

**UNIVERSIDADE FEDERAL DE OURO PRETO – MG
ESCOLA DE MINAS – EM
COLEGIADO DO CURSO DE ENGENHARIA DE CONTROLE E
AUTOMAÇÃO - CECAU**

**CONTROLE AUTOMÁTICO DE PROCESSOS INDUSTRIAIS UTILIZANDO
CONTROLADOR LÓGICO PROGRAMÁVEL (PLC) E SISTEMA
SUPERVISÓRIO PARA CONTROLE DE TEMPERATURA DE LÍQUIDOS EM
TUBOS.**

**MONOGRAFIA DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E
AUTOMAÇÃO**

LUCIANO RECLA SOPRANI

Ouro Preto, 2006

LUCIANO RECLA SOPRANI

**CONTROLE AUTOMÁTICO DE PROCESSOS INDUSTRIAIS UTILIZANDO CONTROLADOR
LÓGICO PROGRAMÁVEL (PLC) E SISTEMA SUPERVISÓRIO PARA CONTROLE DE
TEMPERATURA DE LÍQUIDOS EM TUBOS.**

Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como parte dos requisitos para obtenção de Grau em Engenheiro de Controle e Automação.

Orientador: Professor Dr. Ronilson Rocha

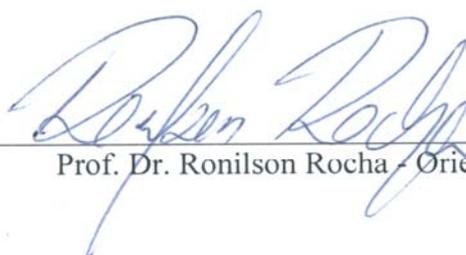
Co-Orientador: Professor Dr. Henor Artur de Souza

Ouro Preto

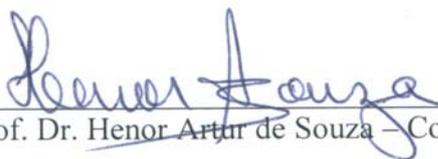
Escola de Minas – UFOP

MAIO / 2006

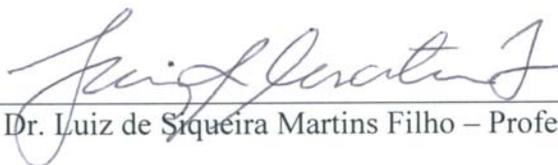
Monografia defendida e aprovada, em 11 de maio de 2006, pela comissão avaliadora constituída pelos professores:



Prof. Dr. Ronilson Rocha - Orientador



Prof. Dr. Henor Artur de Souza - Co-Orientador



Prof. Dr. Luiz de Siqueira Martins Filho - Professor Convidado

*Dedico este trabalho
a meus pais,
os anjos que guiam
meu corpo,
minha mente e
meu espírito.*

*Agradeço a Deus e a todas as pessoas de
boa fé que passaram em minha vida,
me apoiando e incentivando nesses anos de UFOP,
em especial aos professores Henor e Ronilson.*

SUMARIO

LISTA DE FIGURAS	7
LISTA DE QUADROS	9
RESUMO	10
ABSTRACT	11
I INTRODUÇÃO	12
1.1 Uma visão sobre automação industrial utilizando PLC.....	12
1.2 Objetivos.....	12
1.3 Metodologia adotada.....	13
II CONTROLADOR LÓGICO PROGRAMÁVEL – CLP	14
2.1 Arquitetura Geral de um PLC.....	14
2.2 Operação Básica.....	14
2.3 Arquitetura interna do piccolo 104/R Altus.....	15
III LINGUAGEM LADDER	17
3.1 Introdução.....	17
3.2 Conceitos Básicos.....	17
3.2.1 Contatos Lógicos.....	17
3.2.2 Corrente lógica fictícia.....	18
3.2.3 Temporizadores.....	18
3.2.4 Contadores Simples.....	19
3.3 Considerações sobre a Programação no MasterTool.....	20
3.3.1 Operandos Utilizados no MasterTool.....	20
IV CONFIGURAÇÃO DOS MÓDULOS DE FUNÇÕES	22
4.1 Configuração dos Canais Analógicos – Conversão A/D e D/A.....	22
4.1.1 Operandos_ Programação da Função F – ANLOG 006.....	23
4.1.2 Entradas e Saídas da Função F – ANLOG 006.....	23
4.2 Configuração da Função PID.....	24
4.2.1 Esquema de um Controlador PID.....	25
4.2.3.2 Descrição das saídas.....	27
4.2.4 Parâmetros Adicionais da Função F – PID 033.....	27
4.2.5 Características do Funcionamento da Função F- PID 033.....	28

V AÇÕES DE CONTROLE	29
5.1 Ação de Controle Proporcional.....	29
5.2 Ação de Controle Integral.....	29
5.3 Ação de Controle proporcional – derivativo (PD).....	30
5.4 Ação de controle proporcional e integral (PI)	31
5.5 Ação de controle proporcional – integral – derivativa (PID)	34
VI REGRAS DE SINTONIA PARA CONTROLADORES PID	36
6.1 Regras de Ziegler e Nichols para a sintonia de controladores PID.....	36
6.1.1 Primeiro Método de Ziegler-Nichols.....	37
6.1.2 Segundo Método de Ziegler-Nichols.....	39
VII SISTEMA SUPERVISÓRIO	42
7.1 Introdução Sobre Sistemas Supervisórios.....	42
7.1.1 Uma Abordagem sobre Sistemas Supervisórios.....	42
7.1.2 MMI - <i>Man Machine Interface</i>	43
7.1.3 SCADAS - <i>Supervisory Control and Aquisition Data System</i>	43
7.2 Objetivos de um Sistema Supervisório	43
7.3 Descrição dos Softwares Supervisórios.....	44
7.4 Arquitetura Básica	44
7.4.1 Driver de Comunicação	45
VIII RESULTADOS E ANÁLISES	46
8.1 Configuração dos Canais Analógicos e Função PID.....	47
8.2 Configuração dos Módulos de Funções.....	47
8.2.1 Configuração dos Canais Analógicos_ Conversão A/D e D/A	47
8.2.1.1 Parâmetros Utilizados	47
8.2.2 Configuração da Função PID.....	49
8.2.2.1 Parâmetros Utilizados	49
8.2.2.2 Configuração dos Parâmetros Adicionais para o Controle PID	51
8.2.2.3 Gráficos de Desempenho da Função PID – Método das Tentativas.....	51
8.2.2.4 Gráficos de Desempenho da Função PID – Primeiro Método de Ziegler Nichols.....	54
8.2.2.5 Tela de Supervisão para o Método de Ziegler Nichols.....	56
8.3 Considerações finais	56
IX CONCLUSÃO	58

X REFERÊNCIAS BIBLIOGRÁFICAS..... 59**LISTA DE FIGURAS**

Figura 1: a) contatos elétricos, b) contatos lógicos.....	15
Figura 2: Arquitetura interna do Piccolo 104/R.....	16
Figura 3: Sentido da corrente fictícia.....	18
Figura 4: Temporizador	19
Figura 5: Contador Simples	20
Figura 6: Lógica.....	21
Figura 7: Chama Função F - ANLOG 006	22
Figura 8: Chama Função F - PID 033	24
Figura 9: Esquema de funcionamento do controlador PID.....	25
Figura 10: Resposta de um controlador proporcional.....	30
Figura 11: Teste de um controlador PD, utilizando uma rampa.....	31
Figura 12: Teste de um controlador PI, utilizando um degrau	33
Figura 13: Resposta de um controlador PI.....	34
Figura 14: Resposta comparativa dos controladores: P, PI e PID	35
Figura 15: Controlador PID utilizado em um controle de processo	36
Figura 16: Curva de resposta ao degrau unitário mostrando um valor máximo de <i>overshoot</i> de 25%.....	37
Figura 17: Resposta de um processo a uma excitação em degrau unitário.....	38
Figura 18: Curva de resposta em forma de “S”	38
Figura 19: Sistema a malha fechada com controlador proporcional	40
Figura 20: Oscilação mantida com período P_{cr}	40
Figura 21: Arquitetura básica de um sistema supervisório.....	45
Figura 22: Planta piloto.....	46
Figura 23: Diagrama de blocos de um sistema de controle	46
Figura 24: Programação Ladder da configuração dos canais analógicos e da ação de controle PID a) conversão do valor medido analógico-digital, b) função de controle PI, c) conversão do valor de atuação digital-analógico.....	48
Figura 25: Configuração da conexão do PLC ao processo (planta), onde: 1) Canal Analógico DAC 01; 2) Comum e 3) Canal Analógico DAC 02.	50
Figura 26: Comportamento do controlador PI ($G_p=5.0$ e 1000 s/rep).....	52

Figura 27: Comportamento do controlador PI ($G_p=10.0$ e 1000 s/rep).....	52
Figura 28: a) Comportamento do controlador PI ($G_p=15.0$ e 500 s/rep), b) com variação da vazão.	53
Figura 29: a) Comportamento do controlador PI ($G_p=20.0$ e 500 s/rep), b) com a “água viciada”, c) variação do set-point.	55
Figura 30: Resposta no tempo do controle PI para o método Ziegler-Nichols.....	56
Figura 31: Tela para coleta de dados para o método de Ziegler Nichols.....	57

LISTA DE QUADROS

Quadro 1: Principais características dos contatos NA e NF	18
Quadro 2: Operandos Utilizados no MasterTool	20
Quadro 3: Parâmetros Adicionais do Controlador PID	27
Quadro 4: Primeira regra de Ziegler-Nichols	39
Quadro 5: Segunda regra de Ziegler-Nichols	40
Quadro 6: Parâmetros de sintonia para o primeiro Método de Ziegler-Nichols.....	54

RESUMO

Neste trabalho, o controle de um sistema térmico é desenvolvido baseado em controladores Lógicos Programáveis (PLC's) integrado a Sistemas Supervisórios. A planta consiste em um sistema de aquecimento e bombeamento de água, onde a potência de aquecimento e a vazão de líquido podem ser ajustados visando controlar a temperatura de saída da água, a qual é medida utilizando um termopar tipo k. Um algoritmo PID interno ao PLC é usado para processar o sinal de erro gerado a partir da comparação entre a medida e o sinal de referência para determinar a potência de aquecimento. A vazão de líquido é considerada como um distúrbio no sistema. Utiliza-se dois métodos para a sintonia do controlador PID: tentativa e erro e os métodos propostos por Ziegler-Nichols. Um sistema supervisorio é utilizado para observar o comportamento do sistema em malha fechada.

Palavras chaves: Controlador lógico programável (PLC), Sistema supervisorio, Programação do PLC, Aquecimento de água.

ABSTRACT

In this work, a thermal control system is developed based on Programmable Logic Controllers (PLC's) jointed to Supervisory Systems. The plant consists in a water heating and pumping system, where the heating power and liquid flow can be adjusted aiming to control the output water temperature, which is measured using a termocouple type k. An internal PID algorithm of PLC is used to process the error signal generated from a comparison between measure and reference aiming to determine the heating power. The liquid flow is considered as disturbance. The PID controller is tuned from two methods: trial by error and Ziegler-Nichols methods. A supervisory system is used to observe the behaviour of feedback system.

Keywords: Programmable Logic Controller (PLC), Supervisory System, PLC Programming, Water Heating.

I INTRODUÇÃO

1.1 Uma visão sobre automação industrial utilizando PLC

As indústrias dos mais diversificados ramos de atuação tais como mineração, metalúrgicas, papel e celulose, cimento, alimentícias, entre tantas outras, estão cada vez mais preocupadas com a produtividade de suas fábricas e também com a qualidade de seus produtos. Tais condições são vitais ao crescimento destas empresas, sendo que a automação tem sido a grande responsável pelos altos índices de produtividade e qualidade destas indústrias. A base desta automação se deu com a criação dos chamados Controladores Lógicos Programáveis- CLP - ou do inglês *Programmable Logic Controller* - PLC - que substituíram os antigos painéis de réles.

Segundo Georgini (2005, p. 48) “ um PLC pode ser definido como um dispositivo de estado sólido; um computador industrial, capaz de armazenar instruções para implementação de funções de controle (seqüência lógica, temporização, etc.), além de realizar operações lógicas e aritméticas, manipulação de dados e comunicação em rede.”

Os sistemas de controle que usam PLC's, integrados a um *software* supervisorio que realiza a interface entre a máquina (PLC) e o ser humano (operador), tem sido consagrada como a configuração mais difundida e confiável, sendo encontrada praticamente em todas as instalações industriais existentes onde haja algum sistema de controle.

1.2 Objetivos

Desenvolver um sistema de controle da temperatura de líquidos (água) em tubos. No trabalho desenvolve-se nas seguintes etapas:

- a) Implementação e programação do sistema de controle com a utilização do controlador lógico programável (PLC).
- b) Ligação entre a planta piloto e o PLC..
- c) Projeto de controladores PID.
- d) Discussão dos resultados obtidos no sistema real.

Estas etapas proporcionarão uma aplicação real de técnicas de controle em um processo de aquecimento de líquidos

1.3 Metodologia adotada

Então para aplicar esta tecnologia, foram necessários estudos e testes com os equipamentos, PLC, *software* de programação do PLC e com o *software* de supervisão (Eclipse E3), para poder aplicar estes conhecimentos na formulação da ação de controle (Controlador PID – além de um estudo sobre as várias formas de controladores P, PD, PI e PID), em métodos de sintonia de malha (primeiro método de Ziegler Nichols) com a aquisição de dados referentes ao processo e pelo método de tentativas.

São apresentados também, os parâmetros utilizados para configuração dos módulos de funções usados na lógica de controle, ou seja, os principais parâmetros referentes às funções ANLOG 006 (responsável pela conversão A/D e D/A) e os da função PID 033 (referente à ação de controle PID – proporcional integral e derivativo), que além da configuração básica necessita da configuração dos parâmetros adicionais, nos quais são configurados os ganhos proporcional, integral e derivativo. Neste ponto é que entra a sintonia de malha da ação PID. Utilizando-se dos métodos de Ziegler Nichols ou com auxílio dos gráficos de comportamento temporal para cada configuração de valores de ganhos implementados (método das tentativas) é possível melhorar o desempenho do controlador.

II CONTROLADOR LÓGICO PROGRAMÁVEL – CLP

2.1 Arquitetura Geral de um PLC

Praticamente todos os PLCs encontrados no mercado hoje, possuem as seguintes partes integrantes:

- **CPU:** “*Central Processing Unit*”, compreende o processador (microprocessador, microcontrolador ou processador dedicado), o sistema de memória RAM e ROM e os circuitos auxiliares de controle.
- **Circuitos/Módulos I/O (*input-output, entrada-saída*):** podem ser discretos (sinais digitais, 24Vdc, 110Vac) ou analógicos (sinais analógicos, 4-20 mA, 0-10 Vdc).
- **Fonte de Alimentação:** responsável pela tensão de alimentação fornecida à CPU e aos circuitos/módulos de I/O
- **Base:** proporciona a conexão mecânica e elétrica entre a CPU, os módulos de I/O e a fonte de alimentação. Contém o barramento de comunicação entre eles, no qual o sinal de dados, endereço, controle e tensão de alimentação estão presentes.

Um PLC pode ainda ser composto por circuitos e ou módulos especiais: controlador rápido, interrupção por *hardware*, controlador de temperatura, controlador PID, co-processadores, comunicação em rede, etc.

2.2 Operação Básica

O princípio de funcionamento também é o mesmo entre os PLCs existentes no mercado, diferenciando em alguns casos por algum incremento a mais na tecnologia empregada.

A CPU executa a leitura do *status* (condições, estados) dos dispositivos de entrada por meio dos circuitos ou módulos de I/O (módulos de entrada e saída_ *input e output*). Estes *status* são armazenados na memória (RAM) para serem processados pelo programa de aplicação (desenvolvido pelo usuário e armazenado em memória RAM, EPROM ou EEPROM no PLC). Após a execução do programa aplicativo, o

processador atualiza os status dos dispositivos de saída por meio dos módulos de I/O, realizando a lógica de controle.

A programação do PLC é feita por meio de uma ferramenta de programação, que pode ser um programador manual (terminal de programação, *handheld programmer*, etc.) ou um PC com *software* de programação específico. A linguagem *Ladder* (RLL - *Relay Ladder Logic*, lógica de contatos de relé), muito popular entre os usuários dos antigos sistemas de controles a relés, sendo hoje a linguagem mais utilizada. Esta linguagem é a representação lógica da seqüência elétrica de operação, conforme é mostrado na figura 1.

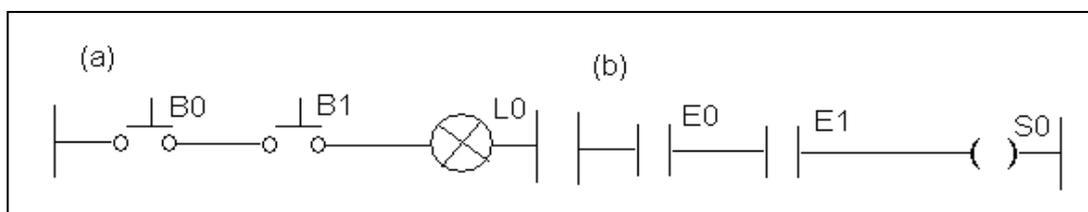


FIGURA 1: a) contatos elétricos, b) contatos lógicos.

FONTE: GEORGINI, 2000, p. 50

A lógica implementada pelo PLC é muito similar à convencional, sendo que os dispositivos de entrada, elementos B0 e B1, são conectados ao módulo de entrada e o dispositivo de saída, elemento L0, ao módulo de saída. O programa de aplicação determina o acionamento da saída em função das entradas ($B0 \cdot B1 = L0$). Qualquer alteração desejada na lógica de controle é feita por alterações no programa aplicativo, permanecendo as mesmas conexões nos módulos de entrada e saída.

2.3 Arquitetura interna do piccolo 104/R Altus

Na figura 2, mostra-se a arquitetura interna da família de PLCs, denominada PICCOLO, da empresa Altus.

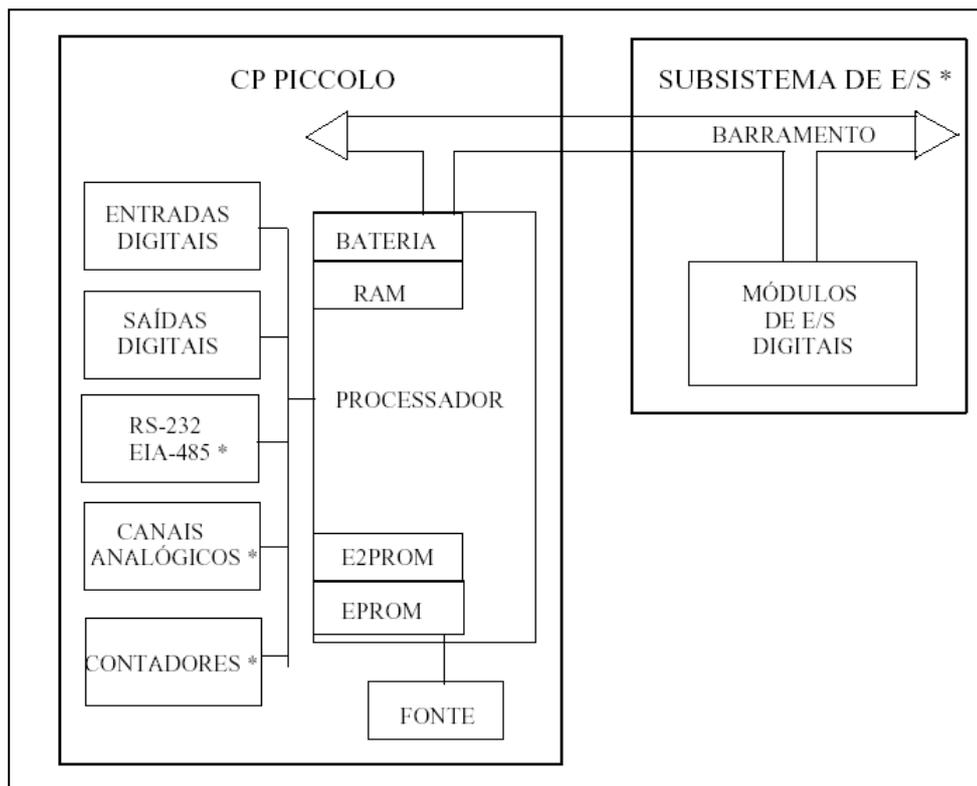


FIGURA 2: Arquitetura interna do Piccolo 104/R

FONTE: MU Altus, 1995, p.2-7.

III LINGUAGEM LADDER

3.1 Introdução

Mesmo tendo sido a primeira linguagem destinada à programação de PLCs, a linguagem *ladder*, se mantém no mercado como a mais utilizada, sendo a linguagem de programação padrão para a maioria dos PLCs existentes. Isto se deve ao fato de que a linguagem *ladder* é baseada em símbolos semelhantes aos usados nos esquemas elétricos (contatos e bobinas), uma vez que as diferenças entre os fabricantes de PLCs, quanto à representação são facilmente assimiladas pelo usuário.

Cada elemento (bobina ou contato), da lógica de controle representa uma instrução, sendo alocada em um endereço específico, cada uma consumindo uma quantidade de memória, ficando armazenada para ser utilizada no programa aplicação.

3.2 Conceitos Básicos

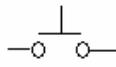
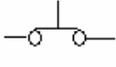
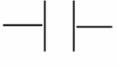
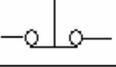
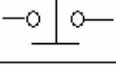
Estes conceitos são necessários para o correto desenvolvimento de programas aplicação, sendo aplicados a todos os PLCs, independente dos fabricantes e de recursos disponíveis na CPU utilizada.

3.2.1 Contatos Lógicos

Existem dois tipos de contatos, os normalmente abertos (NA) e os normalmente fechados (NF). No quadro 1 são apresentadas algumas características destes contatos.

Os contatos NA estarão ativos quando estiverem no estado 1 (um), e os contatos NF quando se encontrarem no estado 0 (zero).

QUADRO 1: Principais características dos contatos NA e NF.

Dispositivo de Entrada	Nível Lógico das Entradas	Símbolo	Atuação do Contato Lógico	Fluxo de Corrente Fictícia
	0		Não	Não
	1		Sim	Sim
	1		Não	Sim
	0		Sim	Não

Fonte: GEORGINI, 2000, p. 106

3.2.2 Corrente lógica fictícia

Para que um elemento da lógica de controle seja acionado, faz-se necessário energizá-la logicamente. Assim, utiliza-se o conceito de corrente lógica fictícia, ou seja, supondo que entre as barras verticais da lógica de controle exista uma tensão aplicada, com a barra da esquerda positiva e da direita negativa, havendo, portanto circulação de corrente somente da esquerda para a direita. Figura 3.

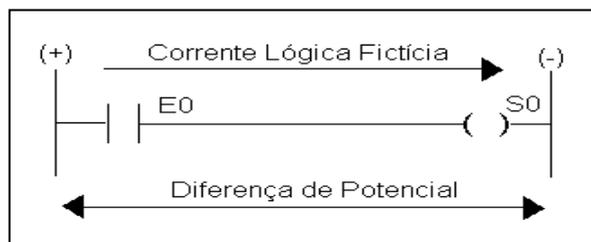


FIGURA 3: Sentido da corrente fictícia

FONTE: GEORGINI, 2000, p. 102

3.2.3 Temporizadores

Os temporizadores executam contagens de tempo com a energização das suas entradas de acionamento.

No caso do PLC Altus a instrução TEE (referente a um temporizador) possui dois operandos, o OPER1 que especifica a memória acumuladora e contagem de tempo, e o OPER2 que indica o tempo máximo a ser contado. Sendo o incremento de 0,1 segundos.

As entradas de acionamento são a “libera” que permite a contagem quando energizada e a “ativa” que ativa a contagem quando energizada, quando as duas entradas estão acionadas, é feita a contagem do tempo, até seu valor limite.

Há também duas saídas, uma a saída indicada pela letra “Q”, que indica que o tempo foi transcorrido e uma pelo símbolo e letra “-Q”, que indica sempre o completo de “Q”, ou seja, o valor inverso.

OBS: existe também o TED, que é o temporizador na desenergização, ou seja, realiza contagem de tempo com a desenergização da sua entrada de acionamento. Figura 4.

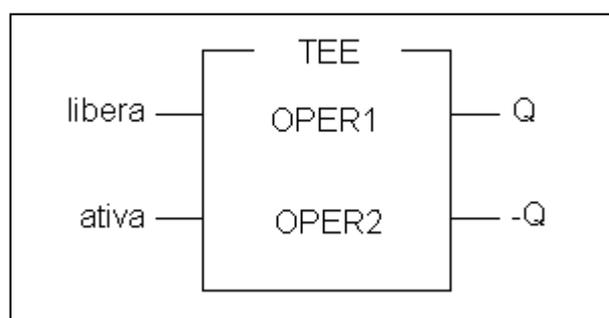


FIGURA 4: Temporizador

FONTE: MP Altus, 1997, p. 3-57.

3.2.4 Contadores Simples

Esta instrução realiza contagens simples de tempo, com incremento de uma unidade em cada acionamento.

Esta instrução é composta por dois operandos de entrada, o OPER1, do tipo memória, especifica a memória que contabiliza os eventos e o OPER2 geralmente do tipo constante decimal, estabelece o valor limite de contagem.

É composto ainda por duas entradas de acionamento, a “incrementa”, entrada que quando sofre uma transição de 0 para 1, estando a entrada “ativa” energizada

aumenta o valor do OPER1 em uma unidade. É uma entrada “ativa”, que habilita a contagem quando ocorre uma transição da entrada “incrementa” de 0 para 1.

Possui também duas saídas, uma chamada “limite” que é energizada quando o valor do OPER1 iguala ao valor do OPER2, e uma chamada “não limite”, que é energizada quando a entrada “ativa” estiver desenergizada ou a contagem ainda não tenha atingido o valor limite, figura 5.

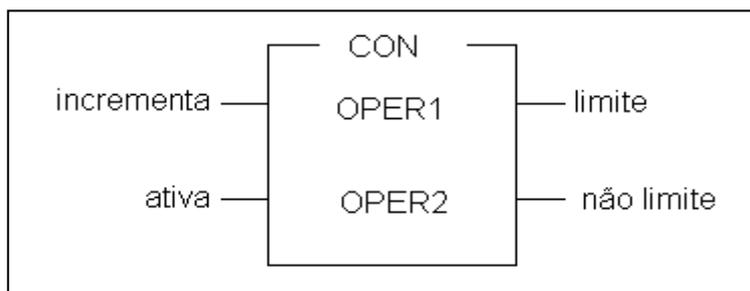


FIGURA 5: Contador Simples

FONTE: MP Altus, 1997, p. 3-53.

3.3 Considerações sobre a Programação no MasterTool

3.3.1 Operandos Utilizados no MasterTool

São mostrados no quadro 2 os operandos disponíveis no MasterTool:

QUADRO 2: Operandos Utilizados no MasterTool

Tipo	Operando	Tipo	Operando
%E	Relés de Entrada	%D	Decimais
%S	Relés de Saída	%KM	Constantes Memórias
%R	Endereço no Barramento	%KD	Constantes Decimais
%A	Relés Auxiliares	%TM	Tabelas Memórias
%M	Memórias	%TD	Tabelas Decimais

FONTE: MP Altus, 1997, p. 2-7.

3.3.2 Lógicas

Chama-se lógica a matriz de programação formada por 32 células (elementos da matriz) dispostas em 4 linhas (0 a 3) e 8 colunas (0 a 7). Em cada uma das células podem ser colocadas instruções, podendo-se programar até 32 instruções em uma mesma lógica. Cada lógica presente no programa simula um pequeno trecho de um diagrama de relés real. Na figura 6 é mostrado o formato de uma lógica do programa aplicativo.

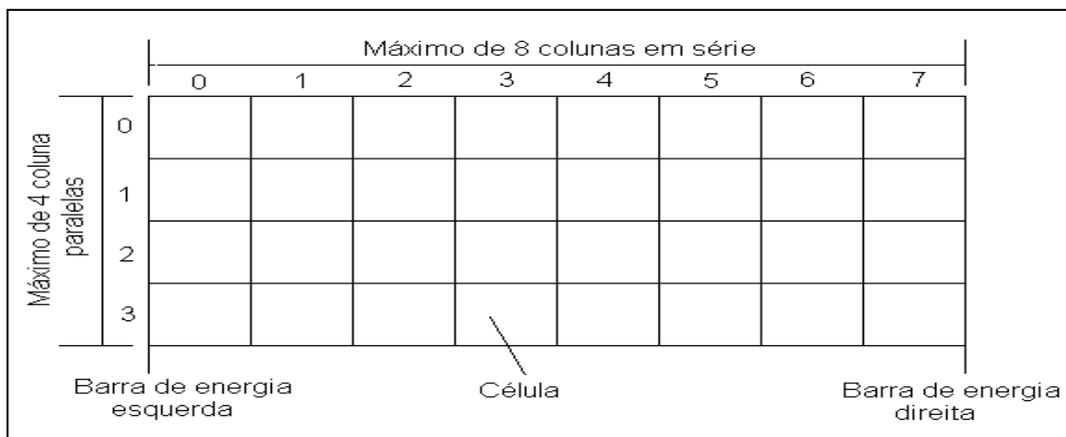


FIGURA 6: Lógica

FONTE: MP Altus, 1997, p. 2-4.

IV CONFIGURAÇÃO DOS MÓDULOS DE FUNÇÕES

“[...] os módulos Função implementam diversas rotinas de uso específico ou para o acesso aos módulos especiais de E/S pelo programa aplicativo, sendo similares às instruções ladder, porém carregados como módulos do programa. A sua execução é disparada por outros módulos por meio da instrução CHF [...] (Manual de Programação do MasterTool, 1997, p. 4-1).

Os módulos que acompanham o MasterTool são programados em linguagem de máquina, não podendo ser lidos para o programador e visualizados como os módulos em diagrama de relés. Devem ser carregados diretamente do disco para o CP. Isso é feito da seguinte maneira: cria-se inicialmente o projeto no qual será abrigada a lógica de controle feita no módulo principal. Dentro desta lógica, chamamos a função desejada (no caso deste relatório as funções usadas foram a F – ANLOG 006 e a F – PID 033). Após ser feito este passo é necessário copiar o módulo em *Assembly* para o diretório do projeto e depois se procede normalmente enviando os módulos para o CP.

4.1 Configuração dos Canais Analógicos – Conversão A/D e D/A

“[...] A função F-ANLOG 006 realiza a conversão A/D (analógico/digital) ou D/A (digital/analógico) dos canais analógicos integrados. Utilizando-se duas instruções CHF, é possível realizar a conversão A/D em um dos canais e D/A no outro canal [...]. (Manual de Programação do MasterTool, 1997, p. 4-24). Na figura 7 é apresentada a função F – ANLOG 006.

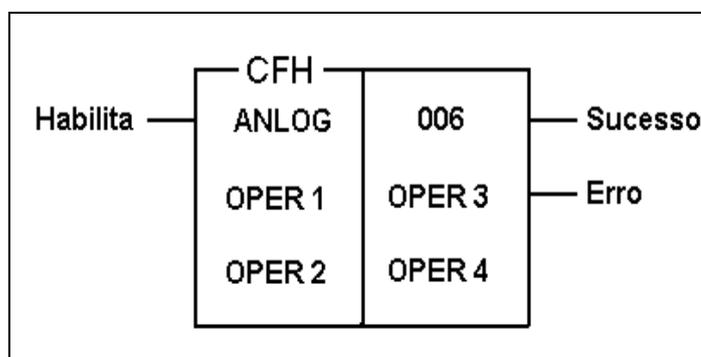


FIGURA 7: Chama Função F - ANLOG 006

FONTE: MP Altus, 1997, p. 4-24.

4.1.1 Operandos_ Programação da Função F – ANLOG 006

As células da instrução CHF utilizada para a chamada da função são programadas do seguinte modo:

- **OPER1** - Especifica o número de parâmetros que são passados para a função em OPER3. Este operando deve ser obrigatoriamente uma constante memória com valor 3 (%KM+00003).
- **OPER2** - Deve ser um operando do tipo constante memória com valor 0 (%KM+00000). Determina o número de parâmetros possíveis de serem programados na janela de edição de OPER4. Como esta função não necessita de nenhum parâmetro em OPER4, o valor de OPER2 é 0.
- **OPER3** - Contém os parâmetros que são passados para a função, declarados utilizando-se uma janela visualizada no MasterTool quando a instrução CHF for editada. O número de parâmetros editáveis é especificado em OPER1, sendo fixo em 3 para este módulo:
 1. **%KM+XXXXX** - Especificação do canal a ser convertido. Deve-se utilizar %KM+00000 para o canal DAC 1 e %KM+00001 para o canal DAC 2.
 2. **%KM+XXXXX** - Tipo de conversão a ser realizada no canal definido pelo parâmetro anterior. Deve-se utilizar %KM+00000 para conversão A/D e %KM+00001 para conversão D/A.
 3. **%MXXXX** - Especificação do operando onde é armazenado o valor a ser escrito no conversor em caso de conversão D/A ou valor lido em caso de conversão A/D.
- **OPER 4** - Não utilizado.

4.1.2 Entradas e Saídas da Função F – ANLOG 006

- **Habilita** - a função é chamada quando esta entrada está energizada, sendo analisados os parâmetros programados na instrução CHF.
- **Sucesso** - é energizada quando a função foi corretamente executada.

- **Erro** - é energizada caso ocorra erro na especificação dos operandos ou tentativa de acesso a operandos não declarados.

4.2 Configuração da Função PID

“A função F-PID 033 implementa o algoritmo de controle proporcional, integral e derivativo. A partir de um valor medido (VM) e do ponto de ajuste desejado (PA), a função calcula o valor de atuação (VA) para o sistema controlado [...]” (Manual de Programação do MasterTool, 1997, p. 4-53).

Este valor é calculado periodicamente, levando em consideração os fatores proporcionais, integrais e derivativos programados. O diagrama em blocos da função é mostrado na figura 8. As características mais importantes apresentadas pelo laço de controle implementado são:

- Desaturação da ação integral (*anti-reset windup*);
- Acompanhamento da saída no modo manual e comutação manual/automática balanceada (*output tracking e bumpless transfer*);
- Ação direta ou reversa;
- Limites de saída máximo e mínimo ajustáveis;
- Ação derivativa calculada sobre várias amostragens;
- Capacidade de realizar integral discreta;
- Deslocamento com sinal;
- Tempo de execução de 1,6 ms;
- Resolução de saída de 1:1000.

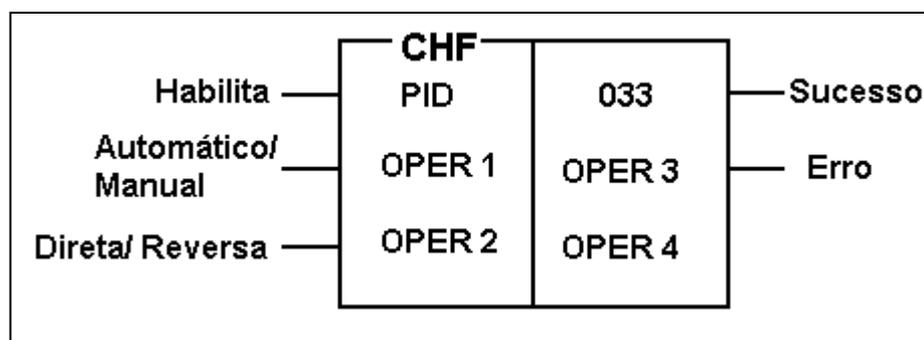


FIGURA 8: Chama Função F - PID 033

FONTE: MP Altus, 1997, p. 4-53.

4.2.1 Esquema de um Controlador PID

Na figura 9 apresenta-se o esquema de funcionamento do controlador PID, implementado no PLC Piccolo 104 R da Altus.

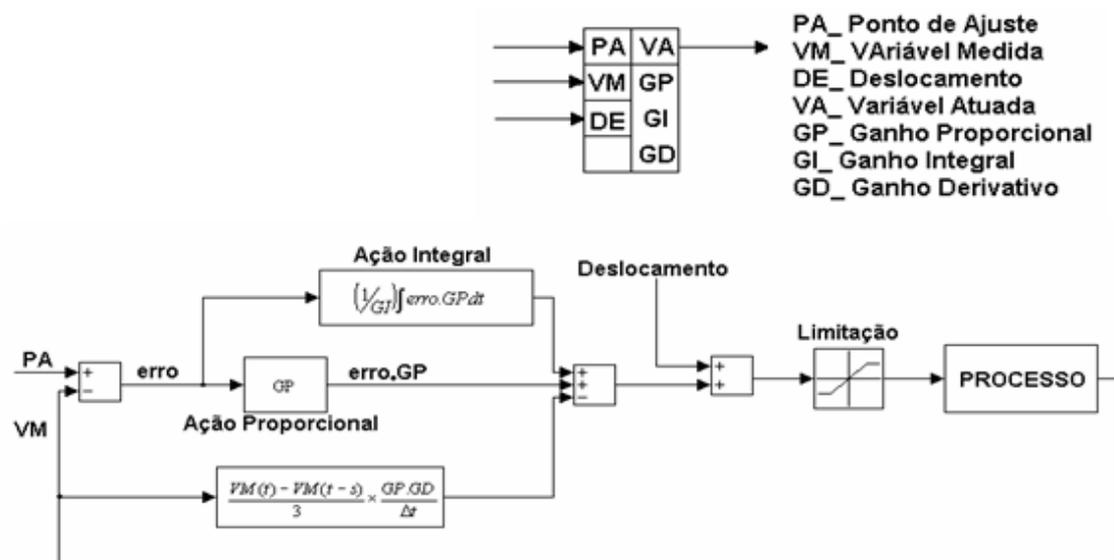


FIGURA 9: Esquema de funcionamento do controlador PID

FONTE: MP Altus, 1997, p. 4-54.

O uso da função PID no programa aplicativo permite uma série de facilidades que são facilmente integradas ao sistema sem o uso de controladores externos. Por exemplo:

- Função automático/manual;
- Inibição do fator integral ou derivativo;
- Laços cascadeados;
- Geração de curvas de ponto de ajuste;
- Modificação dos parâmetros de controle pelo programa;

4.2.2. Operandos_ Programação da Função F- PID 033

As células da instrução CHF utilizada para a chamada da função são programadas do seguinte modo:

- **OPER1** - Especifica o número de parâmetros que são passados para a função em OPER3. Este operando deverá ser obrigatoriamente uma constante memória com valor 5 (%KM+00005).

- **OPER2** - Especifica o número de parâmetros que são passados para a função em OPER4. Este operando deverá ser obrigatoriamente uma constante memória com valor 0 (%KM+00000).
- **OPER3** - Contém os parâmetros que são passados para a função, declarados utilizando-se uma janela visualizada no MasterTool quando a instrução CHF for editada. O número de parâmetros editáveis é especificado em OPER1, sendo fixo em 5 para este módulo:
 - 1) **%TMXXXX** - Tabela que contém os parâmetros utilizados pelo algoritmo de controle. Deve conter 16 posições.
 - 2) **%MXXXX** - Memória que contém o valor medido do processo, normalmente obtido por meio de uma instrução A/D.
 - 3) **%MXXXX** - Contém o ponto de ajuste (*set point*), que é o valor desejado para a variável medida. O seu valor pode ser modificado conforme a política de controle desejada.
 - 4) **%MXXXX** - Memória que contém o valor de atuação no processo, geralmente acionando uma instrução D/A.
 - 5) **%AXXXX** - Octeto auxiliar que contém pontos de controle da função PID.
- **OPER4** - Não utilizado.

4.2.3 Entradas e Saídas da Função F – PID 033

4.2.3.1 Descrição das entradas

- **Habilita_** quando esta entrada está energizada a função é chamada, sendo analisados os parâmetros programados na instrução CHF. Caso o número de parâmetros ou seu tipo sejam diferentes das necessidades da função, todas as saídas da instrução são desenergizadas. Se estiverem corretos, o cálculo do controle PID é realizado.
- **Automático (0)/Manual (1)_** quando energizada, o operando de atuação não recebe o valor calculado pela função (modo manual).
- **Direta (0)/Reversa (1)_** especifica a forma de ação do controle.

4.2.3.2 Descrição das saídas

- **Sucesso_** é energizada quando a função foi corretamente executada.
- **Erro_** é energizada caso ocorra erro na especificação dos operandos ou tentativa de acesso a operandos não declarados.

4.2.4 Parâmetros Adicionais da Função F – PID 033

Além dos operandos programados na instrução de chamada CHF, outros parâmetros devem ser carregados na tabela declarada em OPER3. Esta tabela deve conter 16 posições, sendo utilizada para definir os parâmetros utilizados pelo algoritmo de controle e armazenar resultados intermediários. No quadro 3 são apresentados os parâmetros que devem ser carregados em cada posição de tabela, bem como seus valores mínimos e máximos.

QUADRO 3: Parâmetros Adicionais do Controlador PID

Ps.	Parâmetro Armazenado	Fórmula	Variação	Valor Tabela
00	Ganho Proporcional x 10	GP x 10	GP: 1,0 a 100,0	10 a 1000
01	Fat integral - parte frac.	dt / GI	GI: 1 a 1000 s/rep	0,0001 a 10,000
02	Fat integral - parte int		dt: 0,1 a 10 s	
03	Fat derivativo - parte frac.	GD / 3dt	GD: 1 a 1000 s	0,0333 a
04	Fat derivativo - parte int		dt: 0,1 a 10 s	
05	Deslocamento	DE	0 a 1000	0 a 1000
06	Valor mínimo da saída		0 a 1000	0 a 1000
07	Valor máximo da saída		0 a 1000	0 a 1000
08	Não utilizada			
09	Variável medida N - 1			0 a 1000
10	Variável medida N - 2			0 a 1000
11	Variável medida N - 3			0 a 1000
12	Erro			0 a 1000
13	Ação proporcional x 10			0 a 65535
14	Ação integral - parte frac x			0 a 65535
15	Ação integral - parte int x			0 a 65535

FONTE: MP Altus, 1997, p. 4-57.

4.2.5 Características do Funcionamento da Função F- PID 033

A desaturação da ação integral (*anti-reset windup*) é feita de modo a evitar que o termo integral continue a acumular erro quando um distúrbio no processo causa a saturação da saída do controlador em alguns dos limites. No momento em que o valor de saída atinge algum dos limites (máximo ou mínimo), o termo integral é fixado em seu valor corrente, impedindo o seu crescimento indefinido, sem influenciar na saída. Isto assegura que haverá uma resposta do controlador tão logo desapareça o distúrbio que o levou a saturar a saída.

A função pode ser executada em modo manual, energizando-se a segunda entrada da instrução CHF. Neste modo, a rotina não mais modifica o valor da saída de atuação, mas o acompanha (*output tracking*). Isto é, em função do valor da saída fixo e do valor medido do processo, os termos proporcional e derivativo são calculados e o termo integral é forçado para um valor adequado, de modo que, quando ocorrer a transição de manual para automático, a rotina possa reassumir o controle com o valor inicial da saída igual ao último valor da saída no modo manual. Chama-se este fato de comutação manual/automática balanceada (*bumpless transfer*).

A forma de controle pode ser direta ou reversa. Esta seleção é realizada desenergizando ou energizando a terceira entrada da instrução CHF. Caso o processo seja tal que o valor medido cresce quando o valor da saída de atuação cresce, a ação direta deve ser selecionada. Se o valor medido decresce com o aumento da saída de atuação, então a ação reversa deve ser utilizada.

O intervalo entre amostragens de um laço PID pode variar de 0,1 a 10,0 segundos. Nota-se ainda que o valor do intervalo de amostragem usado para o cálculo dos fatores multiplicativo integral e derivativo devem coincidir com o intervalo de tempo de *scan* do controlador. Como cada execução da rotina pode despende até 3 ms, é aconselhável que cada laço de controle diferente seja disparado em diferentes varreduras do programa.

V AÇÕES DE CONTROLE

5.1 Ação de Controle Proporcional

Para um controlador com ação de controle proporcional, a relação entre o sinal de saída do controlador $u(t)$ e o sinal de erro atuante $e(t)$ (OGATA, 2000, p. 180) é:

$$u(t) = K_p e(t) \quad (5.1)$$

ou no domínio de Laplace,

$$\frac{U(s)}{E(s)} = K_p \quad (5.2)$$

E segundo Bega (2003, p.15) “o modo de controle proporcional não contém nenhum elemento dinâmico, onde sua atuação depende do valor do erro $e(t)$ e independe de sua velocidade ou tempo de duração; a saída do controlador proporcional não varia quando o erro está fixo, ela só varia quando o erro está estacionário [...]”.

Nas curvas de resposta mostradas, figura 10, o sistema de controle está em malha fechada; nela se mostra a curva de resposta de um controlador proporcional com diversos valores de K_p . Nota-se que à medida que o ganho proporcional K_p aumenta, o erro diminui e o sistema responde mais rapidamente às variações. Assim pode-se afirmar que o ganho proporcional deve ser ajustado no maior valor que o processo permitir. À medida que o ganho proporcional é aumentado, o erro diminui, mas as oscilações e o tempo para estabilização aumentam e caso utilize-se um ganho excessivo, o processo poderá se tornar instável.

5.2 Ação de Controle Integral

Em um controlador com ação de controle integral, o valor da saída do controlador $u(t)$ é variado segundo uma taxa proporcional ao sinal de erro atuante,

$$\frac{du(t)}{dt} = K_i e(t) \quad (5.3)$$

ou

$$u(t) = K_i \int_0^t e(t) dt \quad (5.4)$$

onde K_i é uma constante ajustável. A função de transferência do controlador integral é:

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \quad (5.5)$$

Se o valor de $e(t)$ for dobrado, então o valor de $u(t)$ varia duas vezes mais rápido, logo para um erro atuante nulo, o valor de $u(t)$ permanece estacionário. (OGATA, 2000, p. 181)

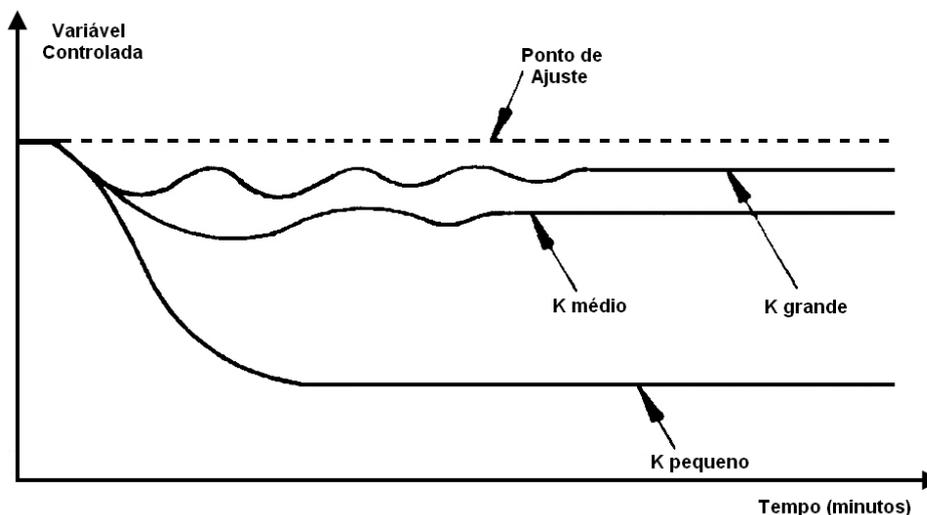


FIGURA 10: Resposta de um controlador proporcional

FONTE: BEGA, 2003, p. 16

5.3 Ação de Controle proporcional – derivativo (PD)

Define-se a ação de controle de um controlador proporcional e derivativo por meio das seguintes equações,

$$u(t) = K_p e(t) + K_p \frac{de(t)}{dt} \quad (5.6)$$

e a função de transferência,

$$\frac{U(s)}{E(s)} = K_p (1 + T_d) \quad (5.7)$$

onde K_p representa o ganho proporcional e T_d é uma constante chamada tempo derivativo, (OGATA, 2000,181).

Segundo Bega (2003, p.20) “o modo derivativo não pode ser utilizado sozinho. Na prática ele virá inevitavelmente acoplado ao modo proporcional [...]”.

A saída do modo proporcional é proporcional ao erro $e(t)$, enquanto a saída do modo derivativo é função da derivada do erro $e(t)$. Pode-se afirmar então, que a saída de um controlador com ação derivativa é proporcional à inclinação da curva da variável controlada, ou que a saída é proporcional à velocidade de variação da variável controlada. Bega (2003, p.20) afirma ainda que “[...] a saída do modo derivativo adianta a saída do modo proporcional de um tempo T_d ; este tempo é chamado de tempo derivativo, sendo por definição o tempo que a saída do modo proporcional leva para repetir a saída do modo derivativo”. Conforme é ilustrado na figura 11.

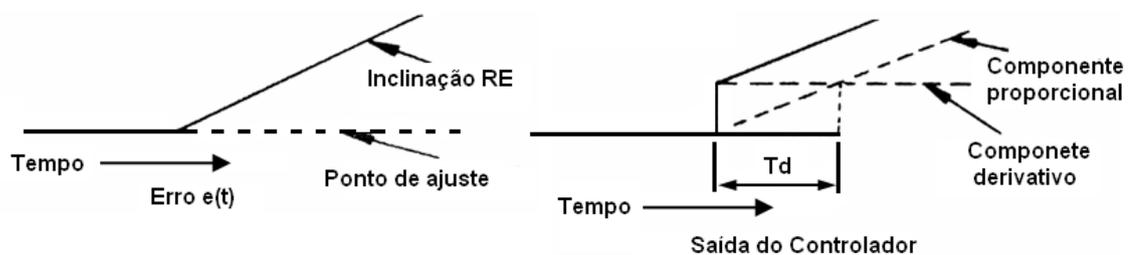


FIGURA 11: Teste de um controlador PD, utilizando uma rampa

FONTE: BEGA, 2003, p. 21

O sistema de controle está em malha fechada e nela se tem uma curva de resposta de um controlador proporcional e derivativo (PD). Nota-se que o aumento do tempo derivativo melhora a estabilidade do processo, embora se diminua a velocidade de resposta. Tem-se também que a ação derivativa não influencia no valor final do erro $e(t)$.

5.4 Ação de controle proporcional e integral (PI)

A ação de controle de um controlador proporcional e integral é definida pela seguinte equação, (OGATA, 2000, p. 181),

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt \quad (5.8)$$

ou pela função de transferência do controlador:

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} \right) \quad (5.9)$$

O controle proporcional seria aceitável na maioria das aplicações de controle de processos se não houvesse a necessidade de eliminar o erro $e(t)$. Embora o operador possa compensar ou eliminar o erro com reajuste manual ou por mudança de polarização, isso se torna incômodo e ineficiente. Assim a maioria dos controladores proporcionais tem um modo de controle adicional, chamado integral, que elimina o erro $e(t)$ automaticamente. Segundo Bega (2003, p. 17) “a saída do modo proporcional é proporcional ao erro $e(t)$, a saída do modo integral é função da integral do erro, ou seja, a velocidade de correção do elemento final de controle é proporcional ao erro $e(t)$ ”.

A ação integral é normalmente utilizada em conjunto com ação proporcional (controlador PI), pois a velocidade de resposta da ação integral sozinha é muito lenta e seu tempo de estabilização é muito longo.

No esquema mostrado, figura 12, o sistema de controle esta em malha aberta. No caso mostrado, introduziu-se um erro de valor ΔE , no tempo $t=0$ minuto. O modo proporcional varia a saída de $K.e(t)$ no momento em que o erro $e(t)$ muda de zero para ΔE , ficando constante a partir daí, uma vez que o erro se mantém fixo.

A saída do modo integral não varia instantaneamente com o surgimento do erro, mas o fato do erro ser diferente de zero faz com que a saída do modo integral varie à medida que o tempo vai passando, esta variação só irá cessar quando o erro voltar a zero. Como o erro é constante (ΔE), a saída da componente integral será uma rampa com inclinação $\left(\frac{Ke(t)}{T_i} \right)$, ou seja, irá variar de acordo com equação:

$$\left(\frac{Ke(t)}{T_i} \right) \quad (5.10)$$

Vê-se, então que o modo proporcional será mais efetivo que o modo integral na resposta a rápidas variações de processo.

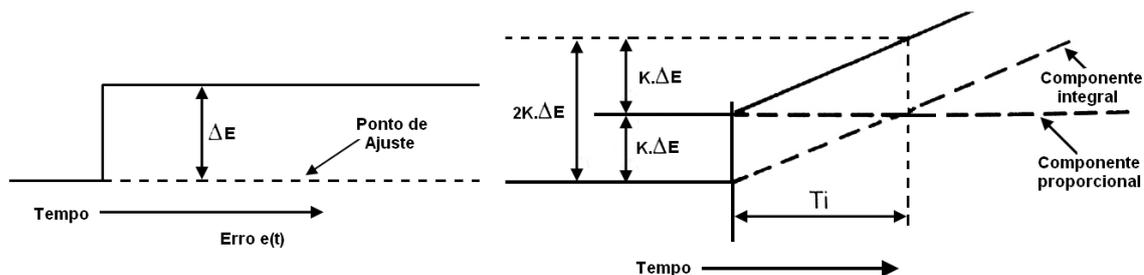


FIGURA 12: Teste de um controlador PI, utilizando um degrau

FONTE: BEGA, 2003, p. 18

Observa-se no esquema mostrado na figura 12, que a componente proporcional dá uma variação instantânea de $K.e(t)$ na saída. O tempo necessário para que a componente integral varie a saída do mesmo valor da variação da componente proporcional é chamado de tempo integral T_i . A ação integral irá repetir a correção da ação proporcional, a cada tempo integral (a unidade de T_i é dada em minutos para repetir – MPR, ou segundos para repetir – SPR).

Ainda segundo Bega (2003, p. 19) “ a existência da ação integral não garante que o processo seja estável, garante sim, que não haverá erro quando o processo atingir um estado estável”. Na figura 13, mostra-se o sistema de controle em malha fechada, e representa-se a curva de resposta de um controlador proporcional mais integral (PI), com ganho proporcional constante e com diversos valores de tempo integral. Nota-se que para tempo integral muito grande, a resposta tem uma “cauda” e a variável se aproxima do ponto de ajuste muito lentamente. Para tempo integral curto, o excesso de correção faz com que a variável ultrapasse o ponto de ajuste e demore a estabilizar (oscile). Como para os valores de tempo integral o erro será eliminado, o ajuste do valor de tempo integral será função, basicamente, do ajuste da componente proporcional. O valor do tempo integral a ser utilizado deverá ser tal que elimine o erro no menor tempo possível, sem afetar, significativamente, a taxa de amortecimento.

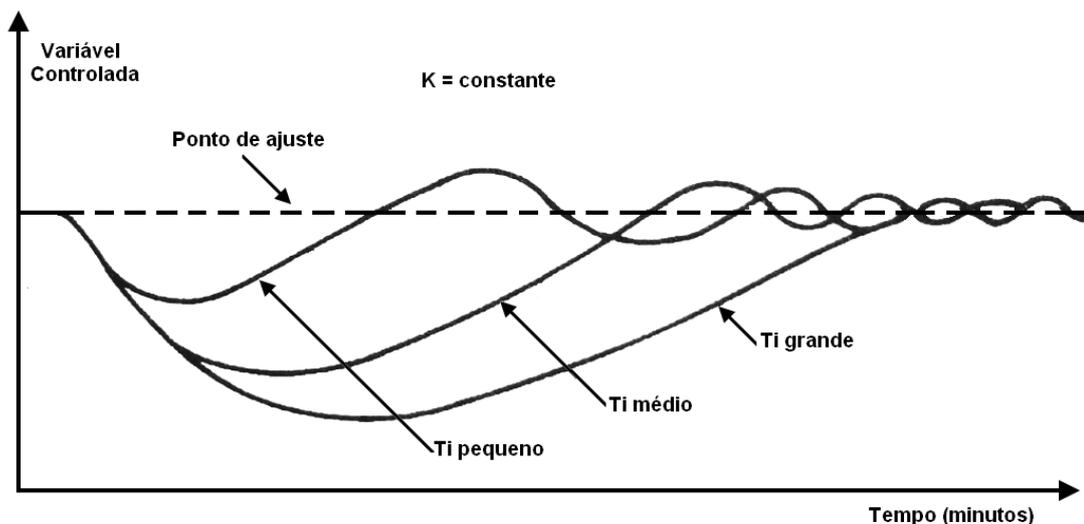


FIGURA 13: Resposta de um controlador PI

FONTE: BEGA, 2003, p. 20

5.5 Ação de controle proporcional – integral – derivativa (PID)

A combinação da ação de controle proporcional, ação de controle integral e ação de controle derivativo é denominada ação de controle proporcional-integral-derivativa. Esta ação de controle combinada possui as vantagens de cada uma das três ações de controle separadas. Ogata (2000, p. 182) define a ação de controle PID pela seguinte equação,

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \quad (5.11)$$

ou pela função de transferência,

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (5.12)$$

Bega (2003, p. 24) faz ainda uma comparação entre as três principais formas de controladores P, PI e PID, conforme é ilustrado na figura 14.

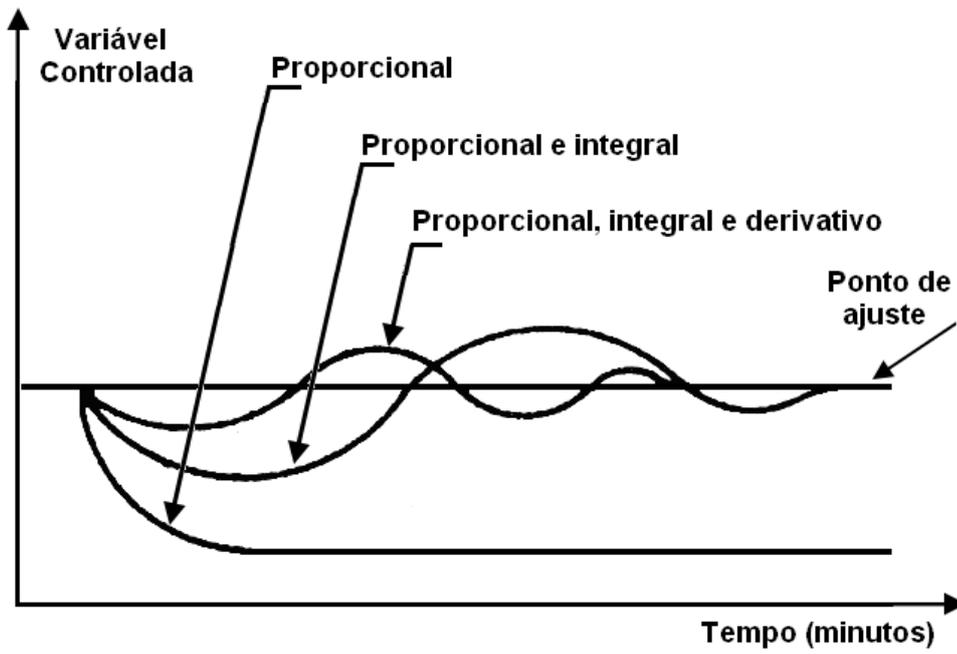


FIGURA 14: Resposta comparativa dos controladores: P, PI e PID

FONTE: BEGA, 2003, p. 24

VI REGRAS DE SINTONIA PARA CONTROLADORES PID

Na figura 15 é mostrado o controlador PID de um processo, “[...] quando se tem um modelo matemático do processo, é possível aplicar várias técnicas visando à determinação dos parâmetros do controlador que atendam às especificações de regime transitório e permanente do sistema a malha fechada” (OGATA, 2000, p. 544). Nos casos, contudo, em que o processo seja tão complicado que seu modelo matemático não possa ser obtido com facilidade, então a abordagem analítica para se projetar um controlador PID deixa de ser viável. Nestes casos deve-se fazer uso das técnicas experimentais de sintonia de controladores PID.

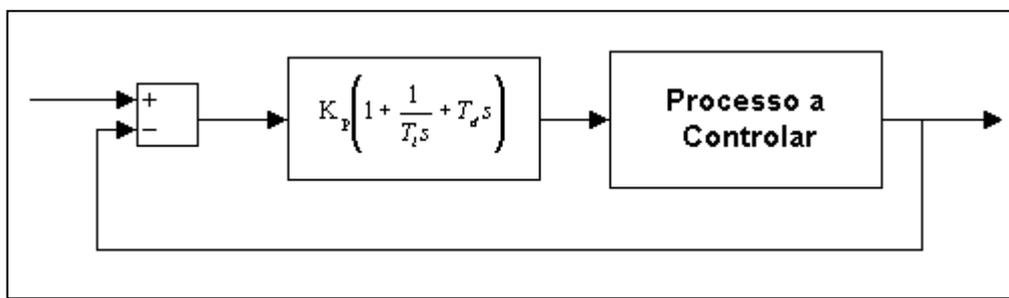


FIGURA 15: Controlador PID utilizado em um controle de processo

FONTE: OGATA, 1998, p.545

O procedimento de seleção dos parâmetros do controlador de modo a serem atendidas as especificações de desempenho é conhecido como sintonia do controlador. Ziegler e Nichols (apud, OGATA, 2000) propuseram as regras para sintonia de controladores PID (significando o ajuste dos valores de K_p , T_i e T_d) baseadas na resposta experimental a uma excitação degrau ou no valor de K_p que resulta em estabilidade marginal quando se utiliza unicamente a ação de controle proporcional. As regras de Ziegler e Nichols (apud, OGATA, 2000) são apresentadas nos itens 6.1.

6.1 Regras de Ziegler e Nichols para a sintonia de controladores PID

Ziegler e Nichols (apud, OGATA, 2000) propuseram regras para se determinarem os valores dos ganhos proporcional K_p , do tempo integral T_i e do tempo derivativo T_d , com base nas características da resposta transitória de um determinado processo a controlar. Tal determinação de parâmetros, ou sintonia, dos controladores PID pode ser feita na própria instalação, ou utilizando-se de experimentos sobre o processo.

Há dois métodos designados como regra de sintonia de Ziegler e Nichols (apud, OGATA, 2000). Em ambos se pretende um valor máximo de *overshoot* de 25% na resposta a uma excitação em degrau, conforme é mostrado na figura 16.

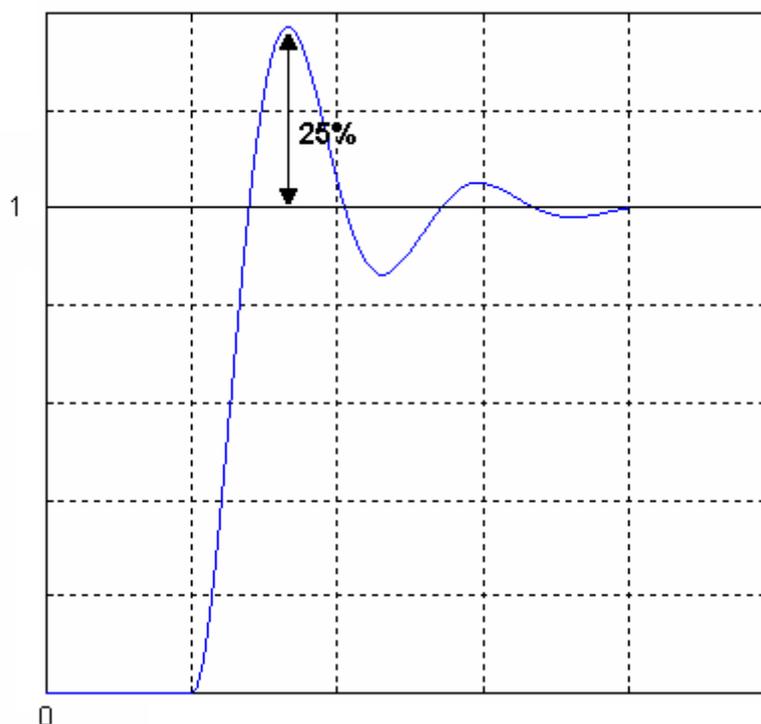


FIGURA 16: Curva de resposta ao degrau unitário mostrando um valor máximo de *overshoot* de 25%

FONTE: OGATA, 1998, p.545

6.1.1 Primeiro Método de Ziegler-Nichols

No primeiro método se obtém experimentalmente a resposta do processo a se controlar a uma excitação em degrau unitário, conforme mostrado na figura 17. Segundo Ogata (2000, p. 545), “[...] quando o processo a controlar não envolver integradores nem pólos dominantes complexos conjugados, a curva de resposta ao degrau unitário pode se assemelhar a uma curva em forma de S [...]”, como é mostrado

na figura 18 (Se a resposta não apresentar esta curva em forma de S, o método não se aplica). Tais curvas de resposta ao degrau podem ser geradas experimentalmente ou a partir de uma simulação dinâmica do processo a controlar.

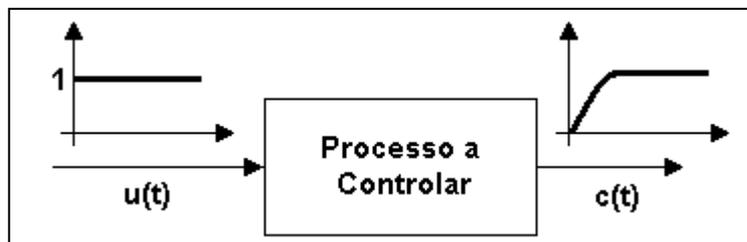


FIGURA 17: Resposta de um processo a uma excitação em degrau unitário

FONTE: OGATA, 1998, p.545

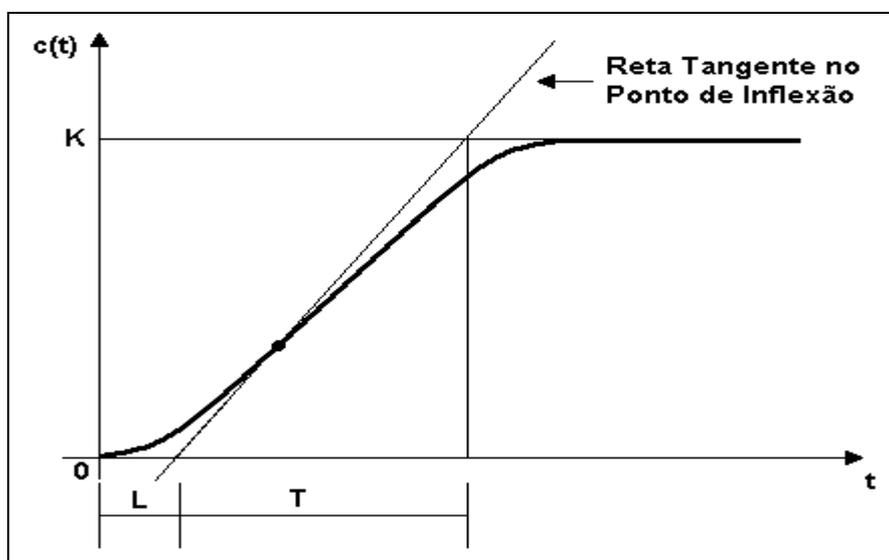


FIGURA 18: Curva de resposta em forma de “S”

Fonte: Ogata, 1998, p.545

A curva em forma de S pode ser caracterizada por duas constantes, o tempo de retardo L e a constante de tempo T . O tempo de retardo e a constante de tempo podem ser determinados traçando-se uma reta tangente à curva em forma de S no ponto de inflexão e determinando-se as interseções com o eixo dos tempos e com a reta $c(t) = k$, conforme assinalado na figura 18. A função de transferência $C(s)/U(s)$ pode ser aproximada à de um sistema de primeira ordem com retardo de transporte, como na expressão:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-Ls}}{Ts + 1} \quad (6.1)$$

Ziegler e Nichols (apud, OGATA, 2000) sugeriram ajustar os valores de K_p , T_i e T_d de acordo com a fórmula mostrada no quadro 4. Observe-se que o controlador PID sintonizado por meio do primeiro método das regras de Ziegler Nichols (apud, OGATA, 2000) fornece:

$$G_c(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (6.2)$$

$$G_c(s) = 1,2 \frac{T}{L} \left(1 + \frac{1}{2Ls} + 0,5Ls \right) \quad (6.3)$$

$$G_c(s) = 0,6T \frac{\left(s + \frac{1}{L} \right)^2}{s} \quad (6.4)$$

QUADRO 4: Primeira regra de Ziegler-Nichols

Tipo de Controlador	K_p	T_i	T_d
P_ Proporcional	$\frac{T}{L}$	∞	0
PI_ Proporcional e Integral	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID_ Proporcional, Integral e Derivativo	$1,2 \frac{T}{L}$	$2L$	$0,5L$

FONTE: OGATA, 1998, p.546.

Assim o controlador PID possui um pólo na origem e um zero duplo em $s = -1/L$.

6.1.2 Segundo Método de Ziegler-Nichols

Neste segundo método ajustam-se primeiro os valores de $T_i = \infty$ e $T_d = 0$. Utilizando-se somente a ação de controle proporcional (figura 19), aumenta-se o valor de K_p de 0 a um valor crítico K_{cr} para o qual o sinal de saída apresenta oscilações mantidas (Se o sinal de saída não apresentar oscilações, quaisquer que sejam os valores de K_{cr} , então o método não se aplica.). Em consequência, são determinados

experimentalmente os valores de ganho crítico K_{cr} e o período crítico correspondente P_{cr} (ver figura 20). Ziegler Nichols (apud, OGATA, 2000) sugeriram ajustar os valores dos parâmetros K_p , T_i e T_d de acordo com os valores mostrados no quadro 5.

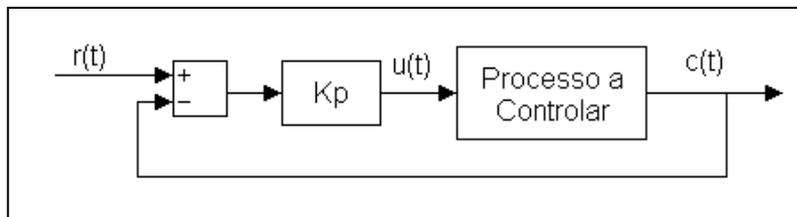


FIGURA 19: Sistema a malha fechada com controlador proporcional

FONTE: OGATA, 1998, p.546

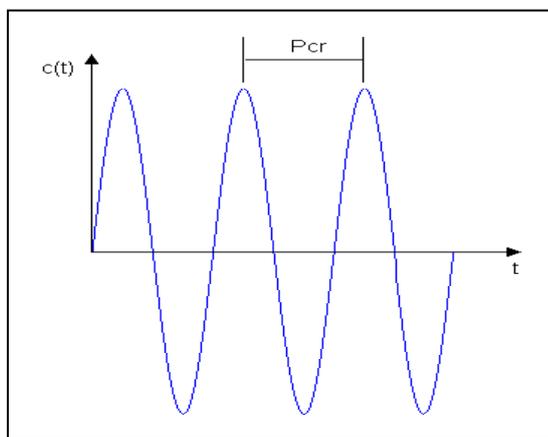


FIGURA 20: Oscilação mantida com período P_{cr}

FONTE: OGATA1998, p.546

QUADRO 5: Segunda regra de Ziegler-Nichols

Tipo de Controlador	K_p	T_i	T_d
P	$0,5 K_{cr}$	∞	0
PI	$0,45 K_{cr}$	$1/1,2 P_{cr}$	0
PID	$0,6 K_{cr}$	$0,5 P_{cr}$	$0,125 P_{cr}$

FONTE: OGATA, 1998, p.545

Observa-se que o controlador PID sintonizado de acordo com o segundo método das regras de Ziegler Nichols (apud, OGATA, 2000) fornece:

$$G_c(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (6.5)$$

$$G_c(s) = 0,6K_{cr} \left(1 + \frac{1}{0,5P_{cr}s} + 0,125P_{cr}s \right) \quad (6.6)$$

$$G_c(s) = 0,075K_{cr}P_{cr} \frac{\left(s + \frac{4}{P_{cr}} \right)^2}{s} \quad (6.7)$$

Assim, o controlador PID possui um pólo na origem e um zero duplo em

$$s = -\frac{4}{P_{cr}} .$$

VII SISTEMA SUPERVISÓRIO

7.1 Introdução Sobre Sistemas Supervisórios

Os sistemas supervisórios são de grande aplicação em todos os ramos industriais que sejam parcialmente ou totalmente automatizados, refletindo numa integração eficiente entre operador e processo, melhorando não somente a produtividade, como também a qualidade de seus produtos. Inicialmente será dada uma explanação geral sobre sistemas supervisórios, uma vez que o *software* escolhido para a interface homem máquina do controle do processo de bombeamento e aquecimento de água da planta laboratorial foi o Eclipse E3.

7.1.1 Uma Abordagem sobre Sistemas Supervisórios

Segundo Silva e Salvador (2005, p.1) “[...] os sistemas supervisórios são poderosos *softwares* aplicativos que permitem que sejam monitoradas e rastreadas informações do processo produtivo [...]”. As informações podem ser visualizadas por intermédio de quadros sinóticos animados com indicações instantâneas das variáveis de processo (vazão, temperatura, pressão, volume, etc.). Os dados são provenientes do controle do CLP, podendo os sistemas supervisórios gerenciar processos de qualquer tamanho ou natureza. Estes auxiliam no processo de implantação da qualidade e de movimentação de informações para gerenciamento e diretrizes. A escolha de um *software* de supervisão Scada/MMI é muito importante na estratégia de automação de uma empresa.

Com a implantação de um processo automatizado sem o sistema supervisório, temos um processo controlado de maneira eficiente, mas não temos a monitoração, supervisão e obtenção de dados de um sistema como esse que é de grande importância para o gerenciamento do processo, das informações de como está se comportando o mesmo; alarmes, curvas de processo, comportamento das variáveis críticas, situações de falhas, etc.

Atualmente os sistemas supervisórios podem ter uma arquitetura aberta, ligados em rede, permitindo que o fluxo de dados do processo ultrapasse o limite das paredes da

empresa e percorra o mundo através dos meios de comunicação existentes. No mercado hoje, os *softwares* de supervisão são divididos em dois grupos: MMI e SCADA.

7.1.2 MMI - *Man Machine Interface*

Os supervisórios MMI são os *softwares* que tem a finalidade de servir como uma interface homem máquina, ou seja, estes *softwares* não têm a função de controlar nenhuma parte da máquina ou processo, portanto se ocorrer qualquer problema durante a sua execução, não prejudicará a automação da máquina ou processo. Normalmente estes *softwares* apresentam facilidades de configuração, mas geralmente são limitados em segurança de dados, comunicação em rede, comunicação remota, controles de processo, etc.

7.1.3 SCADAS - *Supervisory Control and Aquisition Data System*

Os supervisórios SCADA são softwares que possuem as mesmas funções dos softwares MMI mais podem efetuar controle, distribuir informações entre estações via rede com performance e segurança, etc. Estes softwares normalmente são mais robustos e confiáveis para aplicações de grande porte e para aplicações distribuídas em varias estações.

7.2 Objetivos de um Sistema Supervisório

As principais funções de um supervisório são:

- Aquisição de dados: Esta é uma função muito importante para a funcionalidade do sistema. A aquisição de dados consiste na retirada de informações do processo através da conexão que computador terá com o CLP, que controla o processo e detém em sua memória todos os eventos que estão ocorrendo em tempo real.
- Gerenciamento de dados: Após a aquisição de dados do processo, a apresentação dos mesmos deve ser feita de maneira clara e precisa, e apresentar em tempo real

de execução ao operador os valores obtidos. A principal apresentação dos dados é demonstrada nas telas de apresentação dos processos (*view*). Esses dados são manipulados e distribuídos para os módulos do *software* e apresentados em forma de telas, relatórios, históricos, alarmes, etc.

7.3 Descrição dos Softwares Supervisórios

Atualmente, a maioria dos softwares de supervisão é multitarefa preemptivo. Sistemas preemptivos permitem que diversas tarefas sejam executadas simultaneamente, sendo que o controle dos recursos do processador é feito pelo sistema operacional. Desta forma, caso uma das tarefas requeira recursos do processador mesmo quando ela não estiver sendo processada, o sistema operacional interrompe outras tarefas de menor prioridade e envia recursos para executar a tarefa solicitante.

Os benefícios de ser preemptivo, é que com o aumento das funções dos sistemas de supervisão, cada vez mais tarefas são adicionadas a ele. Desta forma, é inadmissível interromper-se a aquisição de dados ou o teste de alarmes do sistema quando se imprime um relatório, ou mesmo quando se efetua *back-up* de seus arquivos. Nos sistemas de supervisão, por exemplo, a aplicação que faz a aquisição e gerenciamento dos dados tem prioridade maior do que aquela que exibe os dados na tela.

7.4 Arquitetura Básica

Na figura 21 é mostrado o fluxo percorrido pelos dados no campo até serem processados pelo sistema supervisorio:

1. O *driver* de comunicação lê os dados do CLP e transfere os valores para os endereços de sua tabela imaginária;
2. O programa da base de dados lê os dados da tabela imaginária do *driver* de comunicação, processa-os e transfere-os para a base de dados;

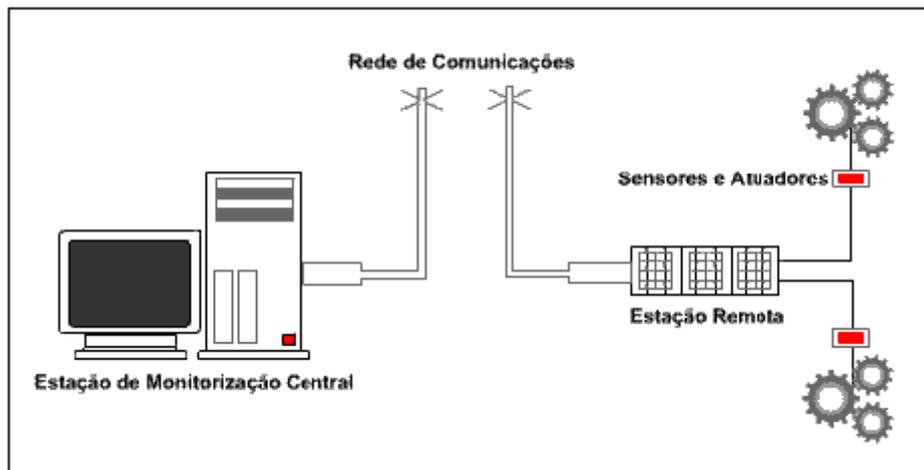


FIGURA 21: Arquitetura básica de um sistema supervisório

FONTE: SILVA e SALVADOR, 2005, p.3

7.4.1 Driver de Comunicação

O driver de comunicação é o programa responsável pela transferência dos dados do CLP ou hardware de processo para o sistema supervisório e vice-versa. Como cada hardware tem seu protocolo de comunicação específico, o driver de comunicação também é diferente para cada hardware.

No driver de comunicação são especificados os parâmetros de comunicação necessários para criar a tabela imagem do driver.

VIII RESULTADOS E ANÁLISES

Os estudos e testes foram realizados numa planta piloto. Esta planta consiste basicamente de um reservatório de água, onde é instalado uma pequena bomba automotiva de 12 Vdc, responsável por bombear a água para o circuito de aquecimento.

O circuito de aquecimento consiste numa resistência elétrica (resistência elétrica de eletrodomésticos – cafeteira), que aquece a água e a devolve através de um tubo de cobre ao reservatório, tomada do valor de temperatura é feita utilizando-se um termopar, implantado dentro do tubo de cobre. Na figura 22 representa-se esquematicamente a planta e o sistema de controle (PLC e sistema supervisório).

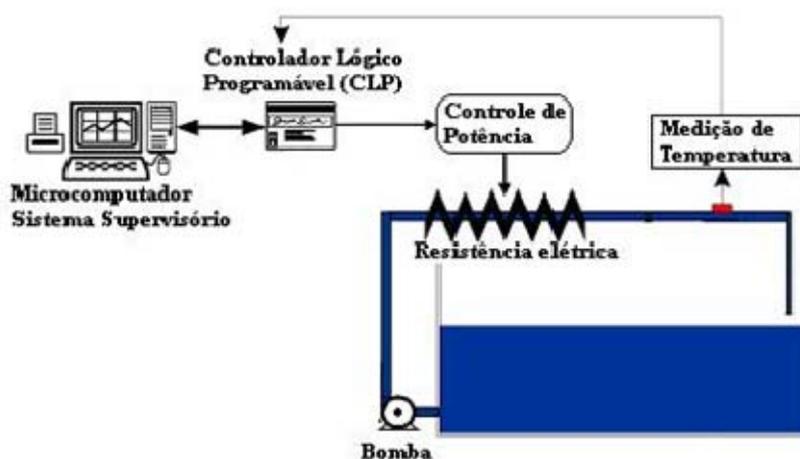


FIGURA 22: Planta piloto

FONTE: COSTA, 2005, p.8

Tem-se ainda a ação de controle, que trata-se de uma malha de controlador PI (proporcional e integral), implementado no PLC, na figura 23 representa-se o diagrama do sistema de controle. Esta malha deve ser configurada de modo a proporcionar o menor tempo de estabilização.

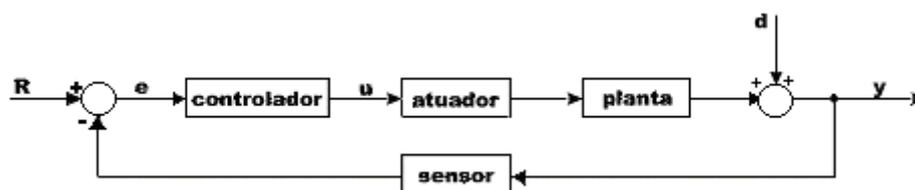


FIGURA 23: Diagrama de blocos de um sistema de controle

Na malha (figura 23) d representa um distúrbio, nesse caso pode-se considerar a vazão como um distúrbio.

8.1 Configuração dos Canais Analógicos e Função PID

Aplicando as ferramentas estudadas referentes à configuração dos módulos de função, pôde-se montar um esboço do que será implementado para controlar a planta de aquecimento e bombeamento de água. Partindo do processo temos um termopar que é um sensor, que conforme exposição a uma determinada temperatura gera uma milivoltagem, por meio de cálculos é possível converter essa diferença de potencial em unidade de engenharia, para isso, um circuito que converta esse valor é utilizado. Os canais analógicos lêem estes valores (variável medida VM), e utilizando-se a função F – ANLOG 006 faz-se a conversão analógica para digital, onde 0V corresponde a 0 e 10V corresponde a 32000.

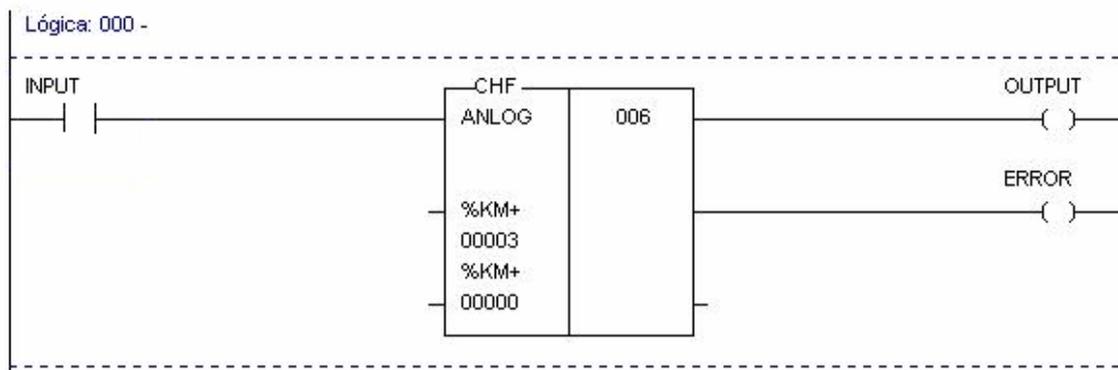
Após ter sido feita a conversão A/D, o valor da variável medida VM é armazenado em uma memória, e este valor será utilizado na função F – PID 033, que fará a ação de controle, especificando a saída no valor de *set point* especificado. Logo este valor de saída é armazenado em uma memória, que será convertida em um sinal analógico, e efetuará a ação de controle (variável atuada VA). Na figura 24 é ilustrado o esquema (programação Ladder) descrito acima.

8.2 Configuração dos Módulos de Funções

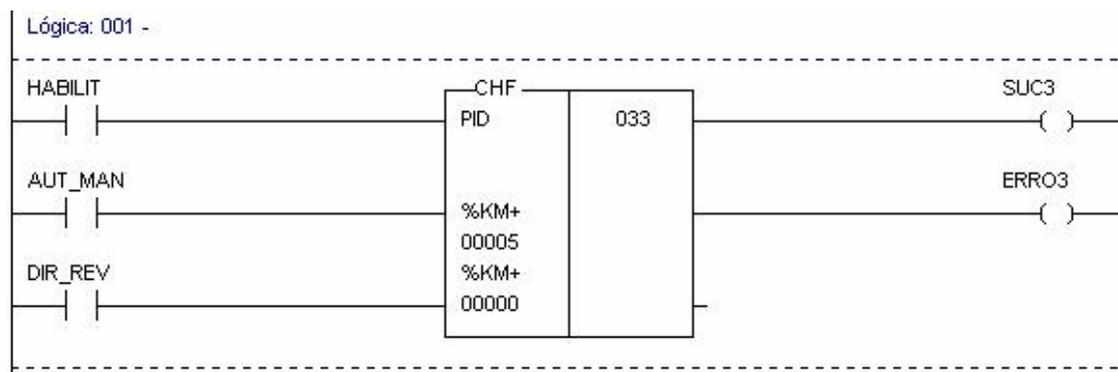
8.2.1 Configuração dos Canais Analógicos_ Conversão A/D e D/A

8.2.1.1 Parâmetros Utilizados

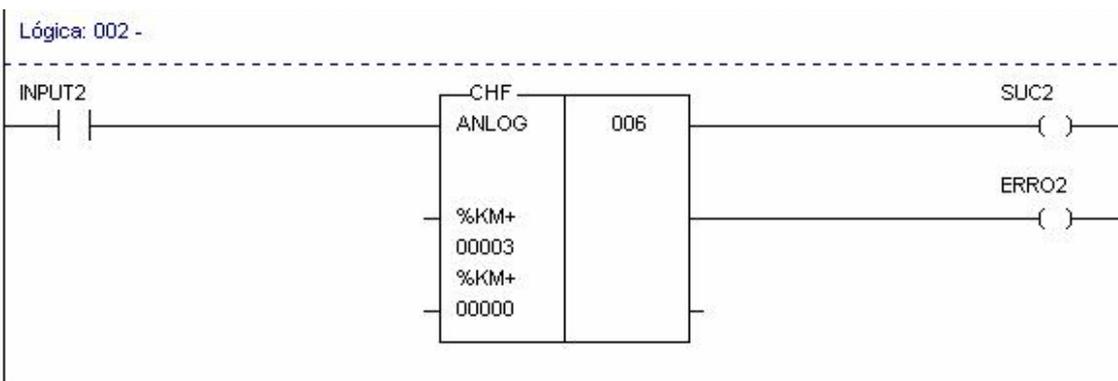
Existem alguns parâmetros a serem configurados nas funções pré-determinadas no MasterTool. Analisando primeiro a função ANLOG 006 (responsável pela configuração dos canais analógicos) baseados na figura 07, têm-se os seguintes parâmetros, no OPER 3:



a)



b)



c)

FIGURA 24: Programação Ladder da configuração dos canais analógicos e da ação de controle PID a) conversão do valor medido analógico-digital, b) função de controle PI, c) conversão do valor de atuação digital-analógico

a) Para o canal DAC 01 foram definidos:

- Especificação do canal a ser convertido: utiliza-se a constante %KM+00000 para designar o canal DAC 01.
- Tipo de conversão a ser realizada no canal definido pelo parâmetro anterior: utiliza-se a constante %KM+00000 para designar a conversão A/D (analógico digital, pois se trata do sinal proveniente da planta)
- Especificação do operando onde é armazenado o valor a ser escrito a partir da leitura do sinal analógico: utiliza-se de uma memória, cujo endereço varia de acordo com o projeto de programação ladder.

b) Para o canal DAC 02 foram definidos:

- Especificação do canal a ser convertido: utiliza-se a constante %KM+00001 para designar o canal DAC 02.
- Tipo de conversão a ser realizada no canal definido pelo parâmetro anterior: utiliza-se a constante %KM+00001 para designar a conversão D/A (digital - analógico, pois se trata do sinal proveniente do PLC enviado à planta)
- Especificação do operando onde é armazenado o valor a ser escrito a partir da leitura do sinal analógico: utiliza-se de uma memória, cujo endereço varia de acordo com o projeto de programação ladder.

A ligação física dos canais analógicos da planta como o PLC é mostrada na figura 25.

8.2.2 Configuração da Função PID

8.2.2.1 Parâmetros Utilizados

Assim como na função Anlog 006, na função PID 033 também existem alguns parâmetros que necessitam de parametrização e definição da tabela que contem os parâmetros utilizados pelo PID.

A tabela é definida por uma tabela de memória (TM), e é criada no projeto de programação no MasterTool, sendo preenchida pelo programa aplicação, podendo ser até modificada pelo programa supervisor.

- Leitura do valor medido na planta (neste caso o valor da temperatura na saída do aquecedor): este valor é o armazenado pela função de conversão analógico-digital, localizado na memória especificada no programa ladder.
- Especificação do ponto de ajuste (*set-point*): é definido como uma memória, cujo valor armazenado é o valor desejado de operação da planta, podendo ser modificado conforme interesse do operador ou processo, sendo normalmente feita com o programa supervisor.
- Escrita do valor de atuação no processo: é o valor que irá atuar na planta sendo definido como uma memória. Este valor aciona uma conversão digital - analógico que atuará na planta.

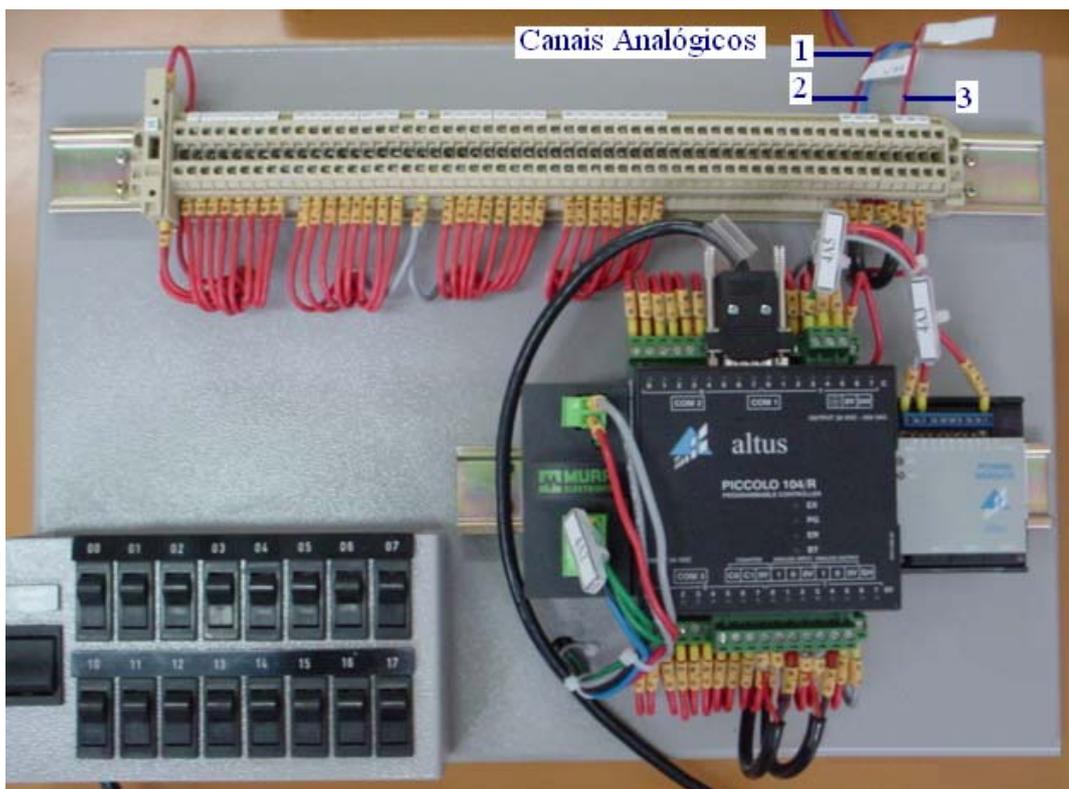


FIGURA 25: Configuração da conexão do PLC ao processo (planta), onde: 1) Canal Analógico DAC 01; 2) Comum e 3) Canal Analógico DAC 02.

8.2.2.2 Configuração dos Parâmetros Adicionais para o Controle PID

Os parâmetros adicionais são os parâmetros configurados dentro da tabela memória de dezesseis posições, na qual se encontram os valores dos ganhos proporcional, integral e derivativo, o que faz com que esta tabela esteja intimamente ligada à sintonia do controlador PID.

Inicialmente a sintonia do controlados foi realizada pelo método mais simples, porém bastante utilizado, o da “tentativa e erro”, este método foi possível graças ao processo ser um processo térmico bastante conhecido e à experiência do suporte técnico da empresa Altus, pois deste foram dadas as primeiras “tentativas”, onde cada tentativa teve seu desempenho acompanhado na forma de gráficos.

8.2.2.3 Gráficos de Desempenho da Função PID – Método das Tentativas

Nas figuras 26 a 29 representa-se a evolução temporal da ação de controle da função PI no intervalo de tempo medido, onde é mostrado o valor do Ganho Proporcional (GP), do Ganho Integral (GI) e do Ganho Derivativo (GD) e o tempo de estabilização em cada curva. Mostram-se ainda os valores do *Set Point* ou Ponto de Ajuste (PA), o Valor Medido (VM) e o Valor Atuado (VA). Nas curvas mostradas tem-se justamente a evolução temporária destes três últimos valores, onde em verde tem-se o ponto de ajuste, em rosa o valor medido e em azul o valor atuado, lembrando que esses valores são sinais digitais, tal que:

- Para o valor medido tem-se a relação:

000 ----- 0V-----0°C

255-----10V-----100 °C

- E para o valor atuado tem-se:

000-----0V----- Potência máxima na resistência de aquecimento

255-----10V----- Potência mínima na resistência de aquecimento

Desta forma as curvas apresentadas mostram como funciona a ação de controle.

Para um ganho proporcional de 5 e um ganho integral de 1000 segundo/repetição (valores iniciais cedidos pelos técnicos da empresa Altus, fornecedora do PLC _ Piccolo 104 /R), pode-se observar que não foi obtido um resultado satisfatório, uma vez que houve um erro muito grande em regime estacionário e muitas oscilações em regime transiente, o que não é conveniente em um processo de regulação automática.

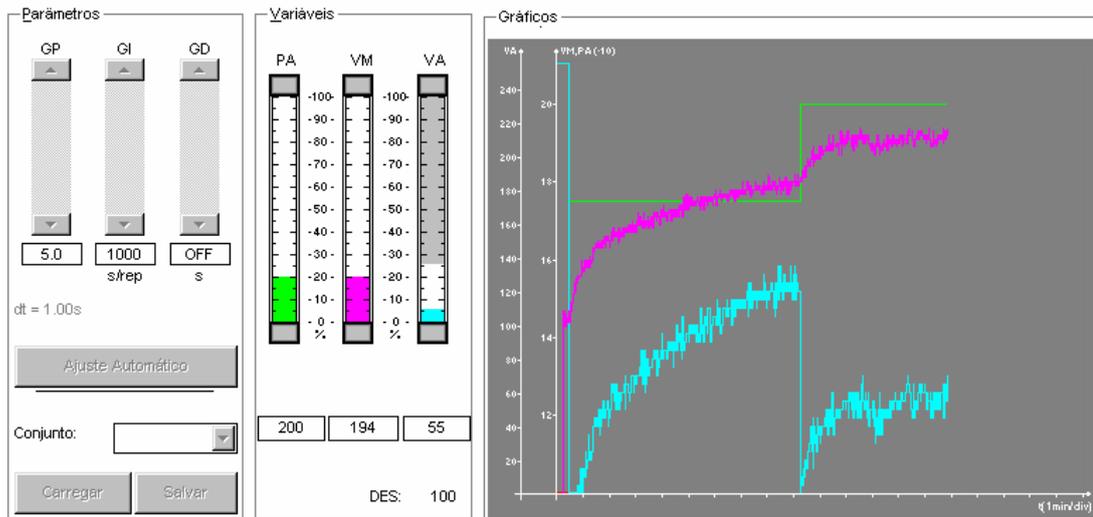


FIGURA 26: Comportamento do controlador PI ($G_p=5.0$ e 1000 s/rep).

Como para os ganhos anteriores não houve resultado satisfatório, a idéia básica seria a de manter o ganho integral e aumentar o ganho proporcional de 5 para 10, com isso aumenta-se a estabilidade do controlador e a velocidade de resposta do processo.

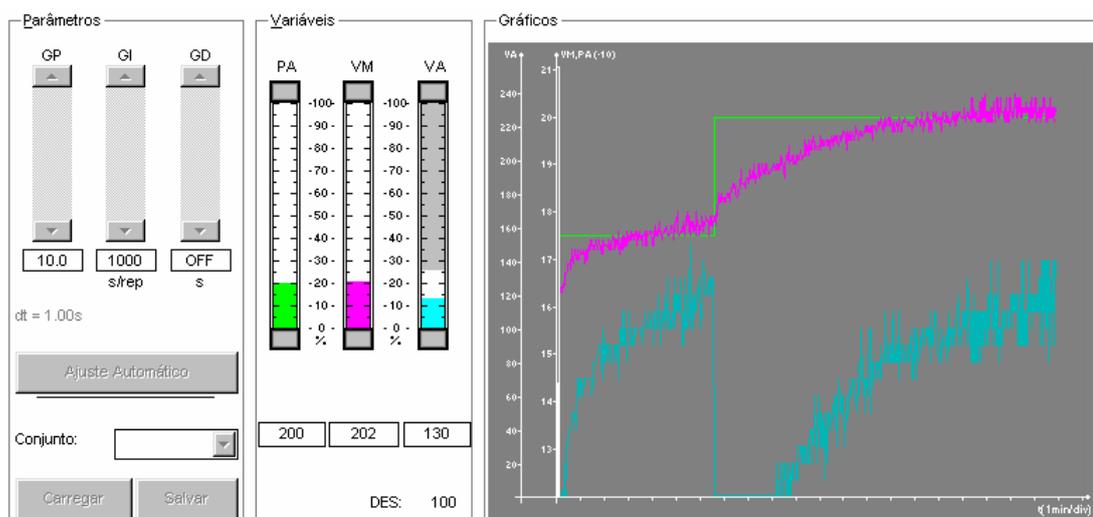
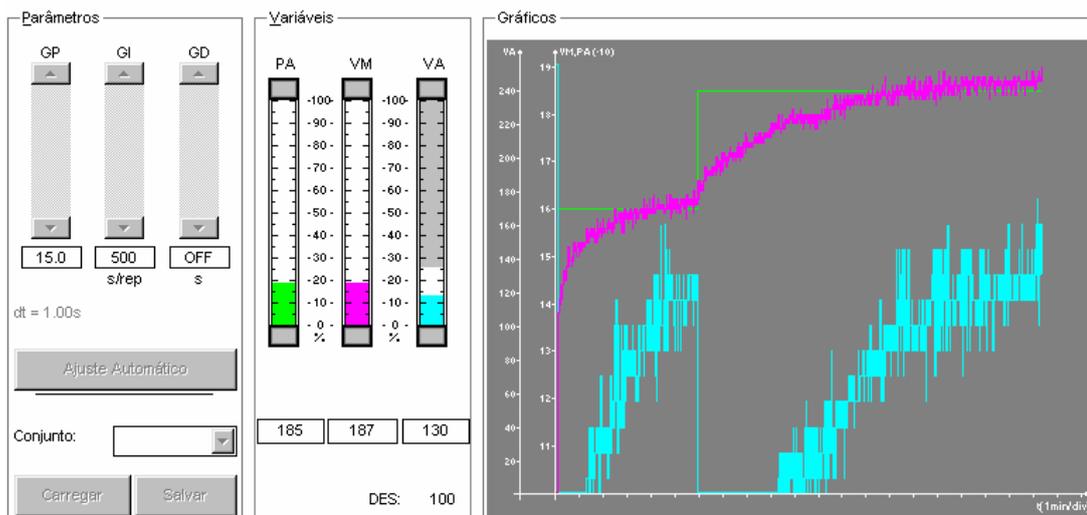
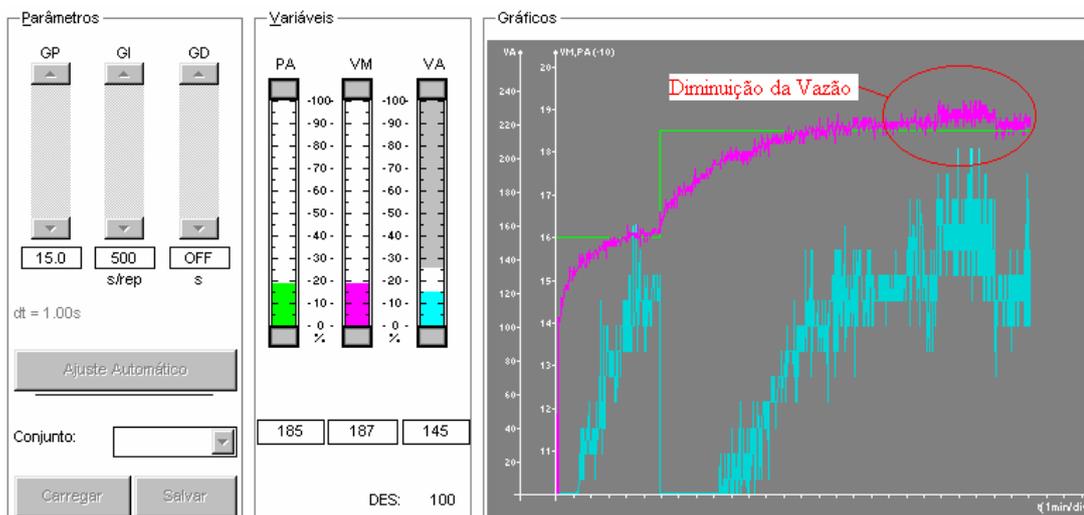


FIGURA 27: Comportamento do controlador PI ($G_p=10.0$ e 1000 s/rep).

Continuando a aumentar o ganho proporcional de 10 para 15 e diminuindo pela metade a constante de tempo integral, foi obtido a curva apresentada na figura 28. Deixando por um tempo maior, notamos que o resultado já é satisfatório, porém um fato interessante é que ao alterarmos a vazão o controlador a enxerga como sendo um distúrbio e tende a controlá-lo, fazendo com que a saída siga a referencia, conforme é mostrado na figura 28 – b.



a)



b)

FIGURA 28: a) Comportamento do controlador PI ($G_p=15.0$ e 500 s/rep), b) com variação da vazão.

Aumentando o ganho proporcional de 15 para 20 e mantendo o mesmo ganho integral, pode-se observar um resultado muito bom, com bom regime transiente e praticamente com erro nulo em regime permanente. As curvas mostradas na figura 28 possuem os mesmos ganhos, porém em circunstâncias diferentes, comprovando o bom resultado do teste de sintonia do controlador PID.

Um problema encontrado deve-se à planta, pois a água fica “viciada” rapidamente, sendo a planta um circuito fechado de pequena capacidade. A água do sistema rapidamente fica superaquecida o que gera um erro de *offset* conforme indica-se na figura 29 - b. Algumas idéias estão sendo analisadas para sanar tal problema, uma delas seria acoplar um outro reservatório para receber a água aquecida aumentando a capacidade do sistema, ou ainda a utilização de um circuito aberto, no qual a água aquecida não retorna à resistência de aquecimento como é feito atualmente.

Deixando a planta “rodar” por um tempo maior e dando sucessivos picos no valor do *set point* o regulador funciona perfeitamente, porém devido ao problema de superaquecimento da água do sistema à temperaturas próximas ao ponto de ebulição da água ocorre alguns distúrbios no controle devido a problemas principalmente no processo de medição.

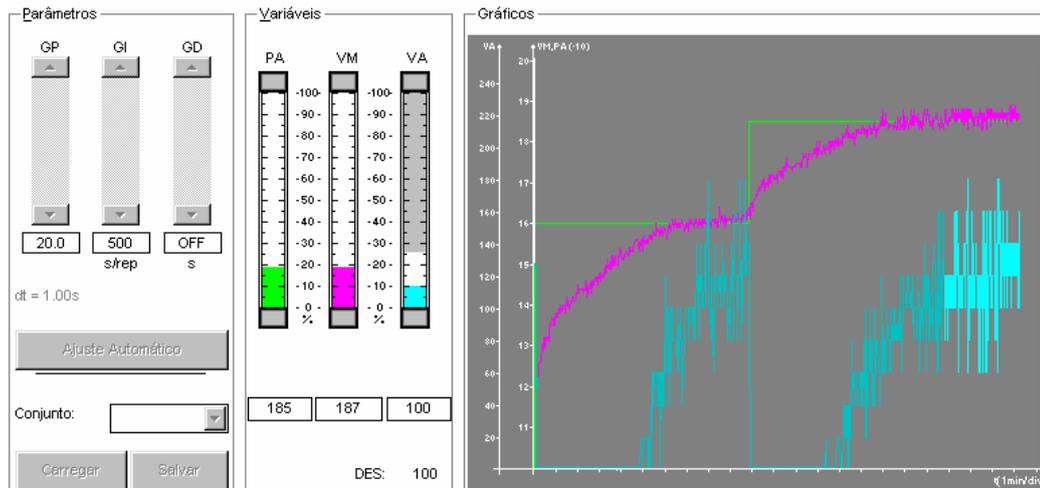
8.2.2.4 Gráficos de Desempenho da Função PID – Primeiro Método de Ziegler Nichols

Considerando (quadro 4), que $R = \frac{K}{T}$, tem-se segundo Costa (2005, p. 37) o quadro de resultados para o primeiro método de Ziegler Nichols, para essa mesma planta de aquecimento, conforme é ilustrado no quadro 6.

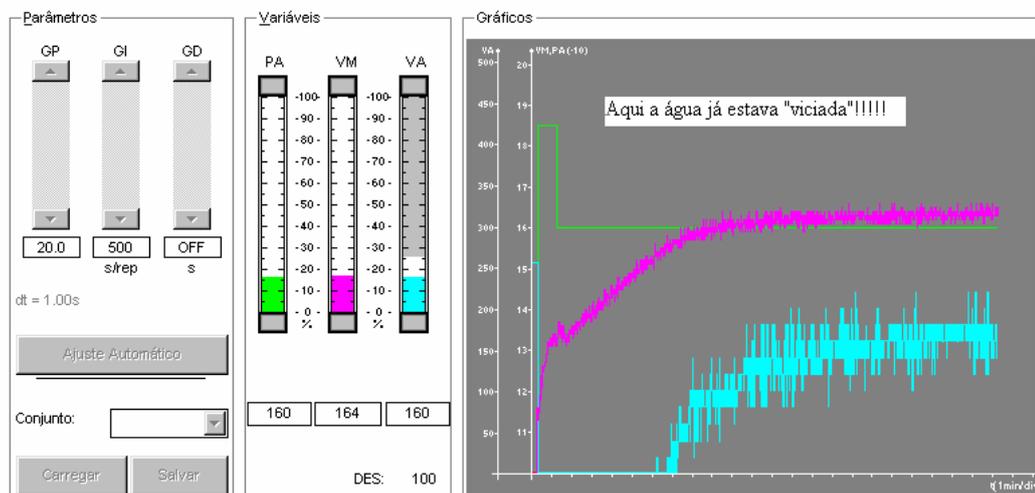
QUADRO 6: Parâmetros de sintonia para o primeiro Método de Ziegler-Nichols

Tipo de Controle	Kp	Ki	T _i	Kd	T _d
Controlador P	24,65	-	-	-	-
Controlador PI	22,18	0,95	23,33	-	-
Controlador PID	29,58	2,11	17	103,55	3,5

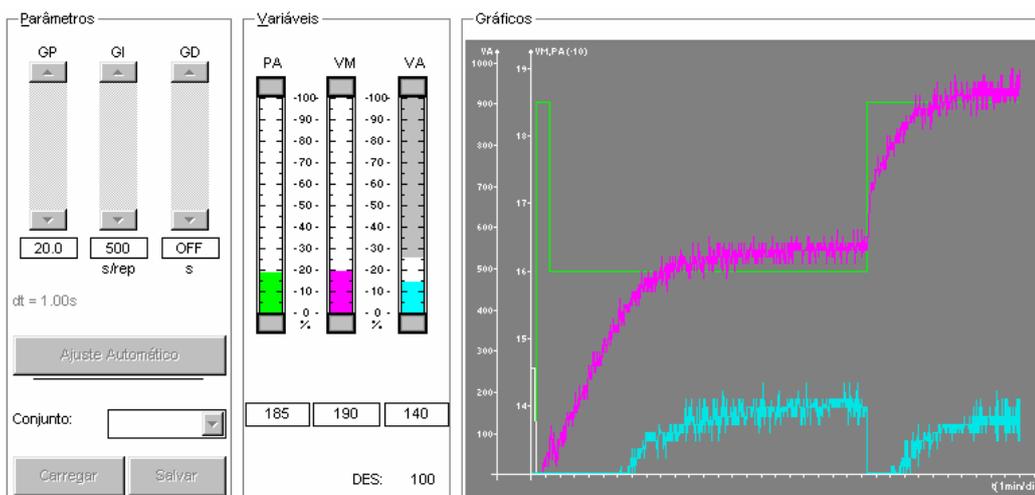
FONTE: COSTA, 2005, p.37



a)



b)



c)

FIGURA 29: a) Comportamento do controlador PI ($G_p=20.0$ e 500 s/rep), b) com a “água viciada”, c) variação do set-point.

Aplicando os parâmetros obtidos para o controlador PI, na planta tem-se a seguinte curva de desempenho, figura 30.

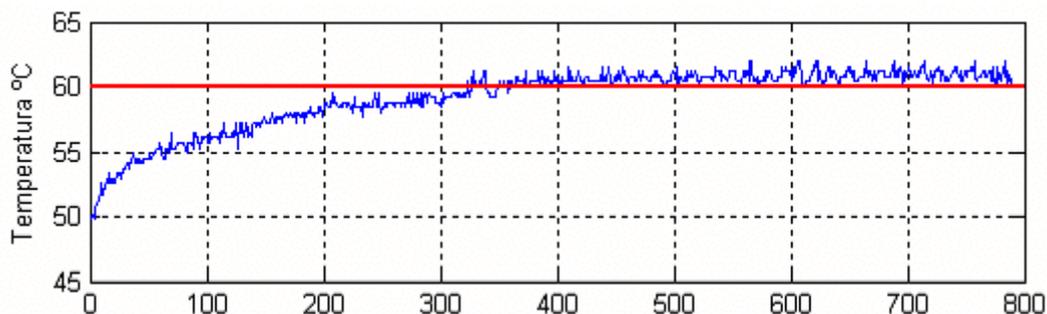


FIGURA 30: Resposta no tempo do controle PI para o método Ziegler-Nichols

FONTE: COSTA, 2005, p.47

8.2.2.5 Tela de Supervisão para o Método de Ziegler Nichols

Na figura 31 tem-se uma tela de supervisão desenvolvida no Elipse E3 para realizar a supervisão e aquisição de dados referentes aos testes para a sintonia de malha utilizando o primeiro método de Ziegler Nichols (apud, OGATA, 2000).

8.3 Considerações finais

O trabalho desenvolvido obteve bons resultados, uma vez que se conseguiu realizar um profundo estudo sobre o funcionamento das três partes envolvidas no projeto: a planta piloto, o PLC e o controlador PI e o *software* supervísório Elipse E3, o que contribuiu para a correta aplicação desse ferramental, respondendo consequentemente pelo bom desempenho do controle da planta de aquecimento. Esses bons resultados puderam ser observados na prática, como é mostrado na curva da figura 29-b, onde o controlador teve um bom desempenho, tanto em sua estabilização como no tempo referente à estabilização, considerando-se que este tempo é relativamente longo por se tratar de uma planta de aquecimento de água.



FIGURA 31: Tela de supervisão para coleta de dados para o método de Ziegler Nichols.

IX CONCLUSÃO

A Engenharia de Controle e Automação é uma ciência que evolui rapidamente, onde novas tecnologias são desenvolvidas diariamente. Portanto, não se pode apoiar o ensino desta ciência baseado em tecnologias ou equipamentos, sendo necessário então dar ao estudante uma forte base teórica para que este possa compreender de forma segura as questões práticas. Contudo é importante mostrar ao estudante montagens práticas, que consolidem o que foi aprendido em sala de aula e também o prepare para o mercado de trabalho.

Este projeto possibilitou um grande contato com as principais tecnologias empregadas na área de controle e automação. Uma vez que o conhecimento do funcionamento do PLC, e do sistema supervisório abre portas para que demais estudantes possam realizar projetos, utilizando-se desta tecnologia para implementar sua ação de controle.

Houve contato com a programação ladder, que é a linguagem mais utilizada dentre os programadores de PLC, devido a sua simplicidade e facilidade de programação e com a programação do sistema supervisório Elipse E3, bastante difundido no mercado, assim como a aplicação de um controlador PID (identificação, sintonia, etc) a uma planta, aplicando os conhecimentos adquiridos em aula, bem como novos conhecimentos buscados em livros, manuais, tutoriais, entre outros.

Como implementação futura, será feito um controlador multivariável, contado agora com a vazão. Este controlador atuará então a partir da vazão de líquido no tudo e da temperatura de saída do líquido após a resistência, a tensão aplicada à resistência para gerar a quantidade correta de calor para manter o valor da temperatura no *set point* desejado.

X REFERÊNCIAS BIBLIOGRÁFICAS

BEGA, E. A.. **Instrumentação Aplicada ao Controle de Caldeiras**. 3ª Edição. Rio de Janeiro: Editora Interciência, 2003. 179 p.

COSTA, Cássio Pascoal. **Avaliação de projetos de controladores**. 2005. 51 f. Monografia – Engenharia de Controle e Automação, Universidade Federal de Ouro Preto, Ouro Preto, 2005.

EMPRESA ALTUS. **Manual de Programação do MasterTool**, Ref. 6399-100.4. Rev. A 11/97. Disponível em: www.altus.com.br/manuais, Acesso em: 2003.

EMPRESA ALTUS. **Manual de Utilização do PLC Piccolo 104/R**, Ref. 6299-014.4. Rev. A 12/95. Disponível em: www.altus.com.br/manuais, Acesso em: 2003.

FRANÇA, J. L.. **Manual para Normalização de Publicações Técnico-Científicas**. 6ª Edição.rev. e amp. Belo Horizonte: Editora UFMG, 2003. 230 p.

GEORGINI, M.. **Automação Aplicada – Descrição e implementação de sistemas seqüenciais com PLC's**. 6ª Edição. São Paulo: Editora Érica, 2000. 228 p.

OGATA, K.. **Engenharia de Controle Moderno**. 3ª Edição. Rio de Janeiro: Editora Prentice-Hall do Brasil, 1998. 813 p.

SILVA, A. P. G., SALVADOR, M.. **O que são sistemas supervisórios**. 2004, 5 f. *Elipse Software*, São Paulo, 2004.

ZIEGLER,J. G.; NICHOLS, N. B. **Optimum Settings for Automatic Controllers**. ASME Trans 64, 1942. 759-68 p.. (apud, OGATA, K.. **Engenharia de Controle Moderno**. 3ª Edição. Rio de Janeiro: Editora Prentice-Hall do Brasil, 1998. 813 p.).

ZIEGLER,J. G.; NICHOLS, N. B. **Process lags in Automatic Control Circuits**. ASME Trans 65, 1943. 433-44 p.. (apud, OGATA, K.. **Engenharia de Controle Moderno**. 3ª Edição. Rio de Janeiro: Editora Prentice-Hall do Brasil, 1998. 813 p.).

ROCHA, R. ; MARTINS FILHO, L. de S. ; PRADO, R. S. ; SOPRANI, L. R..

Proposta de planta piloto: controle de vazão e temperatura de líquidos em tubos.

In: Congresso Brasileiro de Ensino de Engenharia - COBENGE 2004, 2004, Brasília.

Congresso Brasileiro de Ensino de Engenharia - COBENGE 2004, 2004.