

CONTROLE PID UTILIZANDO O ARDUINO

Wesley Yoshiharu Oliveira Suzuki ¹

Egídio Raimundo Neto ²

RESUMO

Este projeto consiste em desenvolver um sistema de controle para uma planta que tem por finalidade manter a temperatura de um ambiente, o controlador utilizado será uma ferramenta muito empregado no meio industrial, o Proporcional, Integral, Derivativo (PID). Para a elaboração do sistema serão utilizados em conjunto, uma placa eletrônica construída com base no microcontrolador Atmel ATmega328, e um circuito integrado capaz de detectar variações térmicas, o sensor LM35. Assim abordado o processamento digital de sinais, além de conceitos eletrônicos tanto da parte analógica quanto da parte digital, assim como estruturas de programação em linguagem C/C++. Todos os conceitos serão aplicados à fim de gerar gráficos matemáticos em função do tempo.

Palavras-Chave: Controle. PID. Processamento. Micro-controlado.

¹ WESLEY YOSHIHARU OLIVEIRA SUZUKI: Graduando em Engenharia de Controle e Automação - FAEX

² EGÍDIO RAIMUNDO NETO: Mestre em Telecomunicações, Engenheiro Eletricista - INATEL

ABSTRACT

This project aims to develop a controller to a system used to keep the temperature of an environment, it will be used a typical industrial tool, the PID. For that will be used, an electronic board based in the microcontroller Atmel ATmega328, and an integrated circuit that detects the temperature variations, the sensor LM35. Will be discussed the digital signal processing, analog and digital concepts of eletronic, and also programming structures in C/C++ language. All concepts will be applied to generate mathematical graphics in function of time.

Palavras-Chave: Control. PID. Processing. Microcontroler.

INTRODUÇÃO

Atualmente os sistemas de controle movem muitos processos industriais que exigem precisão de medidas. Nas indústrias alimentícias é fundamental que a temperatura, a umidade ou mesmo a pressão de diversos processos esteja dentro dos padrões, pois variações além da tolerância podem comprometer gravemente o produto final.

Dentre os sistemas de controle utilizados nos processos de engenharia estão os controladores proporcional, proporcional integral, proporcional derivativo, entre outros. Pode-se destacar entre todos os controladores o PID que é o mais aplicado industrialmente devido a agilidade de parametrização e sua resposta satisfatória.

O objetivo deste artigo é apresentar de forma clara o desenvolvimento de um controlador PID para um sistema de primeira ordem, proveniente de uma planta para monitoramento de temperatura. O processamento será realizado de forma digital com um microcontrolador Atmel Atmega 328.

A fim de possibilitar o desenvolvimento de um algoritmo para implementação no microcontrolador, desenvolveu-se através de artifícios matemáticos a dedução de uma expressão capaz de representar o processo realizado por um controlador PID.

A partir de alguns experimentos elaborados em ambientes reais obteve-se os dados através de um sensor de temperatura em conjunto com uma interface, capaz de permitir a comunicação do sensor com o microcontrolador. A interface constituída por um circuito normalizador baseado em um amplificador operacional. O dispositivo controlado foi acionado com o auxílio de uma interface a base de transistor permitindo a modulação da potência necessária para o aquecimento do ambiente.

Em todo sistema de controle há a possibilidade de ocorrer distúrbios, estes sendo causados devido a condições que o projetista não pode manipular, como a temperatura ambiente ou a circulação de ar presente em determinadas plantas. Para desconsiderar os distúrbios construiu-se uma caixa para isolar a planta do sistema, onde adicionou-se apenas um resistor responsável pelo

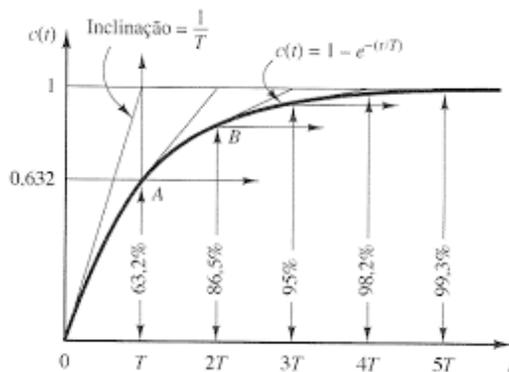
aquecimento do ambiente de estudo e o sensor de temperatura. Sendo assim este artigo visa apresentar de maneira simplificada o projeto de um controlador PID para um sistema em malha aberta.

FUNDAMENTAÇÃO TEÓRICA

Função de transferência

A função de transferência representa o comportamento de um sistema no domínio da frequência, e segundo Bolton (Engenharia de Controle 1ªed. 1995) é possível obter a função de transferência de um sistema de primeira ordem observando sua resposta ao degrau, a seguir um exemplo genérico:

Figura 9 - Resposta de sistema de 1ª ordem



$$H(s) = \frac{1}{T \cdot s + 1} \quad (1)$$

Fonte: Sinais e Sistemas – CASSIANO RECH e RAFAEL CONCATTO BELTRAME

Utilizando a expressão $y(T) = 0,632 \cdot [y(\infty) - y(0)] + y(0)$, é possível obter $y(T)$ e T com os valores de $y(\infty)$ e de $y(0)$ que são visíveis através da resposta ao degrau. Porém na prática há um fator chamado de atraso de transporte, dado pela letra grega θ , e ele não está incluso neste método. Pode-se utilizar o método de Ziegler-Nichols em malha aberta (0) para a obtenção dos dados reais.

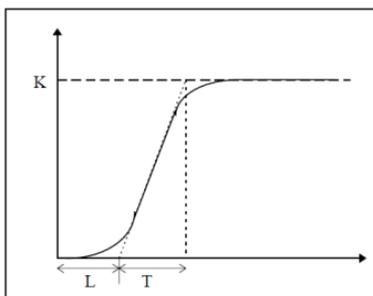
Atraso de transporte

Atraso de transporte é o tempo que um sistema leva para responder a uma entrada, ou seja, o sistema recebe um sinal de excitação em sua entrada, mas sua saída não apresenta uma variação imediata de acordo com esta excitação de entrada.

O atraso de transporte é definido como o tempo após a ocorrência de uma perturbação na entrada até que seja notada uma mudança na saída. (Identificação de Sistemas de Primeira Ordem com Atraso de Transporte, 2006)

Para sistemas cujo o gráfico de resposta se assemelha à um S, como o estudado neste artigo, pode-se utilizar o método da curva de reação, mais conhecido por Ziegler-Nichols de malha aberta.

Figura 10 - Exemplo de curva de reação



$$H(s) = \frac{K}{T \cdot s + 1} \cdot e^{-L \cdot s} \quad (2)$$

Fonte: Métodos de sintonia em controladores PID – FELIPE DE ANDRADE SARAIVA, 2011

Fonte: Autoria própria

O atraso de transporte e a constante de tempo na Figura 10 são dados respectivamente por L e T , mas também são comumente apresentados pelas letras gregas θ e τ , que serão adotadas neste artigo.

Controlador PID digital

A equação de controle PID para sinais digitais, Equação 4, possui algumas diferenças com relação à equação de controle PID para sinais analógicos, Equação 3, pois integrais e derivadas não são implementáveis em microcontroladores e microprocessadores.

$$u(t) = Kp \cdot e(t) + Ki \cdot \int_0^t e(\tau) d\tau + Kd \cdot \frac{de(t)}{dt} \quad (3)$$

Fonte: Engenharia de Controle – W. BOLTON, 1993

Para traduzir estas operações em modelos matemáticos que permitam uma descrição lógica em software é preciso utilizar de alguns artifícios de cálculo numérico, então tomando a regra dos trapézios para a integral, que consiste na média aritmética das somas a direita e a esquerda de Riemann, e para a derivada utilizando do conceito de diferenciação, obtêm-se a Equação 3. Onde n e $n - 1$ referem-se às amostras que serão utilizadas e T_a é a distância entre os pontos.

$$u[n] = Kp \cdot e[n] + \frac{Ki \cdot T_a}{2} \left\{ \sum_0^n e[n-1] + \sum_0^n e[n] \right\} + Kd \cdot \frac{e[n] - e[n-1]}{T_a} \quad (4)$$

Fonte: Controlador PID digital: Uma modelagem prática para microcontroladores – FELIPE NEVES, 2014

onde:

- **Kp**: Coeficiente da ação proporcional;
- **Ki**: Coeficiente da ação integral;
- **Kd**: Coeficiente da ação derivativa
- **T_a**: Período de amostragem;
- **n**: Amostra a ser processada;
- **u[n]**: Sinal de saída do sistema na amostra n;
- **e[n]**: Sinal de erro na entrada do controlador na amostra n;
- **e[n-1]**: Sinal de erro na entrada do controlador na amostra n-1.

Aplicando a transformada Z na Equação 4 é possível encontrar uma

função de transferência e a partir disto, pode-se encontrar uma equação de diferenças que permite uma implementação mais adequada do controle PID.

Considerando $T_a = 1$, pode-se dizer que:

$$KP = Kp \quad (5) \qquad KI = \frac{Ki}{2} \quad (6) \qquad KD = Kd \quad (7)$$

E por fim realizando-se as devidas manipulações matemáticas e reagrupando as variáveis para melhor visualização da equação, chega-se à:

$$G(z) = \frac{U(z)}{E(z)} = \frac{(KP + KI + KD) - (KP - KI + 2 \cdot KD) \cdot z^{-1} + KD \cdot z^{-2}}{(1 - z^{-1})} \quad (8)$$

$$b_0 = KP + KI + KD \quad (9) \qquad b_1 = -KP + KI - 2 \cdot KD \quad (10) \qquad b_2 = KD \quad (11)$$

$$G(z) = \frac{U(z)}{E(z)} = \frac{b_0 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2}}{1 - z^{-1}} \quad (12)$$

Extraindo a equação de diferenças a partir de $G(z)$, obtêm-se a expressão que pode ser implementada em qualquer microcontrolador ou microprocessador, a Equação 13.

$$u[n] = b_0e[n] + b_1e[n - 1] + b_2e[n - 2] - u[n - 1] \quad (13)$$

Fonte: Controlador PID digital: Uma modelagem prática para microcontroladores – FELIPE NEVES, 2014

Fonte: Autoria própria

METODOLOGIA

Obtenção da função de transferência do sistema

Com uma aquisição de dados via serial, foi possível verificar a resposta ao degrau do sistema em malha aberta através de simulações, onde pode-se

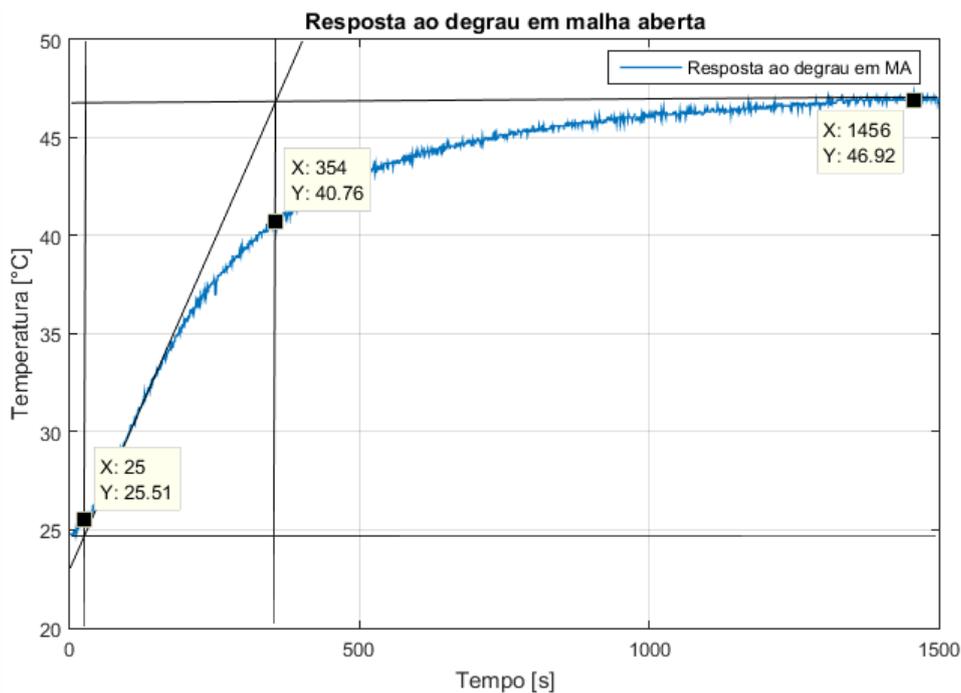
observar os valores para a obtenção de sua função de transferência, e conseqüentemente pode-se obter os dados necessários para a sintonia do controlador PID, então obteve-se $\theta = 25s$, $\tau = 329s$ e $k = 47^{\circ}C$, observar Figura 11.

$$H(s) = \frac{47}{329 \cdot s + 1} \cdot e^{-25 \cdot s} \quad (14)$$

E ainda utilizando a aproximação de Padé para funções de primeira ordem tem-se, após as devidas manipulações matemáticas, a função de transferência da planta:

$$H(s) = \frac{-0,1428 \cdot s + 0,0114}{s^2 + 0,083 \cdot s + 0,0002} \quad (15)$$

Figura 11 - Resposta ao degrau obtida por aquisição



Fonte: Autoria própria

Fonte: Experimento realizado

Tabela 1 - Tipos de parametrização para PID

Método	K_p	T_i	T_d
ZN	$\frac{1,2}{k} \cdot \frac{\tau}{\theta}$	$2 \cdot \theta$	$0,5 \cdot \theta$
CHR - 0%	$\frac{0,6}{k} \cdot \frac{\tau}{\theta}$	τ	$0,5 \cdot \theta$
CHR - 20%	$\frac{0,95}{k} \cdot \frac{\tau}{\theta}$	$1,357 \cdot \tau$	$0,474 \cdot \theta$
CC	$\frac{1}{k} \cdot \left(0,25 + 1,35 \cdot \frac{\tau}{\theta}\right)$	$\theta \cdot \left(\frac{1,35 + 0,25 \cdot \frac{\theta}{\tau}}{0,54 + 0,33 \cdot \frac{\theta}{\tau}}\right)$	$\frac{0,5 \cdot \theta}{1,35 + 0,25 \cdot \frac{\theta}{\tau}}$
ITAE - s	$\frac{0,965}{k} \cdot \left(\frac{\tau}{\theta}\right)^{0,85}$	$\frac{\tau}{0,796 + 0,1465 \cdot \frac{\theta}{\tau}}$	$\frac{0,308}{k} \cdot \tau \cdot \left(\frac{\theta}{\tau}\right)^{0,929}$
ITAE - r	$\frac{1,357}{k} \cdot \left(\frac{\tau}{\theta}\right)^{0,947}$	$\frac{0,965}{k} \cdot \left(\frac{\tau}{\theta}\right)^{0,85}$	$\frac{0,381}{k} \cdot \tau \cdot \left(\frac{\theta}{\tau}\right)^{0,995}$
	$2 \cdot \tau + \theta$	θ	$\tau \cdot \theta$

Fonte: Métodos de sintonia em controladores PID – FELIPE DE ANDRADE SARAIVA, 2011

Fonte: Autoria própria

A alocação dos polos pode ser realizada através de diversos métodos, a seguir pode-se verificar uma tabela (Métodos de Sintonia em Controladores PID, 2011, p. 28) utilizada quando se trabalha com um sistema de malha aberta, a mesma apresenta as equações utilizadas para a obtenção dos parâmetros K_p , T_i e T_d .

É importante ressaltar que T_i e T_d não são os parâmetros finais de ação integral e derivativa, estes são dados pelas relações a seguir:

$$K_i = \frac{K_p}{T_i} \quad (16)$$

$$K_d = K_p \cdot T_d \quad (17)$$

Conhecendo a Tabela 1 e possuindo os valores de θ , τ e k . Pode-se obter os valores dos coeficientes de controle. Para isto, foram selecionados apenas os métodos Ziegler-Nichols, CHR – 0% e CHR – 20%, os resultados foram obtidos por meio de simulação e apresentados na Tabela 2.

Tabela 2 - Coeficientes de sintonia PID

Métodos	Kp	Ki	Kd
ZN	0,336	0,00672	4,2
CHR - 0%	0,168	0,000511	2,1
CHR - 20%	0,266	0,000596	3,1521

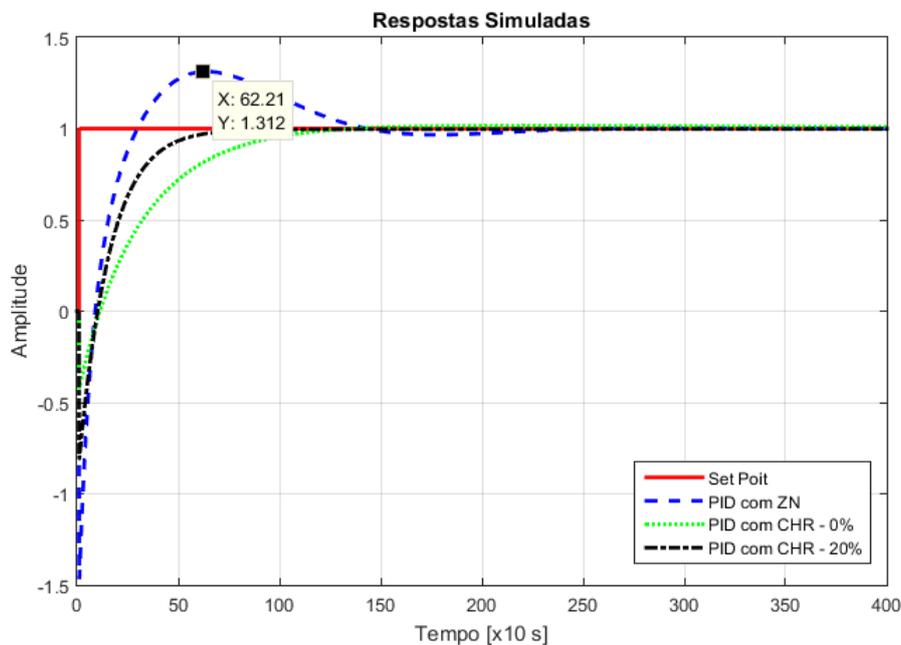
Fonte: Autoria própria

Simulações

Com base nos parâmetros encontrados na Tabela 2 e visando apresentar a resposta ao degrau do sistema quando utilizando alguns dos controladores relacionou-se os gráficos obtidos através de simulações, o que facilitou a escolha do método de sintonia mais interessante para aplicação.

Pode-se observar que a resposta do sistema com o PID parametrizado utilizando o método Ziegler-Nichols apresenta um sobressinal de aproximadamente 31,2%, por sua vez os controladores projetados com os métodos CHR não apresentam nenhum sobressinal.

Figura 12 - Respostas por simulação



Fonte: Experimento realizado

Assim, o controlador projetado com o método Ziegler-Nichols apresenta um tempo de acomodação superior aos outros, o que proporciona uma

visualização melhor sobre o funcionamento de um sistema com controle PID.

A partir dos resultados obtidos, optou-se por utilizar os parâmetros obtidos pelo método Ziegler-Nichols.

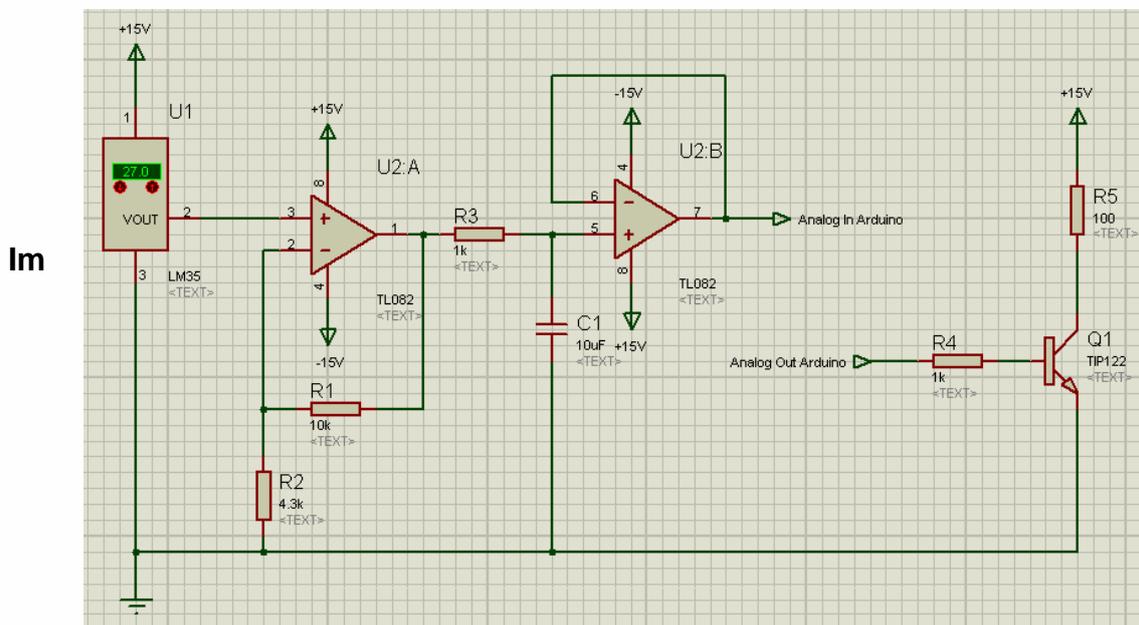
Interfaces

As entradas analógicas do Arduino apresentam 10 bits de resolução, logo tem-se o valor de entrada variando de 0 a 1023, onde 0 equivale a 0 volts de entrada e 1023 equivale a 5 volts de entrada, e as saídas PWM do Arduino apresentam a faixa de valores para modulação variando de 0 a 255, onde 0 equivale a 0 volts de saída e 255 equivale a 5 volts de saída.

O sensor LM35 trabalha na faixa de 0 a 150°C quando alimentado de forma direta com 15 volts, e entrega 10 milivolts para cada °C. Sendo assim realizou-se a montagem de um circuito normalizador para que a entrada do Arduino recebesse os níveis adequados de tensão.

Além do circuito normalizador também foi adicionado um filtro passa-baixa para evitar o maior número possível de ruídos. Para a saída novamente foi preciso adicionar um circuito de interface por conta dos níveis de tensão e corrente do componente responsável por aquecer a planta, um resistor com potência de 100 watts. Para que a saída PWM do Arduino pudesse chavear o resistor utilizou-se um transistor com configuração Darlington.

Figura 13 - Circuitos normalizadores de entrada e saída



A implementação no Arduino realizou-se com a Equação 13 e optou-se pelo método Ziegler-Nichols em malha aberta para sintonia do controlador PID, pois este apresentou a resposta com maiores características de sobressinal e oscilação.

O set point foi ajustado para 37°C e as constantes de controle foram definidas com base nos valores apresentados na Tabela 2. Optou-se por realizar o arredondamento dos valores para 2 casas decimais, o que proporcionou os valores $Kp = 0,34$, $Ki = 0,01$ e $Kd = 4,2$.

Como já mencionado anteriormente os dados que o Arduino recebe em suas portas de entrada e escreve em suas portas de saída não são usuais, pois variam de acordo com regras binárias, assim é preciso trabalhar com a conversão dos valores para a obtenção das respostas esperadas. O processo de saída relaciona o valor do set point já convertido em sinal PWM com o sinal de controle, e assim é gerada a variação da resposta e conseqüentemente o processo de controle.

Outro ponto importante a ser considerado na implementação de um sistema de controle em um dispositivo digital é o tempo de amostragem, que foi definido como sendo de um segundo, ou seja, $Ta = 1$, este valor foi adotado visando o melhor desempenho possível do hardware utilizado.

ANÁLISE E RESULTADOS

Algoritmo

O primeiro passo que o algoritmo implementado realiza é a conversão e o processamento dos dados parametrizados, assim são obtidos os valores de b_0 , b_1 e b_2 . Com estes parâmetros começa o segundo passo, quando a Equação 13 é executada e o controle realizado em torno do set point.

Por fim os valores lidos pelo sensor de temperatura são direcionados via serial ao computador onde é possível com o auxílio de diversos softwares gerar bancos de dados em formatos variados, para este artigo foram gerados arquivos em “.txt” e através de softwares de processamento matemático foram

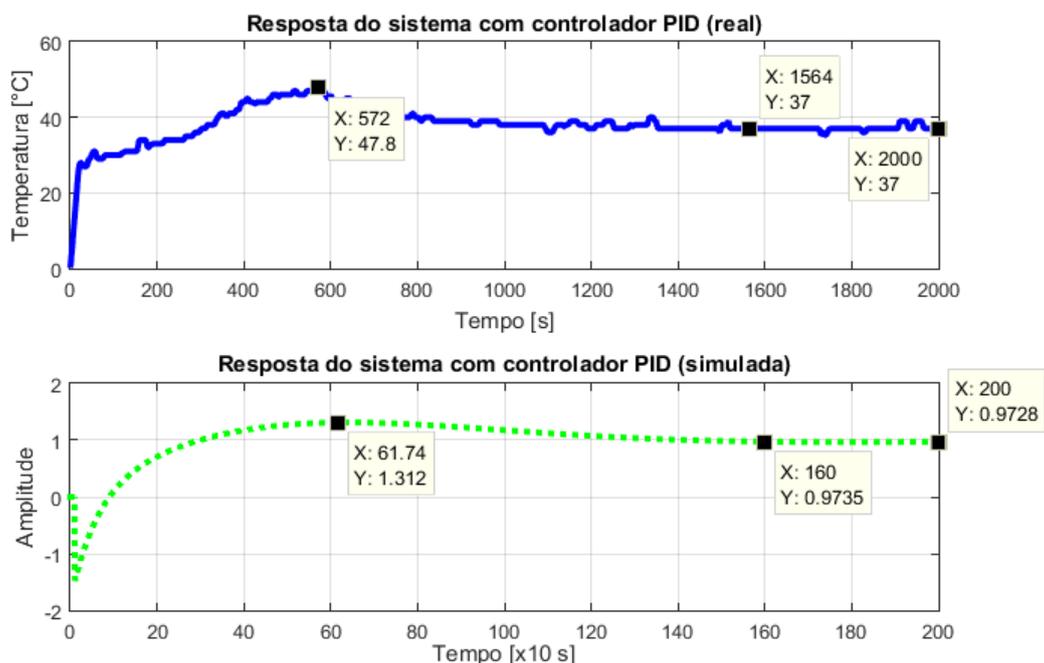
realizadas as plotagens dos gráficos demonstrados.

Gráfico de resposta

É possível verificar através da Figura 14 que a resposta ao degrau do sistema físico apresenta sobressinal semelhante ao da simulação, aproximadamente 29,19%. E o tempo de acomodação também não apresenta grande variação, ocorrendo em 1564 segundos quando na simulação ocorre em 1600 segundos.

A variação referente ao sobressinal e a variação do tempo de acomodação podem ser explicadas devido aos métodos utilizados para a formulação da função de transferência. Os métodos por análise gráfica tendem a apresentar maior chance de erro em relação aos demais, estes erros podendo ser de falha humana ou mesmo por problemas físicos não previstos na planta.

Figura 14 - Resposta ao degrau com controle PID



CONCLUSÕES

A extração da função de transferência de um sistema real de primeira ordem a partir de sua resposta ao degrau é uma tarefa extremamente trabalhosa, que exige atenção em inúmeros detalhes que se não considerados podem causar o mal dimensionamento de um projeto em malha aberta para controle, que por sua vez pode funcionar de forma indevida, chegando até mesmo a instabilidade do sistema.

É de extrema importância conhecer as ferramentas matemáticas adequadas para encontrar uma forma elaborada de implementar o controle PID em um dispositivo de processamento digital. As ferramentas de cálculo são as principais pontes para estruturar uma equação de diferenças, capaz de responder com maior precisão.

Os métodos de alocação dos polos de um controlador apesar de apresentarem uma grande agilidade com cálculos sem grande complexidade, podem causar atrasos caso não sejam manuseados de forma adequada e direta. Pode-se realizar um grande número de simulações para a comparação das ferramentas de parametrização, mas ainda é possível obter resultados falhos capazes de gerar sistemas inadequados.

Com o auxílio de simulações é possível prever a resposta de sistemas físicos de controle de forma muito satisfatória, apesar das disparidades dos gráficos teóricos e reais a resposta segue os mesmos padrões. Também há a possibilidade de erros causados por diversos imprevistos que uma simulação é incapaz de prever, como a temperatura ambiente que pode variar no dia a dia.

O Arduino é uma ferramenta poderosa que proporciona maior flexibilidade e tempo ao usuário para o enriquecimento teórico e a maior fundamentação de outros aspectos, apesar de suas limitações é uma ferramenta completa que proporciona o que se propõem para os fins de projetos práticos e artigos.

Este artigo proporciona uma rápida apresentação de como um controlador PID pode ser implementado, da maneira mais objetiva possível, para um sistema cuja a função de transferência é desconhecida, assim como as disparidades entre os fundamentos teóricos e os eventos práticos através de gráficos e simulações.

REFERÊNCIAS

ARDUINO. **Playground**. Disponível em: < <http://playground.arduino.cc/> >.

Acesso em: 29 de abril de 2016.

BEZERRA, Matheus Sales. **PROJETO, IMPLEMENTAÇÃO E ENSAIOS DE UM CONTROLADOR PID UTILIZANDO FPGA**. Fortaleza: UFC, 2010.

BOLTON, W. **Engenharia de Controle**. 1. ed. São Paulo: MARKON Books, 1995.

BUFFONI, Salete Souza de Oliveira. **Integral definida ou de Riemann**. Niterói: UFF.

CHEVARRIA, Guilherme de Calazans; COELHO, Raphael Rocha.

Identificação de sistemas de primeira ordem com atraso de transporte.

Ijuí: UNIJUÍ, 2006.

FAIRCHILD SEMICONDUCTOR. **Medium Power Linear Switching Applications**. 2001.

KAGUEYAMA, Cintia Ayumi. **Sintonia do controlador PID: Método de Ziegler Nichols Modificado**. Londrina: UEL, 2011.

MATHWORKS. **Serial Port Devices**. Disponível em: <

<http://www.mathworks.com/help/matlab/serial-port-devices.html> >. Acesso em:

07 de maio de 2016.

MATHWORKS. **2-D and 3-D Plots**. Disponível em: <

<http://www.mathworks.com/help/matlab/2-and-3d-plots.html> >. Acesso em: 09

de maio de 2016.

MATHWORKS. **Files and Folders**. Disponível em: <

<http://www.mathworks.com/help/matlab/files-and-folders.html> >. Acesso em: 11 de maio de 2016.

NEVES, Felipe. **Controlador PID digital: Uma modelagem prática para microcontroladores - Parte 1**. Disponível em: <

<http://www.embarcados.com.br/controlador-pid-digital-uma-modelagem-pratica-para-microcontroladores-parte-1/> >. Acesso em: 02 de maio de 2016.

NEVES, Felipe. **Controlador PID digital: Uma modelagem prática para microcontroladores - Parte 2**. Disponível em: <

<http://www.embarcados.com.br/controlador-pid-digital-uma-modelagem-pratica-para-microcontroladores-parte-1/> >. Acesso em: 02 de maio de 2016.

OLIVEIRA, Diego A. **Integração - Regra dos Trapézios**. Vitória da Conquista, 2016.

OLIVEIRA, Gustavo H. Costa. **Prática 4: projeto de controladores pid usando método de ziegler nichols**. Disponível em: <

<http://professorgustavo.weebly.com/praacutetica-4---projeto-de-pid-usando-ziegler-nichols-i.html> >. Acesso em: 6 de outubro de 2016.

PUHLMANN, Henrique. **PID digital - Método de cálculo numérico**. Disponível em: < <http://www.embarcados.com.br/pid-digital-metodo-de-calculo-numerico/> >. Acesso em: 05 de maio de 2016.

RODOR, Fadul Ferrari. **Aplicação de conceitos de conjuntos aproximados na adaptação da constante de filtro de controladores IMC**. Itajubá, 2012.

SAMITHA RANSARA. **Tutorial – Plotting the Serial Port Data with Matlab**.

Disponível em: < <http://mycola.info/2011/04/02/plotting-the-serial-port-data-with-matlab/> >. Acesso em: 06 de maio de 2016.

SARAIVA, Felipe de Andrade. **Métodos de Sintônica em Controladores PID**. Canoas, 2011.

SEIDEL, Ivan. **Arduino + PID**. Disponível em: <
<https://www.youtube.com/watch?v=txftR4TqKYA> >. Acesso em: 22 de setembro
de 2016.

TEXAS INSTRUMENTS. **TL082 Wide Bandwidth Dual JFET Input
Operational Amplifier**. 2013.

TEXAS INSTRUMENTS. **LM35 Precision Centigrade Temperature Sensors**.
1999.