidade: "
  Serial.print(h); //Imprime na serial o valor da umidade
  Serial.println("%"); //Imprime na serial o caractere "%" e salta para a próxima linha
  Serial.print("Temperatura: "); //Imprime no monitor serial a mensagem "Temperatura: "
  Serial.print(t); //Imprime na serial o valor da temperatura
  Serial.println("°C "); //Imprime no monitor serial "°C" e salta para a próxima linha
}
```

Ao elaborar o código observe os seguintes pontos:

1. A instrução `#include <DHT.h>` é utilizada para incluir a biblioteca do sensor DHT11 no código.
2. A instrução `#define DHTPIN 2` define que o pino digital 2 será responsável por receber os dados do sensor DHT11;
3. A instrução `#define DHTTYPE DHT11` define que o tipo de sensor utilizado é o DHT11, uma vez que esta biblioteca também pode ser utilizada para outros tipos de sensor como o DHT22;
4. O comando `DHT dht(DHTPIN, DHTTYPE);` foi utilizado para criar o objeto `dht` do tipo `DHT`, passando como parâmetros o pino que o DHT11 encontra-se conectado e o tipo de sensor utilizado;
5. No loop, inicializamos a comunicação serial e o sensor DHT11;
6. Iniciaremos o setup incluindo um intervalo para que as leituras só ocorram de 2 em 2 segundos, pois este é o tempo de resposta do sensor;
7. Em seguida, criamos duas variáveis do tipo float para receber a leitura da umidade e temperatura;
8. Como forma de evitar erros de leitura, criamos a condicional `if (isnan(h) || isnan(t))` para verificar se os valores de temperatura ou umidade são números. Para isso, utilizamos a função `isnan` (do inglês, *is not a number*) que retorna 1 se o valor não for um número;
9. A instrução `Serial.print("Umidade: ");` é utilizada para que seja impressa no monitor serial o texto "Umidade: ";
10. A instrução `Serial.print(h);` é utilizada para imprimir no monitor serial do Arduino IDE o valor da umidade lida;
11. O comando `Serial.println("%");` é utilizada para imprimir na serial o caractere "%" e saltar para a próxima linha;
12. A instrução `Serial.print("Temperatura: ");` é utilizada para que seja impressa no monitor serial o texto "Temperatura: ";
13. Para imprimir o valor da temperatura captada pelo sensor utiliza-se a instrução `Serial.print(t);`;
14. O comando `Serial.println("°C ");` é utilizada para imprimir na serial o texto "°C";

Após carregar o código para o Arduino, abra o monitor serial. Os dados de temperatura e umidade serão exibidos e atualizados num intervalo de 2 segundos.

