

CONTROLADOR DE TEMPERATURA COM INTERFACE COM O USUÁRIO POR COMPUTADOR

Lucas Nascimento Candido¹, Marcelo Antonio Pavanello²
^{1,2} Departamento de Engenharia Elétrica, Centro Universitário FEI
 lucascandido04@gmail.com, pavanello@fei.edu.br

Resumo: Esse trabalho tem por objetivo o desenvolvimento de uma ferramenta de *software* capaz de realizar a leitura e controle de temperatura em um sistema eletrônico, permitindo a realização de medidas em componentes eletrônicos desde a temperatura ambiente até 300°C. O *software* realizará a programação do microcontrolador, responsável pela leitura e controle da temperatura. O software desenvolvido também gerenciara a interface com o usuário, em ambiente computacional, recebendo os dados medidos e os apresentados graficamente.

1. Introdução

A operação de dispositivos eletrônicos em altas temperaturas está associada a uma série de aplicações, podendo superar os 600 K [3]. Quando submetidos a temperaturas superiores a temperatura ambiente, os dispositivos eletrônicos sofrem uma degradação de suas propriedades elétricas, que se intensifica em maiores temperaturas. No caso de transistores MOS, ocorre um aumento na corrente de fuga das junções PN, redução da tensão de limiar, aumento da inclinação de sub-limiar e redução da mobilidade dos portadores de carga, entre outros efeitos [2,5]. Essas mudanças são importantes para a compreensão dos efeitos físicos associados, e determinar os limites de utilização de uma certa tecnologia e incluí-los em modelos compactos.

A realização de medidas elétricas em dispositivos e circuitos eletrônicos em nível de circuito integrado (CI), é uma etapa fundamental para a produção de conhecimento sobre novas tecnologias e validação dos circuitos projetados. Essas medidas são realizadas em equipamentos denominados de microprovidores, que possuem pontas de prova na forma de agulhas, para acesso dos terminais dessas tecnologias, que são posicionados em uma superfície plana, de ouro, cobre ou titânio, denominada porta-amostras.

Para a obtenção de curvas características dos transistores MOS em função da temperatura, são utilizados sistemas eletrônicos dedicados, que permitem o aquecimento ou resfriamento da amostra, conforme a seleção do usuário, e objetivam garantir que a temperatura do porta-amostras seja uniforme em toda a sua superfície, evitando o aparecimento de gradientes térmicos (também denominado de ruído térmico).

O Centro Universitário FEI possui um equipamento para a realização dessas medidas elétricas, que opera em ambiente fechado, a vácuo e com um porta-amostras retangular, de 10 mm x 15 mm, o que limita o seu emprego em amostras com área maiores. Além disso, a necessidade do posicionamento da amostra em ambiente de vácuo limita bastante sua substituição, devido a posterior retomada do ambiente de vácuo, aumentando o tempo para produção de resultados.

2. Software

Para dar início a escolha de componentes, linguagem de programação e outros requisitos do projeto de pesquisa, foram realizados diagramas para auxiliar a linha de pensamento, e entender melhor cada bloco do *software* e suas finalidades de acordo com os componentes que seriam utilizados, o que contribuiu para o desenvolvimento de um planejamento e divisão do projeto por etapas, sendo elas três:

1ª Etapa – Teste de comunicação com periféricos

2ª Etapa – Verificação de *Inputs* e exibição dos pontos

3ª Etapa – Malha de controle de temperatura

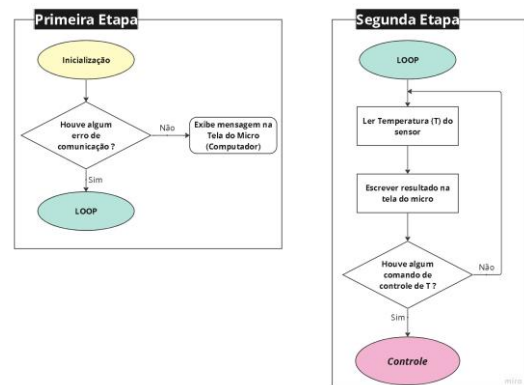


Figura 1 – Diagramas de funcionalidades do *software*

3. Componentes

A escolha dos componentes do projeto, foi determinada pela análise da arquitetura do controlador de temperaturas existente atualmente na FEI, o modelo K-20, especificamente a análise do seu microcontrolador e seus periféricos responsáveis pela leitura de temperatura e aquecimento, levando em conta o nível de processamento do microcontrolador e resolução dos componentes de medição e aquecimento.

Com isso foi determinado o uso do microcontrolador ESP32, devido ao seu poder de processamento, e toda sua arquitetura com diversos protocolos de comunicação e interfaces de desenvolvimento de software frequentemente utilizadas no mercado [6]. E ainda dois periféricos, o conversor digital-analógico MCP4725, com 12 bits de resolução e protocolo de comunicação I2C [1], e o conversor analógico-digital MAX31865 que apresenta 16 bits/14 bits de resolução, protocolo de comunicação SPI além da compatibilidade com o sensor de temperatura PT100 [4,7].

4. Metodologia

Com a definição das funcionalidades do *software* e dos componentes foi possível selecionar o tipo de programação a ser trabalhada nesse projeto, e analisando as linguagens de programação que a placa de

desenvolvimento ESP32 contemplava, optou-se por utilizar a linguagem C com a IDE do Arduino, dada a quantidade de conteúdo e projetos existentes para estudo e verificação acerca dessa linguagem.

Sabendo dos protocolos de comunicação utilizados pelos conversores, foram necessárias bibliotecas específicas de cada componente ou do protocolo vigentes, para realizar as primeiras etapas de verificação de comunicação, onde através das funções contidas nas bibliotecas se tornava possível verificar a comunicação. Em seguida com a bibliotecas e instruções lógicas de programação foi possível implementar o código com a segunda etapa de leitura e exibição dos dados, onde seria possível fazer a verificação da porta serial afim de verificar se houve algum *input* do operador.

5. Resultados

Para coleta dos dados foi utilizado um programa chamado CoolTerm, que permite a comunicação serial via USB e Bluetooth com outros dispositivos, permitindo a leitura e envio de dados, além de exibir e armazenar em arquivos de texto.

Com o emprego dessa ferramenta junto ao *software* desenvolvido, foi possível a realização do teste do sistema de aquecimento e monitoramento, obtendo valores de tempo de leitura, temperatura e tensão, que foram tratados pelo software Microsoft Excel para gerar as curvas de comportamento do sistema, verificado na Figura 1, e assim obter as primeiras análises do sistema integrado com a parte de *software* e *hardware*.

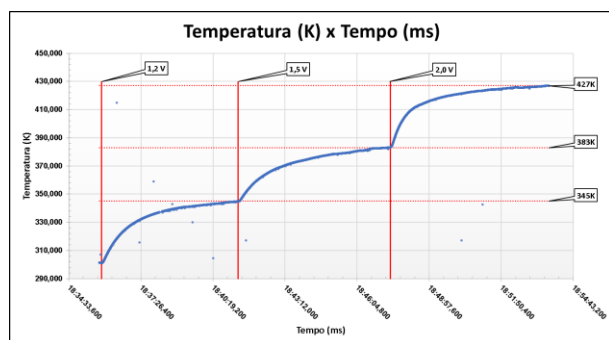


Figura 2 – Gráfico de temperatura em Kelvin em função do tempo.

A curva obtida através da simulação do sistema, se deu pelo seu estado inicial sem alimentação do sistema de potência do aquecedor (0 volts), onde o programa faria somente a varredura da temperatura do porta amostra, em seguida foi inserido um valor de 1,2V via porta serial e com isso a temperatura começou a subir até que estabilizou em uma temperatura aproximadamente de 345K, e em um segundo instante foi inserido uma valor de 1,5V alcançando uma temperatura estabilizada de aproximadamente 383K e por fim inserindo um último valor de 2V chegando a uma temperatura estabilizada de aproximadamente de 427K. Através dos dados coletados também, foi possível verificar a variação da temperatura em função do tempo de acordo com a tensão determinada.

Tabela I – Variação de Tensão, Tempo e Temperatura.

	0V - 1,2V	1,2V - 1,5V	1,5V - 2,0V
Variação de Tensão	1,2 V	0,3 V	0,5 V
Variação do Tempo	00:05:27,774	00:06:04,668	00:06:19,633
Variação de Temperatura	44K	38K	44K

6. Conclusões

Por fim o trabalho abrangeu um *software* que roda na placa ESP32, que é capaz de identificar falhas de comunicação com cada componente periférico através de funções específicas de cada biblioteca utilizada, e ainda realizar a leitura de temperatura do sistema a ser estudado, e através do CoolTerm realizar o controle de temperatura, monitoramento dos dados, e ainda armazenamento do das informações coletadas, entretanto foi possível notar uma não equivalência de temperatura e com o passar do tempo, o que levou a concluir que essa não padronização dos valores originariam do acoplamento térmico do sensor de temperatura ou da giga de teste desenvolvida, que por ser de cerâmica poderia apresentar um bom isolamento térmico em contrapartida uma alta absorção térmica.

7. Referências

- [1] BOB-12918, Data Conversion IC Development Tools I2C DAC Breakout - MCP4725 Datasheet, SparkFun Electronics, 2022.
- [2] D. -. Jeon and D. E. Burk, "A temperature-dependent SOI MOSFET model for high-temperature application (27 degrees C-300 degrees C)," in IEEE Transactions on Electron Devices, vol. 38, n. 9, pp. 2101-2111, 1991, doi: 10.1109/16.83736.
- [3] J. Watson and G. Castro, "A review of high-temperature electronics technology and applications", J Mater Sci: Mater Electron, vol. 26, pp. 9226–9235, 2015, doi: 10.1007/s10854- 015-3459-4
- [4] MAX31865, RTD-to-Digital Converter Datasheet, Maxim Integrated, 2022.
- [5] P. Francis, A. Terao, B. Gentinne, D. Flandre and J. -P. Colinge, "SOI technology for high-temperature applications," 1992 International Technical Digest on Electron Devices Meeting, 1992, pp. 353-356, doi: 10.1109/IEDM.1992.307590.
- [6] MURTA, José Gustavo Abreu. **Conhecendo o ESP32 – Introdução (1)**. 2022. Disponível em: <https://blog.eletrogate.com/conhecendo-o-esp32-introducao-1/>. Acesso em: 19 nov. 2022.
- [7] 3328, Temperature Sensor Development Tools Adafruit PT100 RTD Temp SNSR Amp Datasheet, Adafruit, 2022.

Agradecimentos

À instituição FEI por disponibilizar toda estrutura necessária para realização deste trabalho, e ainda a oportunidade de trabalhar em uma área tão influente na instituição.

¹ Aluno de IT do Centro Universitário FEI. Projeto com vigência de 06/2022 a 05/2023.