

CONTROLE DE TEMPERATURA DE BAIXO CUSTO UTILIZANDO LÓGICA FUZZY E DIMMER DIGITAL

Fulana de tal¹, Fulano de tal², Fulano de tal³, Fulana de tal⁴

¹Discente de graduação em agronomia -IFAL. Bolsista de Iniciação Científica do CNPq/IFAL. e-mail: fulanadetal@gmail.com; ²Discente do curso técnico integrado em agroecologia - IFAL. Bolsista PIBIC do IFAL. e-mail: fulanodetal@hotmail.com; ³Professor do Curso de Agronomia - IFAL. e-mail: fulanodetal@ifal.edu.br; ⁴Professorado curso técnico integrado em agroecologia- IFAL. e-mail: fulanadetal@ifac.edu.br

1 **RESUMO:** Este trabalho apresenta a placa Arduino no controle de temperatura de um
2 ambiente de baixo custo. Um sistema térmico em escala reduzida foi montado para a
3 implementação de um controle de temperatura utilizando a Lógica *Fuzzy*. A placa Arduino
4 ligou uma lâmpada, por meio de um dimmer digital que controlou o brilho da mesma, e assim
5 controlou a temperatura no seu arredor. As leituras de temperatura oscilaram próximo da
6 referência predefinida de 35 °C. Os resultados dos ensaios experimentais mostraram que é
7 possível utilizar a Lógica *Fuzzy* e a placa Arduino para controlar a temperatura de um
8 ambiente fechado, uma vez que esse tipo processo possui uma baixa velocidade na ação de
9 controle.

10 **Palavras-chave:** acionamento, Arduino, *dimmer*, prototipagem, sensor.

11

TEMPERATURE CONTROL FROM LOW COST USING FUZZY LOGIC AND DIGITAL DIMMER

12

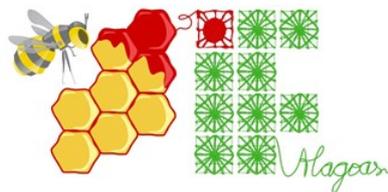
13
14
15 **ABSTRACT:** This work show the Arduino board in temperature control from low cost
16 environment. A termic system at reduce scale was mounted for implementation of the
17 temperatura control using Fuzzy Logic. The Arduino board turn on a lamp, through of the
18 dimmer, which controled the its brightness, and therefore controled the temperatura at its
19 surroundings. The temperature measurements oscillated near to reference of the 35 °C. The
20 results of the experimentals tests showed which is possible to use Arduino board with Fuzzy
21 Logic to controlling the temperatura a closed environment, once that process has a low speed
22 in control action.

23 **KEYWORDS:** actuation, Arduino, *dimmer*, prototyping, sensor.

24

INTRODUÇÃO

25
26 A plataforma Arduino, criada em 2005, é constituída basicamente por uma pequena
27 placa de circuito impresso. Tem como público alvo as pessoas que desejam criar protótipos de
28 eletrônica, objetos ou ambientes interativos. A placa pode receber sinais dos mais variados
29 tipos de sensores em suas entradas e pode interagir com sua vizinhança acionando luzes,
30 motores ou outros atuadores.



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

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

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

44

45 REVISÃO BIBLIOGRÁFICA

46 *Lógica Fuzzy*

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

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

58

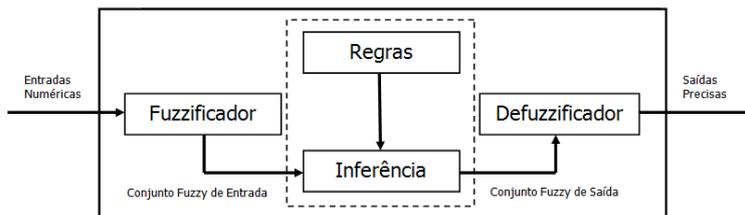
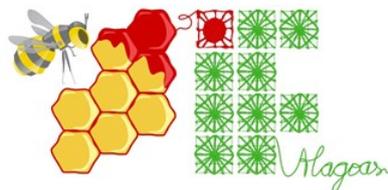


Figura 1: Blocos do Controlador *Fuzzy*.

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.

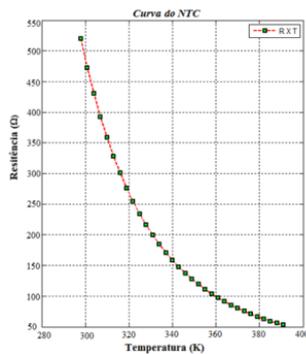
A inferência tem por característica transformar os valores linguísticos de entrada, junto com as regras, em valores linguísticos de saída.

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 e Simões, 1999).

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 Fig. 2(a). Já a Fig. 2(b) mostra o sensor *NTC* utilizado neste projeto.



a)



b)

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).

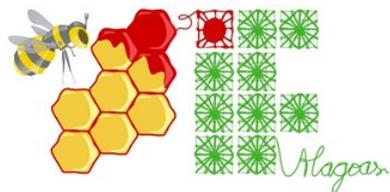
$$T(K) = \frac{1}{a + b \cdot \ln(R_{NTC}) + c \cdot (\ln(R_{NTC}))^3} \quad (3)$$

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).

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

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} ,



106 depois determinar a temperatura em Kelvin e por fim, converter de Kelvin para uma
107 temperatura em graus Celsius, conforme o **Código 1**.

108

```
//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=0.000234125*log(Rntc); // b*LN(RNTC)
c1=0.0000000876741*log(Rntc)*log(Rntc)*log(Rntc); //
c*(LN(RNTC)^3)
tempK=1/(0.001129148+b1+c1); //
1/(a+b*LN(RNTC)+c*(LN(RNTC)^3))
tempC=tempK-273.15; // converte p/ graus C
```

109

Código 1: Trecho de Código para Arduino.

110

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

113

114 *Plataforma Arduino*

115

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

120

121 Segundo **Banzi et al. (2005)** o Arduino é uma plataforma desenvolvida para criação de
122 protótipos de eletrônica. É uma placa de circuito impresso com entradas e saídas digitais;
123 entradas analógicas para utilização com sensores; pinos de alimentação; microcontrolador
124 ATmega328; e um conversor USB/Serial. Para programar uma placa Arduino é necessário
utilizar seu ambiente integrado de desenvolvimento baseado em *Processing*.

125

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

129

130 **METODOLOGIA**

131 *Montagem do Protótipo*

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

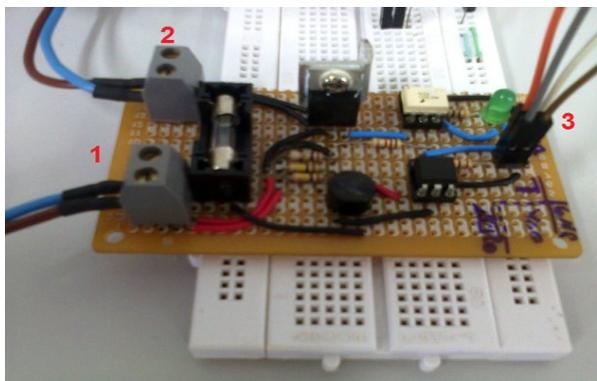
136



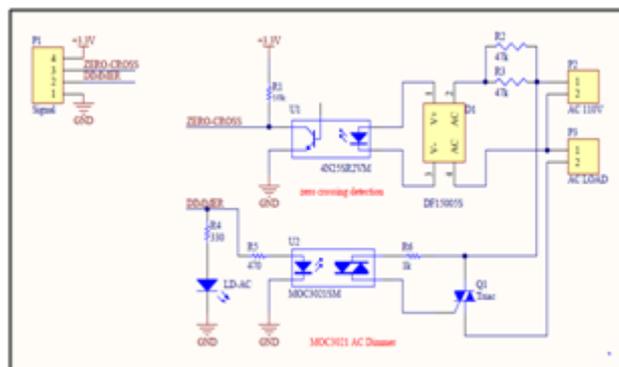
Figura 3: Ambiente em Escala Reduzida.

137
 138
 139
 140
 141
 142
 143
 144
 145
 146

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, passando por fusível. E por fim, os fios de controle que vem da placa Arduino foram ligados no borne 3 da placa dimmer, conforme a **Fig. 4(a)**. O esquema elétrico de montagem da placa dimmer pode ser visualizado na **Fig. 4(b)**.



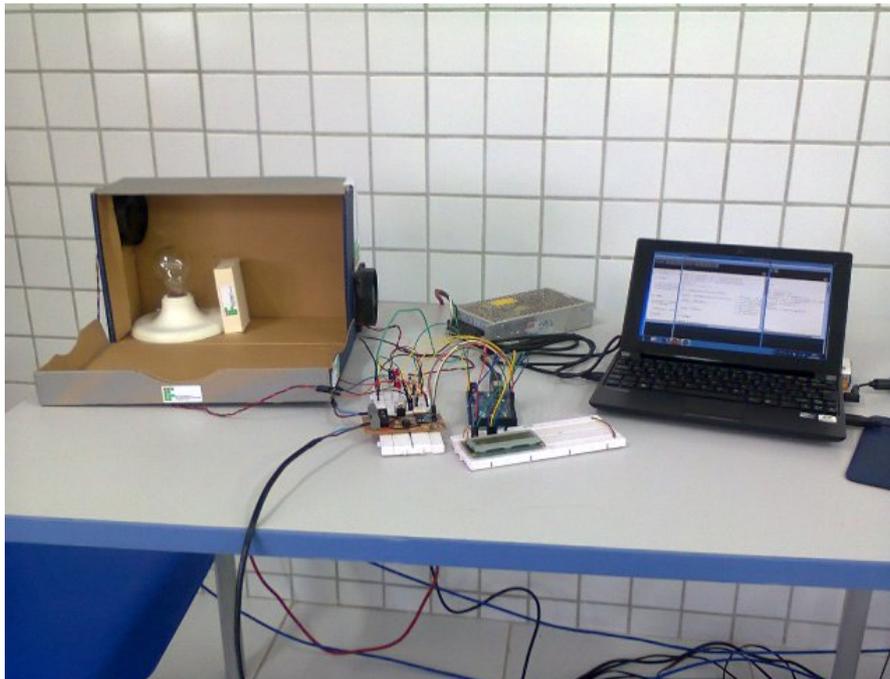
a)



b)

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

147
148 A **Figura 5** mostra todo o sistema proposto montado. Nela é possível observar as
149 conexões de todas as partes apresentadas anteriormente e presença de um netbook para
150 controlar todo o sistema, como também, para produzir os gráficos de temperatura *versus*
151 tempo.
152



153
154 **Figura 5:** Sistema em Escala Reduzida Proposto.
155

156 O funcionamento da placa *dimmer* no controle de brilho da lâmpada pode ser
157 visualizada em dois vídeos publicados na Internet. O primeiro vídeo pode ser visualizado pelo
158 seguinte endereço [LINK EDITADO](#) e o segundo vídeo foi publicado no seguinte endereço
159 [LINK EDITADO PARA NAO IDENFITICAR O AUTOR](#).

160 161 *Controlador Fuzzy*

162 O controlador foi configurado com uma variável de entrada e uma variável de saída. A
163 variável de entrada foi chamada de *temper* e foi definida com valores de 20 a 60 °C. A **Figura**
164 **6** mostra o conjunto *Fuzzy* gerado, onde os valores linguísticos possíveis para *temper* foram:
165 baixo, medbaixo, setpoint, medalta e alta.
166

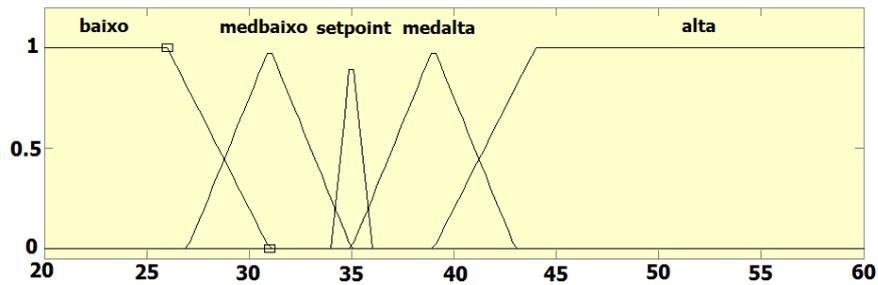


Figura 6: Variável de Entrada temper.

167
168
169
170
171
172
173
174
175

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.

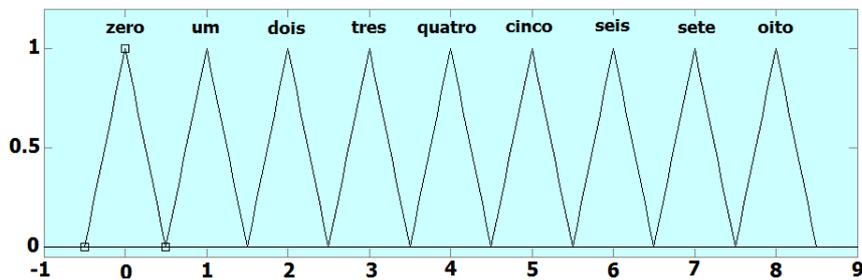


Figura 7: Variável de Saída output.

176
177
178
179
180
181

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

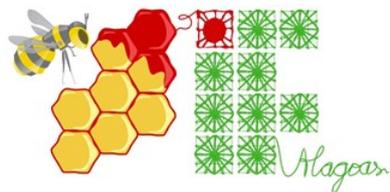
```

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.

182
183
184



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

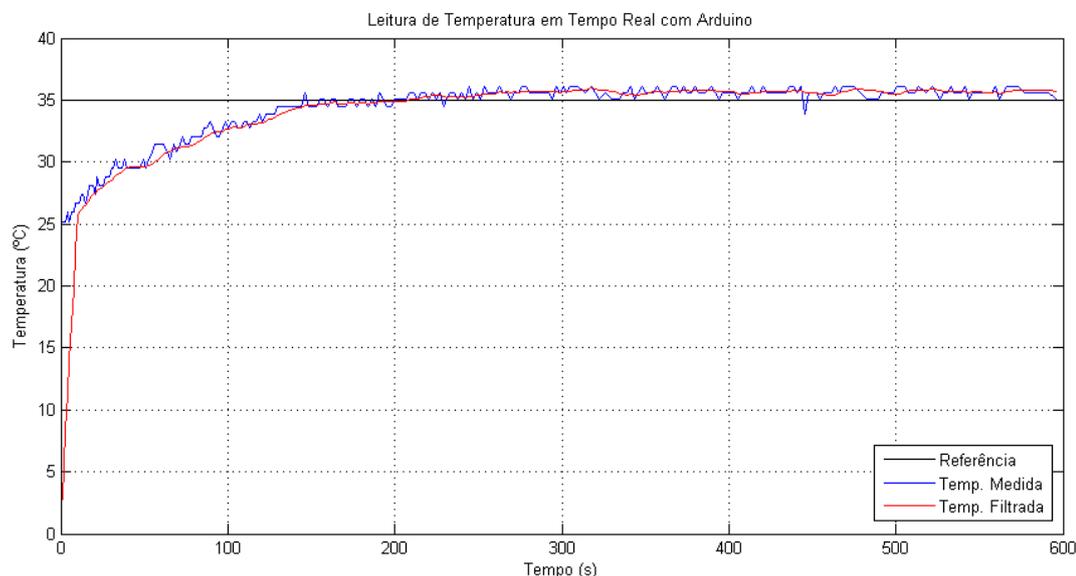
188

189 RESULTADO EXPERIMENTAL E DISCUSSÃO

190 Para comprovação do funcionamento do controle *Fuzzy* foi produzido um gráfico de
191 temperatura *versus* tempo, com base nos dados capturados pela porta Serial da placa Arduino.
192 A **Figura 9** mostra a evolução da temperatura com o controle inteligente funcionando. A
193 referência escolhida foi de 35 °C durante todo o ensaio de aproximadamente 600 segundos
194 (10 minutos).

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

199

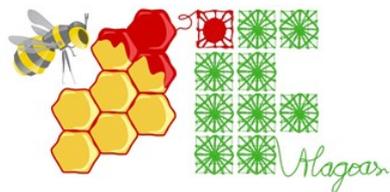


200

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

202

203 Com base no gráfico apresentado é possível visualizar o pleno funcionamento do
204 controle *Fuzzy* proposto. Estratégia de controle esta que pode ser embarcada na placa Arduino
205 por meio de estruturas condicionais IF ... THEN, de baixo custo computacional e de grande



206 aplicabilidade onde for necessário controlar grandezas de evolução lenta como a temperatura
207 ou o nível de um tanque, por exemplo. O que não requer grandes aparatos tecnológicos.

208

209 CONCLUSÕES

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

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

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

225

226 AGRADECIMENTOS

227

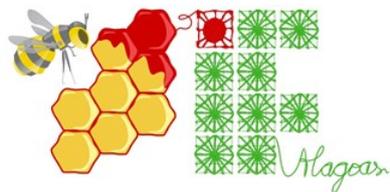
228 REFERÊNCIAS

229 BANZI, M. et al. Arduino - homepage, 2005. Disponível em: <www.arduino.cc>. Acesso em:
230 26 abr. 2013.

231 BILOBROVEC, M.; MARÇAL, R. F. M.; KOVALESKI, J. L. **Implementação de um**
232 **sistema de controle inteligente utilizando a lógica fuzzy**. XI SIMPEP. Bauru, SP,
233 2004.

234 FILHO, F. M. L.; GOSMANN, H. L.; BAUCHSPIESS, A. **Controle Fuzzy para Sistema de**
235 **Nível de Líquidos**. XIV Congresso Brasileiro de Automática. Natal, RN, 2002. p. 3017-
236 3022.

237 SHAW, I. S.; SIMÕES, M. G. **Controle e Modelagem Fuzzy**. 1ª ed. ed. São Paulo, SP:



XI CONNEPI

CONGRESSO NORTE NORDESTE DE PESQUISA E INOVAÇÃO
MACEIÓ · ALAGOAS · 2016

238 Edgard Blücher Ltda, 1999.

239