



Instituto Federal de Educação, Ciência e Tecnologia  
do Rio Grande do Norte – Campus Santa Cruz

I Semana de Ciência, Tecnologia e Extensão do IFRN

XI CONGIC

XI CONGIC

## CONTROLE DE TEMPERATURA DE BAIXO CUSTO UTILIZANDO LÓGICA FUZZY

G. F. Lima<sup>1</sup>

E-mail: gustavo.lima@ifrn.edu.br<sup>1</sup>

### RESUMO

Este trabalho apresenta a placa Arduino no controle de temperatura de um ambiente de baixo custo. Um sistema térmico em escala reduzida foi montado para a implementação de um controle de temperatura utilizando a Lógica *Fuzzy*. A placa Arduino ligou uma lâmpada, por meio de um *dimmer* que controlou o brilho da mesma, e assim controlou a temperatura no

seu arredor. As leituras de temperatura oscilaram próximo da referência predefinida de 35 °C. Os resultados dos ensaios experimentais mostraram que é possível utilizar a Lógica *Fuzzy* e a placa Arduino para controlar a temperatura de um ambiente fechado, uma vez que esse tipo processo possui uma baixa velocidade na ação de controle.

**PALAVRAS-CHAVE:** Arduino, controle, sensor, prototipagem, *dimmer*.

## TEMPERATURE CONTROL FROM LOW COST USING FUZZY LOGIC

### ABSTRACT

This work show the Arduino board in temperATURE control from low cost environment. A termic system at reduce scale was mounted for implementation of the temperatura control using Fuzzy Logic. The Arduino board turn on a lamp, through of the dimmer, which controled the its brightness, and therefore controled the temperatura at its

surroundings. The temperature measurements oscillated near to reference of the 35 °C. The results of the experimentals tests showed which is possible to use Arduino board with Fuzzy Logic to controlling the temperatura a closed environment, once that process has a low speed in control action.

**KEY-WORDS:** Arduino, control, sensor, prototyping, dimmer.



## 1 INTRODUÇÃO

A plataforma Arduino, criada em 2005, é constituída basicamente por uma pequena placa de circuito impresso. Tem como público alvo as pessoas que desejam criar protótipos de eletrônica, objetos ou ambientes interativos. A placa pode receber sinais dos mais variados tipos de sensores em suas entradas e pode interagir com sua vizinhança acionando luzes, motores ou outros atuadores.

Este trabalho tem por objetivo apresentar uma aplicação do Arduino realizando a tarefa de controle e monitoramento de temperatura de um ambiente de baixo custo utilizando a Lógica *Fuzzy*. Um sistema térmico de baixo custo e em escala reduzida foi montado para realização de ensaios experimentais. A modelagem matemática do sensor de temperatura e a descrição dos demais componentes são apresentados também.

A motivação inicial deste trabalho foi propor uma maneira das pessoas em uma residência poderem acompanhar a temperatura de um determinado ambiente. A aplicação da Lógica *Fuzzy* permitiu uma evolução gradual da temperatura, de forma econômica, uma vez que sistema consumiu energia sem desperdício.

Como contribuição pretende-se difundir a filosofia do “Faça Você Mesmo” ou “Do It Yourself” (DIY) para construção de alguns itens, reaproveitamento ou reuso de componentes. E por fim, integrar materiais de disciplinas distintas como Instrumentação Eletrônica, Microcontroladores e Eletrônica Analógica.

## 2 REVISÃO BIBLIOGRÁFICA

### 2.1 Lógica *Fuzzy*

Em 1965, o professor Lotfi Asker Zadeh da Universidade de Berkeley (USA) publicou no jornal acadêmico “*Information and Control*” o trabalho intitulado “*Fuzzy sets*”. Ele propôs uma nova teoria de conjuntos, onde segundo Feitosa citado (BILOBROVEC, MARÇAL e KOVALESKI, 2004), a passagem da pertinência para a não pertinência fosse feita de uma forma lenta e gradual e não abrupta como na teoria usual de conjuntos. Assim, surgiram os Conjuntos *Fuzzy* (nebuloso ou difuso).

Na teoria de Sistemas de Controle, um controlador baseado em Lógica *Fuzzy* pode ter sua ação esquematizada pelos seguintes elementos constituintes (FILHO, GOSMANN e BAUCHSPIESS, 2002): *Fuzzificador* ou Codificador; Regras (base de conhecimento); Inferência (lógica de tomada de decisões); e *Defuzzificador* ou Decodificador. Esta estrutura pode ser visualizada na Figura 1.

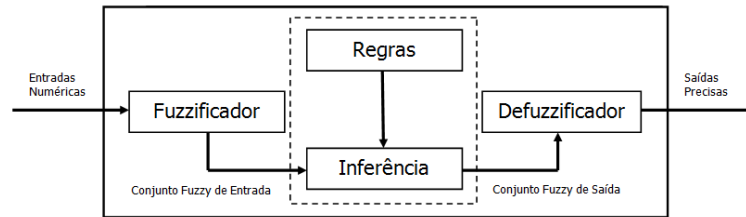


Figura 1: Blocos do Controlador Fuzzy.

O *fuzzificador* tem por função transformar os valores numéricos das entradas em suas respectivas variáveis linguísticas.

A base de dados fornece definições numéricas e a base de regras caracteriza os objetivos do controlador e sua estratégia usada, geralmente fornecida por pessoas especialistas no sistema.

A inferência tem por característica transformar os valores linguísticos de entrada, junto com as regras, em valores linguísticos de saída.

O *defuzzificador* tem por objetivo traduzir os valores linguísticos de saída, inferidos pelas regras, em valores numéricos de saída, utilizáveis numa ação de controle concreta no mundo real.

Em resumo, uma grande vantagem da Lógica *Fuzzy* implementada nos controladores, é a de esses controladores comportem-se conforme o raciocínio dedutivo, isto é, o processo que as pessoas utilizam para inferir conclusões baseadas em informações que elas já conhecem (Shaw & Simões 1999).

## 2.2 Sensor de Temperatura NTC

No projeto foi utilizado um termistor *NTC* (Coeficiente de Temperatura Negativo), semicondutor de baixo custo e que diminui sua resistência elétrica com o aumento da temperatura. Tem como vantagem a medição de variação rápida de temperatura, isso confere ao sensor uma característica não-linear, conforme a **Erro! Fonte de referência não encontrada.**(a). Já a **Erro! Fonte de referência não encontrada.**(b) mostra o sensor *NTC* utilizado neste projeto.

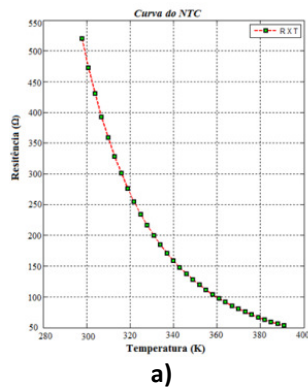


Figura 2: Sensor de Temperatura Utilizado – a) Curva Característica do Sensor; b) Semicondutor NTC.

Um circuito divisor de tensão foi escolhido para converter a resistência do sensor *NTC* em uma tensão mensurável pela placa Arduino. Na configuração escolhida, o divisor de tensão permitiu que um aumento de temperatura implicasse no aumento de tensão  $V_o$ , dada pela Eq. (1).

$$V_o = V_{cc} \cdot \left( \frac{R_1}{R_1 + R_{NTC}} \right) \quad (1)$$

A saída do divisor de tensão  $V_o$  foi conectada à entrada analógica A0 do Arduino e este ao receber essa tensão precisou calcular a resistência  $R_{NTC}$ , pela Eq.(2).

$$R_{NTC} = \left( R_1 \cdot \frac{V_{cc}}{V_o} \right) - R_1 \quad (2)$$

De posse do valor de resistência  $R_{NTC}$  foi possível determinar a temperatura atual do sistema em Kelvin. A relação entre resistência e temperatura no sensor *NTC* é dada pela equação de Steinhart-Hart, conforme Eq.(3).

$$T(K) = \frac{1}{a + b \cdot \ln(R_{NTC}) + c \cdot (\ln(R_{NTC}))^3} \quad (3)$$

onde:  $a = 0,001129148$ ,  $b = 0,000234125$  e  $c = 0,0000000876741$ .

Por fim, a conversão de temperatura de Kelvin para graus Celsius é dada pela Eq. (4).

$$T(^{\circ}\text{C}) = T(K) - 273,15 \quad (4)$$

De posse desses coeficientes foi possível implementar o trecho de código fonte para Arduino capaz de converter a tensão fornecida pelo sensor *NTC*, calcular a resistência  $R_{NTC}$ , depois determinar a temperatura em Kelvin e por fim, converter de Kelvin para uma temperatura em graus Celsius, conforme Código 1.

```
//Conversao da leitura do sensor em temp
sensorRead=analogRead(A0); // le porta A0
V0=sensorRead*0.0048875855; // converte p/ volt
Rntc=(5*300/V0)-300; // RNTC=(Vcc*R1/Vin)-R1
b1=log(Rntc); // LN(RNTC)
b1=0.000234125*b1; // b*LN(RNTC)
c1=log(Rntc); // LN(RNTC)
c1=0.0000000876741*c1*c1*c1; // c*(LN(RNTC)^3)
tempK=0.001129148+b1+c1; // a+b*LN(RNTC)+c*(LN(RNTC)^3)
```

**Código 1: Trecho de Código para Arduino.**

O cálculo da temperatura acontece a cada 1,0 segundo. Seu valor é necessário para determinar a porcentagem de tensão que será aplicada na lâmpada.

## 2.3 Plataforma Arduino

Surgiu em 2005, no *Interaction Design Institute* na cidade de Ivrea, na Itália, com um professor chamado Massimo Banzi. Ele se deparou com dois desafios: ensinar eletrônica e programação para seus alunos de *design*; e a inexistência de placas baratas disponíveis no mercado. Ao discutir esse problema com David Cuartielles, eles decidiram criar sua placa própria, o Arduino.

Segundo [Banzi et. al \(2005\)](#) o Arduino é uma plataforma desenvolvida para criação de protótipos de eletrônica. É uma placa de circuito impresso com entradas e saídas digitais; entradas analógicas para utilização com sensores; pinos de alimentação; microcontrolador ATmega328; e conversor USB <> Serial. Para programar uma placa Arduino é necessário utilizar seu ambiente integrado de desenvolvimento (do inglês, Integrated Development Environment ou IDE) baseado em *Processing*, que utiliza uma linguagem baseada em *Wiring*.

Neste projeto foi utilizado um Arduino Mega 2560, uma placa com o microcontrolador Atmega2560. Ela conta com 54 pinos digitais, 16 pinos analógicos, 4 portas seriais, um cristal de 16 MHz, entrada USB, entrada de alimentação, soquete de comunicação ICSP e um botão reset.

## 3 IMPLEMENTAÇÃO DO SISTEMA

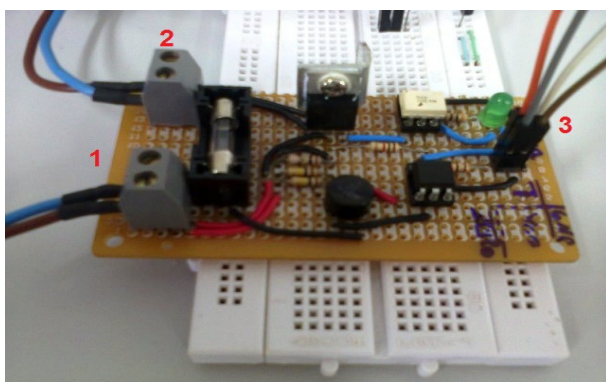
### 3.1 Montagem do Protótipo

A montagem do sistema proposto começou pelo ambiente fechado em escala reduzida. Foi reaproveitada uma caixa de papelão (33 x 21 x 12 cm), onde foi colocada uma base branca para uma lâmpada incandescente de 220 V/60 W e uma base retangular para instalação do sensor de temperatura NTC, conforme [Figura 3](#).

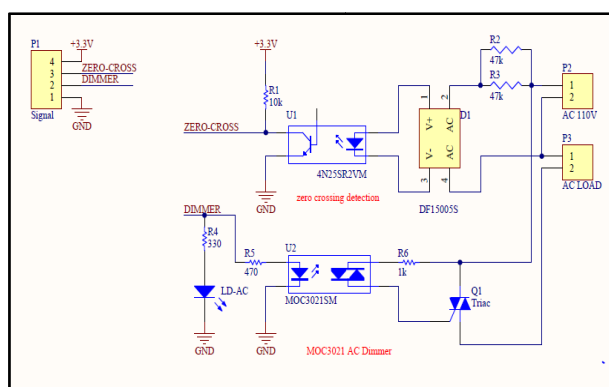


Figura 3: Ambiente em Escala Reduzida.

A fiação que vem da tomada elétrica foi ligada ao borne de força 1 da placa *dimmer*. Em seguida, a fiação que vai para lâmpada foi conectada ao borne 2. E por fim, os fios de controle que vem da placa Arduino foram ligados no borne 3 da placa *dimmer*, conforme a **Figura 4(a)**. O esquema elétrico de montagem da placa *dimmer* pode ser visualizado na **Figura 4(b)**.



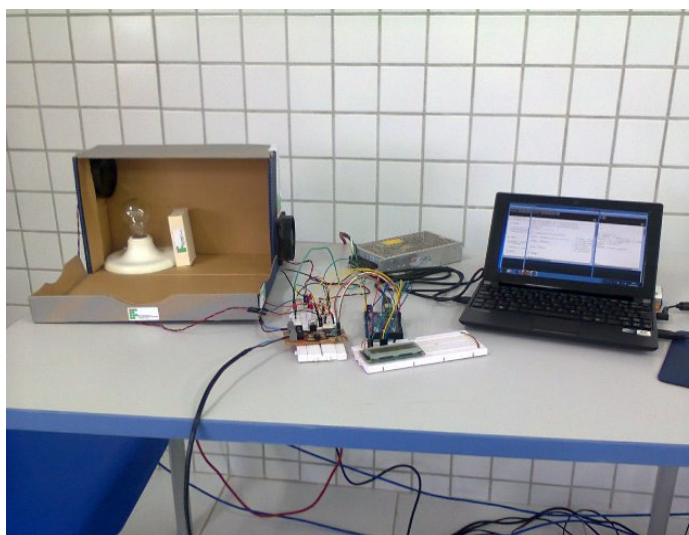
a)



b)

**Figura 4: Placa *Dimmer* para Controle de Brilho da Lâmpada - a) Conexões Elétricas; b) Esquema Elétrico.**

A **Figura 5** mostra todo o sistema proposto montado. Nela é possível observar as conexões de todas as partes apresentadas anteriormente e presença de um *netbook* para controlar todo o sistema, como também, para produzir os gráficos de temperatura *versus* tempo.



**Figura 5: Sistema em Escala Reduzida Proposto.**

O funcionamento da placa *dimmer* no controle de brilho da lâmpada pode ser visualizada em dois vídeos publicados na Internet. O primeiro vídeo pode ser visualizado pelo seguinte endereço <http://youtu.be/5YWZ-cRxhy0> eo segundo vídeo foi publicado no seguinte endereço <http://youtu.be/407Hmb84KAM>.

### 3.2 Controlador Inteligente *Fuzzy*

O controlador foi configurado com uma variável de entrada e uma variável de saída. A variável de entrada foi chamada de *temper* e foi definida com valores de 20 a 60 °C. A **Figura 6** mostra o conjunto *Fuzzy* gerado, onde os valores linguísticos possíveis para *temper* foram: baixo, medbaixo, setpoint, medalta e alta.

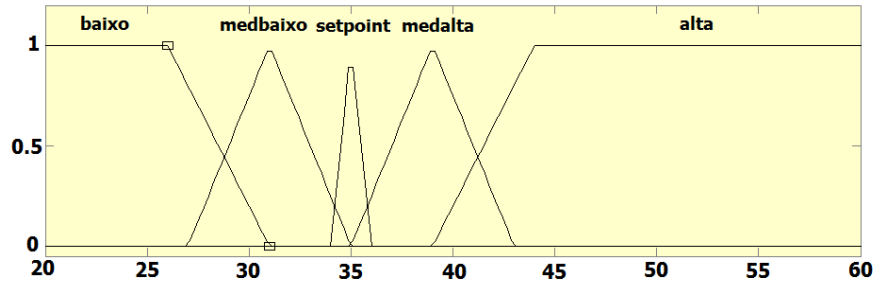


Figura 6: Variável de Entrada *temper*.

A variável de saída foi chamada de *output* e representou o tempo de disparo do TRIAC instalado na placa *dimmer*. Esta variável controlou o tempo para ligar o TRIAC, que variou de 0 a 8 *ms*. Lembrando que um tempo menor de disparo representa um tempo maior da lâmpada ligada e com isso um brilho maior. A **Figura 7** apresenta o conjunto *Fuzzy* gerado, onde os valores linguísticos foram de: zero a oito.

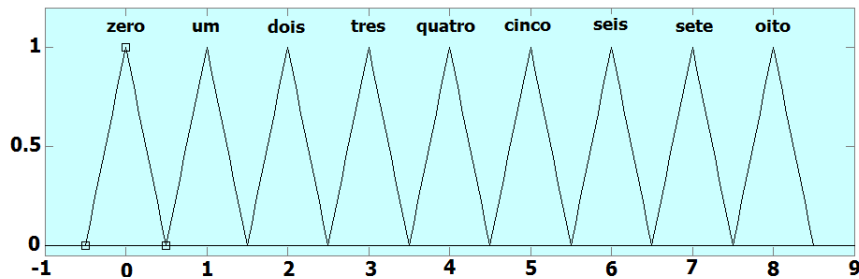


Figura 7: Variável de Saída *output*.

O Bloco de Regras produzido possuiu 5 regras, de maneira a cobrir todas as combinações da variável de entrada. Essas regras foram organizadas na **Figura 8**.

```
1. If (temper is baixo) then (output1 is um) (1)
2. If (temper is setpoint) then (output1 is quatro) (1)
3. If (temper is alta) then (output1 is sete) (1)
4. If (temper is medbaixo) then (output1 is tres) (1)
5. If (temper is medialta) then (output1 is seis) (1)
```

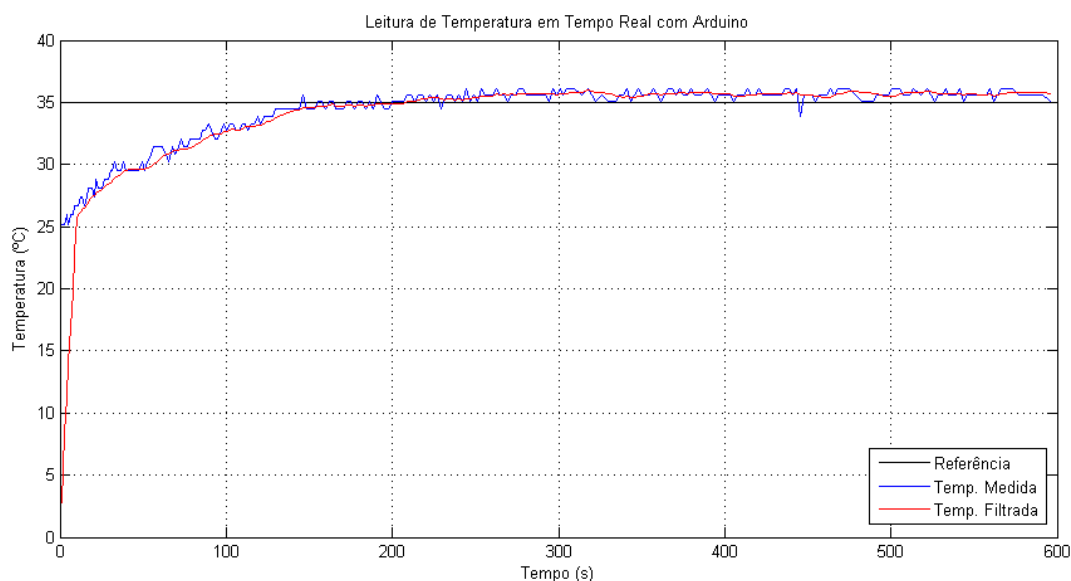
Figura 8: Regras *Fuzzy* Utilizadas.

Para a máquina de inferência foi usada a Mamdani, que utiliza como implicação a t-norma mínimo [min] e como agregação a sua co-norma máximo [max]. A *defuzzificação* foi calculada utilizando a média da pertinência máxima (do inglês, Mean of Maximum – MoM).

#### 4 RESULTADO EXPERIMENTAL E DISCUSSÃO

Para comprovação do funcionamento do controle *Fuzzy* foi produzido um gráfico de temperatura *versus* tempo, com base nos dados capturados pela porta Serial da placa Arduino. A **Figura 9** mostra a evolução da temperatura com o controle inteligente funcionando. A referência escolhida foi de 35 °C durante todo o ensaio de aproximadamente 600 segundos ou 10 minutos.

Outro detalhe sobre o gráfico é que a temperatura real do sistema foi identificada pela cor azul, enquanto que a cor vermelha mostra a média móvel da temperatura. A média móvel foi utilizada com o objetivo de suavizar as oscilações de temperatura do sistema, como também, diminuir os ruídos de leitura.



**Figura 9: Resposta da Temperatura Utilizando o Controlador Inteligente *Fuzzy*.**

Com base no gráfico apresentado é possível visualizar o pleno funcionamento do controle *Fuzzy* proposto. Estratégia de controle esta que pode ser embarcada na placa Arduino por meio de estruturas condicionais IF ... THEN, de baixo custo computacional e de grande aplicabilidade onde for necessário controlar grandezas de evolução lenta como a temperatura ou o nível de um tanque, por exemplo. O que não requer grandes aparatos tecnológicos.





## 5 CONCLUSÕES

Neste trabalho foi apresentada a placa Arduino Mega, pequena placa de circuito impresso que possui 01 microcontrolador Atmega2560, com 54 pinos digitais, 16 pinos analógicos, etc. E a Lógica *Fuzzy*, proposta em 1965, permite dotar os computadores de raciocínio dedutivo, o mesmo utilizado pelo ser humano para concluir sobre algo com base nas informações disponíveis.

O trabalho também mostrou a montagem de um ambiente fechado em escala reduzida e de baixo custo, com o objetivo de transformá-lo em um sistema térmico. Nele a temperatura ao redor de uma lâmpada foi medida por um sensor NTC, numa configuração divisor de tensão. Com essa informação o Arduino conseguiu controlar o acionamento da lâmpada, por meio de uma placa *dimmer*. Esta alterou o brilho da lâmpada de forma a alterar a temperatura do sistema.

Como aplicação um controlador inteligente *Fuzzy* foi configurado com sucesso. E as leituras de temperatura foram capturadas da porta Serial da placa Arduino. A temperatura de referência de 35 °C foi alcançada depois aproximadamente 150 s e mantida em torno desse valor até terminar o ensaio em 600 s.

## 6 REFERÊNCIAS BIBLIOGRÁFICAS

- BANZI, M. et al. Arduino - homepage, 2005. Disponível em: <www.arduino.cc>. Acesso em: 26 abr. 2013.
- BILOBROVEC, M.; MARÇAL, R. F. M.; KOVALESKI, J. L. Implementação de um sistema de controle inteligente utilizando a lógica fuzzy. XI SIMPEP. Bauru, SP: [s.n.]. 2004.
- FILHO, F. M. L.; GOSMANN, H. L.; BAUCHSPIESS, A. Controle Fuzzy para Sistema de Nível de Líquidos. XIV Congresso Brasileiro de Automática. Natal, RN: [s.n.]. 2002. p. 3017-3022.
- SHAW, I. S.; SIMÕES, M. G. Controle e Modelagem Fuzzy. 1ª ed. ed. São Paulo, SP: Edgard Blücher Ltda, 1999.

## AGRADECIMENTOS

O autor agradece ao Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte - IFRN - *Campus Ceará-Mirim* pelos materiais e instalações cedidos para realização deste trabalho.