



ROBOSTEM – A Trainers Toolkit To Foster
STEM Skills Using Microcontroller Applications
2019-1-RO01-KA202-063965



Co-funded by the
Erasmus+ Programme
of the European Union

MÓDULO DE PROGRAMAÇÃO DE MICROCONTROLADORES



Título

Utilização do sensor de temperatura para Arduino aplicado ao COVID19

Elementos Requeridos

- 1x Arduino Uno
<https://ardushop.ro/ro/home/29-placa-de-dezvoltare-uno-r3.html>
- 1x Placa
<https://ardushop.ro/ro/electronica/33-breadboard-830.html>
- 1x Sendor de Temperatura Lm35
<https://ardushop.ro/ro/electronica/192-senzor-temperatura-lm35dz.html>
- 1x Conjunto de fios jumper
<https://ardushop.ro/ro/electronica/28-65-x-jumper-wires.html>

Apps

- Arduino IDE
ou
- Arduino Web Editor

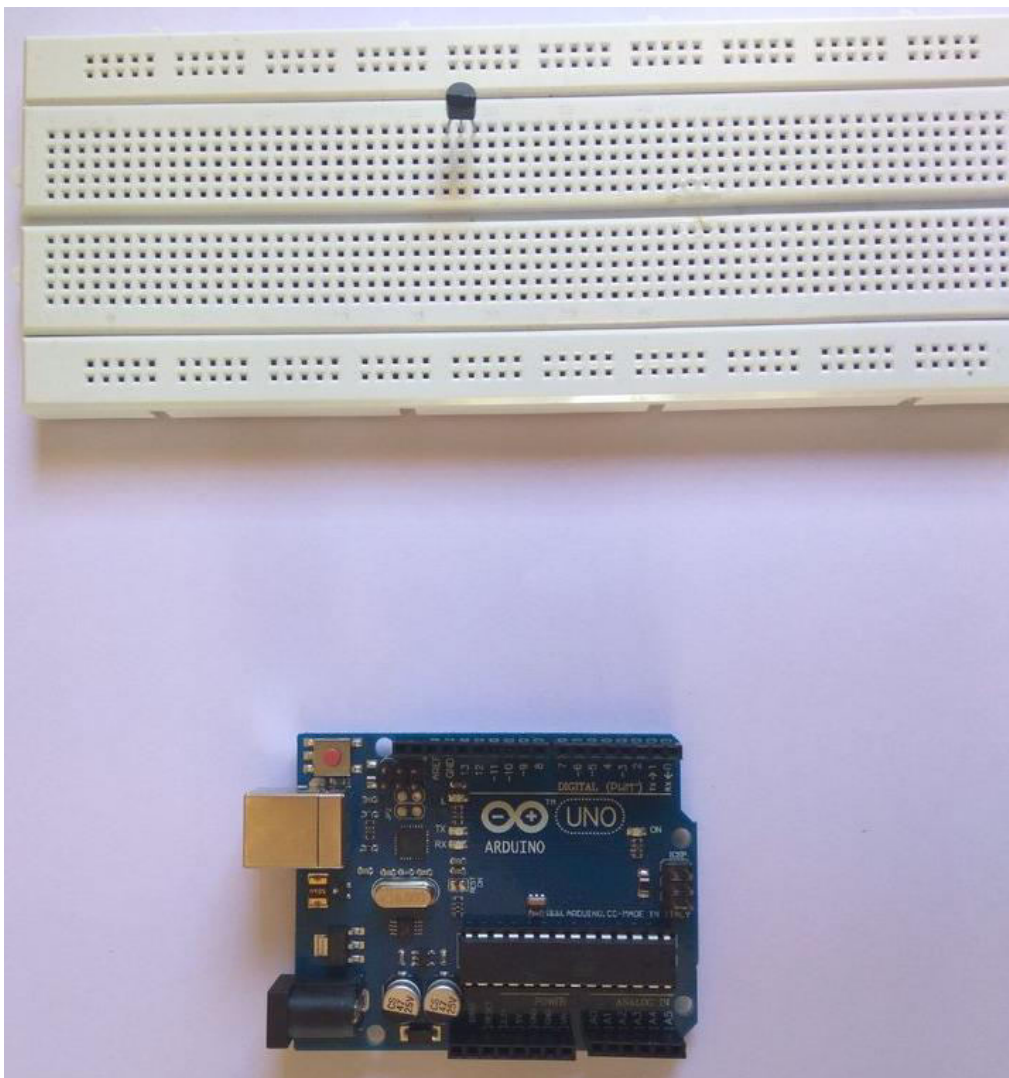
Descrição

O lm35 é um sensor de temperatura analógico linear. Isto significa que a tensão de saída é proporcional à temperatura. A tensão de saída aumenta em 10mv para cada aumento de 1 grau Celsius na temperatura. O Arduino pode ler entradas de 0-5v. O Arduino armazena isso como um número de 10bit (0-1023). O método que vamos usar agora pode ser usado para medir a temperatura a partir de 2 graus Celsius até ao máximo de temperatura que o seu lm35 pode medir.



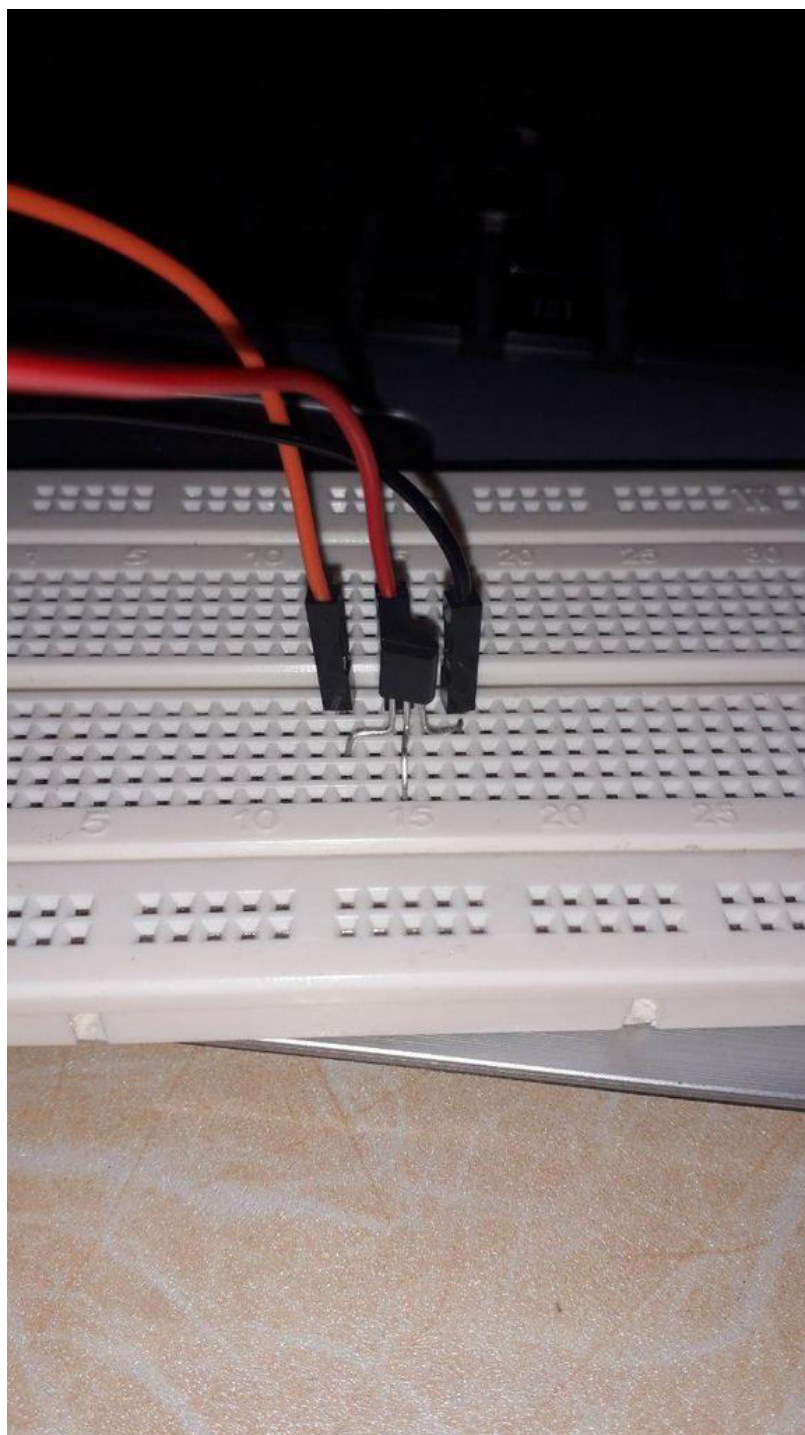
Construção do Projeto

No início vamos ter uma placa com o sensor nela e o Arduino que nós vamos conectar ao sensor.

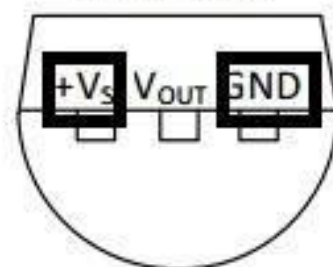




Passo 1: Estabelecendo as Conexões



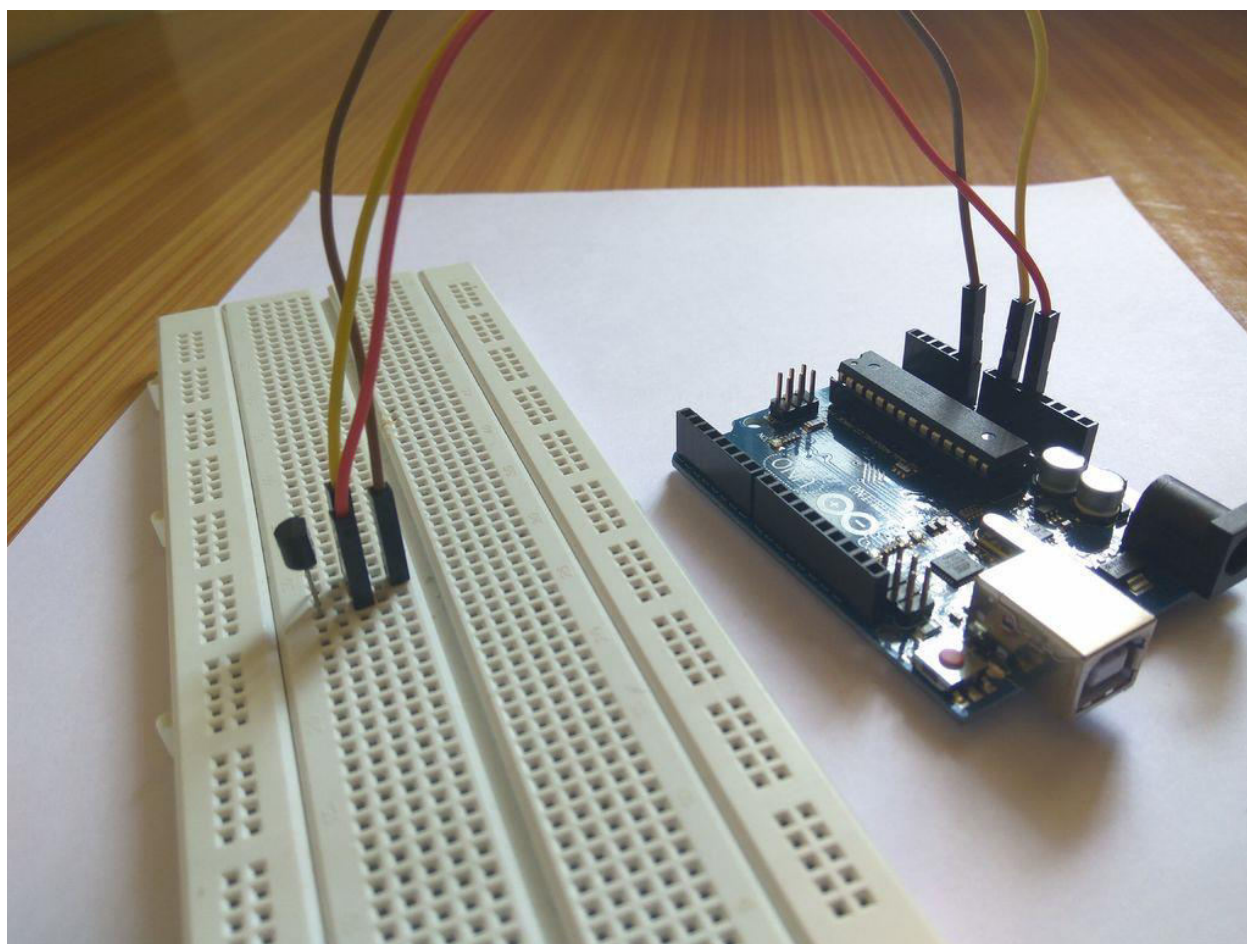
**LP Package
3-Pin TO-92
(Bottom View)**



Vs and
gnd are
intercha
nged in
my case



Conectar o pino Vs ao pino 5v no Arduino e enfiar um dos 2 pinos de terra na fonte de alimentação. Conectar o pino Vout a um dos pinos analógicos, A0 no nosso caso.



O Im35 que nós usámos tem o pino de terra e o pino Vs trocados comparado com o pino do Texas Instruments, a folha de dados para a qual está normalmente disponível. Se trocar os pinos ao conectar o sensor, ele ficará quente para que se saiba que está incorreto.



Dica:

A temperatura que se obtém não é confiável se se conectar muitos cabos de jumper juntos para tornar o fio longo.

Passo 2: O Código

Primeiro precisamos da configuração.

```
int temppin = 0;

float temp;

void setup()
{
    Serial.begin(9600);
}
```

Aqui estamos inicializando a temperatura do pino que iremos trabalhar e a variável que vai armazenar a temperatura da pessoa.

Depois disso, escrevemos uma função chamada loop.

```
void loop()
{
    temp = analogRead(temppin); // Reading data from the sensor. This voltage
    is stored as a 10bit number

    temp = (5.0 * temp * 100.0) / (1024 * 10)
}
```

Esta função permite que a variável temp leia os dados que lhe são fornecidos pelo sensor. E a tensão será armazenada como um número de 10 bits.

Depois disso nos usamos a formula simples, 5.0 (.0 para que eu pegue todos os números reais e seus decimais ao usar os operadores) * temp * 100.0(a mesma história que foi apresentada previamente) / 1024 * 10 -> **(5.0 * temp * 100.0) / (1024. 10)**

A parte 5 * temp / 1024 é para converter o número de 10bit numa leitura de tensão. Esta vai ser multiplicada por 1000 para convertê-la em millivolts e depois dividida por 10 porque cada aumento de grau resulta num aumento de 10 milivolts.



Depois disso escrevemos o sketch no nosso Arduino IDE.

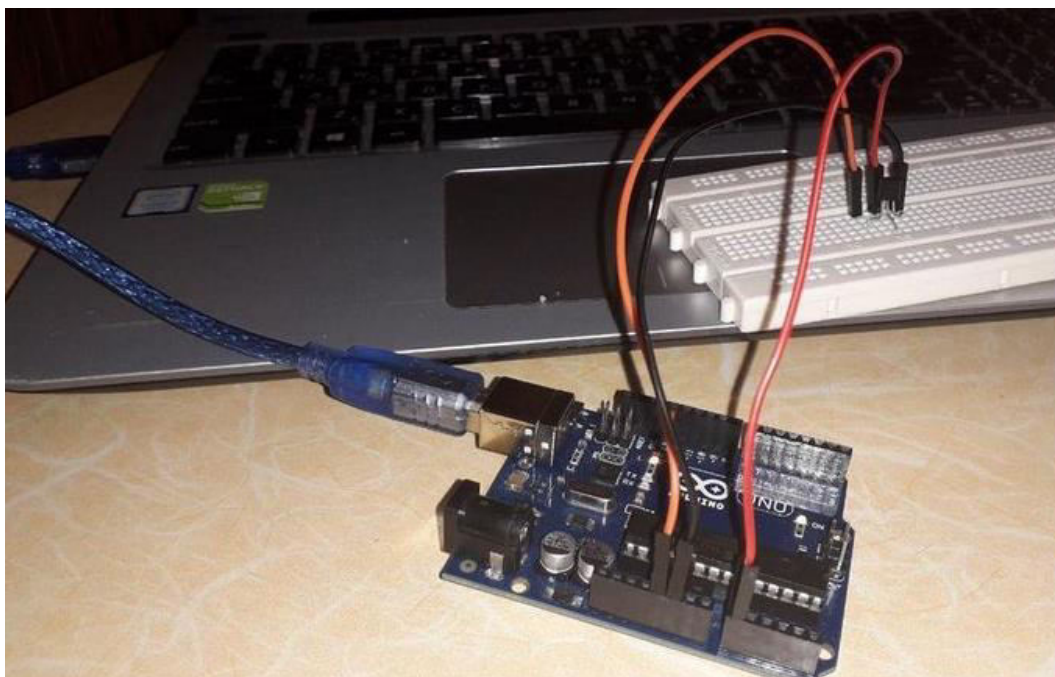
```
temperature_arduino | Arduino 1.8.13 (Windows Store 1.8.39.0)
File Edit Sketch Tools Help

temperature_arduino $
int tempin = 0;

float temp;
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  temp = analogRead(tempin); // Reading data from the sensor. This voltage is stored as a 10bit number
  temp = (5.0 * temp * 100.0)/(1024 *10)
  /* 5 * temp /1024 is to convert the 10bit number to a voltage reading.
  This is multiplied by 1000 to convert it to millivolt.
  We then divide it by 10 because each degree rise results in a 10 millivolt increase*/
}
Serial.println(temp);
delay(800); // This is because we don't want a continuous stream of data
```

Em seguida, enviamos o código para o Arduino Uno, que agora deve ser algo assim:





Passo 3: Verificar a Temperatura

```
COM4 (Arduino/Genuino Uno)
277.34
27.34
27.34
27.34
26.86
27.34
27.34
27.34
26.86
27.34
27.34
27.34
27.34
27.34
27.34
26.86
27.34
27.34
27.34
27.34
27.34
26.86
27.34
27.34
27.34
27.34
27.34
Autoscroll No line ending 9600 baud
```

Aqui podemos ver a temperatura do assunto em que testamos o nosso leitor de temperatura controlada Arduino.



O Código-Fonte

```
int temppin = 0;

float temp;

void setup()
{
    Serial.begin(9600);
}

void loop()
{
    temp = analorRead(temppin); // Lendo os dados do sensor. Esta tensão é ar
    mazenada como um número de 10bit

    temp = (5.0 * temp * 100.0)/(1024 *10)

    /* 5 * temp /1024 é para converter o número de 10 bits numa leitura de ten
    são.
    Esta é multiplicada por 1000 para convertê-la em milivolts.
    Em seguida, dividimos por 10 porque cada aumento de grau resulta num aumen
    to de 10 milivolts */

}

Serial.println(temp);

delay(800); // Isto é porque não queremos um fluxo contínuo de dados
```

Aqui temos o Código Fonte com alguns comentários que devem ajudar todos a entender melhor o que fizemos.



Lista de Preços

- 1x Arduino Uno – 4€
- 1x Breadboard – 2€
- 1x Temperature sensor Lm35 – 3€
- 1x Set of jumper wires – 3€

O preço total dos elementos requeridos é 12€

Links úteis

- <https://arduinomodules.info/>
- <https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>
- <https://www.youtube.com/watch?v=nL34zDTPkcs>
- https://www.youtube.com/watch?v=QO_Jlz1qpDw
- <https://randomnerdtutorials.com/9-arduino-compatible-temperature-sensors-for-your-electronics-projects/>