

WP3-CETELI-3-IoT

Atividade Prática	Experimento06_tbeam_microfone	Período	07 de agosto de 2023
Monitor	Diego Alves Amoedo	Módulo	1
Professor	Andrey Ruben Rieiro Bessa		

Experimento de Monitoramento de Som usando Arduino UNO, KY-038 Microfone e Plotter Serial do Arduino IDE**Sumário**

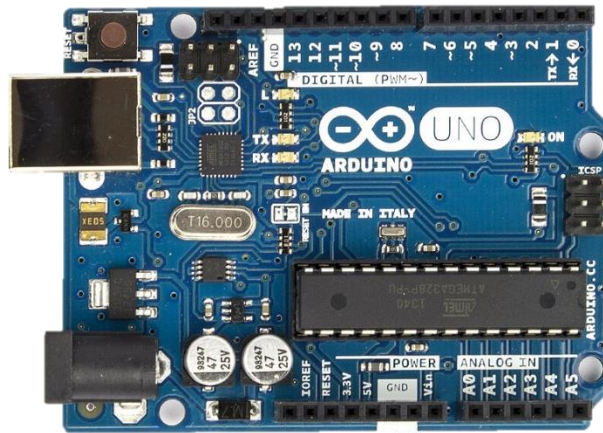
1. OBJETIVOS 1
2. INFORMAÇÃO TEÓRICA **Erro! Indicador não definido.**
3. SOFTWARE ARDUINO IDE – NOÇÕES BÁSICAS ... **Erro! Indicador não definido.**

1. OBJETIVOS

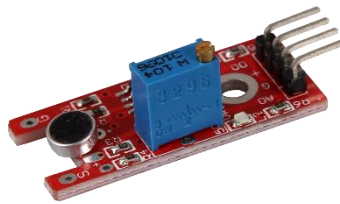
O objetivo deste experimento é utilizar o Arduino UNO em conjunto com o módulo de microfone KY-038 para monitorar os níveis de som ambiente. O experimento visa capturar dados do microfone, tanto em formato analógico quanto digital, e exibi-los em tempo real no Plotter Serial do Arduino IDE. O propósito é entender como o microfone responde a diferentes níveis de som e observar os valores de leitura no Plotter Serial.

2. MATERIAIS

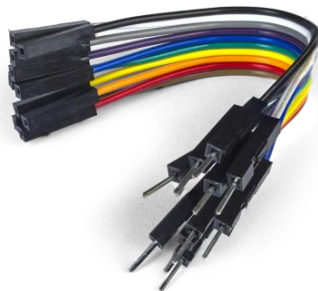
- Arduino MEGA (ou placa compatível)



- Módulo de microfone KY-038



- Cabos de conexão



3. CONFIGURAÇÃO DO AMBIENTE

Antes de começar, é necessário configurar o ambiente de desenvolvimento. Siga os passos abaixo:

- a) Faça o download e a instalação da IDE do Arduino no site oficial (<https://www.arduino.cc/en/software>).
- b) Conecte o Arduino ao computador utilizando um cabo USB.
- c) Abra a IDE do Arduino e verifique se o Arduino é reconhecido corretamente nas portas disponíveis (no menu "Ferramentas" > "Porta").

4. IMPLEMENTAÇÃO DO SCRIPT

O código para este experimento será fornecido previamente. Certifique-se de copiar o código corretamente e salvá-lo com uma extensão ".ino". O código utiliza os pinos A0 (para leitura analógica) e 3 (para leitura digital) para se comunicar com o módulo de microfone KY-038.

```
1 // Projeto: SUPER - Projeto para Educação e Pesquisa 2023
2 // Task: WP3-CETELI-3-IoT
3 // Instrutor: Andrey Bessa
4
5
6 // Defina os pinos do módulo de microfone
7 #define MIC_ANALOG_PIN A0
8 #define MIC_DIGITAL_PIN 3
9
10 void setup() {
11     Serial.begin(115200);
12     pinMode(MIC_ANALOG_PIN, INPUT);
13     pinMode(MIC_DIGITAL_PIN, INPUT);
14 }
15
16 void loop() {
17     int micAnalogValue = analogRead(MIC_ANALOG_PIN);
18     int micDigitalValue = digitalRead(MIC_DIGITAL_PIN);
19
20     // Envia os valores para o Plotter Serial
21     // Serial.print("\t ");
22     // Serial.print(micAnalogValue);
23
24     Serial.print(" ");
25     Serial.println(micAnalogValue);
26
27     delay(50); // Pequeno atraso para evitar leituras muito rápidas
28 }
```

5. EXPLICAÇÃO DO CÓDIGO

O código inicia a comunicação serial a uma taxa de 115200 bps e define os pinos do módulo de microfone como entradas. Em seguida, no loop principal, ele realiza leituras analógicas e digitais do microfone. Os valores lidos são enviados para o Plotter Serial do Arduino IDE usando a função "Serial.print()".

6. MONTAGEM DO CIRCUITO

Conecte o módulo de microfone KY-038 ao Arduino UNO utilizando cabos jumper. Conecte o pino analógico do KY-038 ao pino A0 do Arduino e o pino digital ao pino 3 do Arduino. Certifique-se de conectar corretamente os fios para obter leituras precisas.

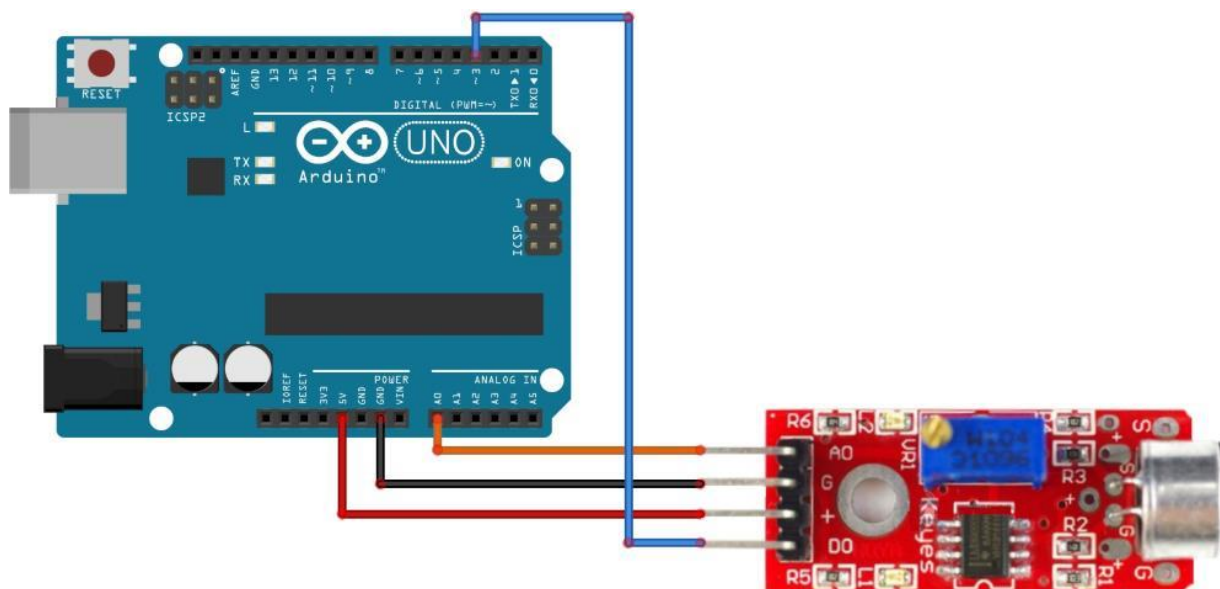


Figura 1 - Interligação dos componentes

7. COMPILAÇÃO E UPLOAD DO CÓDIGO

Verifique se não há erros no código e pressione o botão "Verify" (ou "Compile") no Arduino IDE. Em seguida, faça o upload do código para o Arduino UNO pressionando o botão "Upload".

8. EXECUÇÃO E OBSERVAÇÃO DOS RESULTADOS

Após o upload do código, abra o "Monitor Serial" no Arduino IDE e verifique se os valores analógicos e digitais estão sendo impressos corretamente. Em seguida, clique no botão "Plotter Serial" para visualizar os valores em tempo real no gráfico do Plotter Serial. Observe como os valores variam de acordo com o som ambiente capturado pelo microfone.

Com este experimento, os alunos poderão entender como o microfone KY-038 responde aos diferentes níveis de som e como utilizar o Plotter Serial do Arduino IDE para visualizar dados em tempo real. Eles também terão a oportunidade de explorar outras funcionalidades do Arduino UNO e adaptar o experimento para diferentes aplicações.