

Blog FilipeFlop

Projetos com Arduinos

JUL 23 2011

Sensor Ultrassônico HC-SR04

Este projeto basea-se no Sensor Ultrassônico HC-SR04 (http://www.filipeflop.com/sensor-de-distancia-ultrassonico-arduino-pic_77xJM) comumente usado em projetos com Arduino e com um princípio simples de funcionamento. A construção deste post foi feita com a aquisição do Kit Arduino Start (http://www.filipeflop.com/kit-arduino-start_75xJM) em nossa loja FilipeFlop (<http://www.filipeflop.com/>) que mostrou-se uma ótima ferramenta para iniciar projetos com Arduinos, pois acompanha os seguintes produtos.:

- 1 – Arduino Duemilanove – 2009 (http://www.filipeflop.com/arduino-duemilanove-fonte-cabo-usb-ebook_18xJM)
- 1 – Fonte 9v
- 1 – Cabo USB
- 1 – LCD 16x2 – Escrita Branca e Backlight Azul (http://www.filipeflop.com/lcd-16x2-com-backlight-azul-escrita-branca_87xJM)
- 1 – Mini Protoboard 170 Pontos
- 1 – Sensor de Distância Ultrassônico (http://www.filipeflop.com/sensor-de-distancia-ultrassonico-arduino-pic_77xJM)
- 1 – Ebook (Practical Arduino Cool Projects for Open Source Hardware)
- 2 – Sensores Ópticos Reflexivos
- 2 – Potenciômetros 100kΩ
- 6 – Led's coloridos
- 10 – Resistores 1kΩ
- 10 – Chaves 6x6mm
- 65 – Jumpers para Protoboard (http://www.filipeflop.com/jumpers-protoboard-macho-macho-65-unidades_95xJM)

Princípio de Funcionamento.:

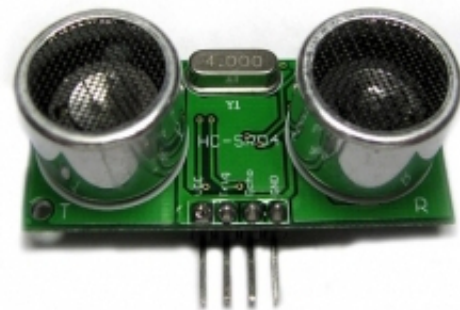
O Sensor Ultrassônico HC-SR04 funciona como um detector de objetos e permite medir distâncias mínimas de 2 centímetros podendo chegar a distâncias máximas de até 5 metros com uma precisão de 3 milímetros. Estes sensores emitem um sinal ultrassônico que reflete em um objeto e retorna ao sensor (eco). O sinal de retorno é captado, permitindo-se deduzir a distância do objeto ao sensor tomando o tempo de trânsito do sinal.

A velocidade do sinal ultrassônico é de aproximadamente 240 m/s, logo se o sensor estiver a uma distância “d” do objeto o sinal percorrerá uma distância equivalente a “2d” para sair e retornar ao sensor. Sabendo esses conceitos é possível calcularmos a distância de um objeto pela fórmula:

$$velocidade = \frac{distancia}{tempo} \Rightarrow v = \frac{2d}{t} \Rightarrow d = \frac{v \cdot t}{2} \Rightarrow d = 120 \cdot t$$

(<http://filipeflop.files.wordpress.com/2011/07/formula1.jpg>) **Pinagem.:**

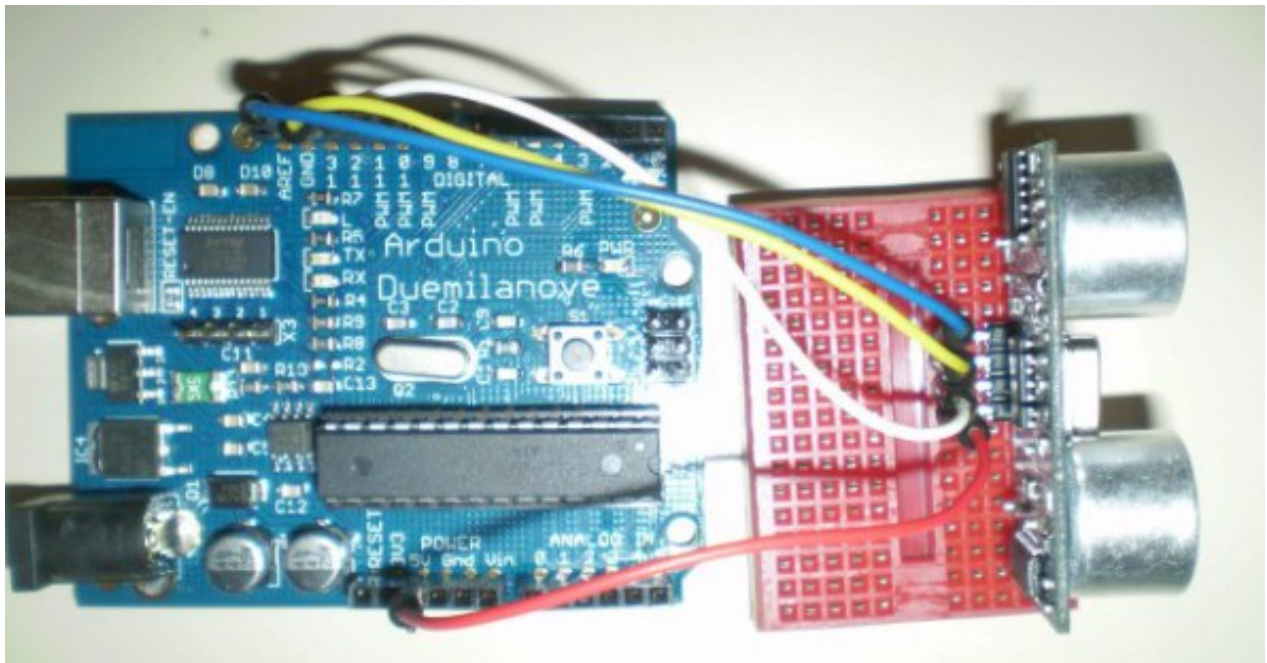
- o VCC : Alimentação 5v
- o TRIG : Pino de Gatilho
- o ECHO : Pino de Eco
- o GND : Terra



(<http://filipeflop.files.wordpress.com/2011/07/12896382070221.jpg>)

Montagem.:

- 1° – O jumper de cor azul é ligado ao pino GND (terra) do arduino e ligado ao pino GND (terra) do sensor.
- 2° – O jumper de cor amarela é ligado ao pino 13 do arduino e ligado ao pino ECHO (eco) do sensor.
- 3° – O jumper de cor branca é ligado ao pino 12 do arduino e ligado ao pino TRIG (emissor) do sensor.
- 4° – O jumper de cor vermelha é ligado ao pino de 5V (corrente) do arduino e ligado ao pino VCC (corrente) do sensor.



(<http://filipeflop.files.wordpress.com/2011/07/p1010020.jpg>)

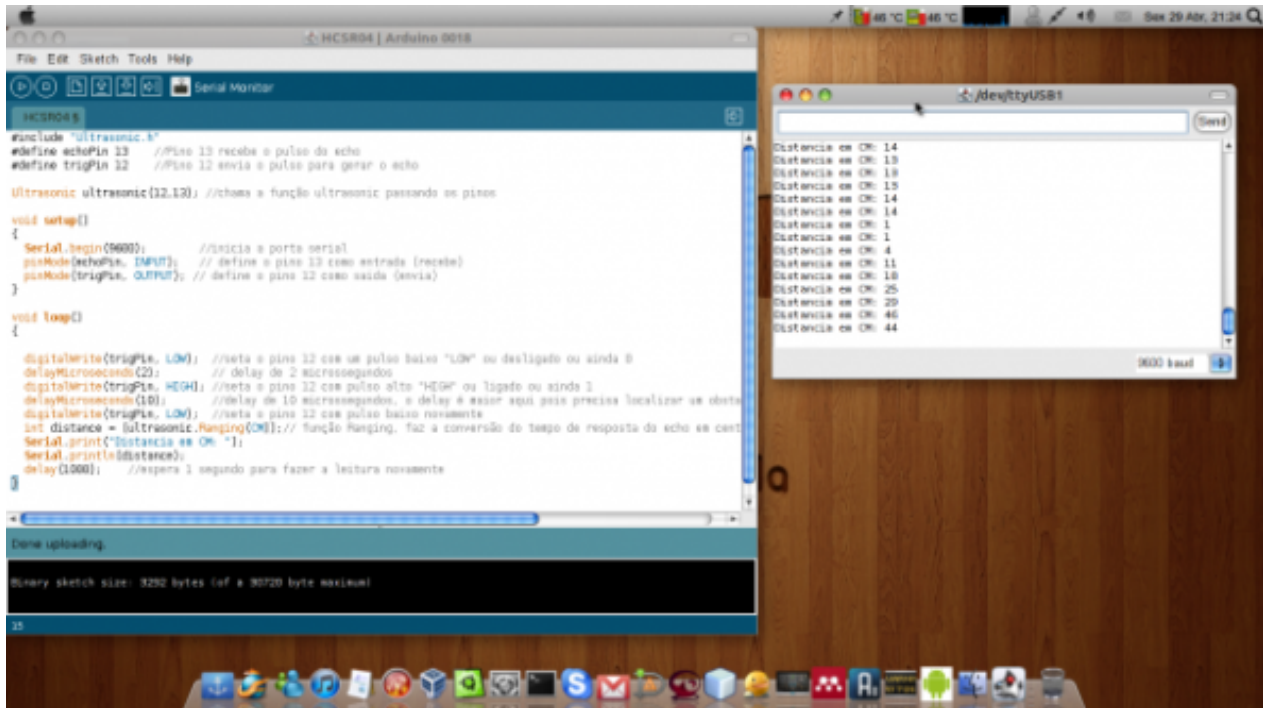
Após o montagem do circuito você deverá carregar o código na IDE do arduino. O código usado foi escrito usando a biblioteca do sensor ultrasonico, o download da biblioteca pode ser feito [AQUI](http://iteadstudio.com/store/images/produce/Sensor/HCSR04/Ultrasonic.rar) (<http://iteadstudio.com/store/images/produce/Sensor/HCSR04/Ultrasonic.rar>) O datasheet do sensor pode ser encontrado [AQUI](http://iteadstudio.com/store/images/produce/Sensor/HCSR04/HCSR04.pdf). (<http://iteadstudio.com/store/images/produce/Sensor/HCSR04/HCSR04.pdf>)

Código.:

```
1  <pre>#include "Ultrasonic.h"
2  #define echoPin 13 //Pino 13 recebe o pulso do echo
3  #define trigPin 12 //Pino 12 envia o pulso para gerar o echo
4  //iniciando a função e passando os pinos
5  Ultrasonic ultrasonic(12,13);
6
7  void setup()
8  {
9      Serial.begin(9600); //inicia a porta serial
10     pinMode(echoPin, INPUT); // define o pino 13 como entrada (recebe)
11     pinMode(trigPin, OUTPUT); // define o pino 12 como saída (envia)
12 }
13
14 void loop()
15 {
16     //seta o pino 12 com um pulso baixo "LOW" ou desligado ou ainda 0
17     digitalWrite(trigPin, LOW);
18     // delay de 2 microssegundos
19     delayMicroseconds(2);
20     //seta o pino 12 com pulso alto "HIGH" ou ligado ou ainda 1
21     digitalWrite(trigPin, HIGH);
22     //delay de 10 microssegundos
23     delayMicroseconds(10);
24     //seta o pino 12 com pulso baixo novamente
25     digitalWrite(trigPin, LOW);
26     // função Ranging, faz a conversão do tempo de
27     //resposta do echo em centímetros, e armazena
```

```
28 //na variavel distancia
29 int distancia = (ultrasonic.Ranging(CM));
30
31 Serial.print("Distancia em CM: ");
32 Serial.println(distancia);
33 delay(1000); //espera 1 segundo para fazer a leitura novamente
34 }</pre>
```

Após o código carregado e enviado para o Arduino, abra o monitor serial na IDE do arduino para verificar as saídas do seu projeto. Repare que as distâncias mudam os valores, nesse momento havia um objeto se aproximando e distanciando do sensor.



(http://filipeflop.files.wordpress.com/2011/07/captura_de_tela-1.png)

Colaborador.: Fernando Krein Pinheiro (ferpinheiro.wordpress.com)

Compartilhe!

About these ads (<http://en.wordpress.com/about-these-ads/>)

Ultrassom

Blog no WordPress.com. | Tema: iTheme2 por Themify.