

## Relatório dos Sensores de Arduino

No dia 27/02/2020 iniciamos as atividades com o microcontrolador Arduino, que porventura utilizaremos em nosso projeto de pesquisa e inovação, no dia em que estamos utilizando o sensor de temperatura, desenvolvemos o código usando a linguagem C, conforme a imagem a seguir:

```
int pinoSensor = 0;
int valorLido = 0;
float temperatura = 0;

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

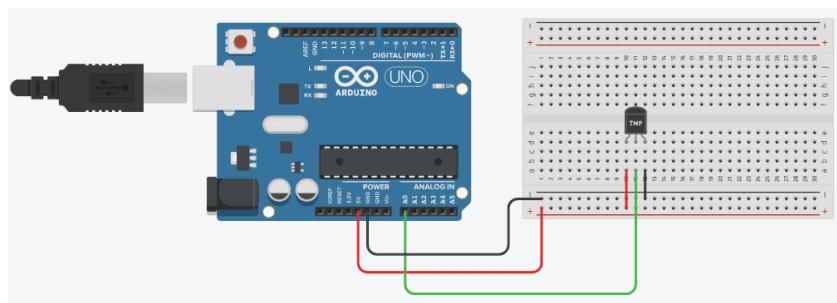
void loop() {

  valorLido = analogRead(pinoSensor);
  temperatura = (valorLido * 0.00488);
  temperatura = temperatura * 100;
  Serial.println(temperatura);
  delay(1000);

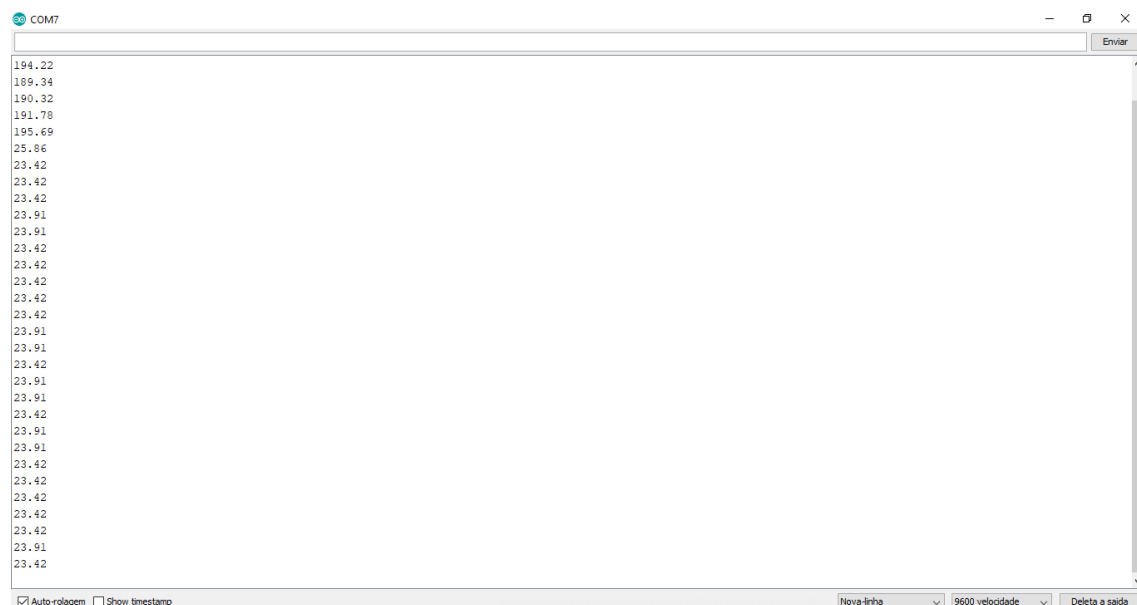
}
```

Acima declaramos as variáveis que serão responsáveis por guardar os valores atribuídos como o cálculo da tensão, valor que está sendo lido pelo arduino. Os componentes utilizados foi: 1 Arduino, 1 Sensor de temperatura LM35 e 3 fios macho-macho.

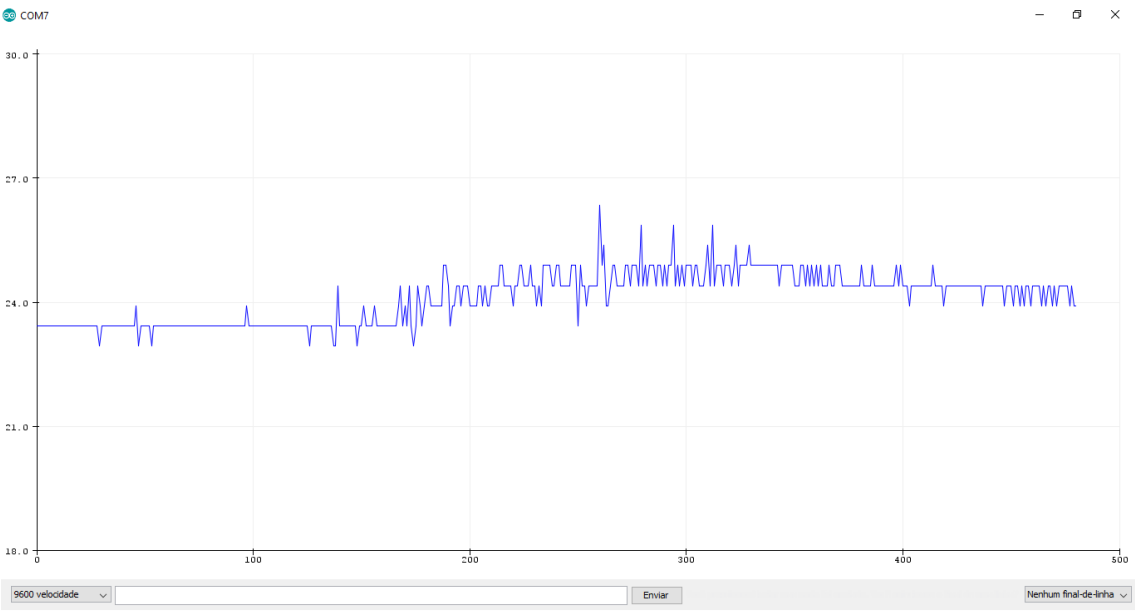
O código desenvolvido é responsável por capturar os dados enviados pelo sensor e exibir a temperatura em uma tela chamada “Monitor Serial”. Conectamos o sensor de temperatura em uma protoboard, e conectamos os cabos no arduino. Os cabos da cor vermelha representam a tensão positiva, os das cores pretas a tensão negativa e o da cor verde ele é responsável por transmitir os dados do sensor. Logo após a montagem fomos testar fisicamente ocorreu um erro porque trocamos a porta que seria lida pelo microcontrolador que seria a porta 0 e não a porta 5, então resolvemos o erro trocando diretamente no Arduino.



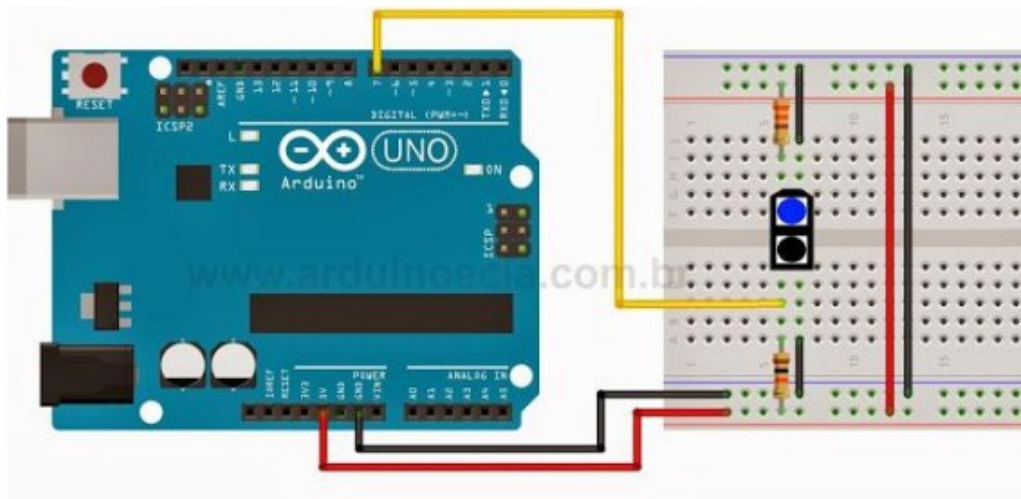
Quando tudo estava correto e funcionando foi exibido no Monitor Serial os seguintes valores da temperatura:



Também podemos ver o quanto a temperatura aumentou ou abaixou através de um gráfico gerado pela ide do Arduino.



No dia 5 de março voltamos com as atividades relacionadas ao Arduino, porém com sensores diferentes, começamos com o sensor de obstáculos que tem por nome técnico TCRT500. Quando começamos a montar o circuito para que o sensor funcionasse de forma correta nos deparamos com um problema, porém a montagem estava correta como especificada na imagem passada pela professora, quando fomos ver direito o que havia acontecido percebemos que trocamos as posições dos resistores, porque cada lado do sensor tem que ter um resistor adequado para aquela parte, depois disso reorganizamos e concertamos o problema que surgiu e agora estava totalmente correto como nos foi passado.



O sensor TCRT500 funciona basicamente dessa forma um dos “leds” emite uma luz e o outro só recebe se estiver algo na frente deles.

Logo após desenvolver o código a seguir, na qual o Arduino irá obedecer às instruções dadas a ele.

```
int objeto = 0;
int sensor = 7;

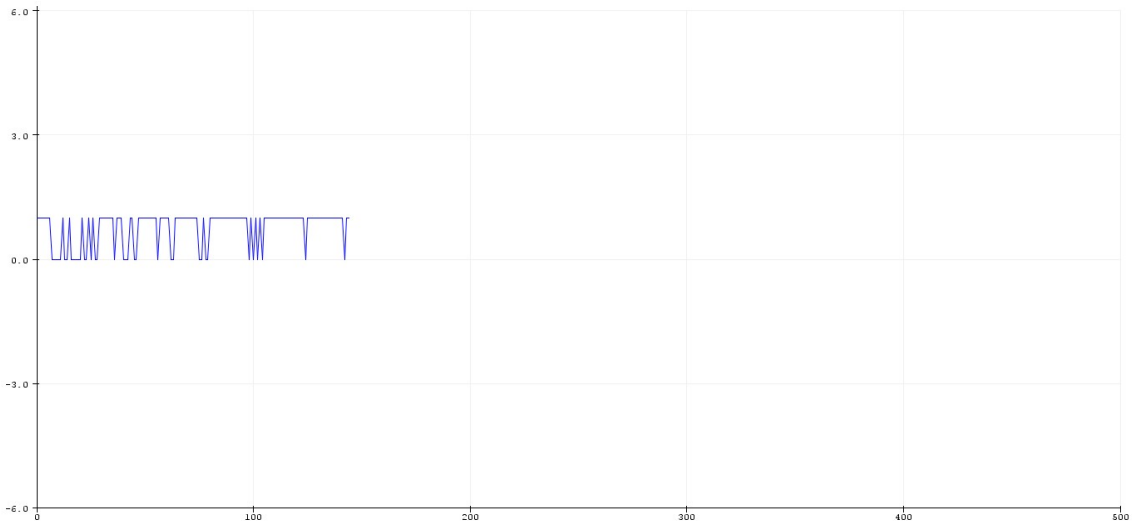
void setup() {
  pinMode(sensor, INPUT);
  Serial.begin(9600);
}

void loop() {
  objeto = digitalRead(sensor);
  Serial.println(objeto);
  delay(1000);
}
```

Com o código feito e as conexões do Arduino corretar podemos ver o resultado quando identificamos um objeto ou não através do monitor serial, quando vemos 0 não passou algo na frente dele, quando vemos 1 quer dizer que passou algo por ele.

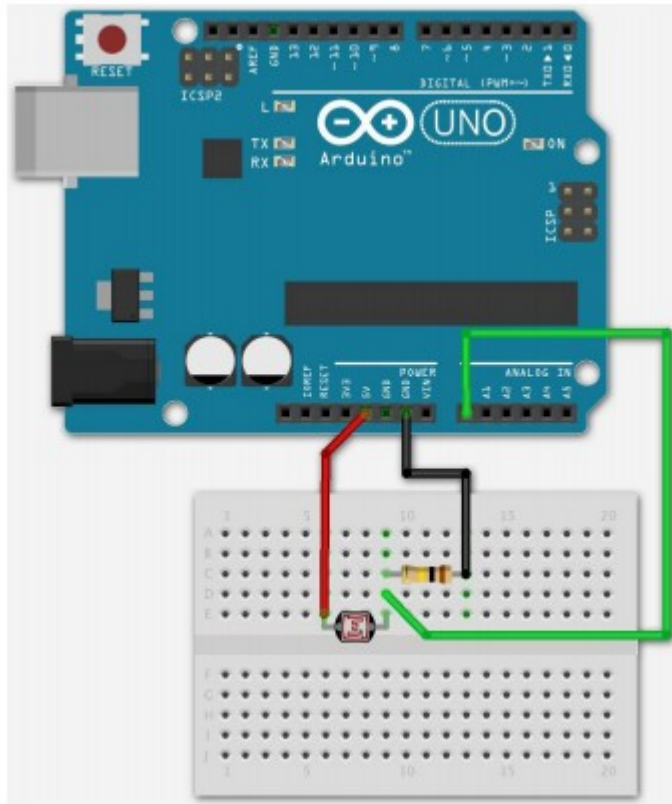


Esse aqui é um gráfico relacionado ao resultado de quando o sensor detecta algo ou não.



Depois de ter concluído esses outros sensores, fomos explorar um outro sensor o LDR ele mede a luminosidade do ambiente. Fomos montar o circuito para que esse sensor receba energia necessária e também passe para a IDE do Arduino as informações que estão sendo lidas pelo sensor, mediante disso partimos para desenvolver o código para que o Arduino entenda o que fazer com aquilo que está recebendo.

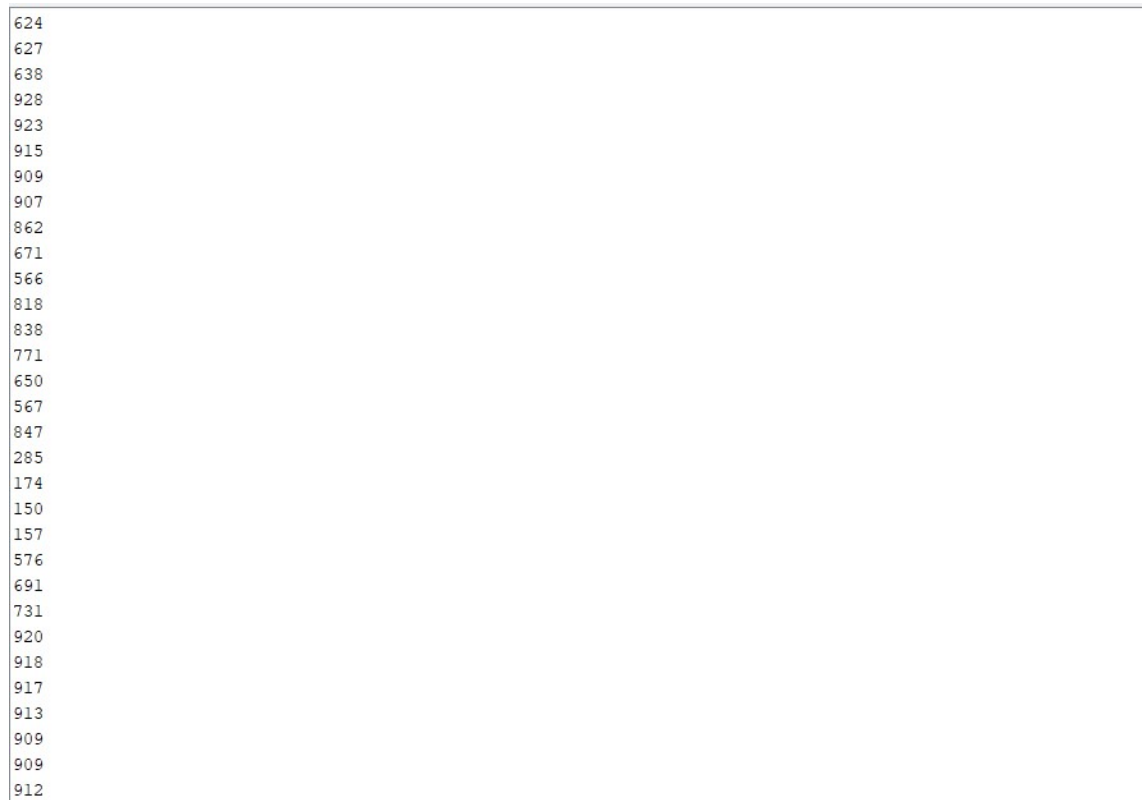
O circuito ficaria assim:



O código:

```
int pinSensor = A0;
int sensorValue = 0;
void setup() {
  Serial.begin(9600);
}
void loop() {
  sensorValue = analogRead(pinSensor);
  Serial.println(sensorValue);
  delay(1000);
}
```

Contudo funcionando conseguimos ver o resultado de quanto de luminosidade o sensor está recebendo, quanto o valor mais baixo menos luminosidade, quanto maior mais luminosidade.



Podemos também ver o gráfico relacionado ao resultado do sensor da seguinte forma, ele aparece com muitas variações pelo fato de que os valores da luminosidade do sensor muda constantemente.

