Unisanta Science and Technology, 2026, 12, June
Published Online 2026 Vol.15 Nº1 http://periodicos.unisanta.br/index.php/sat)

UNISANTA
Science and Technology
ISSN 2317-1316

Sistema IoT de Monitoramento de Elementos do Clima para Condições de Trabalho com Filtro LPA2v

Matheus Cardoso Medeiros[1], Arnaldo de Carvalho Junior[1], Alexandre Maniçoba de Oliveira[2] e João Inácio da Silva Filho[3]

[1] IFSP- Embedded Artificial Intelligence Laboratory (EAILAB), Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Rua Maria Cristina, 50, Jardim Casqueiro – Cubatão/SP.
[2] IFSP- Laboratório Maxwell de Micro-ondas e Eletromagnetismo Aplicado (LABMAX), Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Rua Maria Cristina, 50, Jardim Casqueiro – Cubatão/SP.
[3] UNISANTA–Grupo de Lógica Paraconsistente Aplicada, Universidade Santa Cecília, Rua Oswaldo Cruz, 277 - Boqueirão - Santos/SP

E-mail*:* cardoso.medeiros@aluno.ifsp.edu.br
Received Jun, 2026

Resumo: O monitoramento de variáveis ambientais é fundamental para a avaliação das condições de trabalho e do conforto térmico em ambientes internos e externos. Neste contexto, este trabalho apresenta o desenvolvimento de um sistema IoT capaz de monitorar a temperatura, a umidade relativa do ar e a pressão atmosférica utilizando sensores eletrônicos conectados a uma plataforma Raspberry Pi 3 B+. Os dados adquiridos são processados por meio de umFiltro LPA2v implementado em Python com a biblioteca Paraconsistent-Lib, visando reduzir os efeitos de ruídos presentes nas medições. A partir das grandezas filtradas são calculados o índice de calor e a altitude estimada do ambiente monitorado. O sistema ainda disponibiliza um dashboard em html para exibir as informações na forma gráfica.

Palavras-chave: Internet das Coisas, Lógica Paraconsistente, Paraconsistent-Lib, Monitoramento Ambiental

IoT Weather Element Monitoring System for Working Conditions with PAL2v Filter

Abstract: Monitoring environmental variables is essential for evaluating working conditions and thermal comfort in indoor and outdoor environments. In this context, this work presents the development of an IoT system capable of monitoring temperature, relative humidity, and atmospheric pressure, using electronic sensors connected to a Raspberry Pi 3 B+ plataform. The acquired data is processed using a PAL2v Filter implemented in Python with the Paraconsistent-Lib library to reduce the effects of measurement noise. From the filtered quantities, the heat index and the estimated altitude of the monitored environment are calculated. The system also provides an HTML dashboard to display information graphically.

Keywords: Internet of Things, Paraconsistent Logic, Paraconsistent-Lib, Environmental Monitoring

1. INTRODUÇÃO

O monitoramento contínuo de variáveis ambientais tem se tornado uma ferramenta importante para a promoção da segurança, saúde ocupacional e eficiência operacional em diversos setores industriais. Grandezas como temperatura, umidade relativa do ar e pressão atmosférica influenciam diretamente o conforto térmico dos trabalhadores e podem impactar seu desempenho físico e cognitivo. Estudos demonstram que a exposição prolongada a ambientes com condições térmicas inadequadas pode aumentar o risco de fadiga, estresse térmico e acidentes de trabalho, além de reduzir a produtividade das equipes [1], [2].

Com o avanço das tecnologias de Internet das Coisas (*Internet of Things* - IoT), tornou-se possível realizar o monitoramento ambiental em tempo real por meio de sensores de baixo custo conectados a sistemas computacionais. Diversos trabalhos têm explorado o uso de plataformas IoT para aquisição e processamento de dados ambientais, destacando benefícios como a coleta contínua de

Copyright © 2026 ISESC.

informações, a geração de indicadores para tomada de decisão e a implementação de sistemas inteligentes de supervisão [3], [4]. Entretanto, uma das principais limitações desses sistemas está relacionada à presença de ruídos, incertezas e inconsistências nas medições realizadas pelos sensores, o que pode comprometer a confiabilidade dos dados obtidos [5].

Este artigo apresenta o desenvolvimento de um sistema baseado em IoT destinado ao monitoramento de variáveis ambientais, incluindo temperatura, umidade relativa do ar e pressão atmosférica, com o objetivo de auxiliar na avaliação das condições de trabalho e do ambiente monitorado. O sistema realiza a aquisição contínua dos dados por meio de sensores eletrônicos e aplica técnicas de filtragem baseadas em Lógica Paraconsistente Anotada de dois Valores (LPA2v) para reduzir a influência de ruídos, incertezas e inconsistências inerentes ao processo de medição [6], [7]. Para tal, emprega-se a biblioteca Paraconsistent-Lib para Python, que simplifica a implementação de algoritmos da lógica paraconsistente e possibilita o processamento eficiente dos dados coletados [8]. Após a etapa de filtragem, o sistema calcula parâmetros derivados, como o índice de calor e a altitude estimada, fornecendo informações adicionais para a análise das condições ambientais.

2. METODOLOGIA

O desenvolvimento do sistema foi dividido em três etapas principais: (i) seleção dos componentes de *hardware*, (ii) elaboração e montagem do circuito eletrônico e (iii) desenvolvimento do *software* de aquisição, processamento e visualização dos dados. Considerando que a aplicação tem como objetivo monitorar temperatura, umidade relativa do ar e pressão atmosférica, foi selecionado o sensor BME280, capaz de realizar a medição simultânea dessas três grandezas por meio de uma interface digital I²C. Adicionalmente, foi empregado um sensor TMP36 para a obtenção independente da temperatura ambiente, permitindo a realização de análises comparativas entre diferentes fontes de medição.

O processamento dos dados foi realizado em um Raspberry Pi 3 B+, escolhido por sua capacidade de executar aplicações desenvolvidas em Python, além de oferecer recursos de conectividade sem fio, como Wi-Fi e Bluetooth. Os dados adquiridos são disponibilizados em um painel de monitoramento acessível via navegador web, permitindo o acompanhamento remoto das variáveis monitoradas.

Como mecanismo de redundância para situações de indisponibilidade da rede, foi incorporado ao sistema um *display* OLED SSD1306, responsável pela exibição local das informações mais relevantes. Após a definição dos componentes utilizados, foi elaborado o diagrama eletrônico do sistema, apresentado na Figura 1.

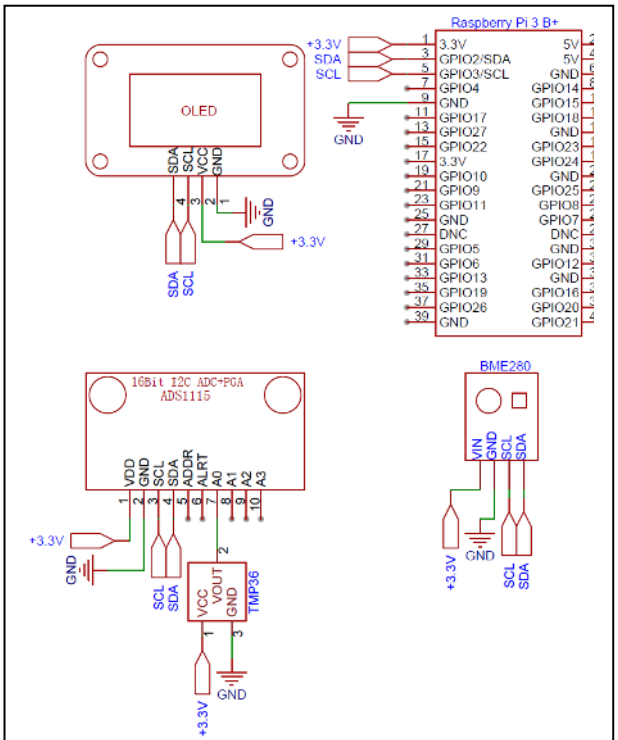

Figura 1. Diagrama do circuito elétrico implementado.

O ADS1115 converte a saída analógica do sensor TMP36 para digital, sendo assim compatível com as entradas do Raspberry Pi 3 B+.

Os componentes reais utilizados podem ser observados na Figura 2. Após a seleção do hardware e montagem do diagrama elétrico, os componentes foram integrados numa placa perfurada, onde foram soldados e conectados entre si, como mostra a Figura 3.

Com todo o circuito funcionando, foram preparadas as rotinas de aquisição e manipulação dos dados dos sensores. Para isso, foi usada a linguagem de programação Python dentro do ambiente do Raspberry Pi.

A programação do sistema foi dividida em quatro partes: (i) leitura dos sinais elétricos advindos dos sensores através de interface I²C, (ii) tratamento da leitura dos sensores através do filtro LPA2v implementado com biblioteca Paraconsistent-Lib, (iii) cálculo do índice de calor e altitude a partir dos valores filtrados e (iv) disponibilização dos dados coletados e filtrados em uma interface web.

Copyright © 2026 ISESC

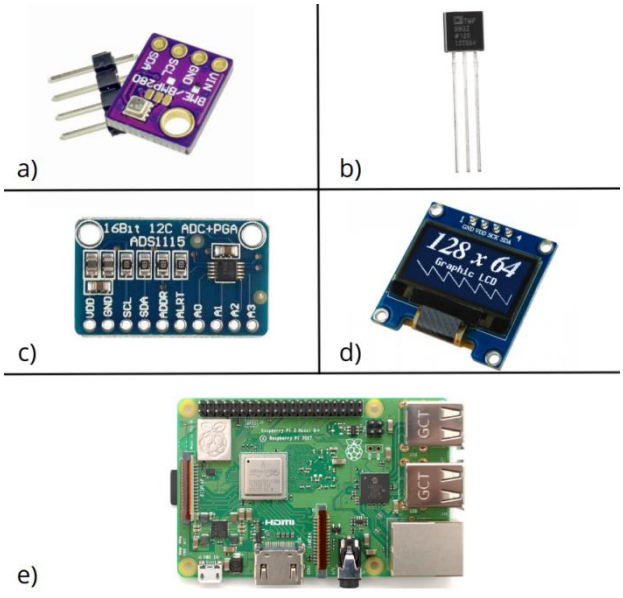

Figura 2. Componentes utilizados: a) BME280, b) TMP36, c) ADC ADS1115, d) *Display* OLED 0.96, e) Raspberry Pi 3 B+.

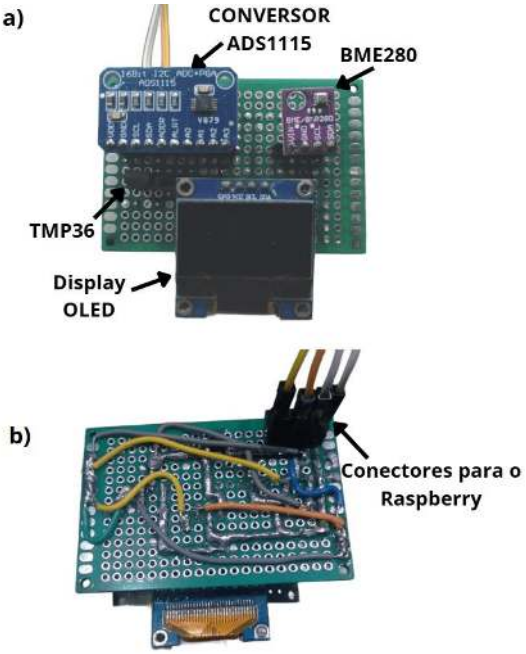

Figura 3. Montagem do circuito eletrônico em placa perfurada dupla face: a) vista superior, b) vista inferior.

A aquisição correta das informações advindas do sensor BME280 foram coletadas através de bibliotecas da Adafruit através de comandos simples e comunicação I²C. O sensor TMP36, entretanto, necessita de converter a tensão lida em um valor correspondente em temperatura. Com esse objetivo, utiliza-se a Equação 1.

$$\text{Temperatura}(°C) = \frac{\text{Tensão}(mV) - 500}{10} \quad (1)$$

Se for lida uma tensão de 750mV, por exemplo, tem-se uma temperatura de 25°C.

Para uma leitura precisa e estável dos sensores, os dados passam por uma cadeia de células de filtragem LPA2v. A Figura 4 mostra a estrutura básica de uma célula de filtragem paraconsistente.

A célula de filtragem, também chamada de célula neural artificial paraconsistente de aprendizagem por extração de contradição ($PANCL_{CTX}$), possui entrada μ (sinal do sensor normalizado na faixa [0, 1]) e saída μ_E (valor filtrado). Ao aplicar o complemento da saída (1 - μ_E) à entrada do grau de evidência desfavorável (λ), a saída tende a seguir a entrada μ com um certo atraso. O parâmetro F_L (fator de aprendizagem) é responsável por ajustar a aceleração ou atraso com que essa operação acontece [9].

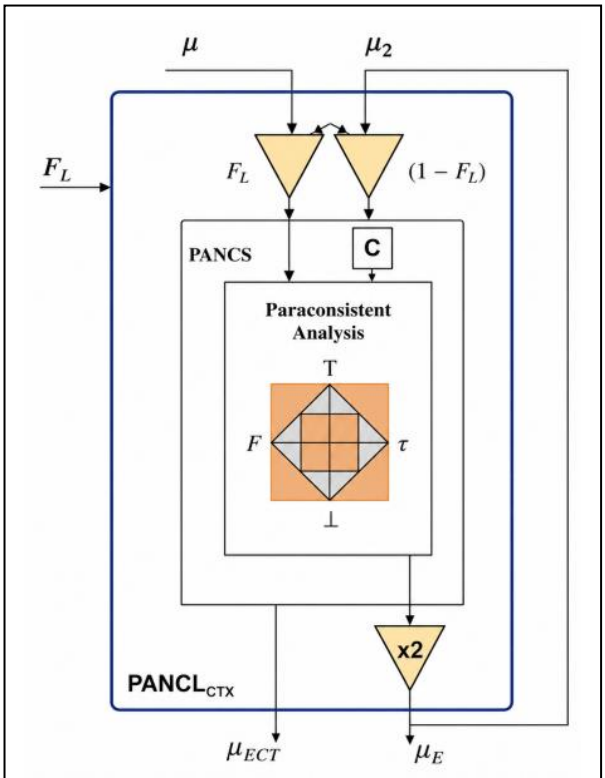

Figura 4. Estrutura da célula de filtragem LPA2v [9].

Copyright © 2026 ISESC

De acordo com a estrutura apresentada na Figura 4. é demonstrado um código em Python que engloba todas as operações necessárias para a implementação do filtro com a Paraconsistent-Lib:

```
from paraconsistent.blocks import Paracon-
sistentBlock

previous_muE = 0.5  # Condição inicial.
PANCLCTX = ParaconsistentBlock()

def filter_panclctx(mu, fl):
 global previous_muE
 PANCLCTX.input.mu = fl*mu # Entrada do fil-
tro      (Sinal do sensor normlizado)
 PANCLCTX.input.lam    =    1-((1-fl)*previ-
ous_muE)          # Realimentação à entrada
lambda (complemento da saída)
previous_muE = PANCLCTX.complete.muE * 2 #
saída do filtro (valor filtrado)
return previous_muE # Função retorna valor
filtrado
```

Como a lógica paraconsistente trabalha com valores entre 0 e 1, então se faz necessário normalizar o valor lido dos sensores em [0, 1] antes de aplicar na entrada no filtro. Para isso, basta conhecer os valores máximo e mínimo da grandeza a ser lida e aplicar a Equação 2:

$$\mu = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{2}$$

Onde x é o valor lido do sensor e μ é o valor normalizado. Além disso, como a saída também está no intervalo normalizado, é necessário desnormalizar para que a grandeza volte à sua unidade de medida original. Aplicando a Equação 3, obtemos a saída do filtro desnormalizada:

$$\mathrm{x} = \mu_{\boldsymbol{E}} \times \left(\boldsymbol{x}_{\max} - \boldsymbol{x}_{\min}\right) - \boldsymbol{x}_{\min} \tag{3}$$

Após a filtragem das grandezas físicas medidas, calcula-se do índice de calor e altitude. Para o índice de calor, é necessário apenas a temperatura do ar e umidade relativa, conforme a Equação 4 [10]:

$$\begin{aligned} HI = &-8.784695 + 1.61139411\,T + 2.338549\,R \\ &- 0.14611605\,TR - 0.01230809\,T^2 \\ &- 0.01642482\,R^2 + 0.00221173\,T^2 R \\ &+ 0.00072546\,TR^2 \\ &- 0.00000358\,T^2 R^2 \end{aligned} \tag{4}$$

Onde: HI (*Heat Index*) é o índice de calor calculado, T é a temperatura do ar (°C) e R é a umidade relativa (%).

O índice de calor representa uma estimativa da sensação térmica percebida pelo corpo humano, considerando os efeitos combinados da temperatura ambiente e da umidade relativa, sendo amplamente utilizado na avaliação de condições de conforto térmico e exposição ao calor. Dependendo da faixa de valor calculada do índice, o sistema exibe alguns sinais de alerta de acordo com a Tabela 1:

Tabela 1. Classificação do índice de calor.

Índice de calor	Sinal de alerta correspondente
27–32 °C	Cuidado — possibilidade de fadiga após exposição e atividade prolongadas.
32–41 °C	Cuidado extremo — hipertermia e cãibras de calor possíveis.
41–54 °C	Perigo — hipertermia e cãibras de calor prováveis.
Acima de 54 °C	Perigo extremo — hipertermia e cãibras de calor iminentes.

A altitude foi estimada a partir dos valores filtrados de pressão atmosférica fornecidos pelo sensor BME280. Para isso, foi utilizada uma forma simplificada da equação barométrica, que relaciona a pressão atmosférica local com a pressão atmosférica padrão ao nível do mar. Essa abordagem permite obter uma estimativa da altitude em relação ao nível médio do mar a partir da redução da pressão atmosférica com o aumento da elevação. O cálculo foi realizado conforme a Equação 5.

$$\mathrm{h} = 44330\left(1 - \left(\frac{P}{P_0}\right)^{\frac{1}{5.255}}\right) \tag{5}$$

Onde: P é a pressão medida e P_0 é a pressão de referência dado em hPa (hectopascal).

Após a etapa de processamento dos dados, foi desenvolvido um painel de monitoramento acessível por navegador web, permitindo o acompanhamento remoto das variáveis monitoradas em tempo real.

Para a implementação do servidor web foi utilizado o *framework* Flask, enquanto a comunicação entre os módulos do sistema empregou bibliotecas da linguagem Python para transmissão e atualização dos dados. A interface apresenta tanto os valores brutos quanto os valores filtrados das grandezas monitoradas, permitindo a comparação direta entre as medições originais e os resultados obtidos pelo filtro paraconsistente.

A Figura 5 apresenta o fluxo de processamento implementado no sistema. Inicialmente, os dados provenientes

Copyright © 2026 ISESC

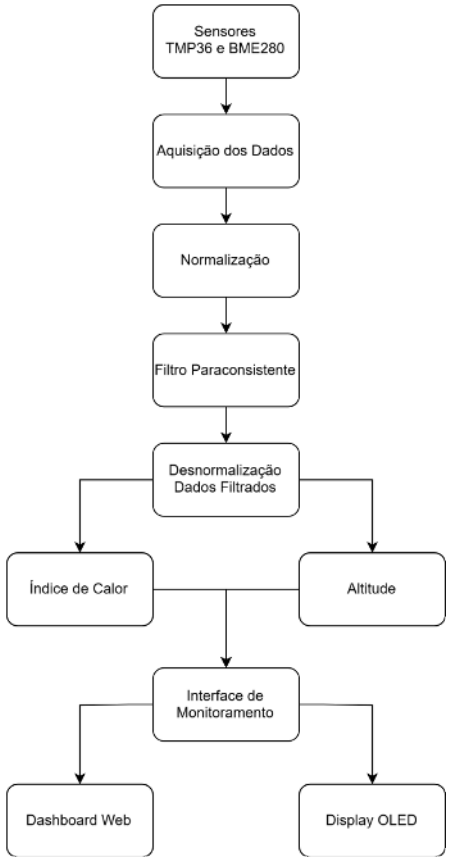

Figura 5. Fluxograma do *software* embarcado.

dos sensores são adquiridos e normalizados para o intervalo de operação do filtro paraconsistente. Em seguida, as amostras são processadas pela biblioteca Paraconsistent-Lib, produzindo valores filtrados utilizados no cálculo do índice de calor e da altitude.

Por fim, as informações processadas são disponibilizadas tanto em uma interface web para monitoramento remoto quanto em um *display* OLED para visualização local.

3. RESULTADOS

A Figura 6 apresenta a interface web desenvolvida para monitoramento remoto das variáveis ambientais. O painel permite a visualização em tempo real dos valores de temperatura, umidade relativa do ar, pressão atmosférica, índice de calor e altitude. Além disso, são exibidos simultaneamente os sinais brutos e filtrados, possibilitando a análise visual do desempenho do filtro paraconsistente.

Ademais da disponibilização dos dados no *dashboard* web, podemos observar a indicação local das variáveis através do *display* OLED como mostrado na Figura 7.

Os testes para avaliação do filtro LPA2v foram conduzidos com um F_L de 0.1 e 4 células em cascata (filtro de 4° ordem). Os ruídos dos sinais dos sensores foram reduzidos consideravelmente, conforme a Figura 8.

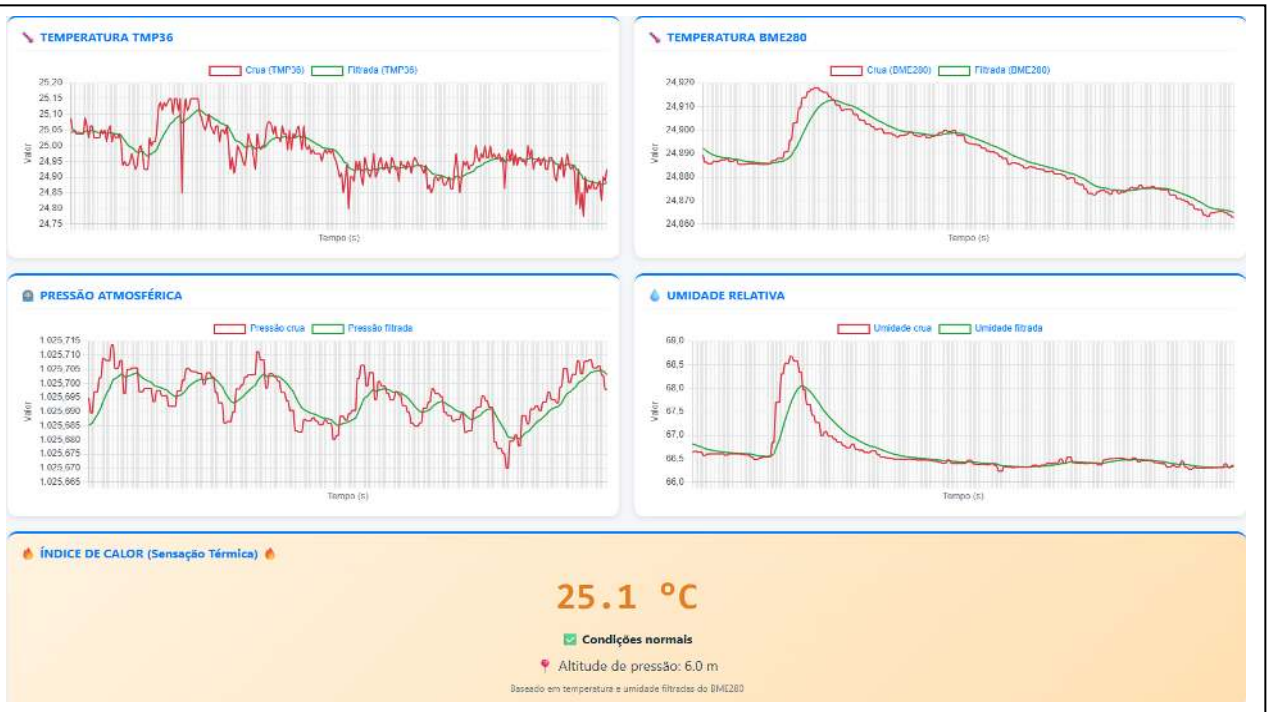

Figura 6. *Dashboard* Web.

Copyright © 2026 ISESC

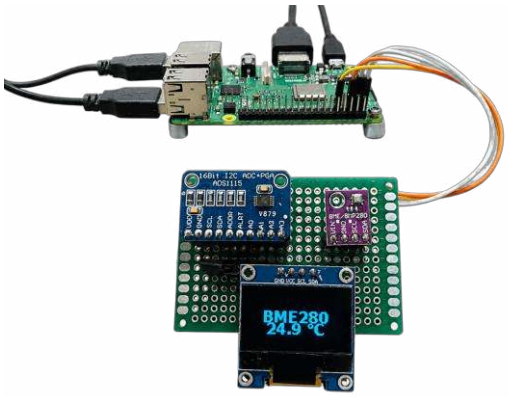

Figura 7. Indicação local no *display* OLED.



Figura 8. Gráfico da temperatura bruta e filtrada obtido do TMP36

Para a validação do sistema de emissão de sinal de alerta baseado nos valores de índice de calor, foi usado um secador de cabelo para simular uma situação de alta temperatura. A Figura 9 mostra as respostas do sistema frente a variação do índice de calor, conforme demonstrado na Tabela 1 da seção anterior.

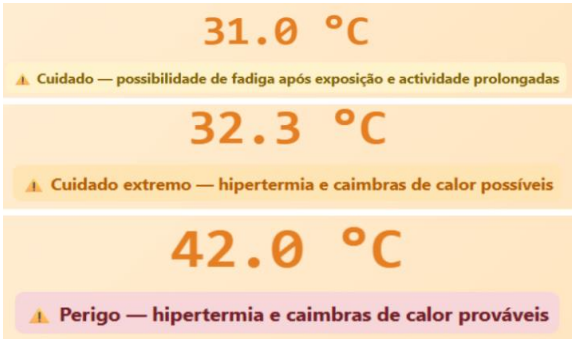

Figura 9. Avisos do sistema conforme o índice de calor.

Verifica-se que o sistema obteve o comportamento desejado e bom desempenho do algoritmo de filtragem.

4. CONCLUSÃO

Este trabalho apresentou o desenvolvimento de um sistema IoT para monitoramento de temperatura, umidade relativa do ar e pressão atmosférica, utilizando sensores eletrônicos integrados a uma plataforma Raspberry Pi. Os dados adquiridos foram processados por meio de um filtro paraconsistente implementado com a biblioteca Paraconsistent-Lib, permitindo a redução dos ruídos presentes nas medições e contribuindo para a obtenção de sinais mais estáveis.

A partir das variáveis monitoradas, foram calculados o índice de calor e altitude estimada do ambiente, ampliando a quantidade de informações disponibilizadas ao usuário. Os resultados foram apresentados tanto localmente, pelo *display* OLED, quanto remotamente por meio de uma *dashboard* web desenvolvida para acompanhamento em tempo real.

Os testes experimentais demonstraram o correto funcionamento do sistema e indicaram que a utilização da Paraconsistent-Lib constitui uma alternativa viável para aplicações práticas de monitoramento ambiental baseadas em IoT. Além da simplificação do desenvolvimento do filtro paraconsistente, a biblioteca possibilitou a integração da lógica paraconsistente e um sistema real de aquisição e processamento de dados.

Destaca-se ainda que este trabalho representa uma das primeiras aplicações da Paraconsistent-Lib embarcada em um sistema real de monitoramento, provando o seu potencial para aplicações de IoT.

REFERÊNCIAS

[1] KJELLSTROM, T.; HOLMER, I.; LEMKE, B. Workplace heat stress, health and productivity: an increasing challenge for low and middle-income countries during climate change. Global Health Action, v. 2, n. 1, p. 2047, 2009. DOI: https://doi.org/10.3402/gha.v2i0.2047.

[2] PARSONS, K. Human Thermal Environments: The Effects of Hot, Moderate, and Cold Environments on Human Health, Comfort and Performance. Boca Raton: CRC Press, 2007. DOI: https://doi.org/10.1201/b16750.

Copyright © 2026 ISESC

[3] DA XU, L.; HE, W.; LI, S. Internet of things in industries: a survey. IEEE Transactions on Industrial Informatics, v. 10, n. 4, p. 2233–2243, 2014. DOI: https://doi.org/10.1109/TII.2014.2300753.

[4] AL-FUQAHA, A. et al. Internet of things: a survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, v. 17, n. 4, p. 2347–2376, 2015. DOI: https://doi.org/10.1109/COMST.2015.2444095.

[5] MAZON-OLIVO, B.; PAN, A. Internet of things: state-of-the-art, computing paradigms and reference architectures. IEEE Latin America Transactions, v. 20, n. 1, p. 49–63, 2021. Disponível em: https://latamt.ieeer9.org/index.php/transactions/article/view/5037.

[6] DE CARVALHO, A. et al. A study of paraconsistent artificial neural cell of learning applied as PAL2v filter. IEEE Latin America Transactions, v. 16, n. 1, p. 202–209, 2018. DOI: https://doi.org/10.1109/TLA.2018.8291474.

[7] DE CARVALHO JUNIOR, A. et al. A comprehensive review on paraconsistent annotated evidential logic: algorithms, applications, and perspectives. Engineering Applications of Artificial Intelligence, v. 127, p. 107342, 2024. DOI: https://doi.org/10.1016/j.engappai.2023.107342.

[8] DE CARVALHO JUNIOR, A. et al. Paraconsistent-Lib: an intuitive PAL2v algorithm Python Library. arXiv preprint arXiv:2511.20700, 2025. Disponível em: http://arxiv.org/abs/2511.20700.

[9] DE CARVALHO JUNIOR, A. et al. Paraconsistent logic approach for active noise reduction. Journal of Mechatronics Engineering, v. 3, n. 1, p. 2–8, 2020. DOI: https://doi.org/10.21439/jme.v3i1.81.

[10] ROTHFUSZ, L. P.; HEADQUARTERS, NWS SOUTHERN REGION. The heat index equation (or, more than you ever wanted to know about heat index). Fort Worth, Texas: National Oceanic and Atmospheric Administration, National Weather Service, Office of Meteorology, 1990. (Technical Attachment SR 90-23). Disponível em: https://www.weather.gov/media/ffc/ta_htindx.PDF.

Copyright © 2026 ISESC