

ESPECIFICAÇÕES TÉCNICAS DO PROJETO DE MONITORAMENTO AMBIENTAL

**PROJETO DE MONITORAMENTO AMBIENTAL
FESA – Faculdade Engenheiro Salvador Arena**

Aline Cristina Ribeiro de Barros – RA: 081230021
Luis Gustavo de Oliveira Carneiro – RA: 081230029
Roger Rocha da Silva – RA: 081230045
João Victor Pereira Andrade – RA: 081230010
Ezequiel Rodrigues Pereira – RA: 081230008

Março, 2025

Sumário

1. Visão Geral	3
2. Componentes Utilizados	4
3. Funcionalidades	5
4. Especificações Técnicas Detalhadas	6
4.1 Sensor DHT22	6
4.2 Sensor LDR	6
4.3 Display LCD 16x2	6
4.4 LEDs	6
4.5 Buzzer	7
4.6 Botão	7
4.7 EEPROM	7
4.8 RTC DS1307	7
5. Lógica de Funcionamento	8
6. Bibliotecas Utilizadas	9
7. Esquema de Conexões	10
8. Considerações Finais	11

1. Visão Geral

O projeto é um sistema de monitoramento ambiental baseado em Arduino Uno para coletar e exibir dados ambientais, incluindo temperatura, umidade e luminosidade. Os dados são apresentados em um display LCD e armazenados na EEPROM para consulta futura. O sistema também conta com alertas visuais e sonoros para indicar condições fora dos limites aceitáveis.

2. Componentes Utilizados

Microcontrolador:

- Arduino Uno.

Sensores:

- DHT22 (Temperatura e Umidade);
- LDR (Sensor de Luminosidade).

Atuadores:

- LEDs (Vermelho, Amarelo, Verde);
- Buzzer.

Exibição:

- Display LCD 16x2 com interface I2C;

Armazenamento:

- EEPROM;
- RTC DS1307 para registro de data e hora.

Outros:

- Pushbutton para alternância de exibição;
- Resistores de 1000Ω para LEDs;
- Breadboard para conexão dos componentes.

3. Funcionalidades

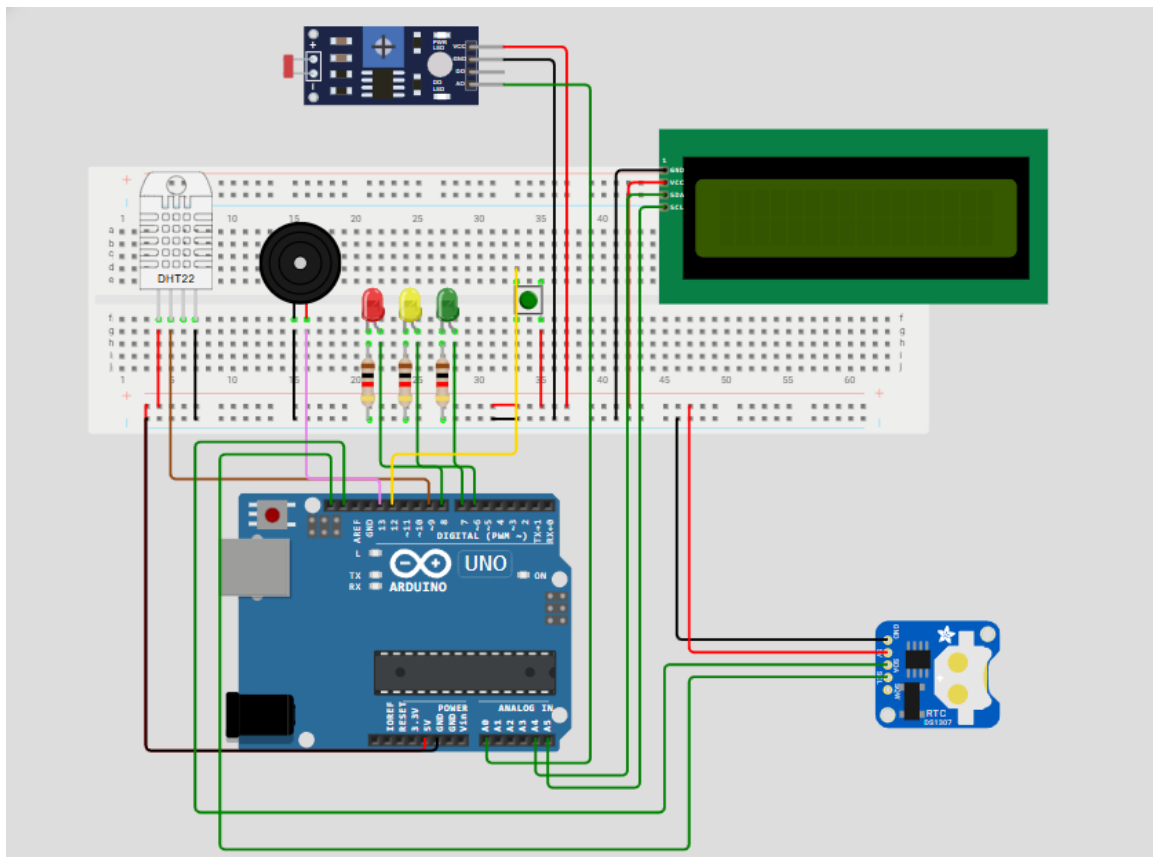


Figura 1 - Configuração do sistema de monitoramento ambiental com Arduino Uno e sensores.

O sistema realiza a leitura de dados de temperatura, umidade e luminosidade por meio dos sensores DHT22 e LDR, respectivamente. Esses dados são processados e exibidos em um display LCD 16x2, permitindo ao usuário alternar entre as informações de temperatura, umidade e luminosidade utilizando um botão. O sistema também emite alertas visuais (através de LEDs) e sonoros (com um buzzer) quando os valores medidos estão fora dos limites pré-definidos. Além disso, os dados são armazenados na EEPROM sempre que estão fora dos limites, permitindo a consulta histórica dos registros.

4. Especificações Técnicas Detalhadas

A seguir, são apresentadas as especificações dos principais componentes utilizados no projeto:

4.1 Sensor DHT22 (Temperatura e Umidade)

- Função: Mede temperatura e umidade do ambiente.
- Faixa de medição:
 - Temperatura: -40°C a 80°C.
 - Umidade: 0% a 100%.
- Precisão:
 - Temperatura: $\pm 0,5^{\circ}\text{C}$.
 - Umidade: ± 2 a 5%.
- Conexão: Utiliza pino digital 9 do Arduino.

4.2 Sensor LDR (Luminosidade)

- Função: Mede a intensidade da luz ambiente.
- Faixa de medição: 0% a 100% (mapeado de 0 a 1023).
- Conexão: Pino A0 do Arduino.

4.3 Display LCD 16x2 (Interface I2C)

- Função: Exibir informações coletadas dos sensores.
- Interface: Comunicação via I2C.
- Conexão:
 - SDA: Pino A4.
 - SCL: Pino A5.

4.4 LEDs Indicadores

- Função: Alertar visualmente sobre as condições ambientais.
- Cores e Significado:
 - Verde: Condições normais.
 - Amarelo: Temperatura fora do intervalo seguro (15°C a 25°C).
 - Vermelho: Umidade fora do intervalo seguro (30% a 50%).
- Conexões:
 - LED Verde: Pino 6.
 - LED Amarelo: Pino 7.
 - LED Vermelho: Pino 8.

4.5 Buzzer

- Função: Emitir alertas sonoros quando valores ultrapassam os limites estabelecidos.
- Conexão: Pino 13 do Arduino.

4.6 Botão (Pushbutton)

- Função: Alternar entre as medições exibidas no LCD.
- Conexão: Pino 12 do Arduino.

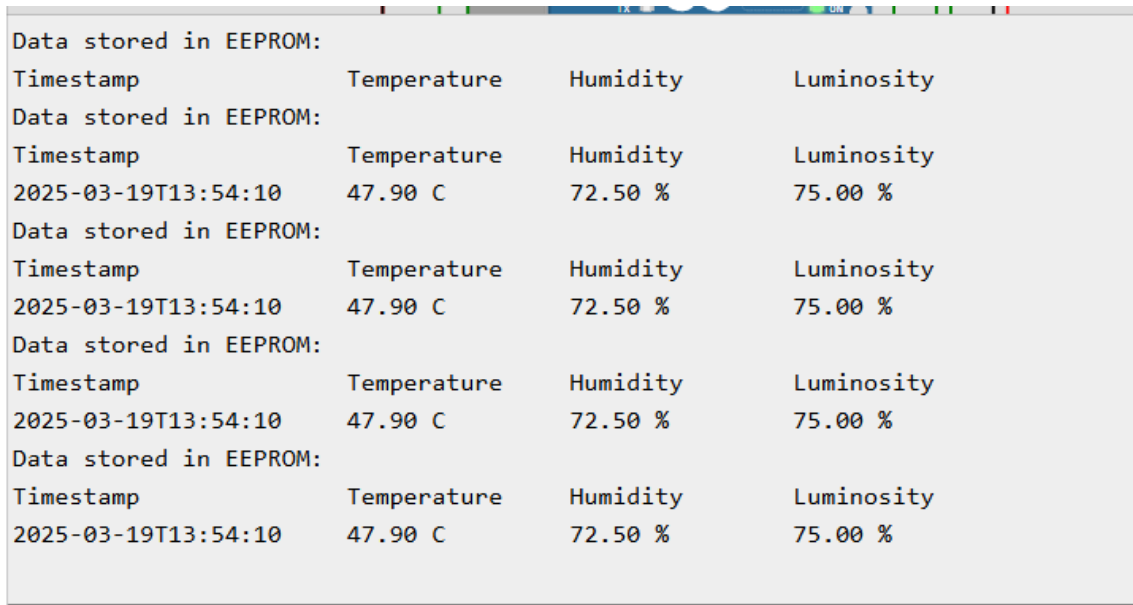
4.7 EEPROM (Memória não volátil)

- Função: Armazenar dados críticos quando fora dos limites estabelecidos.
- Capacidade: Até 100 registros.
- Estrutura de armazenamento:
 - 10 bytes por registro, contendo:
 - Timestamp (data e hora).
 - Temperatura.
 - Umidade.
 - Luminosidade.

4.8 RTC DS1307 (Relógio de Tempo Real)

- Função: Registrar data e hora de cada leitura.
- Interface: Comunicação via I2C.
- Conexão:
 - SDA: Pino A4.
 - SCL: Pino A5.

5. Lógica de Funcionamento



Data stored in EEPROM:			
Timestamp	Temperature	Humidity	Luminosity
Data stored in EEPROM:			
Timestamp	Temperature	Humidity	Luminosity
2025-03-19T13:54:10	47.90 C	72.50 %	75.00 %
Data stored in EEPROM:			
Timestamp	Temperature	Humidity	Luminosity
2025-03-19T13:54:10	47.90 C	72.50 %	75.00 %
Data stored in EEPROM:			
Timestamp	Temperature	Humidity	Luminosity
2025-03-19T13:54:10	47.90 C	72.50 %	75.00 %
Data stored in EEPROM:			
Timestamp	Temperature	Humidity	Luminosity
2025-03-19T13:54:10	47.90 C	72.50 %	75.00 %

Figura 3 - Exibição dos registros de temperatura, umidade e luminosidade salvos.

Durante a inicialização, o sistema exibe uma mensagem no LCD e toca uma melodia no buzzer. O RTC é configurado para a data e hora atuais. Em seguida, o sistema realiza a leitura de dados dos sensores a cada segundo, calculando a média das últimas 10 leituras para temperatura e umidade. Os dados são exibidos no LCD, com a possibilidade de alternar entre as informações utilizando o botão. Caso os valores estejam fora dos limites, os alertas são acionados, com os LEDs e o buzzer indicando a condição anormal. Os dados fora dos limites são armazenados na EEPROM com um timestamp para consulta futura.

6. Bibliotecas Utilizadas

O projeto utiliza diversas bibliotecas para garantir o funcionamento correto dos sensores, do display e do armazenamento de dados. Abaixo, estão as bibliotecas utilizadas e suas respectivas funções:

- **LiquidCrystal_I2C**: Responsável pelo controle do display LCD 16x2 via comunicação I2C, permitindo a exibição das informações coletadas pelos sensores.
- **DHT**: Biblioteca utilizada para fazer a leitura dos dados do sensor **DHT22**, que mede a temperatura e a umidade do ambiente.
- **RTClib**: Gerencia o módulo **RTC DS1307**, que mantém a data e hora atualizadas, garantindo que os registros armazenados na EEPROM sejam marcados corretamente.
- **EEPROM**: Permite o armazenamento dos dados coletados na memória não volátil do Arduino, garantindo que as informações não sejam perdidas após um reset ou desligamento do sistema.

Além disso, essas bibliotecas oferecem métodos e funções específicas que facilitam a implementação do código, reduzindo a complexidade do desenvolvimento e melhorando a eficiência do sistema de monitoramento ambiental.

7. Esquema de Conexões

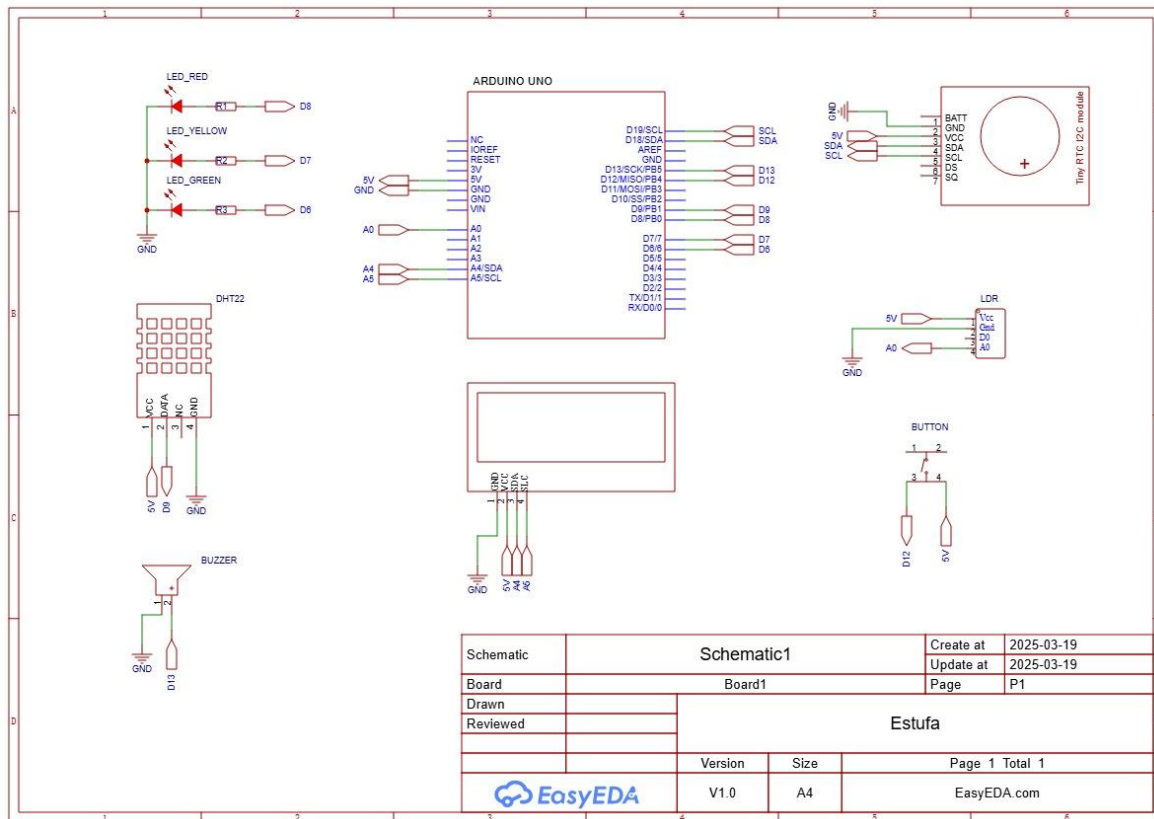


Figura 2 - Representação das conexões elétricas do projeto no EasyEDA.

O esquema de conexões define a interligação dos componentes ao Arduino Uno, assegurando a comunicação correta entre sensores, atuadores e módulos.

- I2C: Utilizado pelo display LCD 16x2 e RTC DS1307 nos pinos A4 (SDA) e A5 (SCL);
- Pinos Digitais: Controlam LEDs, buzzer e botão de navegação;
- Pinos Analógicos: Leitura do sensor LDR para medir luminosidade.

O arquivo diagram.json pode ser visualizado no Wokwi para mais detalhes.

8. Considerações Finais

Este projeto fornece uma solução robusta para monitoramento ambiental, com capacidade de alerta e registro histórico. Ele pode ser facilmente expandido para incluir mais sensores ou funcionalidades, como envio de dados para a nuvem ou integração com outros sistemas. A combinação de hardware e software utilizada garante precisão e confiabilidade nas medições, tornando-o uma ferramenta útil para aplicações de monitoramento em tempo real.