

**Bernardo – 565776**

**Cauã Muniz – 566527**

**João Vitor Anceloti – 563473**

**Matheus – 562765**

**Rafael Ferreira -563285**

**Projeto: Controle de Energia Inteligente com 2 Cargas Prioritárias**

Relatório técnico apresentado ao curso, Ciência da Computação — Turma 1CCPI da FIAP Paulista, orientado pelo prof. Álvaro Alexandre Rezende Gonçalves, como requisito parcial obtenção do título de cientista da computação.

**PAULISTA**

**2025**

## RESUMO

O projeto “Escola Inteligente Sustentável” propõe uma solução inovadora para reduzir o consumo de energia elétrica em escolas públicas por meio da integração de energia solar fotovoltaica e automação inteligente. A proposta utiliza **painéis solares** conectados a um inversor híbrido GoodWe, com possibilidade de uso de **baterias** inteligentes para garantir autonomia mesmo em quedas de energia.

## SUMÁRIO

### Sumário

<b>1 INTRODUÇÃO .....</b>	<b>4</b>
2. Objetivos.....	5
4. Funcionamento do Sistema .....	7
6. Código fonte .....	8
7. Resultados Esperados.....	10

## **1 INTRODUÇÃO**

Este projeto tem como objetivo desenvolver um sistema de automação para priorização de cargas com base na simulação do nível de energia de uma bateria. Através de um potenciômetro, representando a carga da bateria, o sistema decide automaticamente quais dispositivos (LEDs) serão ativados, priorizando os mais essenciais em momentos de baixa energia.

## 2. Objetivos

### Objetivo geral:

Criar uma prova de conceito funcional de um sistema de gerenciamento inteligente de energia com priorização de cargas.

### Objetivos Específicos:

- simular o nível da bateria com um potenciômetro;
- controlar o acionamento de dois LEDs representando cargas prioritárias;
- utilizar um display LCD para exibir o nível de energia em tempo real;
- justificar o uso de automação como solução para eficiência energética.

### 3. Materiais Utilizados

Componente	Quantidade	Função
Arduino uno	1	Microcontrolador
Potenciômetro	1	Simular o nível da bateria
LED azul (primário)	1	Representar carga prioritária
LED roxo (secundário)	1	Representar carga menos prioritária
Resistor 220 $\Omega$	2	Limitar Corrente dos LEDs
Display LCD 16x2 (I2C)	1	Exibir o nível de energia
Protoboard	1	Montagem do circuito
Jumpers	Vários	Conexão dos componentes

#### 4. Funcionamento do Sistema

- O **potenciômetro** é lido pelo pino analógico A0 e representa o nível de energia da bateria (0 a 1023).
- O **LED primário** acende quando o nível é médio (>500) e permanece aceso com níveis altos.
- O **LED secundário** só acende quando o nível é alto (>800).
- O **LCD 16x2** exibe o nível de energia em tempo real.
- Se nível > 800:
  - Aciona LED primário e LED secundário
- Senão se nível > 500:
  - Aciona apenas LED primário
- Senão:
  - Desliga ambos os LEDs

## 5. Simulação no Wokwi

Link da simulação:

<https://wokwi.com/projects/433395785548565505>

## 6. Código fonte

```

1  #include <Wire.h>
2  #include <LiquidCrystal_I2C.h>
3
4  const int potenciometro = A0;
5  const int ledprimario = 9;
6  const int ledsecundario = 3;
7
8  LiquidCrystal_I2C lcd(0x27, 16, 2);
9
10 void setup() {
11
12     pinMode(ledprimario , OUTPUT);
13     pinMode(ledsecundario, OUTPUT);
14     Serial.begin(9600);
15
16     lcd.init();           // Inicializa o LCD
17
18     lcd.backlight();      // Liga a luz de fundo
19
20 }
21
22 void loop() {
23
24     int nivel = analogRead(potenciometro);
25     float porcentagem = nivel * 100.0 / 1023.0; // Converte para %
26
27     Serial.print("Nível: ");
28     Serial.print(nivel);
29     Serial.print(" -> ");
30     Serial.print(porcentagem, 1);
31     Serial.println("%");
32
33     // Lógica das cargas
34
35     if (nivel > 800) {
36
37         digitalWrite(ledprimario , HIGH);
38         digitalWrite(ledsecundario , HIGH);
39
40     } else if(nivel > 500) {
41
42         digitalWrite(ledprimario, HIGH);
43         digitalWrite(ledsecundario, LOW);
44

```



```
} else {  
    digitalWrite(ledprimario , LOW);  
    digitalWrite(ledsecundario, LOW);  
}  
  
// Atualiza o LCD  
  
lcd.setCursor(0, 0);  
  
lcd.print("Bateria: ");  
lcd.print(percentagem, 0);  
  
lcd.print("%    ");  
lcd.setCursor(0, 1);  
  
lcd.print("C1:");  
lcd.print((nivel > 500) ? "ON " : "OFF");  
lcd.print(" C2:");  
  
lcd.print((nivel > 800) ? "ON " : "OFF");  
  
delay(500);  
}
```




## 7. Resultados Esperados

- Em níveis baixos de energia, apenas o LED primário acende.
- Em níveis altos, os dois LEDs acendem.
- O LCD mostra o valor analógico lido em tempo real.
- O sistema é uma simulação funcional de um controle de carga inteligente.

## **8. Conclusão**

O projeto atingiu todos os objetivos propostos. Foi possível simular de forma clara e funcional o conceito de priorização de cargas com automação. A simplicidade e eficiência do sistema mostram sua viabilidade como solução em sistemas reais de energia inteligente.

## 9. Links

-  Wokwi: [Simulação do circuito](#)
-  GitHub: <https://github.com/MUnizZz526/controle-Energia-Inteligente/tree/main#>
-  Vídeo YouTube: [\[https://www.youtube.com/watch?v=QxXpvk3aInQ\]](https://www.youtube.com/watch?v=QxXpvk3aInQ)