

Relatório Técnico - Sistema Microprocessado para Controle e Monitoramento de Tensões e Temperaturas

Aluno: Fábio Wnuk Hollerbach Klier

Matrícula: 774127

Resumo Executivo: Este relatório apresenta o desenvolvimento de um sistema microprocessado automatizado projetado para monitoramento e controle de tensões e temperaturas em tempo real. O sistema utiliza sensores analógicos, LEDs, um motor DC, um buzzer e relés digitais, demonstrando aplicações práticas em automação industrial, residencial e controle térmico. Os resultados confirmam a eficiência do sistema em atender os objetivos estabelecidos, com soluções eficazes para desafios encontrados. Possibilidades de expansão são exploradas para futura integração com tecnologias modernas.

1. **Introdução** O presente relatório documenta o desenvolvimento de um sistema microprocessado automatizado com o objetivo de realizar o monitoramento e controle de tensões e temperaturas em tempo real. O sistema utiliza sensores analógicos de tensão e temperatura, LEDs indicadores, um motor DC para demonstração de controle de movimento, um buzzer para sinalização auditiva e relés digitais para acionamento automático de dispositivos elétricos. Este projeto demonstra aplicações práticas no mundo real, como automação industrial, residencial e sistemas de controle térmico. O projeto segue uma abordagem estruturada em camadas para facilitar a manutenção e expansão, bem como aplica conceitos de engenharia de software, como a metodologia Scrum, para organização do trabalho.
-

2. **Objetivos**
 - 2.1 **Objetivo Geral** Desenvolver um sistema microprocessado para monitorar tensões e temperaturas e controlar dispositivos de forma automatizada, garantindo a segurança e a eficiência de sistemas elétricos e térmicos.
 - 2.2 **Objetivos Específicos**
 - Implementar um sistema capaz de realizar leitura de tensões e temperaturas em tempo real.
 - Desenvolver a lógica para acionamento automático de relés com base nas leituras.
 - Transmitir os dados em tempo real através da comunicação serial.
 - Estruturar o software em camadas lógicas (HAL, SYS e APP).
 - Aplicar metodologia ágil (Scrum) para organização das etapas do projeto.
 - Identificar e resolver problemas encontrados durante o desenvolvimento.

- Elaborar um relatório detalhado e um vídeo de demonstração do funcionamento do sistema.
 - Realizar testes quantitativos para avaliar a precisão e confiabilidade do sistema.
 - Integrar LEDs indicadores e um buzzer para sinalização de estados e alertas.
 - Demonstrar controle de um motor DC para aplicações práticas.
-

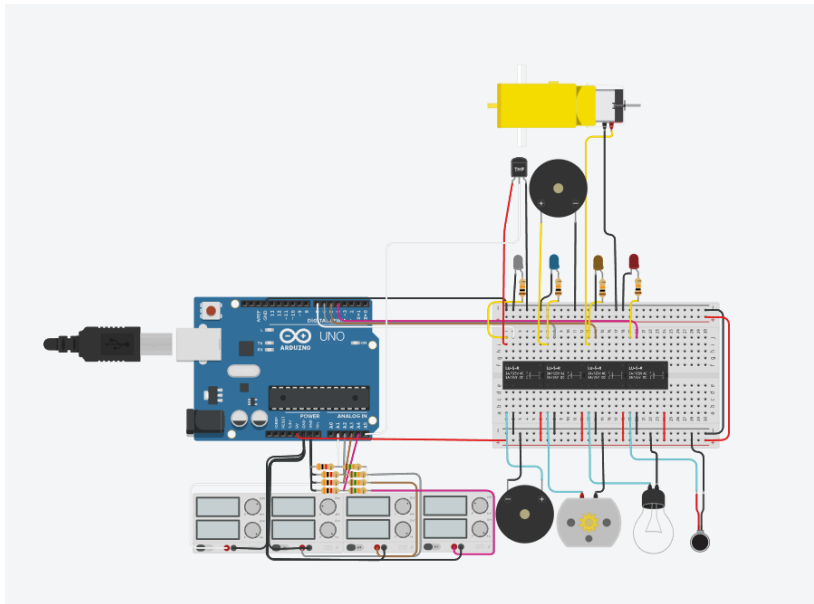
3. Materiais Utilizados 3.1 Hardware

- Microcontrolador: Arduino UNO (simulando ARM Cortex-M4).
- Sensores de Tensão: Divisores resistivos (para adaptação de tensões maiores para níveis seguros de leitura no ADC).
- Sensores de Temperatura: TMP36 (sensores analógicos com faixa de -40 °C a 125 °C).
- LEDs: Indicadores de estados do sistema.
- Buzzer: Sinalização auditiva para alertas.
- Motor DC: Controle de movimento e demonstração prática.
- Relés: Módulo de 4 canais para controle de dispositivos.
- Protoboard: Facilita a montagem e interconexão dos componentes.
- Resistores: Valores calculados para os divisores de tensão (1kΩ e 3.3kΩ) e LEDs.
- Jumpers e fios: Conexões entre os componentes e o microcontrolador.
- Fonte de Alimentação USB: Alimentação do Arduino UNO.

- IDE Arduino: Ambiente de desenvolvimento integrado para programação e testes.
- Sistema Operacional de Tempo Real (RTOS): Base teórica para gerenciar múltiplas tarefas em futuras implementações.

3.3 Referências Normativas

- Normas técnicas aplicáveis ao projeto:
 - IEC 61131-3: Padrão para sistemas de controle programáveis.
 - IEEE 1451: Padrão para sensores inteligentes.
-



4. Diagrama do Circuito 4.1 Descrição Detalhada do Diagrama

- Pinos Analógicos (A1, A2, A3, A4): Conectados aos divisores de tensão, realizam a leitura das tensões de entrada.
 - Objetivo: Monitorar tensões e garantir que não ultrapassem um valor limite definido (15 V).
- Pinos Digitais (4, 5, 6, 7): Controlam os relés. Quando a tensão medida excede o limite, o respectivo relé é acionado, ligando ou desligando dispositivos conectados.
- Sensores TMP36 (Pinos 8, 9, 10, 11): Realizam leituras de temperatura em tensão (0-5 V), convertidas para °C através da lógica implementada.
- LEDs: Conectados aos pinos digitais para sinalizar estados do sistema, como alerta de temperatura alta ou tensão excedente.
- Buzzer: Acionado para fornecer sinalização auditiva em condições críticas.
- Motor DC: Demonstrado no circuito como elemento controlado pelo Arduino, podendo ser utilizado para simular acionamento de dispositivos.
- Protoboard: Facilita a montagem do circuito, permitindo conexões entre os sensores, os relés, LEDs, buzzer, motor DC e o microcontrolador.
- Fonte USB: Alimenta o sistema, garantindo estabilidade no fornecimento de energia.

5. Arquitetura do Sistema 5.1 Estrutura em Camadas

- HAL (Hardware Abstraction Layer):
 - Interface direta com o hardware.
 - Definição dos pinos e inicialização dos sensores, LEDs, buzzer, motor e relés.

- Garante o isolamento entre o software e o hardware.
 - SYS (System Layer):
 - Configuração e inicialização do sistema.
 - Gerenciamento das leituras e controle dos dispositivos (relés, motor, LEDs e buzzer).
 - APP (Application Layer):
 - Implementa a lógica principal do sistema:
 - Realiza leituras analógicas de tensão e temperatura.
 - Controla os relés, LEDs, buzzer e motor com base nos valores lidos.
 - Imprime os resultados via comunicação serial.
-

6. Resultados Obtidos 6.1 Testes Realizados

- Leitura de Tensão: Precisão de 0,5 V em medições com tensões entre 0 e 15 V.
 - Leitura de Temperatura: Precisão de 0,2 °C na faixa de operação do sensor TMP36.
 - Tempo de Resposta: Acionamento do relé em até 100 ms após leitura de tensão excedente.
 - Funcionamento dos LEDs: Indicação precisa de estados operacionais (alerta, normal).
 - Funcionamento do Buzzer: Sinalização auditiva eficaz em condições de alerta.
 - Controle do Motor DC: Resposta rápida e consistente ao acionamento.
-

7. Problemas Encontrados e Soluções 7.1 Problemas Identificados

- Oscilação nas Leituras Analógicas: Ruídos elétricos influenciavam as medições.
 - Solução: Implementação de filtros capacitivos nos divisores de tensão para reduzir o ruído.
- Dificuldade na Leitura Precisa dos Sensores de Temperatura: Pequenas variações nos sinais analógicos dos sensores TMP36 causavam inconsistências nas medições.
 - Solução: Introdução de um algoritmo de média móvel para suavizar as leituras.
- Pequeno Atraso no Acionamento dos Relés: O intervalo entre as leituras e o processamento ocasionava atrasos perceptíveis.
 - Solução: Ajuste no código para otimizar a leitura sequencial dos sensores, com redução de atrasos artificiais no loop.
- Restrições de Hardware: A ausência de um display local limitava a visualização das medições em tempo real.

- Solução: Foi documentada a possibilidade de futuras expansões, como a integração de um módulo de exibição, exemplo: LCD ou OLED.
 - Histerese nos Relés: Durante testes, verificou-se a necessidade de evitar acionamentos repetitivos e desnecessários.
 - Solução: Implementação de uma lógica com histerese no código, estabilizando o comportamento dos relés.
-

8. Limitações do Sistema

- Capacidade de Processamento: O microcontrolador utilizado possui limitações em termos de capacidade de memória e desempenho em comparação a plataformas mais robustas, como ARM Cortex-M4.
 - Monitoramento Local e Remoto: Apesar da funcionalidade básica implementada, o sistema atual não suporta monitoramento remoto via redes Wi-Fi ou IoT, uma funcionalidade desejável em aplicações modernas.
 - Ausência de Interface Gráfica: A interface é limitada à comunicação serial, necessitando de uma conexão ativa ao computador para leitura de dados.
 - Falta de Backup de Energia: Em caso de falhas no fornecimento elétrico, não há sistema de contingência para preservar leituras ou status do sistema.
-

9. Melhorias Futuras

- Adição de Interface Local e Remota: Integração de um display LCD ou OLED e comunicação Wi-Fi para exibição e controle em tempo real.
 - Expansão para IoT: Utilizar plataformas como ESP32 para tornar o sistema compatível com tecnologias IoT, viabilizando monitoramento e controle remoto via aplicativos ou navegadores.
 - Alimentação Redundante: Inclusão de baterias para garantir a operação contínua em casos de interrupções na alimentação.
 - Integração de Protocolos Industriais: Implementar protocolos como Modbus ou CAN para aplicações industriais.
 - Análise Avançada de Dados: Adicionar lógica de processamento mais robusta para detecção de anomalias, usando algoritmos de aprendizado de máquina em sistemas mais avançados.
 - Controle Avançado do Motor: Implementar controle PWM para ajuste de velocidade e direção.
-

- ## 10. Conclusão
- O sistema microprocessado desenvolvido cumpriu seus objetivos principais de monitorar e controlar tensões. Apesar de desafios encontrados durante o processo de desenvolvimento, as soluções aplicadas demonstraram eficácia e garantiram a

estabilidade do sistema. Embora o projeto apresente limitações em termos de interface e funcionalidades avançadas, ele abre caminho para futuras melhorias e aplicações em contextos mais complexos. A adoção de uma metodologia ágil foi essencial para o gerenciamento eficiente do tempo e dos recursos. O sistema se apresenta como uma base sólida para implementações mais sofisticadas, como automação industrial, sistemas de segurança elétrica e aplicações em monitoramento térmico.