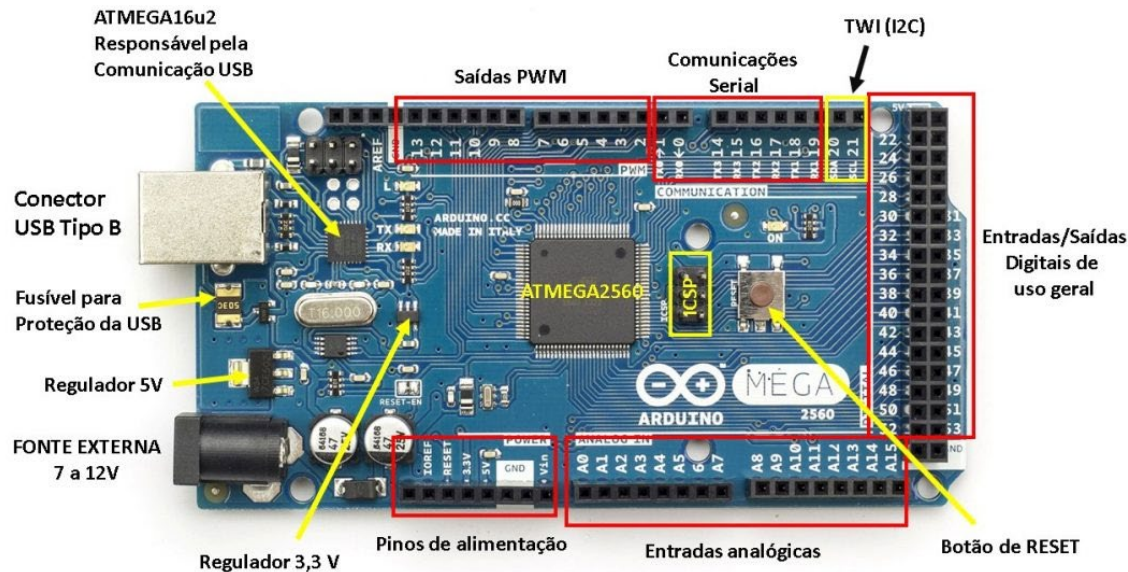


# MONITORAMENTO DE TEMPERATURA E UMIDADE EM AMBIENTES

Clarissa Bilyk  
Myllena Lacerda  
Vanessa Roncovsky



# Arduino Mega 2560

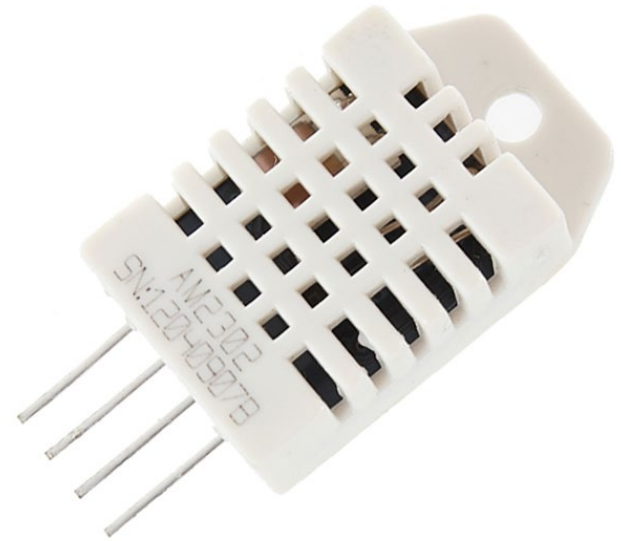


Fonte: <https://www.embarcados.com.br/arduino-mega-2560/>



# Sensor de Umidade e Temperatura DHT22

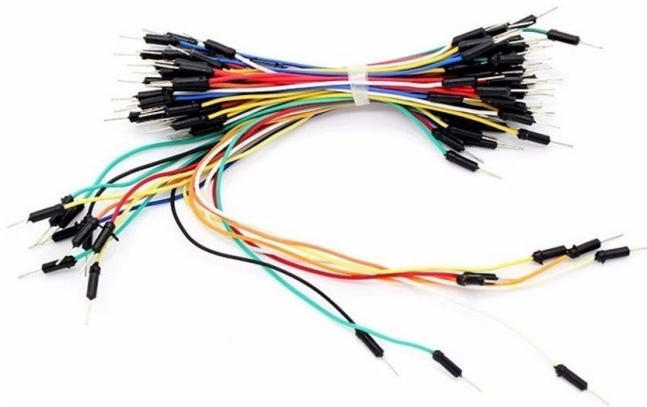
- Temperatura de -40° até 80° celsius
- Umidade do ar nas faixas de 0 a 100%



Fonte: <https://www.filipeflop.com/produto/sensor-de-umidade-e-temperatura-am2302-dht22/>

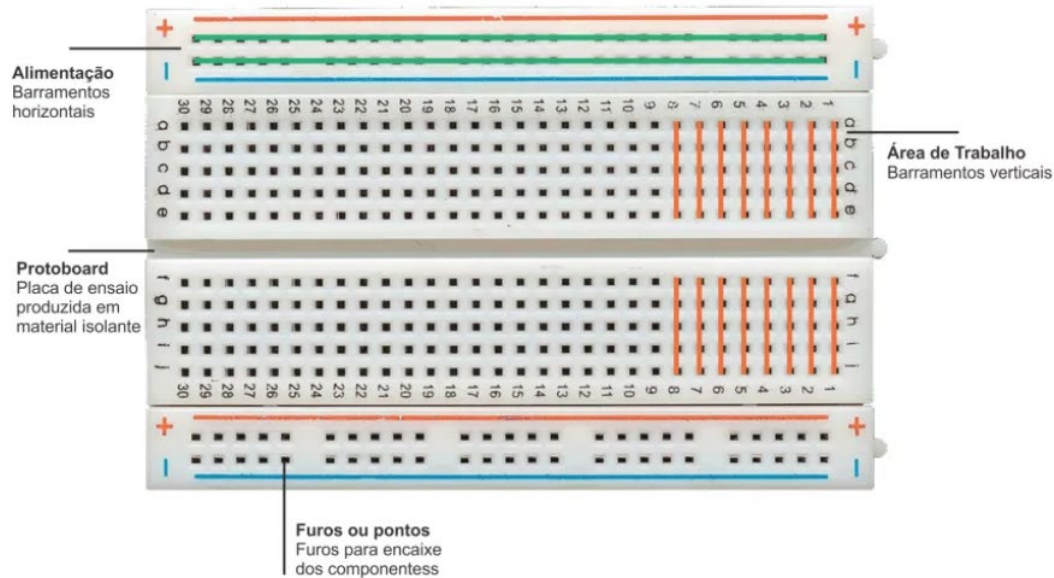


# Jumpers



Fonte: <http://www.baudaeletronica.com.br/acessorios/cabos/jumpers>

# Protoboard



Fonte: <https://portal.vidadesilicio.com.br/protoboard/>



## Objetivo

O sistema monitora, por meio de uma interação do sensor DHT22 (temperatura e umidade), Arduíno Mega 2560 e python, a variação de temperatura e umidade em tempo real, informações que são enviadas ao comando do usuário pelo Telegram.



# Programa Arduino

```
#include <Adafruit_Sensor.h>

#include <DHT.h>
#include <DHT_U.h>

//Define o macro DHTPIN de acordo com o pino digital correspondente.
#define DHTPIN 7

//Selecionando o tipo de sensor (no nosso caso, o DHT22)

#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
```



# Programa Arduino

```
//Cria e inicializa os dados do sensor.  
DHT dht(DHTPIN, DHTTYPE);  
  
float umidadeAtual;  
float temperaturaAtual;  
  
// Função que executa uma vez para configurar  
// variáveis e parametros  
  
void setup() {  
  
    // Inicializa a porta serial a uma velocidade de 57600bps  
    Serial.begin(57600);  
    //Inicializa o sensor.  
    dht.begin();  
}
```





# Programa Arduino

```
// Esta função executa em loop indefinidamente após a execução da função
//setup
// mas pode ser parado por certas funções e interrupções.
void loop()
{
    // Aguarda-se 2s antes de realizar uma nova leitura do sensor.
    delay(2000);

    umidadeAtual = dht.readHumidity();
    temperaturaAtual= dht.readTemperature();
```



# Programa Arduino

```
// Checa se alguma das leituras falhou.  
if (isnan(umidadeAtual) || isnan(temperaturaAtual))  
{  
    Serial.println("ERRO: Falha ao ler os dados do sensor DHT!");  
    return;  
}  
  
// Envia através da porta serial os valores de umidade e temperatura.  
  
Serial.print(umidadeAtual);  
Serial.print("\t");  
Serial.print(temperaturaAtual);  
Serial.print("\n");
```



# Programa Python

```
import telebot
import serial
from serial.tools import list_ports

bot = telebot.TeleBot("859832840:AAGCqrBMfri4iC9_4prenY6Q3jML4BUfL-w")

# Salva todas as leituras de umidade/temperatura
lista = []

def le_porta():
    #Primeiramente precisamos determinar as portas seriais disponiveis na maquina
    # para isso, obtemos a lista de portas seriais e escolhemos manualmente
    # aquela com index 0.

    selectedPortIndex = 0
    selectedDevice = ""

    ports = list_ports.comports()
    #print("Avaialbe ports:\n%s"%"\n".join(["\t%d: %s"%
    #(portIndex, str(ports[portIndex])) for portIndex in range(len(ports))]))

    selectedDevice = ports[selectedPortIndex].device
    #print(f"Selected device: {selectedDevice}")

    ser = serial.Serial(selectedDevice, 57600)

    # Cada execução lê uma linha da porta serial
    # e separa os dois valores (umidade e temperatura)
```



# Programa Python

```
try:
    for line in ser:
        try:
            entry = line.decode("utf-8").split("\t")
            humidity = float(entry[0])
            temperature = float(entry[1])
            #print(f"T: {temperature} - H: {humidity}")
            lista.append([humidity,temperature])
            return humidity, temperature
        except ValueError as e:
            print(f"E: {line}")
            return -1, 0
        except IndexError as e:
            print(f"E: {line}")
            return -1, 0
except KeyboardInterrupt:
    # Ao abortar a execução do programa esta exception é chamada
    # deve-se então fechar a porta serial para novas comunicações
    ser.close()
    return -1, 0
except:
    #Caso seja um erro não especificado é importante fechar a porta
    # serial para permitir comunicação futura
    ser.close()
    return -1, 0
```



# Programa Python

```
except:
    #Caso seja um erro não especificado é importante fechar a porta
    # serial para permitir comunicação futura
    ser.close()
    return -1, 0

# mensagem de Início ao comando /start do Bot
@bot.message_handler(commands=['start', 'help'])
def send_welcome(message):
    bot.reply_to(message, "Olá! Eu sou um Bot criado com a finalidade de captar as informações transmitidas pelo Sensor DHT22 integrado com um Arduino Mega 2

# Bot informa a umidade captada pelo sensor do arduino (ultima info passada pelo sensor) e passa para o usuário
@bot.message_handler(commands=['umidade'])
def analisa_umidade(message):
    umidade, temperatura = le_porta()

    if umidade == -1:
        bot.reply_to(message, "Erro ao obter informações")
    else:
        bot.reply_to(message, "Umidade atual: {}".format(umidade))

# Bot informa a Temperatura captada pelo sensor do arduino (ultima info passada pelo sensor) e passa para o usuário
@bot.message_handler(commands=['temperatura'])
def analisa_temperatura(message):
    umidade, temperatura = le_porta()

    if umidade == -1:
        bot.reply_to(message, "Erro ao obter informações")
    else:
        bot.reply_to(message, "Temperatura atual: {}".format(temperatura))

bot.polling()
```

# RESULTADOS

