

CONTROLE *ON-OFF* DE UMA PLANTA DE CONTROLE DE TEMPERATURA

ON-OFF CONTROL OF A TEMPERATURE CONTROL PLANT

TARRASCO, Maite do Amaral ¹
WERNECK, Marcelo Pinheiro ²

RESUMO

O controle *on-off* apresenta baixo custo e é amplamente utilizado em aplicações industriais mais simples, como o controle de temperatura de um forno industrial. Atribuiu-se esse nome pelo motivo da variável manipulada poder ser alterada somente para o valor máximo ou mínimo. Este artigo consiste em apresentar os fundamentos do controle *on-off* e a aplicação da histerese, em malha fechada, na prática, por meio de um protótipo de controle de temperatura com Arduino Uno, um sensor de temperatura, um cooler para resfriamento da planta e uma lâmpada para aquecimento. Será apresentada a curva de funcionamento do sistema, as vantagens e desvantagens desta ação de controle, aplicação na indústria, além de um breve comparativo teórico com o controle PID.

Palavras-chave: Controle *on-off*; fundamentos; protótipo; controle de temperatura.

ABSTRACT

On-off control is low cost and widely used in simpler industrial applications, such as the temperature control of an industrial furnace. This name was attributed because the manipulated variable could be changed only to the maximum or minimum value. This article presents the fundamentals of on-off control and application of hysteresis in practice, using a prototype temperature control with Arduino Uno, a temperature sensor, a cooler for plant cooling and a light bulb for heating. The system operating curve will be presented, the advantages and disadvantages of this control action, application in the industry, and a brief theoretical comparison with the PID control.

Keywords: *On-off control, Methodology, Standards.*

¹Graduanda do Curso de Especialização em Automação e Controle Industrial da Universidade de Taubaté, maiteatarrasco@gmail.com; ²Professor orientador: Mestre, Universidade Federal de Itajubá, mpwerneck@gmail.com
São José dos Campos, outubro de 2018.

SIBi – Sistema Integrado de Bibliotecas / UNITAU

T192c Tarrasco, Maite do Amaral
Controle on-off de uma planta de controle de temperatura / Maite do
Amaral Tarrasco. -- 2018.
15 f. : il.

Monografia (Especialização) – Universidade de Taubaté, Departamento de
Engenharia Mecânica e Elétrica, 2018.

Orientação: Prof. Me. Marcelo Werneck, Departamento de Engenharia
Elétrica.

1. Controle de temperatura. 2. Controle on-off. 3. Protótipo. I. Título. II.
Especialização em Automação e Controle Industrial (Mecatrônica).

CDD – 621.5

INTRODUÇÃO

O controle automático de um processo industrial surgiu com a necessidade de se produzir com menor custo, maior qualidade, maior quantidade e menor tempo. O primeiro controlador automático industrial com realimentação, o regulador de esferas, foi desenvolvido por James Watt, em 1769, para o controle de velocidade das máquinas à vapor.

Todo processo industrial tem suas particularidades quanto às variáveis que devem ser controladas (temperatura, pressão, umidade) e o tipo de ação de controle a ser utilizada. A ação de controle é definida de acordo com o nível de robustez e precisão desejada para o processo. Existem 4 tipos de ações básicas de controle que podem ser utilizadas isoladamente ou associadas entre si, são elas: *on-off*, proporcional, integral e derivativa.

A ação de controle *on-off*, ou comumente chamada por ação de duas posições, é amplamente utilizada em aplicações industriais e domésticas por apresentar um baixo custo e menor complexidade de implementação. Neste tipo de controle a variável manipulada é alterada para o valor máximo (100%) ou mínimo (0%) de acordo com o valor desejado. Como exemplo de aplicação doméstica tem-se o controle de temperatura de forno elétrico e para aplicação industrial um controlador de nível de água de tanques.

Este trabalho apresentará o princípio de funcionamento da ação de controle *on-off* em malha fechada e os conceitos de histerese, por meio de uma aplicação em um protótipo de controle de temperatura. O *hardware* utilizado no protótipo é um Arduino Uno, para leitura da temperatura do ambiente foi utilizado um sensor de temperatura LM35 e como atuadores foram utilizados um cooler 12V (para resfriamento da planta) e uma lâmpada 12V (para aquecimento da planta), além de seus respectivos circuitos elétricos. A programação foi implementada através do *software* Arduino IDE, o qual utiliza uma linguagem baseada no C/C++.

MATERIAL E MÉTODOS

O protótipo (figura 1) desenvolvido para este trabalho foi construído em uma caixa de plástico para simulação de um ambiente fechado e melhor observação dos resultados. Como dito anteriormente, para resfriar o sistema foi utilizado como atuador um cooler 12V que foi fixado em uma das laterais da caixa. Na superfície da caixa, no lado oposto do cooler, foi fixada a lâmpada CC 12V, para atuar no aquecimento do sistema. O sensor de temperatura LM35 foi fixado no meio da caixa, entre a lâmpada e o cooler e tem como finalidade a medição e realimentação do valor de temperatura do ambiente para ser comparado com o setpoint definido na programação. Nas superfícies inferior, superior e laterais da caixa foi colada uma manta térmica com a finalidade de diminuir a troca de calor com o ambiente.

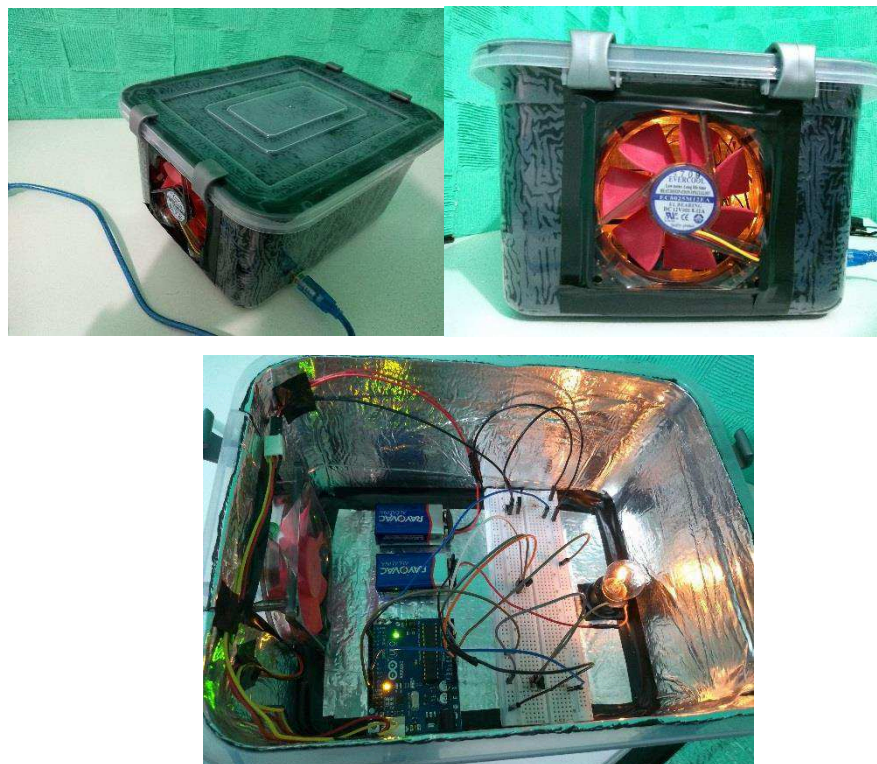


Figura 1. Protótipo para controle de temperatura

Fonte: (AUTOR, 2018)

Em uma protoboard foi montado o circuito elétrico necessário para atuação do sistema. Na figura 2 encontra-se o circuito utilizado neste protótipo.

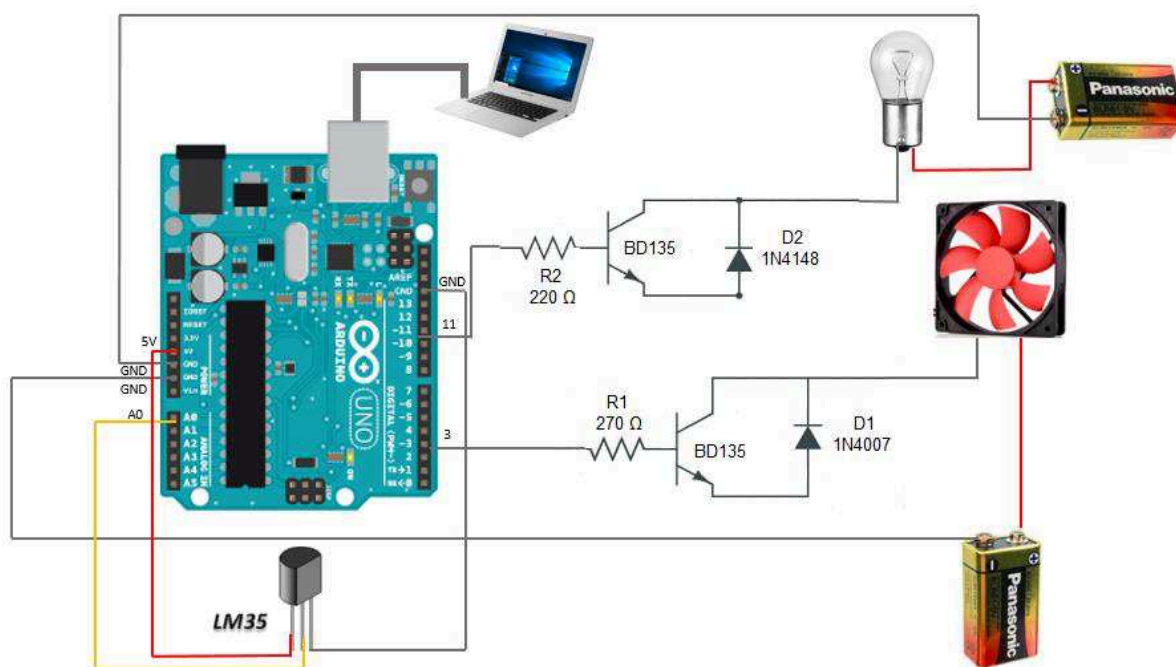


Figura 2. Circuito elétrico

Fonte: (AUTOR, 2018)

Para controle da lâmpada e do cooler foram utilizados dois transistores BD135 que permitem a “regulagem” do fluxo de corrente. Além disso, estes atuadores estão conectados aos pinos PWM 3 e 11 do Arduino, respectivamente. Esta definição foi feita com a finalidade de uma futura implementação de controle de velocidade do cooler e da intensidade da lâmpada por meio de PWM e controle PID. Para esta aplicação de controle *on-off* está sendo considerado o valor 0 (mínimo) do pwm como desligado e o valor 255 (máximo) como ligado.

Com a finalidade de proteção do circuito foram utilizados os diodos 1N4007 (para o circuito do cooler) e 1N4148 (para o circuito da lâmpada), que só conduzem corrente por um sentido e não permitem que a corrente reversa, gerada quando o circuito é desligado, danifique o circuito.

Para alimentação dos circuitos da lâmpada e do cooler foram utilizadas 2 baterias 9V e para o Arduino Uno a alimentação é através do notebook.

No quadro 1 tem-se a lista de componentes utilizados para construção do protótipo.

Componentes	Quantidade
Arduino Uno	1
Protoboard	1
Sensor de temperatura LM35	1
Cooler 12V	1
Lâmpada CC 12V	1
Transistor BD135	2
Diodo 1N4007	1
Diodo 1N4148	1
Bateria 9V	2

Quadro 1. Componentes utilizados no protótipo

Fonte: (AUTOR, 2018)

DESENVOLVIMENTO

O controle *on-off* (liga-desliga)

Basicamente o objetivo de uma ação de controle de malha fechada é medir a variável do processo (através do sensor), comparar com o valor desejado (setpoint) e corrigir por meio da variável manipulada por meio da ação de controle. Assim, o circuito deste projeto tem como função medir a temperatura do ambiente (através do sensor de temperatura), comparar com a temperatura definida como setpoint e corrigir enviando um valor de tensão para os atuadores de acordo com os cálculos do controle *on-off*. Na figura 3 tem-se o diagrama de blocos deste trabalho.

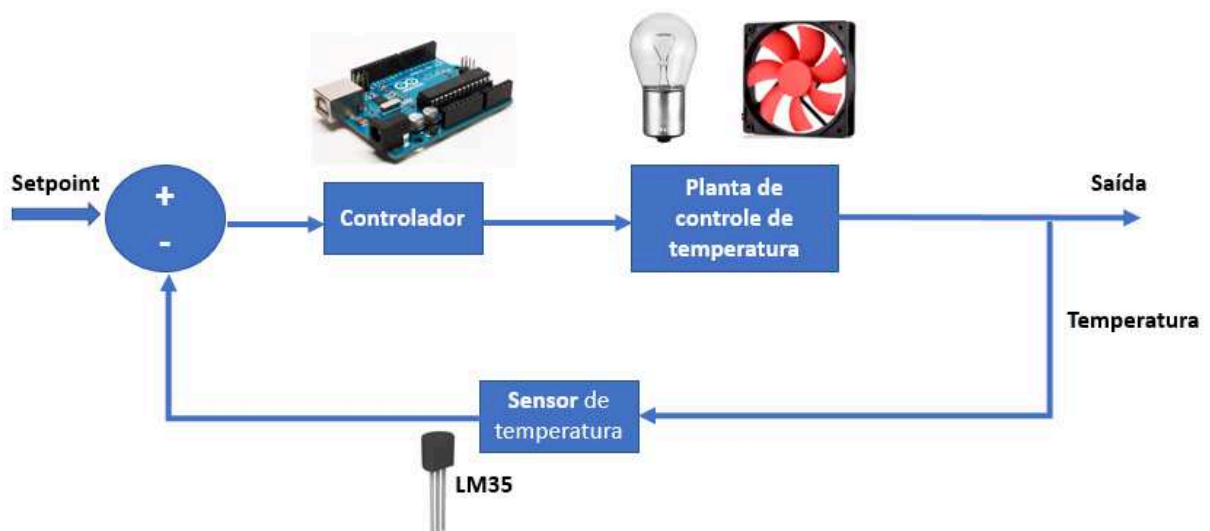


Figura 3. Diagrama de blocos

Fonte: (AUTOR, 2018)

Como dito inicialmente, o controle on-off, ou controle de duas posições, apresenta somente dois valores: ligado (100%), desligado (0%). Esses valores são enviados para os atuadores do sistema após o cálculo do erro entre o valor da variável do processo (temperatura lida pelo sensor) e o setpoint definido. O erro é calculado através da seguinte equação:

$$\text{Erro} = \text{VP} - \text{SP}$$

onde:

SP: Setpoint

VP: Variável do processo (temperatura lida pelo sensor)

Para o controlador de temperatura deste projeto, o valor do erro indicará qual dos atuadores deverá ser acionado: cooler ou lâmpada.

Se $\text{erro} < 0$, significa que a temperatura lida pelo sensor (VP) está mais baixa que o setpoint, logo é necessário que a lâmpada acenda (100%, ou seja, na intensidade mais alta) para iniciar o aquecimento do sistema.

Se $\text{erro} > 0$, significa que a temperatura lida pelo sensor (VP) está mais alta que o setpoint, logo é necessário que o cooler ligue (100%, ou seja, na velocidade mais alta) para iniciar o resfriamento do sistema.

Histerese

Segundo James Alfred Ewing, a histerese é a capacidade de preservar uma deformação gerada por um estímulo. A histerese de um sensor é importante quando é necessário a leitura de variáveis que mudam rapidamente e a utilização desses valores para o controle de algum dispositivo. Um exemplo típico é o caso de um sensor de temperatura utilizado em um termostato, que sem a histerese ou com uma histerese muito pequena, o circuito oscila rapidamente ligando e desligando em função da rápida variação de temperatura.

Quando se inclui a histerese ao sistema, o acionamento dos dispositivos é efetuado baseado na comparação da variável do processo com o setpoint +/- a histerese, como pode ser visto nas equações abaixo e na figura 4, que ilustra o conceito de histerese:

Se $VP < (SP - \text{histerese})$, ou $\text{erro} < (0 - \text{histerese})$: aciona lâmpada;

Se $VP > (SP + \text{histerese})$, ou $\text{erro} > (0 + \text{histerese})$: aciona cooler.

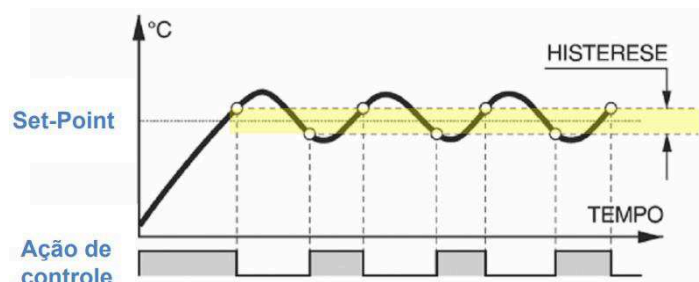


Figura 4. Gráfico explicativo do conceito de histerese

Fonte: (Prof. Me. Eng. RODRIGO CARDOZO FUENTES, 2018)

Na figura 5 encontra-se o fluxograma deste projeto.

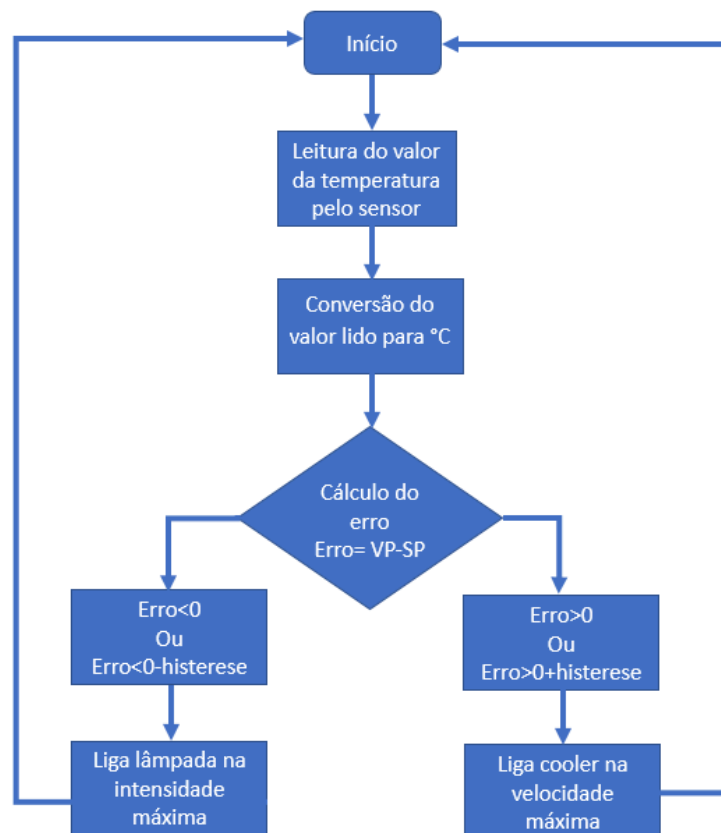


Figura 5. Fluxograma da programação

Fonte: (AUTOR, 2018)

Testes

Os testes realizados com o protótipo consideraram sempre um setpoint mais próximo da temperatura média do dia, devido à pouca variação de temperatura permitida pelo cooler e lâmpada utilizados. Assim, os testes apresentados neste trabalho foram com o setpoint de 22°C, sendo que a temperatura média do dia estava em torno de 22,5°C.

Para os gráficos mostrados ao longo da descrição dos testes que foram realizados (figuras 7, 8 e 9), considerar as legendas abaixo:

Legenda do gráfico do teste 1		Legenda dos gráficos dos testes 2 e 3	
	Curva de variação da temperatura em relação ao tempo		Setpoint
	Setpoint		Limite inferior da histerese
			Limite superior da histerese
			Curva de variação da temperatura em relação ao tempo

Figura 6. Legenda dos gráficos

Fonte: (AUTOR, 2018)

Teste 1: Sem histerese

Inicialmente o teste foi realizado sem considerar a histerese, logo a lógica de programação foi construída somente em relação ao erro, como descrito acima.

Se $\text{erro} < 0$, então liga a lâmpada, ou seja, a lâmpada será acionada quando a temperatura estiver menor que o setpoint, 22°C.

Se $\text{erro} > 0$, então liga o cooler, ou seja, o cooler será ligado quando a temperatura estiver maior que o setpoint, 22°C.

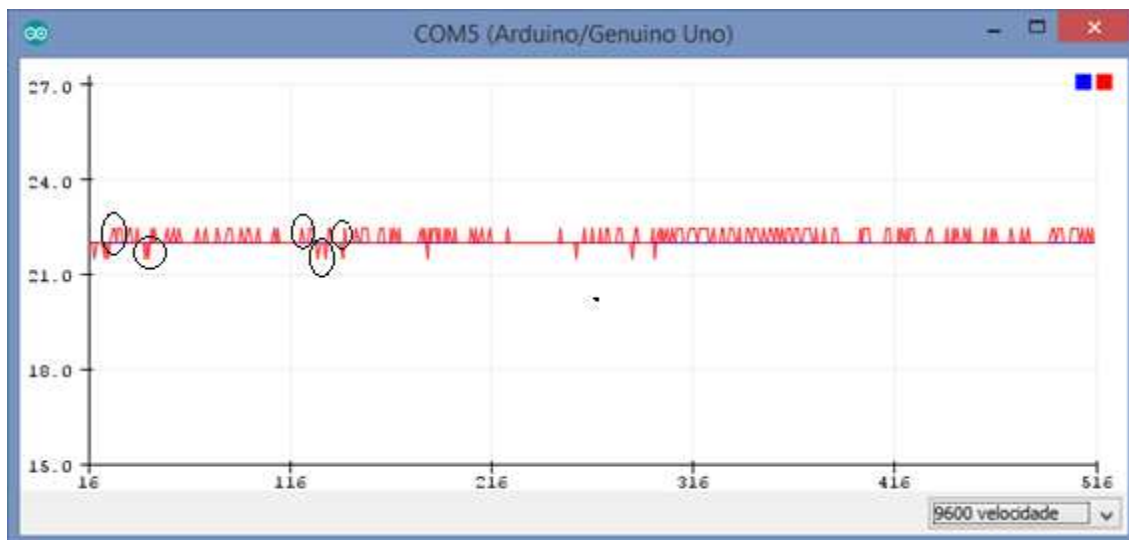


Figura 7. Gráfico do teste realizado sem considerar a histerese

Fonte: (ARDUINO IDE, 2018)

Assim, o ventilador e a lâmpada ficam ligando e desligando em toda inversão do gráfico, ou seja, em todo momento que a temperatura é maior que o setpoint o cooler aciona e fica acionado até o momento que a temperatura se torna menor que o setpoint. Ou seja, qualquer variação mínima em torno do setpoint já aciona a lâmpada ou o cooler.

No gráfico da figura 7, estão circulos em preto alguns momentos de inversão do gráfico para mostrar os acionamentos repetitivos da lâmpada e do cooler em um curto intervalo de tempo. Este movimento faz com que a lâmpada fique piscando e o cooler tenha acionamentos muito rápidos que não permitem à ele alcançar a velocidade máxima antes de desligar.

Teste 2: Histerese de 0,25°C.

Neste teste foi considerada uma histerese de 0,25°C para observar o comportamento do sistema quanto aos movimentos repetitivos de acionamento da lâmpada e do cooler. A programação foi construída conforme descrito acima, quando se considera a histerese.

Se $Erro < -0,25$: aciona lâmpada, ou seja, a lâmpada só será acionada quando a temperatura estiver abaixo de 21,75 °C (SP - histerese).

Se $\text{Erro} > +0,25$: aciona cooler, ou seja, o cooler só será ligado quando a temperatura estiver acima de $22,25^{\circ}\text{C}$ ($\text{SP} + \text{histerese}$).

Logo, o acionamento permanecerá no estado anterior até que atinja essa margem de variação de no máximo $0,5^{\circ}\text{C}$, fazendo com que os dispositivos sejam comutados num intervalo de tempo um pouco maior que o descrito no teste 1, pois não basta somente atingir o setpoint e sim o setpoint \pm a histerese, como pode ser visto no gráfico da figura 8.

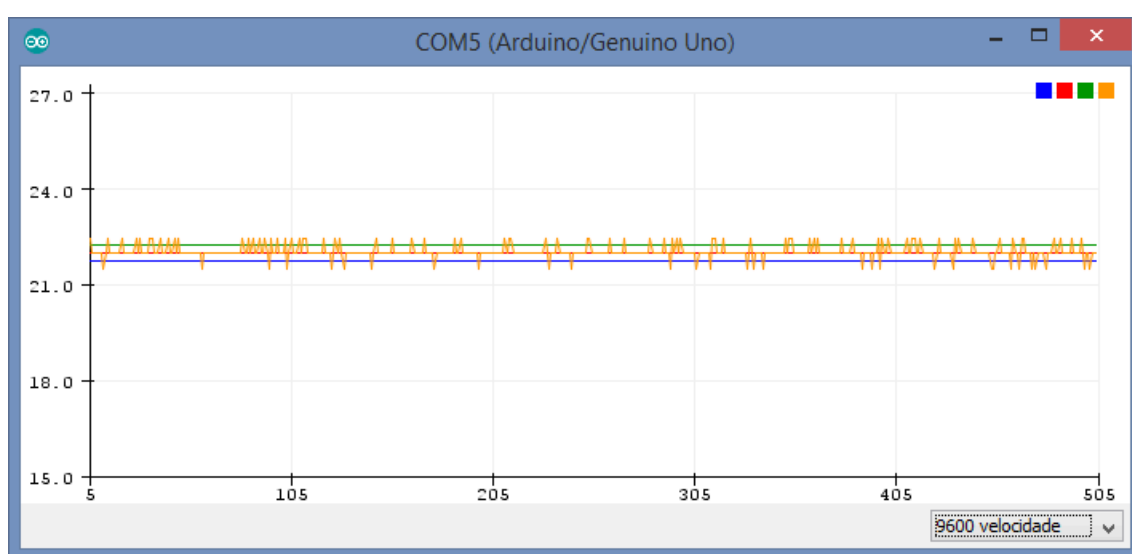


Figura 8. Gráfico do teste realizado considerando histerese de $0,25^{\circ}\text{C}$.

Fonte: (ARDUINO IDE, 2018)

Teste 3: Histerese de $0,50^{\circ}\text{C}$.

Como no teste 2 ainda foi observado uma frequência alta de acionamentos repetitivos dos dispositivos (apesar de ter diminuído o intervalo de tempo desses acionamentos em comparação com o teste 1), foi realizado mais um teste aumentando ainda mais a histerese com a finalidade de eliminar os movimentos repetitivos de acionamento da lâmpada e do cooler. Logo:

Se $\text{Erro} < -0,50$: aciona lâmpada, ou seja, a lâmpada só será acionada quando a temperatura estiver abaixo de $21,5^{\circ}\text{C}$ ($\text{SP} - \text{histerese}$).

Se $\text{Erro} > +0,50$: aciona cooler, ou seja, o cooler só será ligado quando a temperatura estiver acima de $22,5^{\circ}\text{C}$ ($\text{SP} + \text{histerese}$).

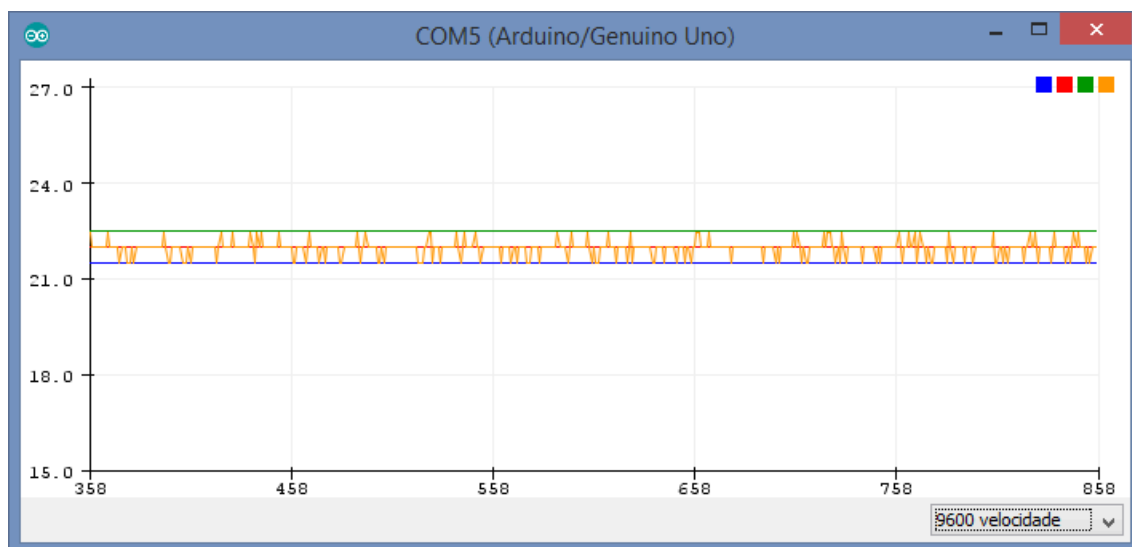


Figura 9. Gráfico do teste realizado considerando histerese de $0,50^{\circ}\text{C}$.

Fonte: (ARDUINO IDE, 2018)

Neste caso, o acionamento permanecerá no estado anterior até que atinja essa margem de variação de no máximo 1°C . Como pode ser visto no gráfico da figura 9, a frequência de acionamento dos dispositivos diminuiu bastante em relação aos gráficos dos testes 1 e 2, tornando o sistema menos suscetível à desgaste dos componentes.

RESULTADOS E DISCUSSÃO

Através dos testes realizados concluiu-se que para processos onde as variáveis (como a temperatura) tenham uma rápida mudança de valores, é essencial que seja incluída a histerese para se obter um controle de temperatura onde os dispositivos não tenham acionamentos repetitivos frequentes que possam deteriorá-los ou ocasionar danos no circuito com o tempo. Além disso, o valor da histerese deve ser determinado de acordo com as particularidades de cada projeto.

Além disso, os componentes, dispositivos e circuitos devem sempre ser dimensionados considerando-se a aplicação e quantidade de troca de calor necessária para que se atinja o valor da variável desejada no tempo estipulado.

O protótipo desenvolvido atendeu às necessidades desse projeto possibilitando a visualização de uma aplicação de controle on-off de temperatura, os efeitos da histerese, a influência do ambiente no controle de temperatura, além do aprendizado na implementação de um projeto com a plataforma Arduino, que se mostrou satisfatória para um projeto de especialização.

CONSIDERAÇÕES FINAIS

A vantagem de se utilizar o controle *on-off* é a praticidade de implementação e o baixo custo e esse controle é utilizado em aplicações que permitem variação em torno do setpoint. A principal desvantagem do uso do controle *on-off* é a provocação de oscilações no processo, devido ao fato de a correção da variável do processo não ser feita considerando-se a intensidade do desvio (erro), o que gera um erro de off-set, ou seja, um erro residual no sistema.

Na indústria o controle *on-off* é muito utilizado para controle de temperatura por meio de termostatos eletromecânicos ou controladores eletrônicos, um exemplo dessa aplicação é em fornos industriais de convecção e o principal objetivo do controle de temperatura neste caso é garantir a qualidade dos alimentos processados.

Como trabalho futuro, sugere-se a melhoria deste protótipo para utilização em aulas práticas de cursos de Engenharia ou pós graduação, com a finalidade de deixá-lo independente do notebook, implementando um potenciômetro para alteração do setpoint, um display para visualização dos valores de setpoint, histerese, erro e temperatura lida pelo sensor e uma fonte para alimentação do Arduino Uno.

Sugere-se ainda, como próximos trabalhos, a determinação dos ganhos e a implementação do controle PID para que se obtenha um resultado menos oscilatório (somente até estabilizar) e mais preciso, eliminando o erro de off-set. Na figura 10

apresenta-se graficamente uma breve comparação do controle *on-off* com o controle PID.

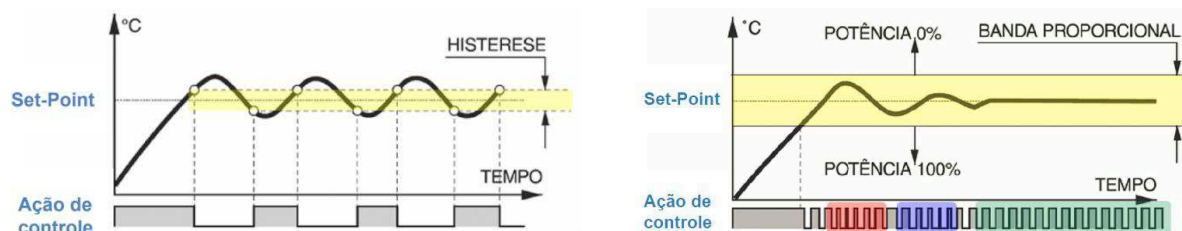


Figura 10. Gráfico comparativo controle *on-off* x PID

Fonte: (Prof. Me. Eng. RODRIGO CARDOZO FUENTES, 2018)

A implementação controle PID neste protótipo possibilitará o aprendizado em uma técnica de controle muito utilizada em diversos processos industriais.

REFERÊNCIAS

ARDUINO. **Documentação de referência da linguagem arduino**. Disponível em: <<https://www.arduino.cc/reference/pt/>>. Acesso em 23 ago. 2018.

BELTRAME, R. C; FUENTES, R.C. **Automação industrial – estratégias de controle**. UNIVERSIDADE FEDERAL DE SANTA MARIA, 2017, São Paulo. p. 5-10

BORGES, M. M. **Sistema didático para aplicação de técnicas de controle**. 2016. 60 p. Trabalho de conclusão de curso – Centro Federal de Educação Tecnológica de Minas Gerais. Minas Gerais, 2016.

BRAZILAUTOMATION. **Comparação entre sistemas de controle para fornos industriais**. SÃO PAULO, 2014. Disponível em: <www.brazilautomation.com.br> Acesso em 10 out. 2018.

BUENO, G. **Controle de iluminação e temperatura pela plataforma Arduino via celular**. 2015. 70 p. Trabalho de conclusão de curso – Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2015.

CITISYSTEMS. **Como funciona o controle de temperatura?** Disponível em: <<https://www.citisystems.com.br/controle-de-temperatura/>> Acesso em 27 jul. 2018.

CORREA,P. **Fundamentos de instrumentação e controle**. Rev. Jul. 2012

LABDEELETRONICA. **Controle de velocidade de motor CC com arduino**. Disponível em:<<http://labdeeletronica.com.br/controle-de-velocidade-motor-cc/>> Acesso em 22 ago. 2018.

MEDIGA. **Controle utilizando PWM**. Disponível em: <<https://mediga.com.br/controle-utilizando-pwm/>> Acesso em 31 ju. 2018

NOVUS. **Controle PID básico**. Disponível em: <www.novus.com.br> Acesso em 05 set. 2018.

OLIVEIRA, A. **Instrumentação: Fundamentos de controle de processos**. SENAI – SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL, 1991, Espírito Santo.

ORLEM, L. D. S.; JÓ, D. S. M. J.; MENDELSSON R. M. N. **Sistema de controle de temperatura para uma estufa com monitoramento via aplicativo**. RCT – Revista de ciência e tecnologia. UFRR. v.3.n.4. Roraima, 2017.

VIDADESILICIO. **Sensor de temperatura LM35 – Medindo temperatura com arduino**. Disponível em: <<https://portal.vidadesilicio.com.br/lm35-medindo-temperatura-com-arduino/>> Acesso em 27 jul. 2018

WIKISCIENCEBR. **A história do controle automático**. Disponível em: <<http://wikisciencebr.blogspot.com/2015/03/a-historia-do-controle-automatico.html>>. Acesso em 03 out. 2018.