



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



**SISTEMA DE MONITORAMENTO DO BOMBEAMENTO DE**  
**LÍQUIDOS CONTROLADOS ENTRE BACIAS DISTANTES**

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

**RODRIGO SOUZA BATISTA**

**Ouro Preto, 2007**

# **SISTEMA DE MONITORAMENTO DO BOMBEAMENTO DE LÍQUIDOS CONTROLADOS ENTRE BACIAS DISTANTES**

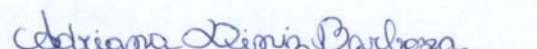
Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como parte dos requisitos para obtenção de Grau em Engenheiro de Controle e Automação.

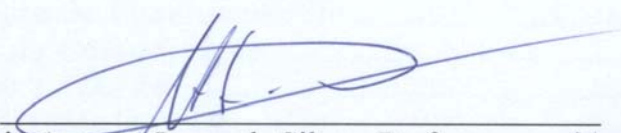
**Orientadora:** Adriana Diniz Barbosa

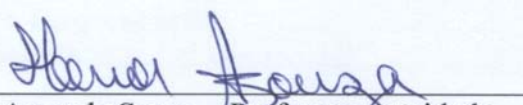
**Ouro Preto**  
**ESCOLA DE MINAS - UFOP**  
**AGOSTO / 2007**

Monografia defendida e aprovada, em 31 de agosto de 2007, pela comissão avaliadora constituída pelos professores:

Agradeco a Deus por todas as oportunidades que foram concedidas, a Fundação Gorceix pelo apoio, a professora Adriana pela dedicaçõ e a todos os funcionários da empresa INB que contribuíram direta ou indiretamente para esse trabalho.

  
Adriana Diniz Barbosa – Orientadora

  
Sávio Augusto Lopes da Silva – Professor convidado

  
Henor Artur de Souza – Professor convidado

## **AGRADECIMENTO**

Agradeço a Deus por todas as oportunidades que foram concedidas, a Fundação Gorceix pelo apoio, a professora Adriana pela dedicação e a todos os funcionários da empresa INB que contribuíram direta ou indiretamente para esse trabalho.

## SUMÁRIO

|  |      |
|--|------|
| <b>LISTA DE FIGURAS</b> .....  | vi   |
| <b>LISTA DE QUADROS</b> .....  | vii  |
| <b>RESUMO</b> .....  | viii |
| <b>ABSTRACT</b> .....  | ix   |
| <b>I – INTRODUÇÃO</b> .....  | 1    |
| 1.1 – Objetivo.....  | 3    |
| 1.1.1 - Objetivo Geral.....  | 3    |
| 1.1.2 - Objetivos Específicos.....                                     | 3    |
| 1.3 – Metodologia Adotada.....   | 4    |
| <b>II – REVISÃO BIBLIOGRÁFICA</b> .....                                | 5    |
| 2.1 - Controlador Lógico Programável - CLP.....                        | 5    |
| 2.1.1 - Arquitetura Básica de um CLP.....                              | 5    |
| 2.1.2 - Princípio de Funcionamento.....                                | 6    |
| 2.1.3 - Redes de Comunicação.....                                      | 8    |
| 2.2 – Linguagem <i>LADDER</i> .....                                    | 8    |
| 2.2.1 - Corrente Lógica Fictícia.....                                  | 9    |
| 2.2.2 - Instruções de BIT.....   | 10   |
| 2.2.3 - Temporizadores.....  | 11   |
| 2.2.4 - Contadores Simples.....  | 11   |
| 2.2.5 - Instruções de Movimentação.....                                | 12   |
| 2.2.6 - Instruções de Comparação.....                                  | 12   |
| 2.3 – Sistema Supervisório.....  | 13   |
| 2.4 – Telemetria.....  | 15   |
| <b>III – DESENVOLVIMENTO</b> .....                                     | 17   |
| 3.1 – Atuadores e Sensores Utilizados.....                             | 19   |
| 3.1.1 – Sensor de Nível.....   | 19   |
| 3.1.2 – Transdutor de Pressão.....                                     | 22   |
| 3.1.3 – Chave de Fluxo.....  | 24   |
| 3.1.4 – Válvula Controladora de Bomba.....                             | 25   |
| 3.2 – Programação do Controlador.....                                  | 27   |
| 3.2.1 - Características do Controlador <i>CompactLogix 5332E</i> ..... | 27   |
| 3.2.2 - Configuração do Controlador <i>CompactLogix 3225E</i> .....    | 30   |
| 3.2.3 - Programação do Controlador <i>CompactLogix 3225E</i> .....     | 32   |
| 3.2.3.1 - Monitoramento do Nível.....                                  | 32   |

|   |           |
|---|-----------|
| 3.2.3.2 - Acionamento da Válvula Controladora de Bomba..... | 34        |
| 3.2.3.3 - Bomba da Bacia de Finos .....                     | 35        |
| 3.2.3.4 - Horômetro .....                                   | 35        |
| 3.2.3.5 - Transdutor de Pressão .....                       | 36        |
| 3.2.3.6 - Chave de Fluxo .....                              | 37        |
| 3.2.3.7 - Telemetria .....                                  | 37        |
| 3.3 – Sistema de Telemetria.....                            | 38        |
| 3.4 – Sistema Supervisório.....                             | 40        |
| 3.4.1 – <i>Tags</i> e o <i>Tag data base</i> .....          | 40        |
| 3.4.2 – Criando <i>Displays</i> Gráficos.....               | 42        |
| 3.4.2.1 - Criando Botões .....                              | 43        |
| 3.4.2.2 - <i>Display</i> Numérico .....                     | 44        |
| 3.4.2.3 - <i>String Display</i> .....                       | 45        |
| 3.4.2.4 - Entrada Numérica .....                            | 46        |
| 3.4.3 – Criando Animações .....                             | 47        |
| 3.4.3.1 - Visibilidade .....                                | 47        |
| 3.4.3.2 - Cor.....  | 48        |
| 3.4.3.3 - Preenchimento .....                               | 48        |
| <br>  |           |
| <b>IV – CONCLUSÃO .....</b>                                 | <b>49</b> |
| <br>  |           |
| <b>V - REFERÊNCIAS BIBLIOGRÁFICAS .....</b>                 | <b>50</b> |
| <br>  |           |
| ANEXO A – <i>TAGS</i> UTILIZADAS NO RSLOGIX5000.....        | 55        |
| ANEXO B – PROGRAMAÇÃO EM <i>LADDER</i> .....                | 58        |
| ANEXO C – <i>TAGS</i> UTILIZADAS NO RSVIEW32 .....          | 64        |

## LISTA DE FIGURAS

|  |    |
|--|----|
| Figura 2.1- Diagrama de blocos simplificado de um CLP .....  | 7  |
| Figura 2.2- Exemplo de instruções em linguagem LADDER. ....  | 9  |
| Figura 2.3 - Sentido da corrente fictícia. ....  | 10 |
| Figura 2.4 - Utilização de instruções de Bit.....  | 10 |
| Figura 2.5 – Utilização de temporizadores .....  | 11 |
| Figura 2.6 - Utilização de Contatores.....   | 12 |
| Figura 2.7 – Instrução de Movimentação. ....   | 12 |
| Figura 2.8 – Instruções de Comparação. ....  | 13 |
| Figura 3.1 - Atual sistema de drenagem da INB.....   | 17 |
| Figura 3.2 – Vista Aérea da planta da INB .....  | 19 |
| Figura 3.3 - Sensor/Transmissor Ultrasônico .....  | 20 |
| Figura 3.4- Sensor/Transmissor de pressão piezo-resistivo.....   | 23 |
| Figura 3.5 – (a) Distancias mínimas recomendadas; (b) Fixação da<br>palheta na haste.....                          | 25 |
| Figura 3.6 – Esquema funcional da Válvula controladora de Bomba .....  | 26 |
| Figura 3.7 – CPU L32E da família <i>CompactLogix3225EI</i> .....   | 27 |
| Figura 3.8 – Esquema elétrico do módulo de entrada discreto (a) e, do<br>módulo de saída discreto (b).....         | 29 |
| Figura 3.9 – Esquema elétrico do módulo de entrada analógico .....   | 30 |
| Figura 3.10 - Criando um novo projeto (a). Configurando módulos de<br>entradas e saídas (b).....                   | 31 |
| Figura 3.11 – Principais tipos de dados. ....  | 32 |
| Figura 3.12 – Exemplo de uma rede de telemetria com 2 transceptores..  | 38 |
| Figura 3.13 – Interligação dos módulos de entrada/saída do sistema de<br>telemetria com dispositivos externos..... | 39 |
| Figura 3.14- Configuração de <i>tags</i> analógicas .....  | 41 |
| Figura 3.15 – Configuração de <i>tags</i> digitais. ....   | 42 |
| Figura 3.16 – Tela geral do sistema supervisorio. ....   | 42 |
| Figura 3.17- Configuração de botões .....  | 44 |
| Figura 3. 18 – (a) Configuração de displays numéricos (b) configuração<br>das entradas numéricas.....              | 45 |
| Figura 3.19 - Diagrama de blocos simplificado de um CLP .....  | 46 |
| Figura 3.20 - Configuração da animação da bomba .....  | 48 |

**LISTA DE QUADROS**

|   |    |
|---|----|
| Tabela 3.1– Dados construtivos aproximados da bacia de finos..... | 21 |
| Tabela 3.2 – Dados do sensor ultrassônico.....                    | 22 |
| Tabela 3.3 – Dados do transmissor de pressão. ....                | 23 |



## RESUMO

Neste trabalho, o monitoramento de um sistema de bombeamento de líquidos controlados se dá por meio de uma rede de telemetria que faz a comunicação entre os equipamentos de campo e o controlador lógico programável (CLP) integrado a um sistema supervisório. A planta consiste em três bacias que recebem líquidos controlados provenientes de um processo industrial ou de precipitação pluviométrica que carregam os mesmos, e também de bombas, medidores de nível, medidores de vazão e atuadores elétricos acoplados a válvulas. O sistema de telemetria é utilizado por se tratar de situações onde é difícil o acesso para manutenção da rede de transmissão de dados e comandos. Um sistema de intertravamento de partida da bomba é implementado no CLP para dar uma maior segurança ao processo e evitar erros de operação. Utiliza-se um sistema supervisório para facilitar a operação disponibilizando leituras das variáveis de campo, animações representando o andamento do processo, além de alarmes de falhas de equipamentos e de desconformidades.

Palavras Chaves: CLP, Ladder, Supervisório, Telemetria e bombeamento.

## ABSTRACT

In this work, the monitoring of the controlled liquids pumping system is done by a telemetry network that makes the communication between field equipments and the programmable logical controller (PLC) integrated to a supervisory system. The plant consists in three basins that receive controlled liquids proceeding from an industrial process or pluviometric precipitation which loads them as well, and also pumps, level gauges, flow measurers and “electric actuators” connected to valves. The telemetry system is used by treating situations where the maintenance access to the data and command transmission network is very difficult. A pump start interlocking system is implemented in the PLC to give higher security to the process and to avoid operation errors. A supervisory system is used in order to facilitate the operation, making available field variables readings, animations representing the course of the process, as well as equipment flaws and unconformities alarms.

**Keywords:** PLC, Ladder, Supervisory, Telemetry and Pumping.

## I – INTRODUÇÃO

Nas últimas décadas, tem sido observado um crescimento exponencial da utilização da tecnologia de comunicação e informação em todas as áreas do conhecimento. Também é crescente a busca pelo controle de processos em empresas que manufacturam produtos ou serviços que fazem necessário o controle rígido.

Os sistemas de automação e controle de processos têm um papel fundamental na indústria atual, sendo desenvolvidos de forma a maximizar a produção, minimizar o custo, oferecer uma melhor qualidade do produto final, além de garantir uma maior segurança de todos os envolvidos no processo.

A automação industrial deu um grande salto com o advento dos controladores lógicos programáveis (CLP's) que são equipamentos físicos capazes de executar tarefas de intertravamento, temporização, contagem, operações lógicas e matemáticas, controle em malha aberta ou fechada e supervisão em máquinas e processos de médio e grande porte, controlando plantas industriais completas, modificando significativamente o *layout* das instalações, reduzindo o espaço necessário para as mesmas. Outra característica que o processo ganhou depois dos CLP's foi uma maior flexibilidade, pois ficou mais fácil, rápido e barato uma implementação ou modificação no mesmo e, diminuição das paradas para manutenção.

Como era inexistente uma interface amigável entre o operador de processo e o CLP, tornou-se necessário um sistema para facilitar esse convívio. O *software* supervisorio pode ser definido como “um programa que tem por objetivo ilustrar o comportamento de um processo através de figuras e gráficos, tornando-se assim, uma interface objetiva entre um operador e o processo, desviando dos algoritmos de controle” (JESUS, 2002).

Hoje já é possível a utilização de supervisórios em todos os segmentos do comércio e da indústria, desde sistemas de alarmes, escolas, hospitais, lojas de departamento, climatização de escritórios e laboratórios, pequenas fábricas e em muitos processos industriais.

## 1.1 – Objetivo

### 1.1.1 - Objetivo Geral

Desenvolver um sistema de monitoramento do bombeamento de líquidos controlados entre bacias distantes.

### 1.1.2 - Objetivos Específicos

- Estudo dos diferentes sistemas de monitoramento de bombeamento existentes;
- Levantamento de material sobre os principais sensores e atuadores que atendem a necessidade do projeto;
- Pesquisa das CPU's, cartões de entrada e saída e fontes para o controlador *Compact Logix* ;
- Estudo de técnicas de programação, intertravamento e comunicação com o supervisor utilizando o *software* RSLogix5000;
- Confeção das telas e animações e endereçamento de dados, entradas, saídas e alarmes no sistema supervisorio.

### 1.3 – Metodologia Adotada

Foram necessárias pesquisas e estudos sobre a arquitetura do CLP (*Compact Logix, Rockwell Automation*), uma revisão e aprofundamento no *software* de programação do CLP (RSLogix 5000, *Rockwell Software*), e no *software* de supervisão (RSView32, *Rockwell Software*), para posteriormente aplicá-los na formulação de intertravamentos, comparações, controles, comunicações, aquisição de dados e visualização.

Sendo este um trabalho multidisciplinar, fez-se necessária uma pesquisa sobre perda de carga e rendimento de sistemas de bombeamento e normas que regem o controle de líquidos radioativos para melhor entendimento do processo e detecção de falhas para melhoria do mesmo.

Para a utilização de um sistema de telemetria necessitou-se de um levantamento topográfico para evitar perda de sinal causada por barreiras geográficas e a escolha de um protocolo que trabalhe em uma faixa livre de rádio frequência isenta de interferências de outros rádios como os utilizados na indústria.

## II – REVISÃO BIBLIOGRÁFICA

### 2.1 - Controlador Lógico Programável - CLP

Segundo a norma DIN 19237 o controlador lógico programável pode ser definido como um “equipamento eletrônico programável por técnicos em instrumentação industrial destinado a controlar em tempo real e em ambiente industrial, máquinas ou processos seqüenciais”.

#### 2.1.1 - Arquitetura Básica de um CLP

A grande maioria dos CLP's industriais encontrados no mercado são constituídos dos seguintes elementos:

- **Unidade Central de Processamento - CPU:** Tem como principais componentes o processador e a memória. O primeiro é responsável pelo gerenciamento total do sistema, controla a comunicação com dispositivos externos, faz a aquisição das entradas e geração de saídas controlando os barramentos de endereços, de dados e de controle. A memória tem como principal função armazenar e recuperar as instruções do programa, estados das entradas e saídas, estados internos e dados (GEORGINI, 2000).
- **Módulos de Entrada e Saída:** Os módulos de entrada e saída fazem a conexão física entre a CPU e o mundo externo por meio de circuitos de interfaceamento, além de garantir isolamento e proteção à CPU. Os módulos de entrada recebem os sinais, analógicos ou digitais, e os convertem em níveis adequados para serem processados pela CPU. Os módulos de saídas enviam os sinais, resultantes da lógica de controle ou intervenção do operador, aos dispositivos (PINTO, 2004).

- **Fonte de Alimentação:** Fornece vários níveis de tensão exigidos pelos módulos de entrada e saída e pela CPU. Em algumas aplicações os módulos de entradas e saídas necessitam de uma alimentação externa, seja por necessitar de uma maior potência ou por alimentar circuitos externos (MORAES et al., 2001).

### 2.1.2 - Princípio de Funcionamento

Para melhor compreensão do seu princípio de funcionamento, são apresentados alguns conceitos associados, os quais são de fundamental importância que sejam assimilados. São eles:

- **Variáveis de entrada:** são sinais externos recebidos pelo PLC, que podem ser oriundos de fontes pertencentes ao processo controlado (chaves de acionamentos, sensores, etc.) ou de comandos gerados pelo operador (comando local, emergência, etc.) (PINTO, 2004).
- **Variáveis de saída:** são dispositivos (contatores, válvulas, lâmpadas, etc.) controlados por cada ponto de saída do CLP. Tais pontos podem servir para intervenção direta no processo controlado por acionamento próprio, ou também para sinalização de estados (PINTO, 2004).
- **Programa:** seqüência específica de instruções que irão efetuar as ações de controle desejadas, ativando ou não as memórias internas e pontos de saídas do PLC a partir da monitoração do estado das mesmas memórias internas e/ou dos pontos de entrada do CLP (SILVEIRA et al., 2002).

O princípio fundamental de funcionamento do CLP é a realização cíclica leitura dos estados dos dispositivos de entrada, execução do programa de controle do usuário e atualização dos estados dos dispositivos de saída, realizando assim a lógica de controle.



A programação do CLP é feita por meio de uma ferramenta de programação, que pode ser um Programador manual (Terminal de Programação), ou um computador pessoal (PC) com *software* de programação específico. A linguagem *Ladder* (RLL – *Relay Ladder Logic*, Lógica de Comandos de Relé), muito popular entre os usuários dos antigos sistemas de controle a relés é a mais utilizada (GEORGINI, 2000).

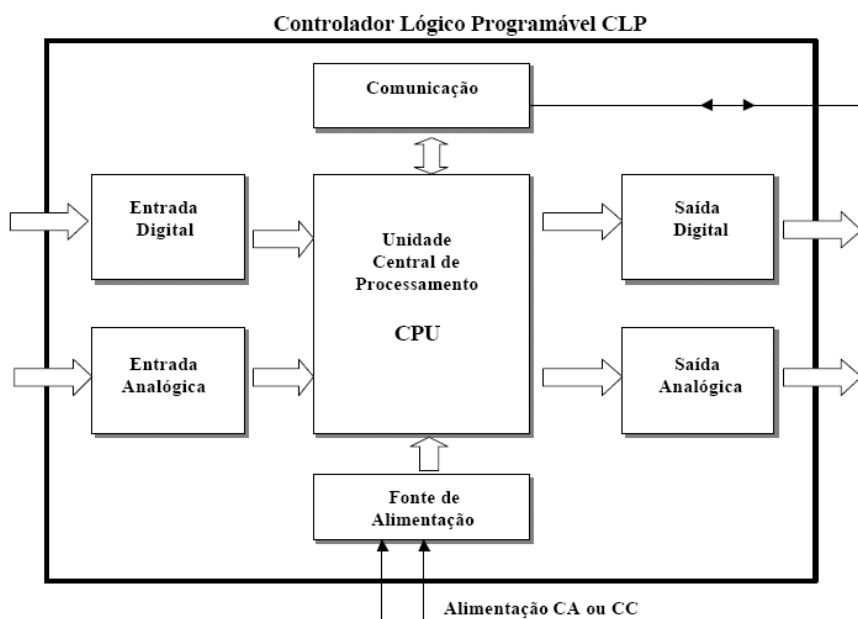


Figura 2.1 – Diagrama de blocos simplificado de um CLP

FONTE: ESTEVES, 2003.

O tempo de varredura (tempo total da execução dessas tarefas) depende da velocidade e características do processador utilizado, do tamanho do programa de controle do usuário, além da quantidade e tipo dos pontos de entrada e saída. Em algumas situações críticas (que envolve segurança, por exemplo) é feita uma interrupção do ciclo de varredura e a rotina é executada imediatamente, não tendo que aguardar o término do mesmo (BARCELLOS, 2005).

### 2.1.3 - Redes de Comunicação

Os sistemas de automação e controle tem se apoiado cada vez mais em redes de comunicação industriais, seja pela crescente complexidade dos processos industriais, seja pela necessidade de integração de todo o conjunto de informações contido na indústria ou pela evolução dos computadores e seus periféricos. Assim, praticamente todos os sistemas implementados atualmente, incluem alguma forma de comunicação de dados, seja local, através de redes industriais, seja remota, implementadas em sistemas supervisórios (JESUS, 2002).

Conforme o número de CLP's e computadores aplicados em automação, os tipos de sistemas existentes são classificados em concentrados e distribuídos. O primeiro pode entender-se como um computador gerenciando um processo constituído por várias unidades remotas (unidades que possuem apenas cartões de entrada/saída e a fonte de alimentação), onde todo o processamento é realizado em apenas uma mala. Para sistemas distribuídos, tanto o gerenciamento da informação como a automação é realizada por máquinas localizadas ao longo da planta. As remotas deixam de ser executoras e passam a participar do processamento (MORAES et al., 2001).

## 2.2 – Linguagem LADDER

“Mesmo tendo sido a primeira linguagem destinada especificamente à programação de CLPs, a Linguagem *Ladder* mantém-se ainda como a mais utilizada, estando presente praticamente em todos os CLPs disponíveis no mercado. Por ser uma linguagem gráfica, baseada em símbolos semelhantes aos encontrados nos esquemas elétricos (contatos e bobinas), as possíveis diferenças existentes entre os fabricantes de CLPs, quanto à representação das instruções, são facilmente assimiladas pelos usuários” (GEORGINI, 2000).

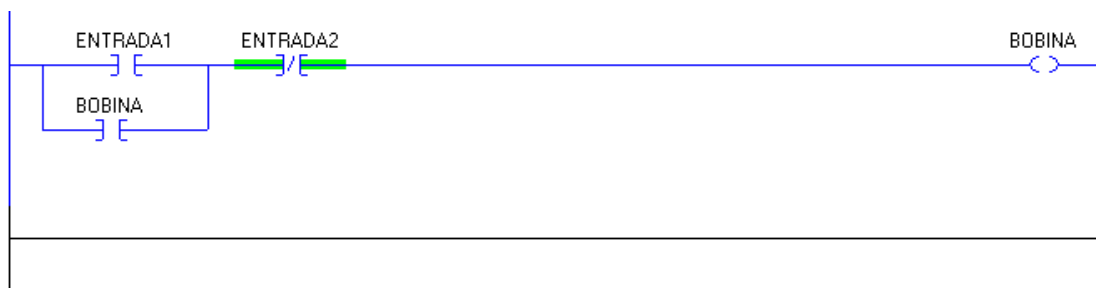


Figura 2.2 - Exemplo de instruções em linguagem LADDER.

O diagrama elétrico parte de duas linhas verticais (Figura 2.2) também chamadas de linhas de alimentação. A programação se dá inserindo elementos controlados (bobina de relé) e um conjunto de condições para o controle desse elemento (lógica de contatos) nas linhas horizontais. Cada contato pode assumir dois estados (energizado ou desenergizado) representando assim uma variável booleana, ou seja, uma variável que assume os valores de verdadeiro ou falso (SOPRANI, 2006).

Abaixo serão estudados conceitos básicos de programação em *ladder* que são necessários para o correto desenvolvimento de programas de aplicação, sendo usados em todos CLPs, independente dos fabricantes e dos recursos na CPU utilizada.

### 2.2.1 - Corrente Lógica Fictícia

Para que uma bobina (ou outro Elemento de Saída - temporizador, contador ou bloco de função, por exemplo) seja acionada (Instrução executada), faz-se necessário “energizá-la logicamente”. Assim, utiliza-se o conceito de Corrente Lógica Fictícia, ou seja, supondo que entre as barras verticais que 'sustentam' toda a Lógica de Controle haja uma diferença de potencial (a barra da esquerda com potencial positivo e a barra da direita com potencial negativo, por exemplo), haverá a circulação de corrente da esquerda para a direita se a Lógica de Controle der condições para tal. A este conceito dá-se o nome de Corrente Lógica Fictícia (GEORGINI, 2000).

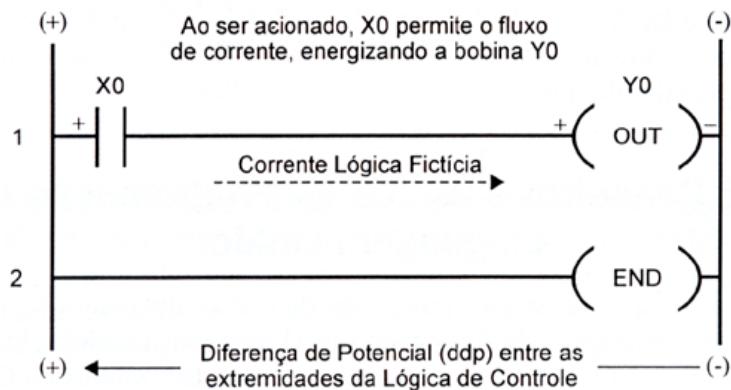


Figura 2.3 - Sentido da corrente fictícia

FONTE: GEORGINI, 2000.

### 2.2.2 - Instruções de BIT

O grupo formado pelos contatos e pelas bobinas recebe o nome de instruções de BIT. Existem dois tipos de contatos, os normalmente abertos (NA) e os normalmente fechados (NF). Nos contatos normalmente abertos existe a continuidade lógica quando o estado lógico da variável associada for 1 (um), e nos contatos normalmente fechados existe continuidade lógica quando o estado da variável associada é 0 (zero). O estado lógico da bobina depende da continuidade lógica da sua linha. Se houver continuidade o estado lógico da bobina será 1 (Bobina energizada), caso contrário será zero (bobina desenergizada) (ROCKEWLL, 2007b).

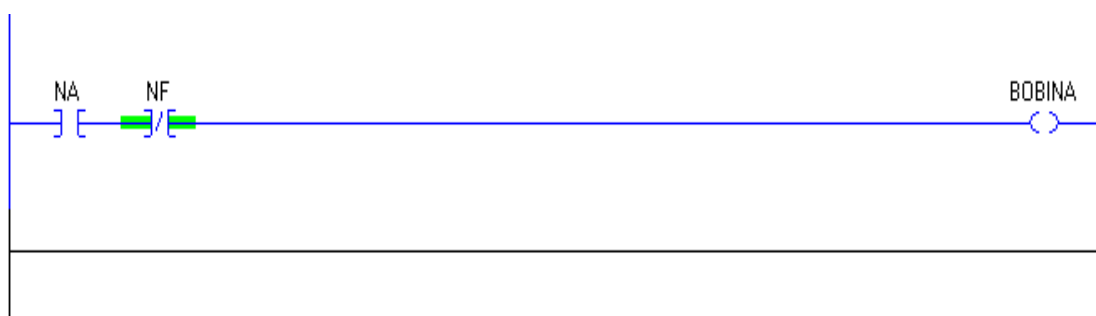


Figura 2.4 - Utilização de instruções de Bit.

### 2.2.3 - Temporizadores

São utilizados, normalmente, para temporizar condições e/ou eventos controlados pelo Programa de Aplicação. Na lógica de controle implementada na Figura 2.5 (RSLogix5000), ao ser atuada a entrada, será energizada o bit Temporizador.EN e iniciará a temporização de “Temporizador”, o qual terá o valor de *Preset* fixo de acordo com a aplicação (no *software RSLogix5000* esse valor é dado em milisegundos). Quando esse valor de preset for atingido o bit “Temporizador.EN” se desenergizará e o bit “Temporizador.DN” permanece ativado até que o temporizador seja desativado (ENTRADA=0). Neste momento reseta o temporizador apagando o valor acumulado (ROCKWELL, 2007b).

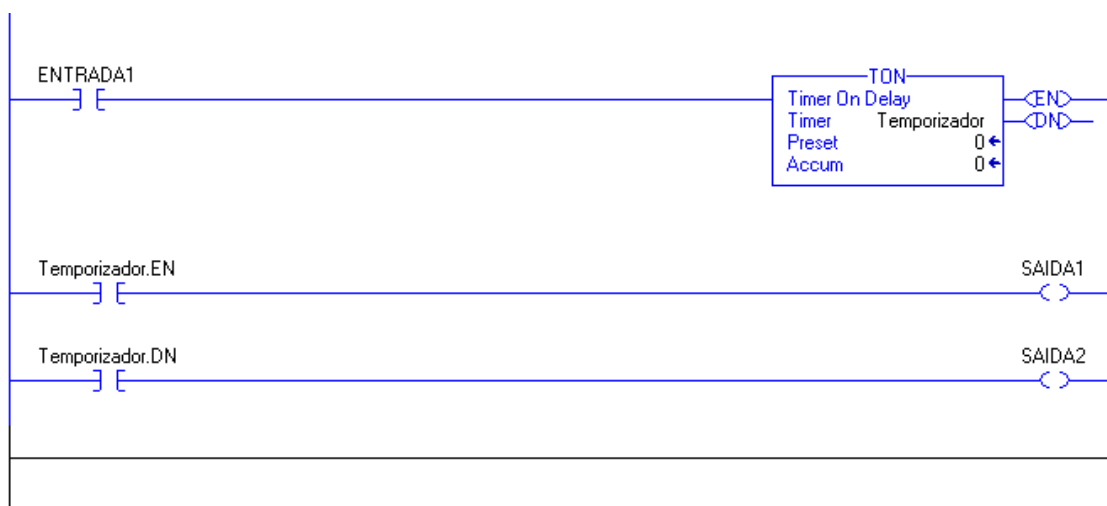


Figura 2.5 – Utilização de temporizadores

### 2.2.4 - Contadores Simples

Normalmente são utilizados para contagem de condições e/ou eventos controlados pelo Programa de aplicação. Toda vez que a linha passa a verdadeiro, o valor acumulado é incrementado de uma unidade. Ao ser atingido o valor de *Preset* é acionado o bit de saída que permanecerá ativado até que o contador seja resetado por meio de mais

uma atuação da entrada. O valor acumulado pode ser resetado também através da instrução RES (*Reset*).

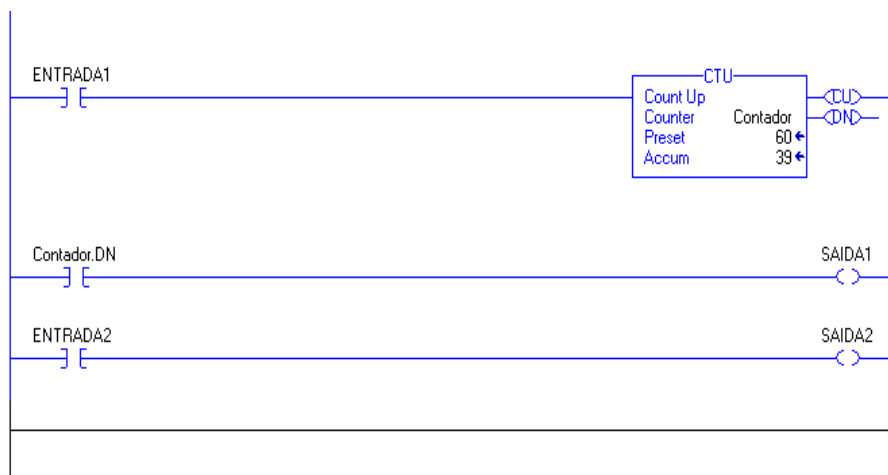


Figura 2.6 – Utilização de Contadores.

### 2.2.5 - Instruções de Movimentação

Quando o bit “ENTRADA” for acionado será transferido o conteúdo do endereço A para o endereço B. Na Figura 2.7 será transferido de 500 para “VARIÁVEL”.

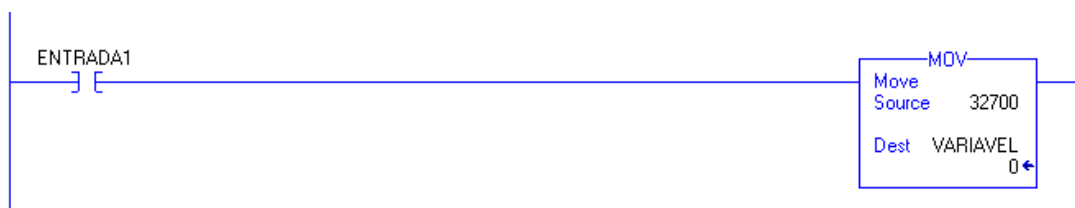


Figura 2.7 – Instrução de Movimentação.

### 2.2.6 - Instruções de Comparação

Quando disponíveis, normalmente são ao menos três instruções de comparação: Igualdade, inferioridade e superioridade. O contato de comparação de igualdade (EQU) será habilitado se o valor contido em A

for igual ao contido em B. O contato de comparação de Inferioridade (LES) será habilitado se o valor contido em A for menor que o contido em B. E, por fim, o contato de comparação superior (GRT) será habilitado se o valor contido em A for maior que o contido em B. Na Figura 2.8 temos uma ilustração das instruções citadas acima utilizando RSLogix5000, onde VARIÁVEL representa A e o valor 30 representa B.

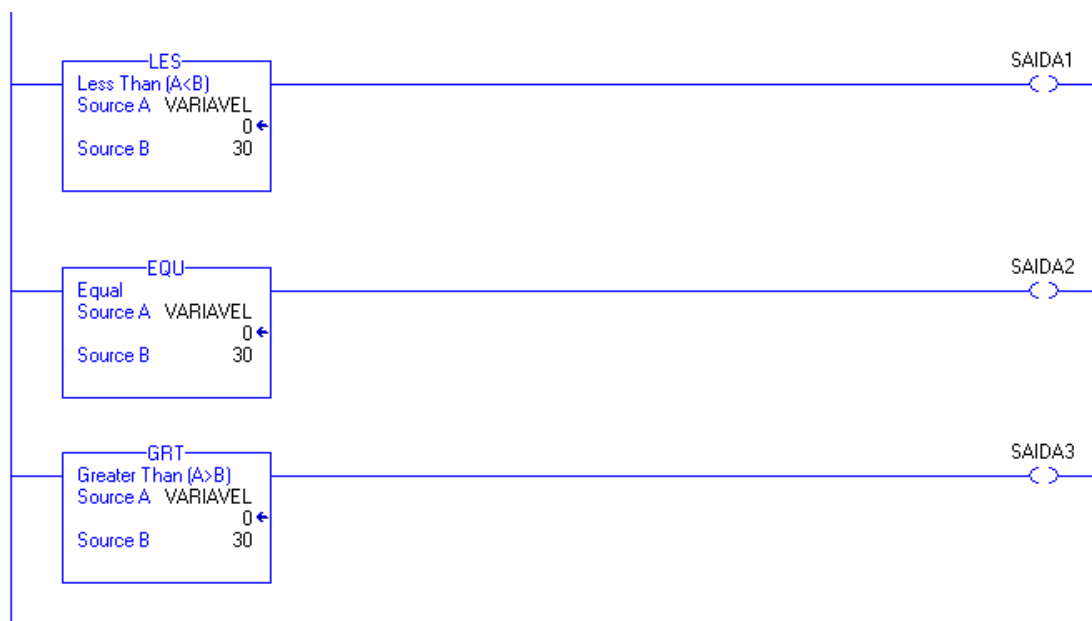


Figura 2.8 – Instruções de Comparação.

### 2.3 – Sistema Supervisório

Os sistemas supervisórios permitem que sejam monitoradas e rastreadas informações de um processo executado em uma planta industrial, através da visualização de variáveis da planta que está sendo automatizada, bem como das ações tomadas pelo sistema de automação. Tais informações são coletadas através de equipamentos de aquisição de dados (sensores, por exemplo) e, em seguida, manipuladas, analisadas, armazenadas e, posteriormente, apresentadas ao usuário. Outra importante característica de sistemas supervisores é o reconhecimento de prováveis falhas em componentes da planta antes que as mesmas ocorram efetivamente e a verificação das condições de alarmes, identificando

quando um valor da variável ultrapassa uma faixa pré-estabelecida. Estes sistemas também são chamados de SCADA (*Supervisory Control and Data Acquisition Systems*).

Segundo OGATA (1993), o software supervisorio é visto como o conjunto de programas gerados e configurado no software básico de supervisão, implementando as estratégias de controle e supervisão com telas gráficas de interfaceamento homem-máquina que facilitam a visualização do contexto atual, a aquisição e tratamento de dados do processo e a gerência de relatório e alarmes. Este software deve ter entrada de dados manual, através de teclado. Os dados serão requisitados através de telas com campos pré-formatados que o operador deverá preencher. Estes dados deverão ser auto-explicativos e possuírem limites para as faixas válidas. A entrada dos dados deve ser realizada por telas individuais, seqüencialmente, com seleção automática da próxima entrada. Após todos os dados de um grupo serem inseridos, esses poderão ser alterados ou adicionados pelo operador, que será o responsável pela validação das alterações.

Os sistemas supervisorios vêm ganhando espaço em praticamente todos os segmentos de controle e monitoração, onde até então seu uso era inviável, seja pelo preço dos sistemas (que eram totalmente importados), seja pela falta de profissionais habilitados em projetá-los e implementá-los. Por isso as aplicações de supervisorios eram restritas às grandes plantas industriais, aos pólos petroquímicos, à geração de energia e ao controle de tráfego (como nos metrô). A sua implementação era dispendiosa e sua manutenção dependia de profissionais treinados no exterior. Mas com a crescente gama de empresas nacionais de alta tecnologia, surgiram no mercado sistemas supervisorios altamente competitivos, com protocolos de comunicação e drives de aquisição de dados desenvolvidos especialmente para CLP's de fabricação nacional (JURIZATO et al., 2003).

A partir da estrutura física de instrumentação é possível definir pontos de conexão entre os dispositivos e o sistema supervisorio, denominados *tags*. Segundo SILVIA et al. (2005) os *tags* "são todas as



variáveis numéricas ou alfanuméricas envolvidas na aplicação, podendo executar funções computacionais (operações matemáticas, lógicas, com vetores ou *strings*, etc.) ou representar pontos de entrada/saída de dados do processo que está sendo controlado. Neste caso, correspondem às variáveis do processo real (ex: temperatura, nível, vazão etc.), se comportando como a ligação entre o controlador e o sistema.”.

Um sistema supervisorio deve apresentar ao operador os dados coletados pelos elementos de campo ligados aos CLP de forma simples e objetiva. O processo é representado por meio de telas animadas, botões, gráficos de tendência e ícones coloridos. A apresentação das telas é modificada de acordo com a variação dos valores dos *tags*.

#### 2.4 – Telemetria

As empresas possuem um custo significativo monitorando equipamentos manualmente. Por se tratar, normalmente, de áreas distantes da planta principal e de difícil acesso, custos adicionais incidem quando as máquinas ou os alarmes falham. Por exemplo, um operador pode dar partida em uma bomba e depois de alguns minutos a mesma parar de funcionar ou o reservatório estando com um nível muito baixo e a bomba começar a succionar lama ou ainda fazer o monitoramento da vazão de saída do fluido, mas não ter certeza que está chegando ao seu destino, a menos que se desloque para o mesmo. Uma solução de telemetria poderia solucionar esses problemas e reduzir substancialmente os custos operacionais.

No caso específico do telecomando, a telemetria pode ser definida como a transferência e utilização de dados provindos de múltiplas máquinas remotas, distribuídas em uma área geográfica de forma pré-determinada, para o seu monitoramento, medição e controle (TELECO, 2007).

Apesar de existirem soluções que utilizam de comunicações com fio, as mesmas não farão parte desse estudo. Por se tratar de áreas distantes e de difícil acesso, necessitaria de um investimento

considerável com posteamento e com cabos de comandos o que tornaria o custo de implantação muito alto. Outro problema é o custo de manutenção desse sistema.

A tecnologia escolhida para a transmissão de comandos e dados foi a de Espalhamento Espectral por Saltos em Freqüência (FHSS), onde “os sinais são digitalizados, codificados e espalhados dentro da faixa de freqüência, passando a utilizar uma banda maior que a ocupada pela informação original” (NIVETEC, 2005).

Essa tecnologia utiliza a largura de banda de forma ineficaz, o que reduz sua velocidade em comparação com outras, mas garante, através da redundância, a segurança e a confiabilidade tão necessárias no meio industrial. A freqüência em que o sinal é transmitido é alterada diversas vezes por segundo seguindo uma seqüência pseudo-aleatória, possibilitando deste modo que as informações sejam enviadas sem sofrer congestionamentos ou interferências entre os rádios. Somente os transceptores que compõem a mesma rede operam em sincronia, sem interferir em outros sistemas de rádio existentes no mesmo ambiente.

### III – DESENVOLVIMENTO

Os estudos foram realizados tomando como base o projeto atual do sistema de drenagem da INB (Indústrias Nucleares do Brasil). Atualmente esse sistema tem sua operação completamente manual, e existe o interesse torná-lo operável remotamente. O atual sistema de drenagem pode ser visualizado na Figura 3.1:

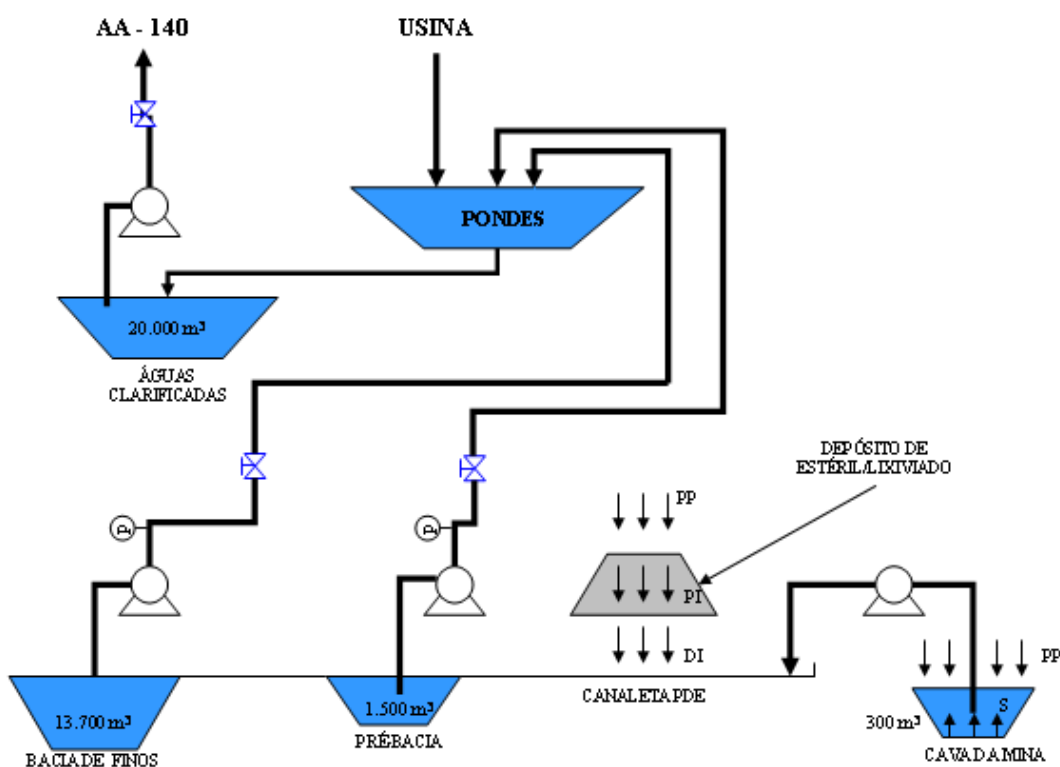


Figura 3.1 – Atual sistema de drenagem da INB.

No depósito de estéril/lixiviado, são armazenados os minérios que não possuem concentração suficiente para seguir o processo (estéril) ou que sofreram o processo de lixiviação (o minério de urânio é “lavado” por uma solução rica de ácido sulfúrico para a obtenção do licor) e, por isso, encontram-se com baixa concentração. Porém, apesar de inviável economicamente, esses minérios possuem concentração suficiente para contaminar precipitações pluviométricas (PP) infiltradas (PI) na pilha.

Essas águas infiltradas são conduzidas através de drenos de fundo (DI) para a canaleta PDE e despejadas na Pré-bacia, de onde são bombeadas para os *ponds* ou transportada por transbordo para a bacia de Sedimentação de Finos. A água armazenada na bacia de sedimentação de finos pode ser bombeada para os *ponds*. Os efluentes da bacia de drenagem de águas clarificadas dos *ponds* da usina (Águas Clarificadas) serão monitorados e, constatando-se que os mesmos se encontram dentro dos limites estabelecidos para a liberação, serão lançados no canal periférico, de onde serão conduzidos para o meio ambiente. Se os efluentes da Bacia de Águas Clarificadas estiverem acima dos limites para a liberação, estes serão bombeados para a área 140 para a reutilização no processo de lixiviação ou para a área 220 para o tratamento dos mesmos.

As águas afluentes e de surgência acumuladas nas cavas da mina são bombeadas para a canaleta PDE e também conduzidas até a Pré-bacia, seguindo o mesmo sistema de drenagem esquematizado na Figura 3.1. No entanto, a transferência da água da cava será efetuada somente após a transferência ou liberação das águas advindas da pilha de estéril/lixiviado. Assim, após serem transferidos estes volumes para os *ponds*, será iniciado o bombeamento das águas das cavas da mina.

Contudo, junto com projeto de monitoramento e controle remoto serão implementadas mudanças, podendo utilizar bacias que não estão inseridas no atual sistema ou retirar alguma existente. Levando em consideração que tratasse de um trabalho acadêmico e o principal objetivo é estudar os instrumentos e as ferramentas utilizados em uma possível solução, e não a logística de armazenamento, será analisada a situação de bombeamento da bacia de finos para os *ponds*. Esse caso foi selecionado por abranger todos os sensores e atuadores previstos para o projeto e representar bem o sistema.

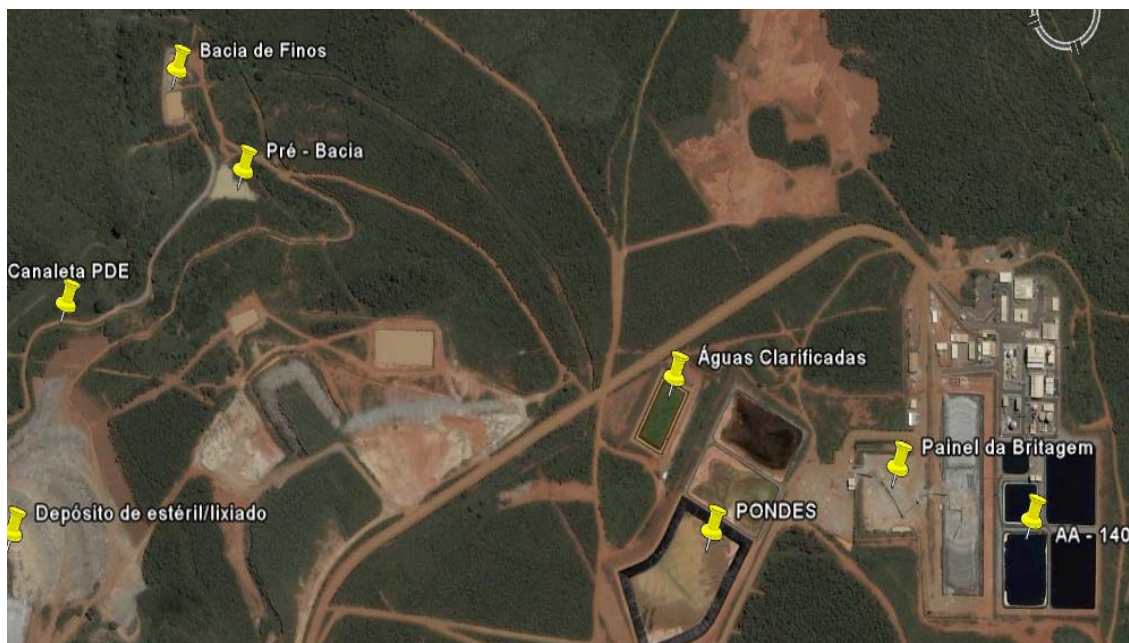


Figura 3.2 – Vista Aérea da planta da INB

FONTE: GOOGLE EARTH, 2006.

### 3.1 – Atuadores e Sensores Utilizados

Os sensores e atuadores possibilitam que o controlador programável possa ler os dados da planta e atuar na mesma, realizando o controle. Neste trabalho os sensores servem para informar ao controlador programável o nível da bacia, o estado da bomba e da válvula, a pressão de saída do fluido e verificar se há rompimento na tubulação através da chave de fluxo na chegada do fluido. E os atuadores servem para que o controlador possa acionar a bomba e abrir ou fechar a válvula. Estes dispositivos serão detalhados a seguir.

#### 3.1.1 – Sensor de Nível

Para a medição de nível da bacia de sedimentação de finos foram analisados vários tipos de sensores/transmissores presentes no mercado, como capacitivos, resistivos, hidrostáticos, bóias magnetoresistivas e ultrassônicos, sendo o ultimo escolhido por não possuir nenhuma parte

em contato com o processo. Os sensores ultrassônicos operam emitindo e recebendo ondas sonoras em alta frequência. A frequência é geralmente em 200kHz, acima da que é percebida pelo ouvido humano.

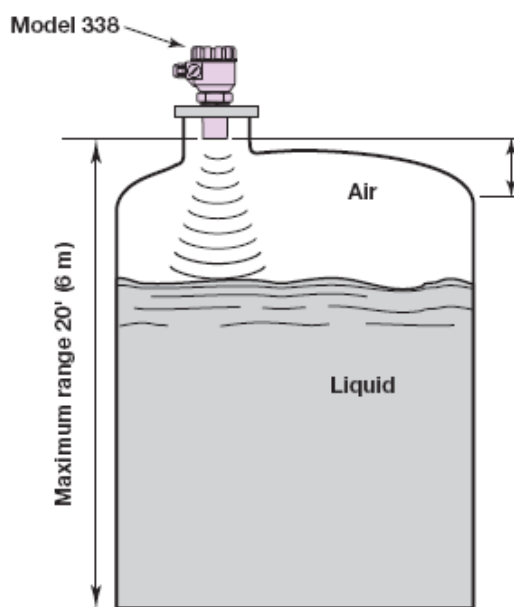


Figura 3. 3 – Sensor/Transmissor Ultrassônico

FONTE: MAGNETROL, 2005.

O sensor ultrassônico estudado é do tipo difuso, isto é, ele emite um sinal ultrassônico e aguarda pelo eco. Quando um objeto entra na zona detectável, ocorre um eco que é percebido pelo sensor. Levando em consideração o tempo gasto pela onda sonora para concluir o trajeto de ida e volta, consegue-se determinar a distância da superfície do fluido ao sensor ( $d_R$ ). Possuindo a distância entre o piso do reservatório/bacia e o sensor ( $H$ ), encontra-se o nível do líquido ( $h$ ) através da fórmula abaixo:

$$h = H - d_R \quad (3.1)$$

Os sensores ultrassônicos possuem uma zona cega localizada na face de detecção. O tamanho da zona cega depende da frequência do transdutor e objetos localizados dentro de um ponto cego podem não ser

confiavelmente detectados. Por este motivo o sensor vai estar a uma distancia  $d$  suficiente para na situação da bacia estar quase cheia, não ocorrer um erro de leitura.

Para encontrar o volume em função do nível ( $h$ ) são necessários dados construtivos do reservatório/bacia como comprimento e largura da base e o fator de declive dos taludes.

Tabela 3.1 – Dados construtivos aproximados da bacia de finos.

| Medida                              | Bacia de finos          |
|-------------------------------------|-------------------------|
| Comprimento da base ( $L_B$ )       | 71,00m                  |
| Largura da base (b)                 | 55,00m                  |
| Comprimento da superfície ( $L_S$ ) | 81,00m                  |
| Largura da superfície (B)           | 65,00m                  |
| Profundidade (P)                    | 3,00m                   |
| Fator de declive (z)                | 1,67                    |
| Volume Total estimado (V)           | 13.709,00m <sup>3</sup> |

FONTE: INB, 2007.

Utilizando as formulas 3.2 e 3.3, encontra-se o fator de declive e o volume estimado.

$$z = \frac{(L_S - L_B)/2}{P} \quad (3.2)$$

$$V_h = L_B b h + (L_B + b) z h^2 + \frac{4}{3} z^2 h^3 \quad (3.3)$$

Substituindo os valores da tabela 3.1 nas formulas supracitadas, tem-se a fórmula 3.4 para a bacia de finos.

$$V_h = 3905h + 210,4h^2 + 3,72h^3 \quad (3.4)$$

Esse sensor é ligado a uma entrada analógica e fornece um sinal de 4 a 20mA, agindo também como um transmissor. Para esse trabalho o regulamos para fornecer mais corrente em função do aumento do tempo de eco. A tabela 3.2 apresenta os principais dados do sensor utilizado neste projeto.

Tabela 3.2 – Dados do sensor ultrassônico.

|                    |          |
|--------------------|----------|
| Tensão de operação | 24V DC   |
| Potência consumida | 60mW     |
| Faixa de medição   | 0 a 6m   |
| Zona cega          | 0 a 0,20 |
| Resolução          | 0,02m    |
| Ângulo do feixe    | 5°       |

FONTE: NIVETEC, 2007.

Instalado o sensor à 1m acima do plano da superfície superior da bacia (evitar a zona cega), espera-se as correntes 6,66 e 14,64mA para a superfície e fundo da bacia de finos , respectivamente.

### 3.1.2 – Transdutor de Pressão

Para a medição da pressão de saída da bomba utiliza-se um transdutor piezo-resistivo. Este foi selecionado por apresentar uma boa linearidade e baixo custo. Não há necessidade de um indicar de pressão, pois a válvula controladora de bomba possui um manômetro em sua estrutura.

O sensor está ligado ao processo através de um diafragma. Este se encontra em contato direto com o meio medido e à medida que a pressão



varia, o diafragma é pressionado proporcionalmente. Um pequeno sensor localizado na parte interna do diafragma converte a variação de pressão em uma variação proporcional de resistência elétrica. Um pequeno circuito eletrônico converte este tipo de informação em um sinal proporcional de 4-20 mA (NIVETEC, 2004).

O transdutor de pressão possui várias aplicações. No sistema apresentado para monografia, a principal função é transmitir para o CLP a pressão de saída do fluido. Outra função do transmissor de pressão é a identificação de possíveis problemas na tubulação como entupimento e rompimento, através de uma variação inesperada da pressão na linha.

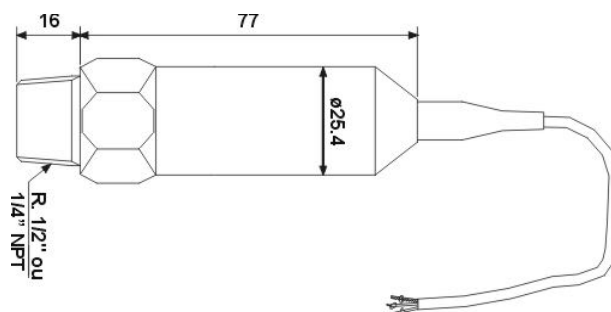


Figura 3.4 - Sensor/Transmissor de pressão piezo-resistivo.

FONTE: NIVETEC, 2004.

Na maioria dos modelos, não há necessidade de efetuar qualquer tipo de ajuste, pois o sinal de saída analógico do tipo 4-20 mA já encontra-se configurado de fábrica.

Segue abaixo a tabela 3.3 que possui os principais dados técnicos do transmissor de pressão utilizado.

Tabela 3.3 – Dados do transmissor de pressão.

|                   |                           |
|-------------------|---------------------------|
| Tipo              | Piezo-resistivo           |
| Faixa de Operação | 0 a 14Kgf/cm <sup>2</sup> |
| Exatidão          | 0.07Kgf/cm <sup>2</sup>   |
| Alimentação       | 24Vcc                     |
| Sinal de Saída    | 4-20mA (a 2 fios)         |

FONTE: NIVETEC, 2004.

### 3.1.3 – Chave de Fluxo

Neste trabalho a principal função da chave de fluxo é de proteção e segurança, detectando se há ou não fluxo na saída da tubulação.

Tendo em vista que o fluido em manejo é controlado, uma chave de fluxo auxiliaria o transmissor de pressão na detecção de rompimentos e possíveis falhas. Por exemplo, se o rompimento ocorrer muito próximo à extremidade oposta ao transmissor de pressão, a variação desta não será o suficiente para caracterizar o rompimento.

A palheta da chave de fluxo permanece inserida na tubulação de modo a se opor ao movimento do fluido. No momento em que o fluxo está presente ou se este aumenta/diminui além do ponto ajustado, a haste (que se encontra presa à palheta) sofre um deslocamento que provoca a atuação do contato elétrico localizado no interior do invólucro. O ajuste do grau de oposição da palheta em relação ao fluxo é determinado por um parafuso que comprime uma mola (NIVETEC, 2007).

Na Figura 3.5 exemplifica uma possível instalação para a chave de fluxo. Alguns cuidados devem ser tomados antes de escolher o local de instalação: escolher um trecho que possua, pelo menos, cinco vezes o diâmetro da tubulação para evitar erro decorrente a turbilhamento, sua instalação não deve ocorrer próximo a válvulas, curvas, reduções ou qualquer outro obstáculo que possa comprometer o desenvolvimento normal do fluxo e o tamanho de palheta a ser utilizado deverá ser escolhido de forma que jamais toque o lado oposto da parede da tubulação. Se isto acontecer, a chave terá seu funcionamento comprometido.

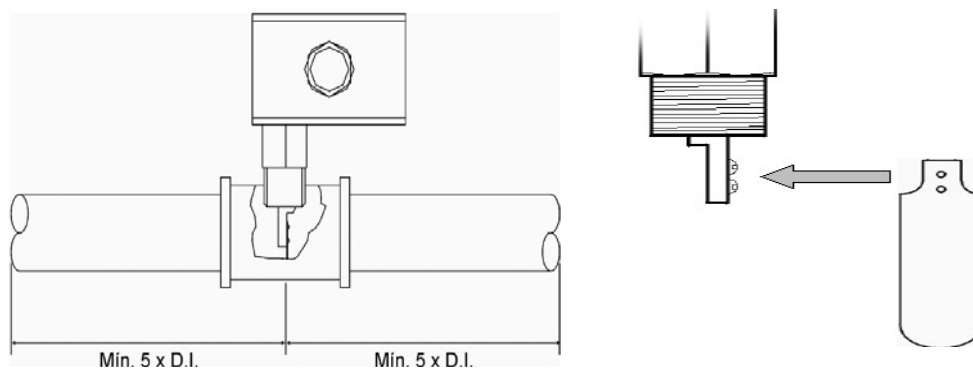


Figura 3.5 – (a) Distancias mínimas recomendadas; (b) Fixação da palheta na haste. FONTE: NIVETEC, 2007.

Neste trabalho ela é configurada como uma chave normalmente aberta (NA), ou seja, fecha o contato quando detecta a presença de fluxo na linha.

#### 3.1.4 – Válvula Controladora de Bomba

As Válvulas Controladoras de Bombas são auto-operadas hidráulicamente através de um atuador tipo diafragma. Ela consiste de uma válvula principal para aplicações diversas, e dispositivos de controle com pilotos que dá a característica da mesma.

O atual procedimento de partida da bomba da bacia de finos consiste em ligar a mesma e através de um manômetro local e anterior a válvula, observar a pressão da linha até que a mesma atinja o valor de trabalho. A partir deste ponto abre-se gradualmente a válvula. Esse procedimento tem o objetivo de evitar a sobrecorrente no motor da bomba causada pelo esforço que o mesmo terá que realizar para vencer o fluxo inverso do fluido presente na tubulação.

A válvula controladora de bomba faz o mesmo papel do operador. Liga-se a válvula e imediatamente após dar-se a partida da bomba, a válvula só abrirá quando a pressão de trabalho (setada anteriormente) da bomba for atingida. A velocidade da abertura da válvula pode ser regulada através da abertura/fechamento de uma válvula de controle de fluxo.

Para desativar a bomba o operador desliga a válvula e a mesma fecha-se com velocidade regulável. Quando atinge o fim de curso da válvula, ela desliga a bomba e passa a funcionar como uma válvula de retenção, mantendo assim a rede estancada. O ajuste da velocidade de fechamento é feito de forma simular a de abertura (Figura 3.6).

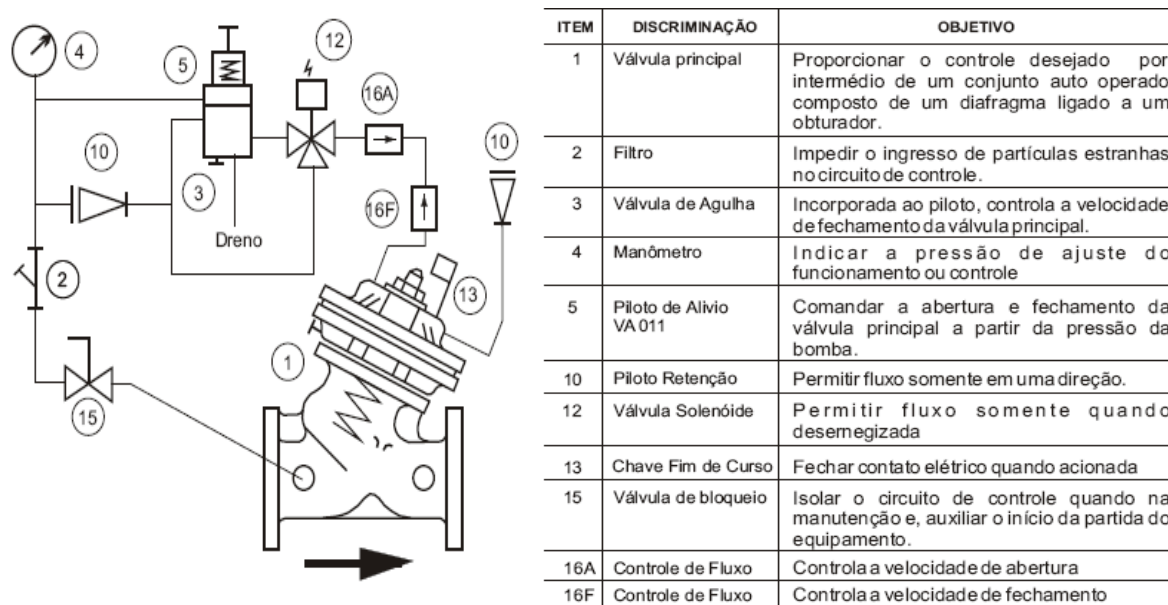


Figura 3.6 – Esquema funcional da Válvula controladora de Bomba

FONTE: VALLOY, 2002.

“A câmara do atuador opera segundo um sistema de controle de três vias. Uma válvula solenóide de três vias aplica alternadamente sobre o diafragma, a pressão a montante para fechar seguramente a válvula ou deixar a pressão da câmara superior escapar para a atmosfera a fim de abrir completamente a válvula. Se um corte do fornecimento de energia elétrica ocorrer, às condições de pressão se invertem, a mola interna fechará a válvula auxiliada pela ação da pressão de jusante. Quando se desenergiza a solenóide, esta pressuriza a câmara de controle do atuador de modo a fechar a válvula enquanto a bomba permanece ligada. Quando a válvula está próxima ao fechamento completo, o indicador de abertura ativará a chave de fim de curso que desligará a bomba diretamente no painel de comando” (SAINT-GOBAIN, 2001).

### 3.2 – Programação do Controlador

O principal objetivo desse tópico é fazer um estudo das características de configuração e programação do controlador programável *CompactLogix 5332E* e do programa *RSLogix 5000*, ambos produzidos pela empresa *Rockwell Automation*.

#### 3.2.1 – Características do Controlador *CompactLogix 5332E*

Na Figura 3.7 pode ser observada a CPU *L32E* da família *CompactLogix 5332E* utilizada nesse projeto, fabricado pela *Rockwell Automation*.

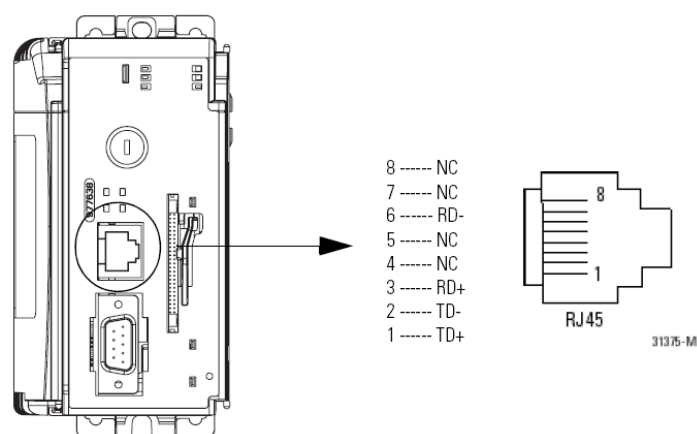


Figura 3.7 – CPU L32E da família *CompactLogix3225EI*

FONTE: ROCKWELL, 2007a.

Este controlador, como a maioria produzida pela *Rockwell*, possui três modos de operação, que podem ser selecionados por uma chave de posição (ROCKWELL, 2007a).

Na posição *RUN* (executar), é permitido apenas *uploads* (descargas), ou seja, o operador pode acompanhar o processo através de um computador contendo um *software* de programação, mas não pode criar, modificar ou excluir tarefas, programas ou rotinas. Também não é

possível fazer alterações em funcionamento de *tags* ou linhas ou mudar a forma de operação através do *software* de programação.

Na posição PROG (programação), as saídas não podem ser acionadas, é possível fazer *uploads* e *downloads* (cargas) de projetos entre o controlador e o computador, é permitido criar modificar e excluir tarefas, programas e rotinas. O controlador não faz varreduras neste modo e também não se consegue mudar o modo de operação através do software de programação.

Na posição REM (remoto) pode-se fazer *uploads* e *downloads* e modificar, através do *software* de programação, entre três modos:

- **Remote Run:** o controlador executa o programa, as saídas podem ser acionadas e permite edições em funcionamento.
- **Remote Prog:** o controlador não executa as tarefas, as saídas não estão habilitadas, é possível fazer *downloads* de projetos, criar, modificar e excluir tarefas, programas e rotinas.
- **Remote teste:** executa tarefas com as saídas desabilitadas e permite edições em funcionamento.

Para a comunicação entre o controlador e o computador é necessário um cabo *ethernet* com conectores RJ45 em suas extremidades. Se a comunicação for feita diretamente entre o controlador e o computador há a necessidade de usar um cabo *crossover* (cabo com a seqüência de fios invertida próprio para ligações diretas), porém se for utilizado um *switch* para essa conexão pode-se usar um cabo *ethernet* convencional (ROCKWELL, 2007a).

A fonte fornece tensão contínua para o barramento, sendo responsável pelo fornecimento de tensão estabilizada para a CPU e para os módulos. A alimentação da fonte 1769-PA4 é feita em tensão alternada com valores entre 120 e 240Vca (com chave seletora) e trabalha com uma faixa de *voltagem* entre 85 a 132Vca ou 170 a 265Vca e em frequência entre 47 e 63Hz. A fonte disponibiliza 4A a 5Vcc O aterramento é conectado em um conector específico (ROCKWELL, 2000b).

O módulo de entrada digital 1769-IA16 (ROCKWELL, 2000a) possui 16 canais de entradas 120Vca em 60Hz, Cada canal tem até 12mA, sendo que a somatória não ultrapasse 115mA.

O módulo de saída digital 1769-OA16 (ROCKWELL, 2001) possui 16 canais de saídas a *triac* que fornece 100 a 240Vca. Cada módulo tem 2 grupos isolados com 8 saídas cada, permitindo assim, conciliar saídas de 120 e 240Vca em um mesmo módulo.

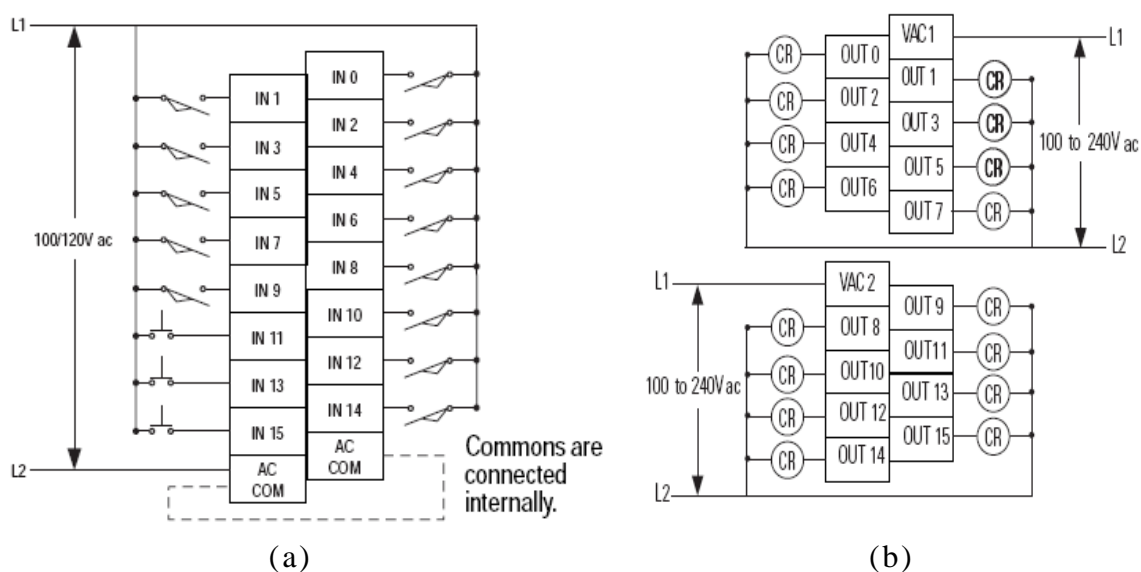


Figura 3.8 - Esquema elétrico do módulo de entrada discreto (a) e, do módulo de saída discreto (b). FONTE: ROCKWELL, 2000a e 2001.

O módulo de entrada analógica 1769-IF8 (ROCKWELL, 2005a) possui 8 canais de entradas ( $\pm 10V_{cc}$ , 0 a 10Vcc, 0 a 5Vcc, 1 a 5Vcc ou 0 a 20mA, 4 a 20mA). Os canais trabalharam na faixa de 4 a 20mA, por ser a faixa de corrente que os sensores estudados fornecem. A corrente total de cada módulo não pode ultrapassar 120mA.

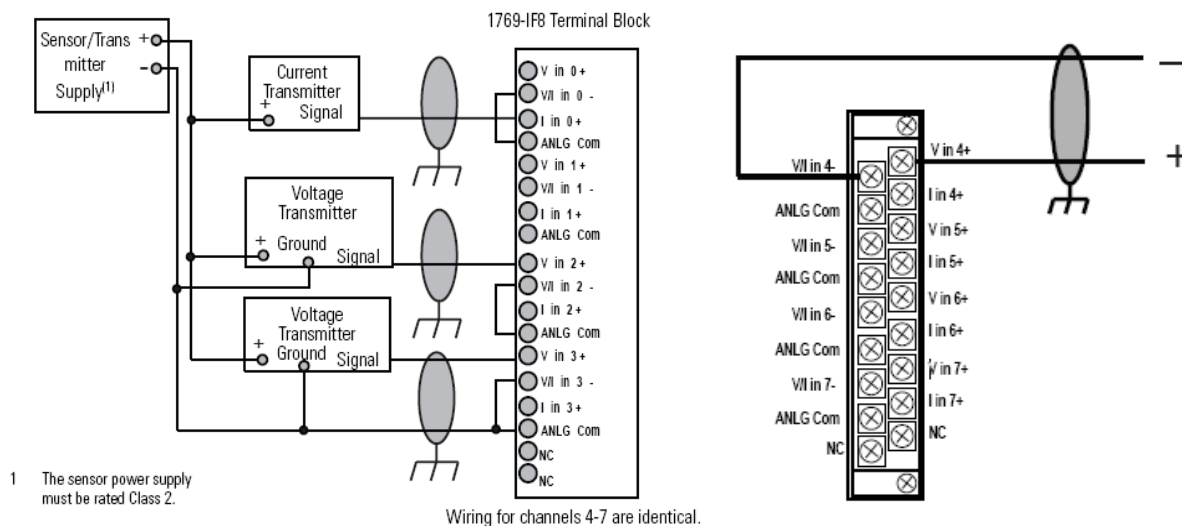


Figura 3.9 – Esquema elétrico do módulo de entrada analógico  
 FONTE: ROCKWELL, 2005a.

### 3.2.2 - Configuração do Controlador *CompactLogix 3225E*

Este tópico tem por objetivo mostrar os passos básicos para a configuração do controlador usando o *software* de programação RSLogix5000.

Para a criação de um projeto, entrar no programa e selecione a opção “*File>> New...*” e aparecerá uma janela para escolher um processador e para nomear o seu projeto, Figura3.10 (a).

Para configuração dos módulos de entrada e saída, basta clicar com o botão direito do mouse no campo *CompactBus Local* e selecionar *new module*, Figura 3.10 (b). O primeiro *slot* (encaixe) após o processador é a bateria, foi configurado para módulo de entrada analógica 1769-IF8, o segundo para o módulo de entrada discreta 1769-IA16 e o terceiro para o módulo de saída discreto 1769-OA16.



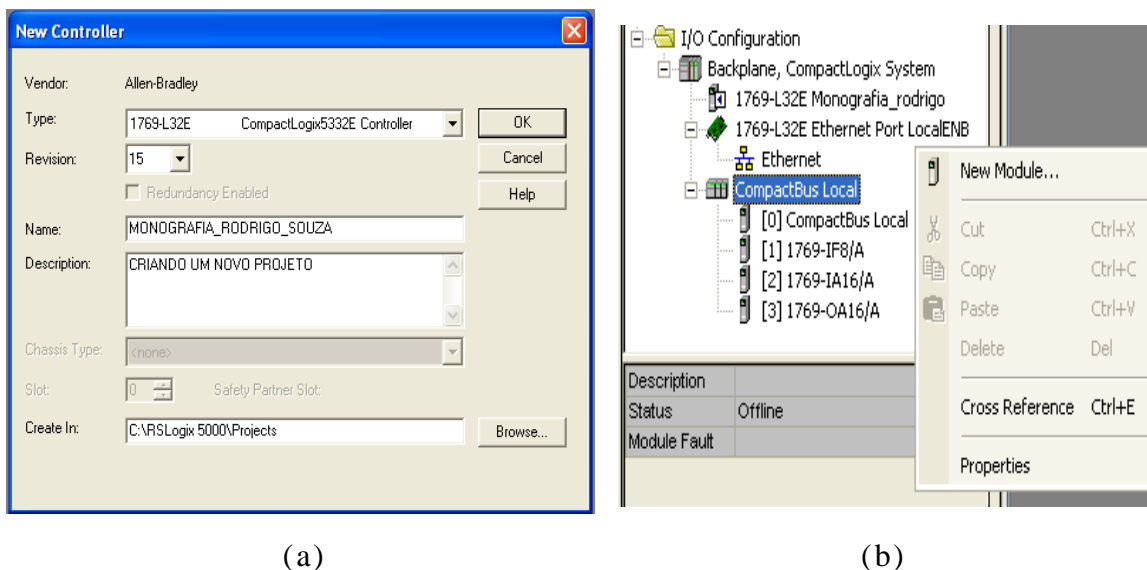


Figura 3.10 - Criando um novo projeto (a). Configurando módulos de entradas e saídas (b).

Podemos interpretar um *tag* como sendo “uma área definida da memória do controlador em que os dados são armazenados. É o mecanismo básico para alocar memória, fazer referência aos dados a partir da lógica e monitorá-los” (BARCELLOS, 2005).

Para criar um *tag*, clique com o botão direito do mouse em *Program tags* localizado dentro do subdiretório *Main Program*. Será necessário fornecer o nome, uma descrição e o tipo de dados. Os tipos de dados mais usados, como a faixa de operação de cada tipo, podem ser observados na Figura 3.11.

Quando se está criando um *tag* existe o campo *alias for* que serve para referenciar a outro *tag*. É dessa forma que se faz um *tag* referenciar o valor de uma saída do controlador ou ler o valor de uma entrada.

| Data type | Bits     |    |    |   |   |   |   |
|-----------|----------|----|----|---|---|---|---|
|           | 31       | 16 | 15 | 8 | 7 | 1 | 0   |
| BOOL      | not used |    |    |   |   |   | 0 or 1  |
| SINT      | not used |    |    |   |   |   | -128 to +127  |
| INT       | not used |    |    |   |   |   | -32,768 to +32767   |
| DINT      |          |    |    |   |   |   | -2,147,483,648 to +2,147,483,647  |
| REAL      |          |    |    |   |   |   | -3.40282347E <sup>38</sup> to -1.17549435E <sup>-38</sup> (negative values)<br>0<br>1.17549435E <sup>-38</sup> to 3.40282347E <sup>38</sup> (positive values) |

Figura 3.11 – Principais tipos de dados.

FONTE: ROCKWELL, 2007c.

### 3.2.3 - Programação do Controlador *CompactLogix 3225E*

Neste trabalho serão utilizados dois tipos de entradas (analógica e digital) e um tipo de saída (digital). As variáveis de entrada podem assumir os valores 0 e 1 e as variáveis analógicas podem assumir valores de -32768 a 32767, pois a resolução do módulo analógico 1769-IF8 é de 16 bits ( $2^{16} = 65536$ ). Esta variação é proporcional aos valores de fundo de escala do módulo (-10.5 a 10.5V e -21 a 21mA), sendo assim a faixa de operação dos sensores será de 6250 à 31200 (número adimensional, pois é a representação dos bits na base decimal). O programa completo encontra-se no Anexo B, sendo apenas referenciado nos tópicos abaixo.

#### 3.2.3.1 - Monitoramento do Nível

Para se fazer o monitoramento do nível da bacia de finos, teremos que resolver os seguintes problemas: definir a programação na faixa de 4 a 20mA, transformar o valor fornecido pelo sensor à entrada analógica em valor de nível (h), obter o volume da bacia em função de h e configurar o alarme de nível baixo e de nível alto.

Primeiramente, transforma-se os valores de corrente ‘I’ (6,66 e 14,64mA) em valores lidos no *tag* de entrada NIVEL\_FINOS ‘N’ (10392 e 22838), usando a fórmula 5.5.

$$N = I(mA) * \frac{32767}{21mA} \quad (3.5)$$

Se ocorrer uma corrente *Under Range* (abaixo da escala) ou *Over Range* (acima da escala), os valores 10400 e 22800 respectivamente, serão movidos para a variável Nivel\_finos\_AUX e sinalizados no *supervisórios* através dos *tags* INSTRUMENTOS.11 e INSTRUMENTOS.10 (ver linhas 0 e 1 do Anexo 2). Se não acontecer nenhuma dessas situações, o valor de NIVEL\_FINOS será movido para Nivel\_finos\_AUX (linha 2).

O valor de Nivel\_finos\_AUX será utilizado pela instrução CPT para calculo da Altura\_finos (linha 3), como mostra na fórmula 3.6:

$$Altura\_finos = 4 - [(Nivel\_finos\_AUX - 6250) * (4/16588)] \quad (3.6)$$

Onde 4 é a altura em metros do piso da bacia até o sensor e o número 6250 representa 4mA, pois o sinal fornecido pelo sensor só é linear na faixa de 4 a 20mA. Com o valor de Altura\_finos (em metros) obtem-se o volume atual da bacia de finos (linha 4), como visto na formula abaixo:

$$V = [3905Altura\_finos + 210.4(Altura\_finos^2) + 3.72(Altura\_finos^3)] * (100/13709.04)$$

O valor de Nivel\_finos\_superv (V) será transmitido para o *superviório*. Através das instruções GRT e LES o controlador compara o valor atual do nível (Nivel\_finos\_superv) com a estabelecida como nível alto e nível baixo, respectivamente, e informa ao *supervisório* através dos *tags* INSTRUMENTOS.12 E INSTRUMENTOS.13 (linhas 5 e 6) se existe alarme de nível alto ou alarme de nível baixo.

### 3.2.3.2 - Acionamento da Válvula Controladora de Bomba

Para o acionamento da válvula controladora deve-se ter o cuidado de verificar se a bacia não está com nível baixo, evitando que a bomba trabalhe a vazio (puxando ar). Outra preocupação é com as falhas de acionamento e de desligamento da bomba.

Nas três primeiras linhas (7, 8 e 9) são mostrados os estados da válvula, que pode ser *funcionando*, *defeito* e *pronto para ligar*. O estado *funcionando* é caracterizado quando a válvula aciona a bomba. O estado *defeito* é energizado quando ocorre algum problema na partida ou desligamento da bomba e o *pronto para ligar* ocorre quando a válvula não tem nenhum impedimento para ser ligada.

Como foi dito no tópico 3.1.4 a válvula aciona a bomba quando é ligada e quando desligada, fecha com velocidade regulável até acionar o fim de curso que desligará a bomba. Para ter um maior controle do processo esses comandos são levados da válvula para o controlador, ou seja, em vez da válvula controlar diretamente a bomba, estes comandos serão enviados para a entrada do controlador para detecção de falhas. Nas linhas 10 à 14 são testadas duas possíveis falhas da válvula. Quando o operador der o comando para ligar a válvula, a mesma tem 5 segundos para acionar a bomba, caso contrário disparará o alarme de falha na partida. A segunda falha ocorre quando o operador desliga a válvula e esta demora mais de 30 segundos (pode variar de acordo com a velocidade de fechamento *setado*) para desligar a bomba. As duas falhas são informadas ao *supervisório* através dos *tags* 'FINOS\_FALHAEQUIP.0' e 'FINOS\_FALHAEQUIP.2'. Na linha 14 o operador zera as falhas e rearma a válvula para uma nova partida.

A linha 15 mostra o acionamento da válvula, onde ela só vai ser energizada se o operador der o comando LIGA, o alarme de nível baixo estiver desativado e a válvula estiver pronta para ser ligada. Como o comando só ficará energizado durante o curto intervalo de tempo em que o operador selecioná-lo, utiliza-se a técnica de colocar a saída do controlador que liga a válvula em paralelo com o comando. Nessa técnica

a saída só se desenergizará quando o comando *desliga* for selecionado, quando o nível baixo for atingido ou se ocorrer uma falha.

#### 3.2.3.3 - Bomba da Bacia de Finos

A bomba também pode assumir os três estados funcionando, defeito no equipamento e pronto para partir. Se a entrada de estado da bomba estiver energizada, a bomba estará funcionando (linha 17). Se a bomba não entrar em funcionamento após 10 segundos de seu acionamento (linhas 20 e 21), existe um defeito no equipamento (contator, CCM, gavetas, motor etc). Agora se a bomba não estiver funcionando e sem nenhum defeito ela está pronta para partir (linha 19).

A bomba poderá partir se a válvula estiver ligada, se não houver algum defeito no equipamento, e se a botoeira de emergência não estiver pressionada. Neste caso não é necessário fazer a realimentação, pois a válvula mantém o contato fechado enquanto estiver ligada.

#### 3.2.3.4 - Horômetro

O horômetro é um excelente instrumento de monitoramento para bombas de uso pouco freqüente. Para fazer um plano de manutenção preventiva (dar a manutenção antes que ele apresente problema) em qualquer equipamento é indispensável saber quantas horas ele trabalhou depois da última manutenção. Utiliza-se um temporizador de retenção (difere do convencional por não zerar o acumulado com a desenergização da entrada) para contar os segundos, este por sua vez alimenta o contador toda vez que o acumulado chega a 60000 e se auto zera (linhas 23 e 26). Quando o acumulado do contador de minutos chega a 60 ele energiza a entrada do contador de horas e se auto zera (linhas 24 e 27). O contador de horas, por sua vez, fornece o valor do seu acumulado para a variável auxiliar 'HORAS\_BOMBA\_FINOS' que será utilizado no supervisório (linhas 25 e 29). O operador autorizado

pode zerar o horômetro caso tenha que trocar a bomba ou dar manutenção na mesma.

### 3.2.3.5 – Transdutor de Pressão

Para fazer o monitoramento da pressão da bacia de finos, teremos que resolver os seguintes problemas: definir a programação na faixa de 4 a 20mA, transformar o valor fornecido pelo transdutor à entrada analógica em valor de pressão (Kgf/cm<sup>2</sup>) e configurar o alarme de pressão baixa e de pressão alta.

O transmissor de pressão trabalha com toda a faixa de operação limitada de fábrica, ou seja, ele fornecerá 4mA para a pressão mínima e 20mA para a máxima pressão (diferente do ultrasônico que neste projeto trabalha entre 6,66 e 14,64mA). Então os valores transferidos para o *tag* *Pressao\_finos\_AUX* quando ocorrer *Under Range* e *Over Range* serão de 6250 e 31200 respectivamente (linhas 30 e 31) e serão sinalizados no supervisórios através dos *tags* INSTRUMENTOS.1 e INSTRUMENTOS.0. Se não ocorrer nenhuma das duas situações, o valor de PRESSAO\_FINOS será movido para *Pressao\_finos\_AUX* (linha 31).

O valor de *Pressao\_finos\_AUX* será utilizado pela instrução CPT para cálculo da *Pressao\_finos\_superv* (linha 33), como mostra na fórmula 5.7:

$$Pressao\_finos\_superv = (Pressao\_finos\_AUX - 6250) * (14 / 249500) \quad (5.7)$$

O valor de *Pressao\_finos\_superv* será transmitido para o superviório. Através das instruções GRT e LES o controlador compara o valor atual da pressão (*Pressao\_finos\_superv*) com a estabelecida como pressão alta e pressão baixa, respectivamente, e informa ao supervisório através dos *tags* INSTRUMENTOS.2 e INSTRUMENTOS.3 (Linhas 34 e 35).

### 3.2.3.6 - Chave de Fluxo

A principal função da chave de fluxo é verificar se o fluido está chegando à saída da tubulação. Então precisa-se estabelecer um tempo máximo em que o fluido demore a circular pela tubulação (1 minuto por exemplo) e indicá-lo no *preset* do temporizador “Vazamento” (linha 36). Então se a chave de fluxo não energizar depois de 1 minuto que a bomba estiver em funcionamento, será disparado o alarme de vazamento na tubulação (linha 37).

Existem situações em que o alarme pode acusar vazamento sem que o mesmo tenha ocorrido. Isto ocorre quando não existe fluido na tubulação de recalque, seja por manutenção na linha, por manobra de operação ou por ser o primeiro teste. Levando em consideração que a velocidade mínima do fluido seja 2m/s e que a tubulação que une a bacia de finos aos *pondes* tenha o comprimento de 1000m, demoraria aproximadamente 8 minutos para o sensor acusar fluxo quando a tubulação estiver completamente vazia.

Na linha 38 a entrada da chave de fluido energiza uma variável auxiliar que tem por função transferir o estado do sensor para o supervisor. Não é recomendado utilizar a própria entrada como inferência no sistema supervisor, pois em uma situação em que a mesma tenha que ser substituída ou transferida para outro cartão, além da modificação no programa do RSLogix5000 o programador terá de realizar modificações no RSView32.

### 3.2.3.7 - Telemetria

Quando o módulo de rádio frequência percebe que perdeu o link (não “enxerga” mais as outras unidades), ele ‘abre’ o relé de estado (normalmente fechado) indicando a falha. Na última linha utiliza-se o *tag* auxiliar FALTA\_DE\_LINK\_AUX para transmitir essa falha ao sistema supervisor.

### 3.3 – Sistema de Telemetria

O sistema de telemetria apresenta basicamente os seguintes componentes: módulo de rádio frequência, módulos de entrada e saída, fonte de alimentação e antena.

No trabalho utilizar-se apenas 2 transceptores, um localizado próximo a bacia de finos e outro painel da britagem de onde o operador vai controlar todo o bombeamento.

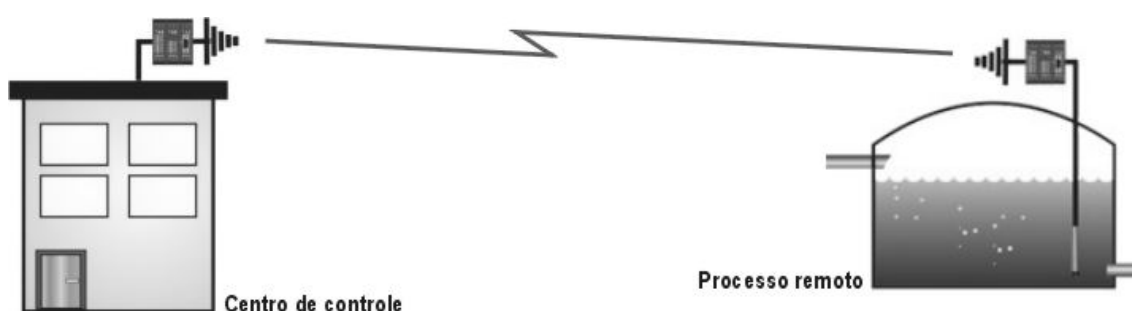


Figura 3.12 - Exemplo de uma rede de telemetria com 2 transceptores

FONTE: NIVETEC, 2005.

Cada módulo de rádio frequência, bem como cada módulo de entrada e saída, possui uma identificação única na rede, inexistindo assim a possibilidade de uma informação ir para outro transceptor (se houvesse mais de 2), ler uma entrada ou acionar uma saída errada. Esta identificação é gravada em uma memória não volátil existente em cada módulo, permitindo que a identificação aconteça imediatamente após cada um deles ser energizado. Normalmente essa identificação vem gravada de fábrica, ou seja, é informado ao fabricante à logística da rede de telemetria e o mesmo personaliza o sistema para melhor atender. Porém, o fabricante também disponibiliza ao cliente um software de configuração de rede, para o mesmo implementar melhorias e modificações no seu sistema.

Cada módulo de rádio frequência possui um relé normalmente fechado de falta de link para sinalização no CLP, ou seja, enquanto os



dois transceptores estiverem se “enxergando” a entrada do CLP estará acionada.

A fonte a alimentação fornece 24Vcc as módulos de entrada e saída e ao módulo de rádio freqüência. Cada módulo de entrada possui 2 entradas digitais 110Vca, 1 analógica de 4-20mA e uma porta de comunicação serial RS-485. Os módulos de saídas possuem 2 relés 3A/250Vca, 1 saída analógica 4-20mA e uma porta de comunicação serial RS-485.

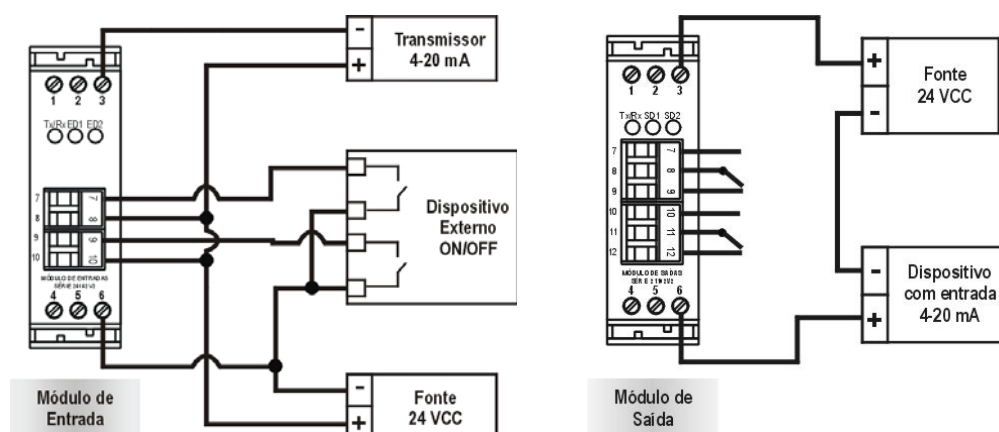


Figura 3.13 – Interligação dos módulos de entrada/saída do sistema de telemetria com dispositivos externos. FONTE: NIVETEC, 2005.

Para cada módulo de entrada que a unidade ‘finos’ possua, a unidade ‘painel de britagem’ deve ter um de saída, e vice-versa. Por exemplo, o sinal de pressão tem que passar por um módulo de entrada na unidade ‘finos’, transferido pelo módulo de rádio freqüência, chega ao módulo de saída da unidade ‘painel de britagem’ e depois vai para um cartão de entrada analógica do CLP. Infelizmente não encontramos no mercado um sistema em que o CLP comunique diretamente com o módulo de rádio freqüência. Isso economizaria além dos módulos de entrada e saída do sistema de telemetria, os cartões de entrada e saída do CLP.

### 3.4 – Sistema Supervisório

Neste tópico serão vistas funções e ferramentas básicas do *software* RSView32, produzido pela *Rockwell Automation*. Dentre outras, terá uma atenção especial a criação de variáveis e o desenvolvimento de telas no supervisório.

Para um novo projeto, deve-se fazer duas configurações iniciais. A primeira é especificação do canal (*drive*) e o protocolo (PCP/IP neste projeto) de comunicação utilizada que pode ser implementada com a ferramenta *channel*. A segunda é a definição da comunicação entre as estações e o sistema de supervisão que será configurada em *node*.

#### 3.4.1 – *Tags* e o *Tag data base*

Com o *tag data base*, define-se qual dado necessita ser monitorado. Cada entrada no *tag data base* recebe o nome do *tag*, que é um nome lógico para uma variável de um instrumento ou memória (ROCKWELL, 2005b). O valor atual de um *tag* é obtido do instrumento e armazenado em um local na memória, para poder ser utilizado em diversas partes do RSView32.

Existe dois tipos de fonte de dados no RSView32, a *device* e a *memory*. A primeira recebe os dados de uma fonte externa ao RSView32, como o RSLogix por exemplo, já a segunda é obtida por tabelas internas ao RSView32.

O RSView32 usa os seguintes tipos de *tags*:

- **Analogic:** Representa variáveis de estados, tais como nível e pressão. Possui um faixa (*range*) de atuação;
- **Digital:** Representa instrumentos que podem assumir valores 0 e 1, como chaves e contatores.
- **String:** Representa instrumentos que usam texto, com códigos de barra, etc;
- **System:** São criadas pelo RSView32 quando rodamos um projeto e não podem ser modificadas pelo usuário.

Para configurar um *tag* analógico é necessário dar um nome para esse *tag*, a restrição de acesso (se houver), uma descrição, o valor mínimo e o valor máximo, a escala de variação, a unidade (opcional), o tipo (inteiro, ponto flutuante, etc), o nome do nó que ela pertence e o endereço dela neste nó (*tag* do RSLogix por exemplo). A Figura 3.14 mostra a configuração do *tag* de nível da bacia.

The image shows a 'Tag Configuration' dialog box with the following fields and options:

- Tag Section:**
  - Name:
  - Type:  (dropdown)
  - Security:  (dropdown)
  - Description:
  - Minimum:
  - Scale:
  - Units:
  - Maximum:
  - Offset:
  - Data Type:  (dropdown)
- Data Source Section:**
  - Type:  Device  Memory
  - Node Name:  (with a browse button '...')
  - Address:
- Buttons:** Close, Prev, Next, New, Help, and an unchecked checkbox for Alarm...

Figura 3.14 – Configuração de *tags* analógicas.

No caso de um *tag* digital o procedimento é parecido: Declara o nome do *tag*, seleciona a restrição, faz uma descrição do *tag*, faz uma descrição de até 20 caracteres do estado *off* e do estado *on*, o nó que ela pertence e o endereço dela no nó. A Figura 3.15 mostra-se a configuração da chave de fluxo.

Podem-se associar alarmes tanto nos *tags* analógicos ou nos digitais. Ao executar o programa, o RSView32 varre os valores dos *tags* no *tag database* e os compara com o limite estabelecido para os *tags*. Se um valor ultrapassar o limite, um alarme será disparado.

Outras características da função *alarme* serão visto em tópicos posteriores.

Tag

Name:  Type:  Security:

Description:

Off Label:  On Label:

Data Source

Type:  Device  Memory

Node Name:  ...

Address:

Close

Prev

Next

New

Help

Alarm...

Figura 3.15 – Configuração de *tags* digitais.

### 3.4.2 – Criando *Displays* Gráficos

Uma tela deve representar o processo como se o operador estivesse realmente em todos os setores da planta. Além do processo, as telas têm que mostrar todos os dados de instrumentação (nível, pressão, estados, etc) e possibilitar que o operador mande comandos para o CLP (liga, desliga, *set* de limites, etc).

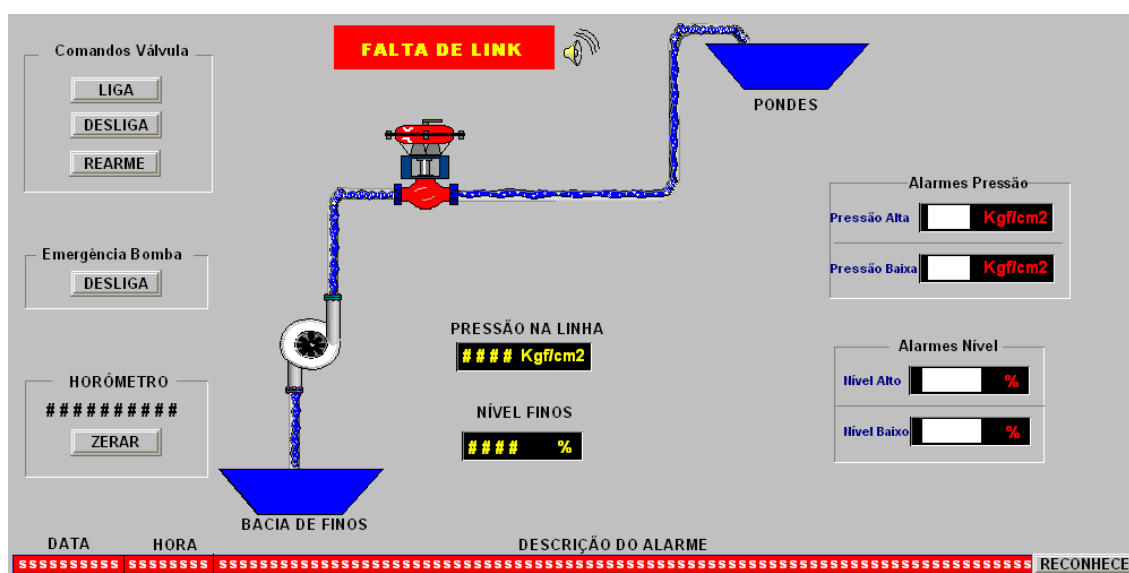


Figura 3.16 – Tela geral do sistema supervisório.

Na Figura 3.16, pode-se ver o processo estudado, onde foi considerado apenas o trecho do bombeamento da bacia de finos para os pondeis da usina.

O editor gráfico é bastante simples e familiar, pois possui muitas funções que lembram as ferramentas de desenho do *Microsoft Office*, como retângulos, polígonos, rotações, retas, círculos, elipses, arcos, etc. O RSVIEW32 também permite a importação de figuras e desenhos feitos em outros programas como Autocad e Coreldraw.

#### 3.4.2.1 - Criando Botões

A principal função dos botões é a de controlar ações que podem ser executadas pelo operador no modo *runtime*. Para adicionar um botão, primeiramente selecione a opção *button*, depois escolha estilo de botão (3D, recessed, etc). O próximo passo é definir qual ação o botão submete o *tag* que ele refere. As ações *set* e *reset* forçam o valor do *tag* para 1 e para 0, respectivamente. A ação *toggle* implica uma mudança no valor de uma variável digital, ou seja, se o valor inicial é zero ele se tornará 1 e vice versa. Os botões configurados para *momentary on* e *momentary off*, assumem estes valores apenas enquanto o botão estiver pressionado pelo operador. O botão com a ação *command* executa um comando pré-definido pelo programador como, abrir outra tela, incrementar ou decrementar um determinado valor de variável, etc. No trabalho todos os botões são do tipo *set/reset*.

O último passo é configurar a aparência do botão quando selecionado ou não selecionado, ou seja, texto, cor de fonte cor de preenchimento, etc.

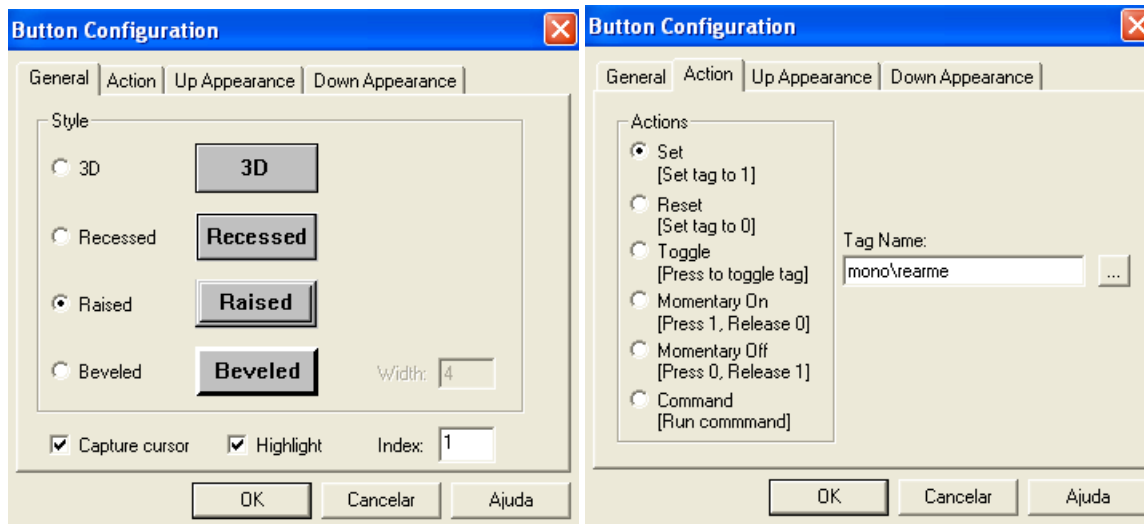


Figura 3.17 – Configuração de botões

#### 3.4.2.2 - *Display* Numérico

Essa ferramenta é normalmente usada para mostrar ao operador os valores das variáveis digitais, analógicas ou valores baseados em uma expressão. Essa ferramenta foi aplicada na leitura das variáveis pressão, nível e no totalizador do horômetro.

Para adicionar *display* numérico, basta clicar na ferramenta *numeric display*. O próximo passo é fornecer o nome do *tag* que será mostrada ou a expressão que será processada e depois expõe o resultado final. Outros parâmetros a serem configurados são: Quantidade de dígitos, casas depois da vírgula (ponto flutuante), formato (inteiro, binário, ponto flutuante), aproximação (para divisões inexatas, raízes, etc) e disposição do texto (esquerda, centralizado, direita).

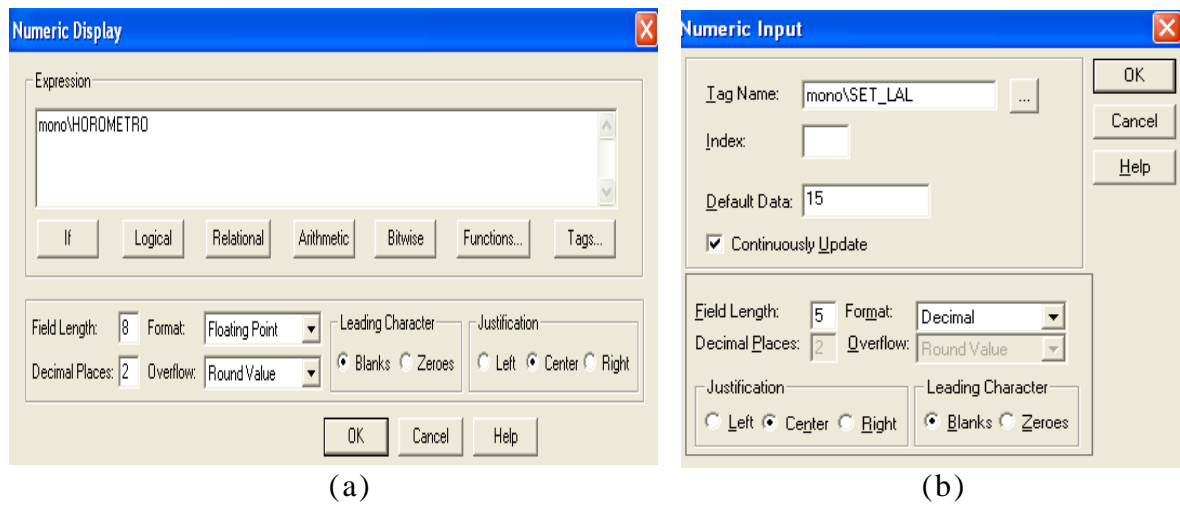


Figura 3. 18 – (a) Configuração de displays numéricos (b) configuração das entradas numéricas.

#### 3.4.2.3 - *String Display*

A ferramenta *string display* é usada para criar campos onde o operador pode acompanhar os valores atuais de *tags* tipo *string*. Apesar de não ter sido cadastrado nenhum *tag* tipo *string*, o mesmo foi gerado automaticamente pelo sistema supervisorio quando associamos alarmes a *tags* analógicos ou digitais. Neste trabalho serão utilizados três tipos de *tags* tipo *system*:

- ***AlarmMostRecentDate***: Mostra a data que ocorreu o último alarme.
- ***AlarmMostRecentTime***: Mostra a hora que ocorreu o último alarme.
- ***AlarmMostRecentTagDesc***: Mostra a descrição do último alarme detectado pelo sistema.

Quando o CLP envia a informação de um alarme para o sistema supervisorio, o mesmo “imprime” no *String Display* (localizado na parte inferior da Figura 3.16 ) a data, a hora e a descrição do alarme e aciona um alarme sonoro na estação de trabalho. Para “silenciar” o alarme o

operador deverá apertar o botão *reconhece* localizado na parte inferior da tela geral do sistema de bombeamento.

Na configuração de um *string display* além de fornecer o nome do *tag* e selecionar a disposição do texto, têm se que delimitar o mesmo informando a largura e a altura (Figura 3.19).

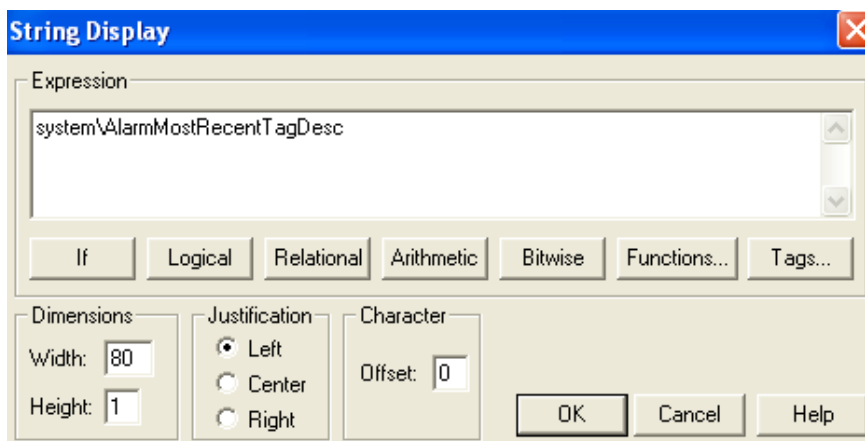


Figura 3. 19 – Configuração de um *string display*.

#### 3.4.2.4 - Entrada Numérica

Use-se a ferramenta entrada numérica (*numeric input*) quando se deseja que o operador entre com dados no supervisão, que são transferidos para variáveis digitais e analógicas. Essa ferramenta é bastante usada para setar níveis de alarmes ou regular velocidade e posição de atuadores com entrada analógica. No *set* de pressão alta, pressão baixa, nível alto e nível baixo foram utilizadas essa ferramenta.

Para criar uma entrada numérica basta selecionar a ferramenta *numeric input* e esperar abrir a janela de configuração. O primeiro campo é destinado ao nome do *tag* para onde irá a informação da entrada, o segundo campo não precisa ser configurado, pois não será utilizada a função *index*. No campo *default data* indica-se o valor inicial da entrada. Marcar o campo *Continuously Update*, para que o campo só possa ser mudado se o operador clicar nele. Os outros campos devem ser configurados conforme explicado na ferramenta anterior.



### 3.4.3 – Criando Animações

A animação é uma ferramenta bastante eficiente para dar uma maior vida a tela e torná-la o mais próximo possível da aplicação real, facilitando a operação. Praticamente todos os objetos gráficos são passíveis de conter uma ou mais animações, porém devemos avaliar a necessidade e o benefício que a animação trará para a operação a fim de evitar animações desnecessárias de efeito apenas estético que serve apenas para sobrecarregar o sistema.

Abaixo serão explicadas apenas as ferramentas de animações que foram utilizadas neste trabalho, tendo em vista o grande número de opções que o programa disponibiliza.

#### 3.4.3.1 - Visibilidade

Com a ferramenta *visibility*, um objeto fica visível ou invisível baseado em valores de *tags* ou do resultado final de uma expressão lógica. Na Figura 3.16, pode ser observado um líquido azul dentro da tubulação. O mesmo só será visível se a chave de fluxo informar ao supervisor que está havendo fluxo na mesma. Outra aplicação dessa ferramenta pode ser observada no bloco vermelho de alarme de falta de link, que fica visível se o sistema de telemetria estiver com problemas de comunicação. Foi dada uma atenção extra à este alarme por forçar o operador a se deslocar para o local enquanto é solucionado o problema.

Clicando com o botão direito do mouse em cima do objeto, que será animado, aparece uma lista de opções de animação, dentre elas a *visibility*. Têm-se duas opções de animação. Na primeira (*invisibly*), se o *tag* ou expressão for verdadeiro (diferente de 0), o objeto fica invisível. Na opção *visibly*, acontece exatamente o contrário, ou seja, quando o *tag* ou expressão for *verdadeiro* o objeto fica visível.

### 3.4.3.2 - Cor

Com essa ferramenta o objeto muda de cor baseado no valor de um *tag* ou de uma expressão. A cor pode ser uma cor sólida ou piscando (*blinking*) a uma taxa selecionável (*blink rate*). Neste trabalho essa ferramenta foi aplicada na bomba e na válvula. Nos dois objetos a programação é feita de forma idêntica (mudando apenas o nome dos *tags*) podendo eles assumir três cores, a verde para o estado “funcionando”, a vermelha para “defeito no equipamento” e a branca para o estado “pronto para partir”. A configuração utilizada para a animação da bomba pode ser vista na Figura 3.20.

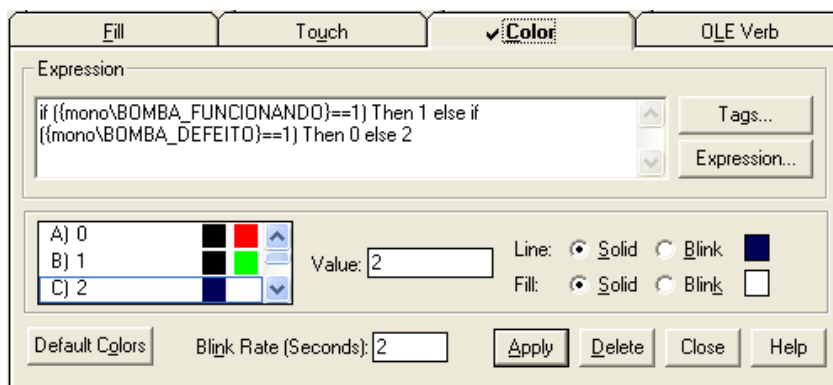


Figura 3. 20 – Configuração da animação da bomba

### 3.4.3.3 - Preenchimento

Na animação de preenchimento (*fill animation*), o nível do objeto é aumentado ou diminuído proporcionalmente ao valor de um *tag*. Essa proporcionalidade se dá informando o valor máximo e mínimo do *tag*. Por exemplo, se no nível da bacia de finos estiver em 50%, o trapézio que o representa estará com 50% preenchido, ou seja, até a metade da altura estará azul e a partir daí estará branco.

Para configurar essa ferramenta além de informar o nome do *tag* e os valores de máximo e mínimo, deve selecionar de que forma será feito o preenchimento (esquerda para direita, direita para esquerda, de cima para baixo ou de baixo para cima).

## IV – CONCLUSÃO

O trabalho desenvolvido obteve bons resultados, uma vez que se conseguiu realizar um profundo estudo sobre três partes envolvidas no projeto: instrumentos propostos, o CLP e o sistema supervisório.

O meio industrial exigiu cuidados e atenções especiais que talvez poderiam ser ignorados em plantas laboratoriais. Por exemplo, o tempo de trabalho de uma bomba para a realização de uma manutenção preventiva, aumentando assim a vida útil do equipamento ou o estudo dos processos manuais e entrevistas com os operadores para saber realmente quais suas dificuldades operacionais e suas sugestões de melhoria, pois apesar do pouco conhecimento técnico a sua experiência e vivência com o processo revela detalhes que passam despercebidos aos olhos da engenharia.

O sistema de telemetria apesar de extremamente simples, mostrou-se muito eficiente e prático economizando tempo e custo de instalação, pois além de não existir posteamento entre os locais, a distância percorrida pela afiação seria extremamente longa e custosa.

É prevista também uma grande economia na mão de obra de operação (operador e transporte do mesmo) e uma redução de pagamentos de horas extras destinadas a manutenção corretiva. Mas o principal ganho é será aumentar segurança do processo com o monitoramento contínuo das variáveis relevantes, reduzindo drasticamente a possibilidade de vazamentos para o meio ambiente.

Procurando dar continuidade ao desenvolvimento de um sistema de monitoramento remoto, sugere-se:

- Configuração de um algoritmo de registro de valores dos principais *tags* do sistema;
- Confecção de uma tela de tendências para facilitar análise das variáveis analógicas.

## V - REFERÊNCIAS BIBLIOGRÁFICAS

BARCELLOS, F. P. **Sistema Didático para Controle de Nível e Temperatura**. 2005. 95p. Monografia (Trabalho final de curso de Automação Industrial), Centro Federal de Educação Tecnológica de Pelotas, Pelotas, 2005.

ESTEVES, M., RODRIGUEZ, J. A. V., MACIEL, M..**Sistema de Intertravamento de Segurança**. 2003.106p. Monografia (Trabalho final de curso de Engenharia Elétrica-Eletrônica), Associação Educacional Bom Bosco, Resende, 2003.

GEORGINI, M.. **Automação Aplicada – Descrição e Implementação de Sistemas Seqüenciais com PLC's**. 3ª Edição. São Paulo: Editora Érica, 2000. 228p.

GOOGLE EARTH. 2006. Disponível em: <<http://earth.google.com/intl/pt/>>. Acesso em: 01/07/2007.

INB - Indústrias Nucleares do Brasil. Catálogo de dados construtivos da Bacia de Finos, 2007.

JESUS, Carlos Eduardo. *Amperes Automation*. Cubatão, 2002. Disponível em: <[www.amperesautomation.hpg.ig.com.br/index0.html](http://www.amperesautomation.hpg.ig.com.br/index0.html)>. Acesso em: 20/05/2007.

JURIZATO, L. A.; PEREIRA P. S. R.. **Sistemas Supervisórios**. Nova Odessa, 2003. 10p.

MAGNETROL. *Echotel® Model 338 Non-Contact Ultrasonic Two-Wire Transmitter for Level, Volume, or Open Channel Flow*, 2005. Disponível em: <[www.magnetrol.com/v2/pdf/MII/51-139.pdf](http://www.magnetrol.com/v2/pdf/MII/51-139.pdf)>. Acesso em: 01/08/2007.

MORAES, C. C.de; CASTRUCCI, P.de L.. **Engenharia de Automação Industrial**. 1ª Edição. Rio de Janeiro: Editora LTC, 2001.295p.

NATALE, F.. **Automação Industrial**. 4ª Edição. São Paulo: Editora Érica, 2001. 237p.

NIVETEC. **Transmissor de pressão série 790 - Manual de instruções**. 2004. Disponível em: <[www.nivetec.com.br/a\\_pdfmanual/Pressao/Transmissor%20de%20pressao%20uso%20geral\\_790\\_manual.pdf](http://www.nivetec.com.br/a_pdfmanual/Pressao/Transmissor%20de%20pressao%20uso%20geral_790_manual.pdf)>. Acesso em: 01/08/2007.

NIVETEC. **Chave de fluxo tipo palheta série 100-v01-at - Manual de instruções**. 2007. Disponível em: <[www.nivetec.com.br/a\\_pdfmanual/Vazao/Chave%20de%20fluxo%20palheta\\_100-V01-AT\\_manual.pdf](http://www.nivetec.com.br/a_pdfmanual/Vazao/Chave%20de%20fluxo%20palheta_100-V01-AT_manual.pdf)>. Acesso em: 01/08/2007.

NIVETEC. **Sistema de telemetria via rádio para monitoramento remoto de processos série 500-v02 - Manual de instalação e programação**. 2005. Disponível em: <[www.nivetec.com.br/a\\_pdfmanual/Telemetria/sistema%20wireless\\_500-V02\\_manual.pdf](http://www.nivetec.com.br/a_pdfmanual/Telemetria/sistema%20wireless_500-V02_manual.pdf)>. Acesso em :01/08/2007.

OGATA, Katsuhiko. **Engenharia de controle moderno**. Rio de Janeiro: Prentice-Hall do Brasil, 1993.

PINTO, J. R. C.. **Técnicas de Automação**. Lisboa: Editora ETEP, 2004. 348p.

ROCKWELL AUTOMATION. **Compact™ 120V ac Input Module**. 2000a. DISPONÍVEL EM: <[http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in061\\_-en-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in061_-en-p.pdf)>. Acesso em: 01/08/2007.

ROCKWELL AUTOMATION. *Compact™ 16-Point 100 to 240V ac Solid State, Output Module*. 2001. Disponível em :

<[http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in061\\_-en-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in061_-en-p.pdf)>. Acesso em: 01/08/2007.

ROCKWELL AUTOMATION. *Compact™ Expansion Power Supplies*.

2000b Disponível em : <[http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in028\\_-en-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in028_-en-p.pdf)>.

Acesso em: 01/08/2007.

ROCKWELL AUTOMATION. *Compact I/O 1769-IF8 Analog Input Module*. 2005a. Disponível em :

<[http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in067\\_-en-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in067_-en-p.pdf)>.

Acesso em: 01/08/2007.

ROCKWELL AUTOMATION. *Installation Instructions 1769-L32E, 1769-L35E CompactLogix Controller*. 2007a. Disponível em:

<[http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in020\\_-en-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/in/1769-in020_-en-p.pdf)>. Acesso em: 01/08/2007.

ROCKWELL AUTOMATION. *Logix5000 Controllers Common Procedures*. 2007b. Disponível em: <[www.literature.rockwellautomation.com/idc/groups/literature/documents/pm/1756-pm001\\_-en-p.pdf](http://www.literature.rockwellautomation.com/idc/groups/literature/documents/pm/1756-pm001_-en-p.pdf)>.

Acesso em: 01/08/2007.

ROCKWELL AUTOMATION. *Logix5000 Controllers General Instructions*. 2007c. Disponível em:

<[http://literature.rockwellautomation.com/idc/groups/literature/documents/rm/1756-rm003\\_-en-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/rm/1756-rm003_-en-p.pdf)>. Acesso em: 01/08/2007.

ROCKWELL AUTOMATION. *RSView32 User's Guide*. 2005b.

Disponível em: <[http://literature.rockwellautomation.com/idc/groups/literature/documents/um/vw32-um001\\_-en-e.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/um/vw32-um001_-en-e.pdf)>. Acesso em:

01/08/2007.

SAINT-GOBAIN. **Válvulas de controle (série E2001)**. 2001. Disponível em: <[www.saint-gobain-canalizacao.com.br/lpor/valvula/vse2001-port06.asp](http://www.saint-gobain-canalizacao.com.br/lpor/valvula/vse2001-port06.asp)> Acesso em: 01/08/2007.

SILVEIRA, P. R. da; SANTOS, W. E.. **Automação e Controle Discreto**. 4ª Edição. São Paulo: Editora Érica, 2002. 229p.

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

SOPRANI, L. R.. **Controle Automático de Processos Industriais Utilizando Controlador Lógico Programável (PLC) e Sistema Supervisório para Controle de Temperatura de Líquidos em Tubos**. 2006. 60p. Monografia (Trabalho final de curso de Engenharia de Controle e Automação), Escola de Minas, Universidade Federal de Ouro Preto, Ouro Preto, 2006.

SCHUTTENBERG, R. M.; LEÃO, J. A.; COSTA E.. **PLC S7 200**. PUC Minas, Belo Horizonte, 2004. 41p.

TELECO. **Telemetria: Descrição do setor**. 2007. Disponível em: <[www.teleco.com.br/tutoriais/tutorialmtelemetria/pagina\\_2.asp](http://www.teleco.com.br/tutoriais/tutorialmtelemetria/pagina_2.asp)>. Acesso em: 01/08/2007

VALLOY. **Válvula controladora de bomba e retenção com solenóide**. 2002. Disponível em: <[www.valloy.com.br/PDF/YCR3D01.pdf](http://www.valloy.com.br/PDF/YCR3D01.pdf)>. Acesso em: 01/08/2007

# **ANEXOS**



## ANEXO A – TAGS UTILIZADAS NO RSLOGIX5000

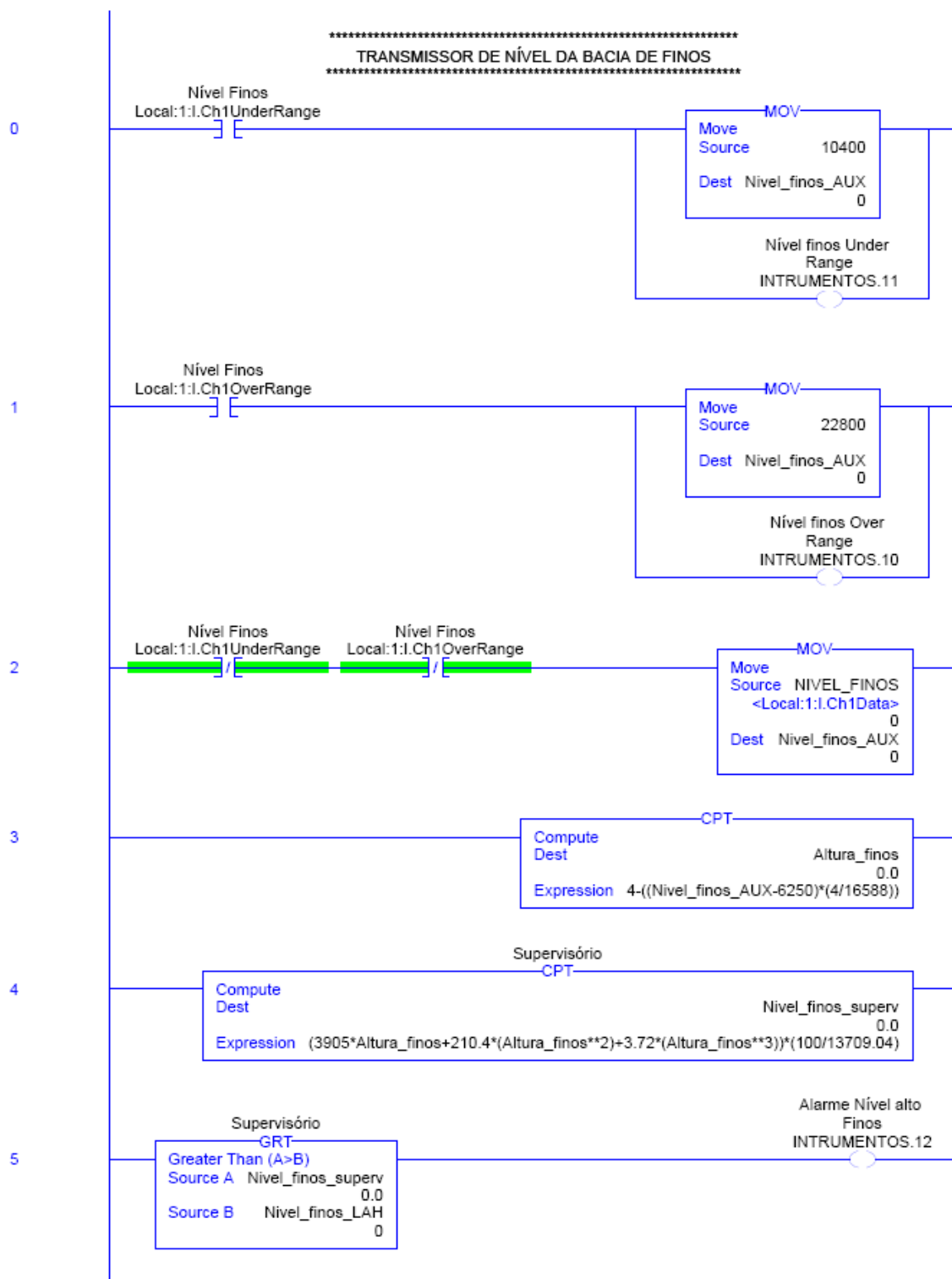
| Name  | Value   | Data Type | Scope              |
|---|---|-----------|--------------------|
| <b>Altura_finos</b><br><i>Altura_finos - MainProgram/MainRoutine - *3(CPT), 4(CPT)</i>  | 0.0   | REAL      | MainProgram        |
| <b>BA_FINOS_comandos</b><br><i>BA_FINOS_comandos.0 - MainProgram/MainRoutine - *16(OTE), 22(XIC)</i>  | 0   | INT       | Monografia_rodrigo |
| <b>BA_FINOS_estados</b><br><i>BA_FINOS_estados.0 - MainProgram/MainRoutine - *17(OTE), 12(XIC), 23(XIC), 35(XIC)</i><br><i>BA_FINOS_estados.1 - MainProgram/MainRoutine - *18(OTE)</i><br><i>BA_FINOS_estados.2 - MainProgram/MainRoutine - *19(OTE), 22(XIC)</i> | 0   | INT       | Monografia_rodrigo |
| <b>BA_FINOS_HORA</b><br><i>BA_FINOS_HORA - MainProgram/MainRoutine - *25(CTU), *28(RES)</i><br><i>BA_FINOS_HORA.ACC - MainProgram/MainRoutine - 29(MOV)</i><br><i>BA_FINOS_HORA.DN - MainProgram/MainRoutine - 28(XIC)</i>  |   | COUNTER   | MainProgram        |
| <b>BA_FINOS_LIGA</b><br><b>PARTIDA NA BOMBA DA BACIA DE FINOS</b><br>AliasFor:<br>Base Tag:<br><i>BA_FINOS_LIGA - MainProgram/MainRoutine - *22(OTE), 20(XIC)</i>   | 0<br>Local:3:O.Data.0(C)<br>Local:3:O.Data.0(C) | BOOL      | MainProgram        |
| <b>BA_FINOS_MINUTOS</b><br><i>BA_FINOS_MINUTOS - MainProgram/MainRoutine - *24(CTU), *27(RES)</i><br><i>BA_FINOS_MINUTOS.DN - MainProgram/MainRoutine - 25(XIC), 27(XIC)</i>  |   | COUNTER   | MainProgram        |
| <b>BA_FINOS_SEGUNDOS</b><br><i>BA_FINOS_SEGUNDOS - MainProgram/MainRoutine - *23(RTO), *26(RES)</i><br><i>BA_FINOS_SEGUNDOS.DN - MainProgram/MainRoutine - 23(XIO), 24(XIC), 26(XIC)</i>  |   | TIMER     | MainProgram        |
| <b>CHFluxo_finos</b><br>AliasFor:<br>Base Tag:<br><i>CHFluxo_finos - MainProgram/MainRoutine - 36(XIO), 38(XIC)</i>   | 0<br>Local:2:I.Data.2(C)<br>Local:2:I.Data.2(C) | BOOL      | MainProgram        |
| <b>ChFluxo_finos_superv</b><br><i>ChFluxo_finos_superv - MainProgram/MainRoutine - *38(OTE)</i>   | 0   | BOOL      | MainProgram        |
| <b>Emergencia_SUPERVISORIO</b><br><i>Emergencia_SUPERVISORIO - MainProgram/MainRoutine - 22(XIO)</i>  | 0   | BOOL      | MainProgram        |
| <b>FALHA_BA</b><br><i>FALHA_BA - MainProgram/MainRoutine - *20(TON)</i><br><i>FALHA_BA.DN - MainProgram/MainRoutine - 21(XIC)</i>   |   | TIMER     | MainProgram        |
| <b>FALHA_VA</b><br><b>PARTIDA</b><br><i>FALHA_VA - MainProgram/MainRoutine - *10(TON)</i><br><i>FALHA_VA.DN - MainProgram/MainRoutine - 11(XIC)</i>   |   | TIMER     | MainProgram        |
| <b>FALHA_VA_DESL</b><br><i>FALHA_VA_DESL - MainProgram/MainRoutine - *12(TON)</i><br><i>FALHA_VA_DESL.DN - MainProgram/MainRoutine - 13(XIC)</i>  |   | TIMER     | MainProgram        |
| <b>FALTA DE LINK</b><br><b>FALTA DE LINK DO SISTEMA DE TELEMETRIA</b><br>AliasFor:<br>Base Tag:<br><i>FALTA_DE_LINK - MainProgram/MainRoutine - 39(XIO)</i>   | 0<br>Local:2:I.Data.4(C)<br>Local:2:I.Data.4(C) | BOOL      | MainProgram        |
| <b>FALTA DE LINK_AUX</b><br><b>AUXILIAR PARA SUPERVISÓRIO</b><br><i>FALTA_DE_LINK_AUX - MainProgram/MainRoutine - *39(OTE)</i>  | 0   | BOOL      | MainProgram        |
| <b>FINOS_FALHAEQUIP</b><br><i>FINOS_FALHAEQUIP - MainProgram/MainRoutine - *14(CLR)</i><br><i>FINOS_FALHAEQUIP.0 - MainProgram/MainRoutine - *11(OTE), 8(XIC), 9(XIO)</i>   | 0   | INT       | Monografia_rodrigo |

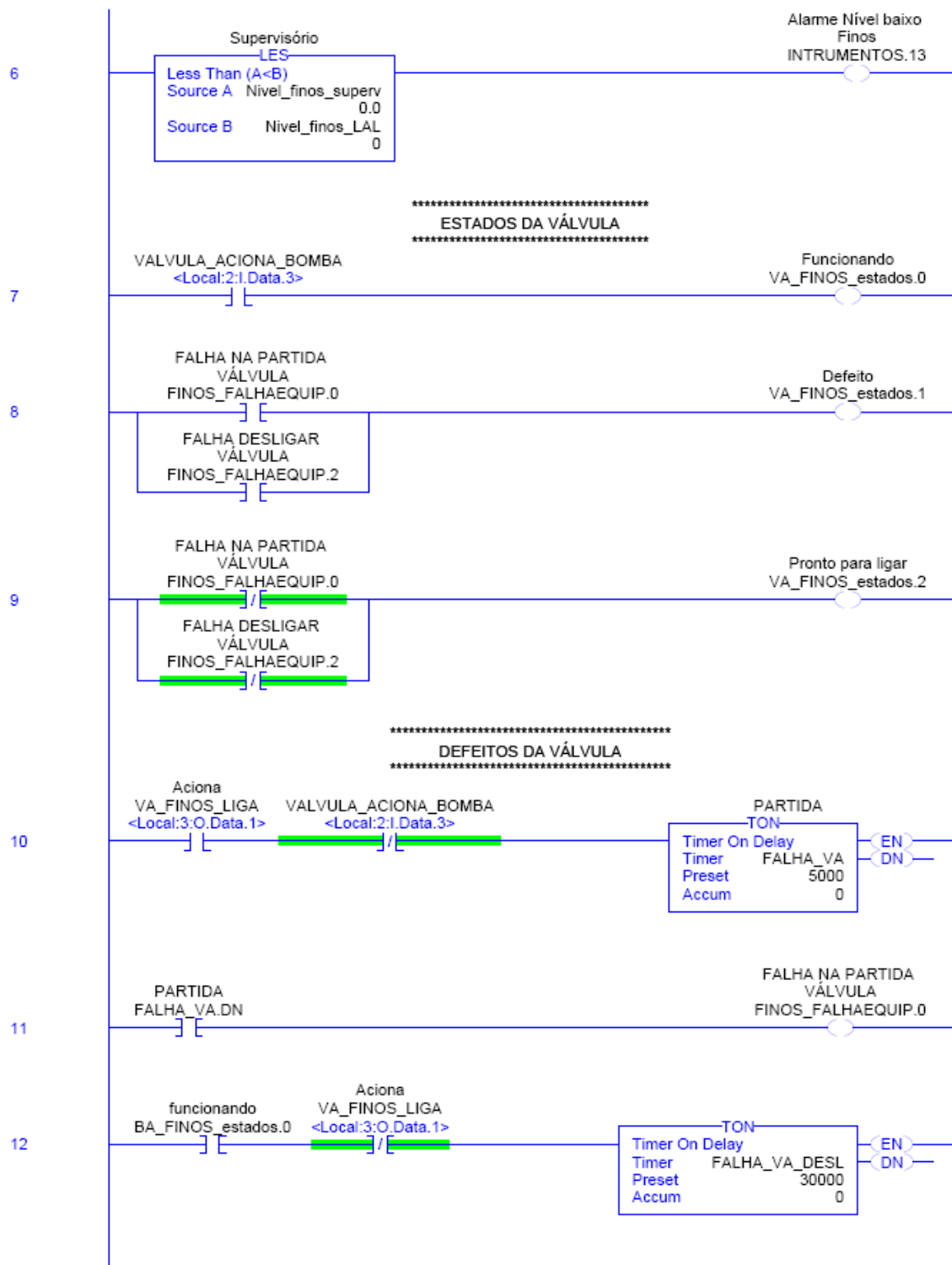
**FINOS FALHAEQUIP (Continued)***FINOS\_FALHAEQUIP.1 - MainProgram/MainRoutine - \*21(OTL), 18(XIC), 19(XIO)**FINOS\_FALHAEQUIP.2 - MainProgram/MainRoutine - \*13(OTE), 8(XIC), 9(XIO)*

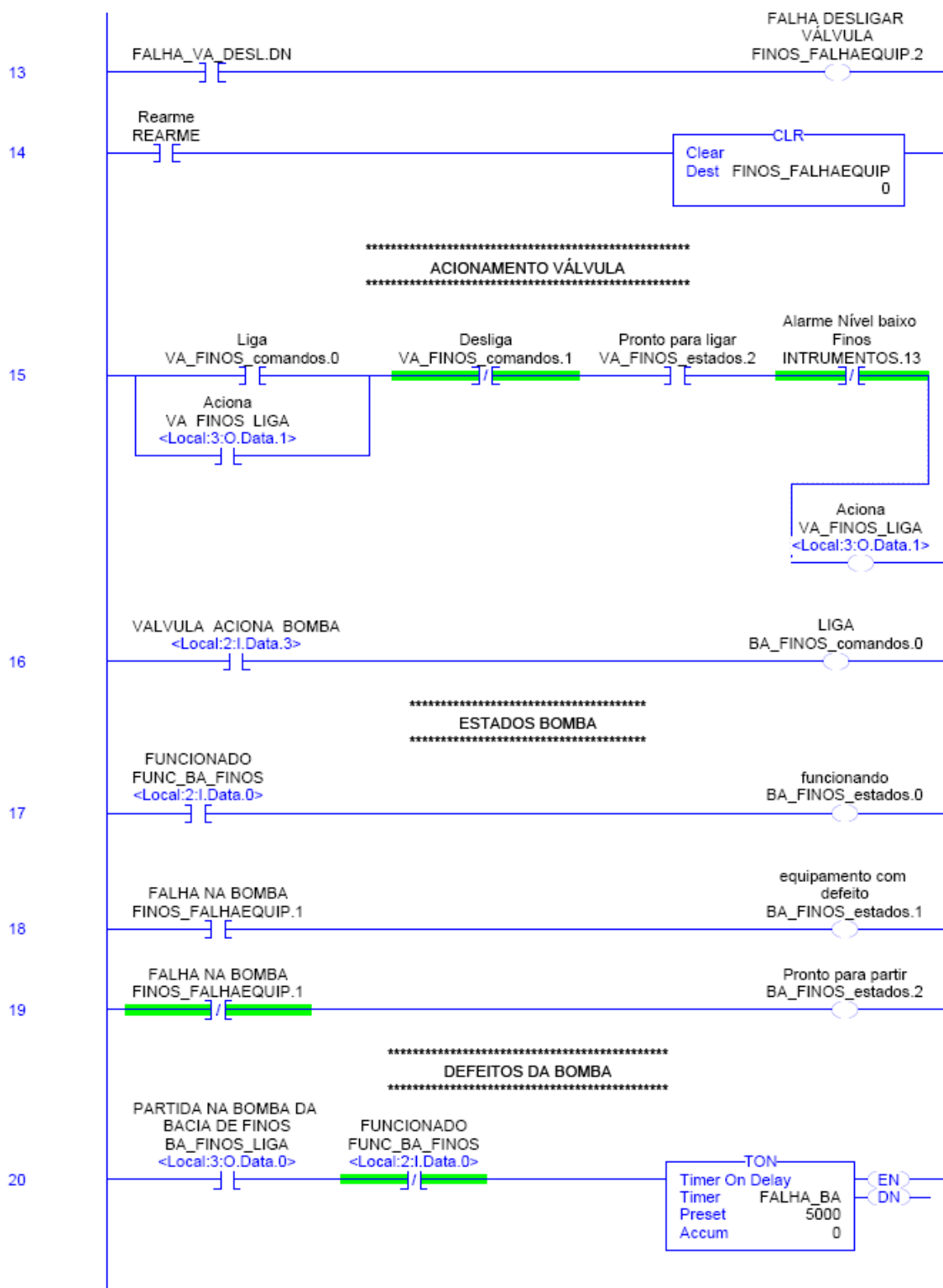
|  |                      |                 |                    |
|--|----------------------|-----------------|--------------------|
| <b>FUNC_BA_FINOS</b>   | 0                    | BOOL            | MainProgram        |
| FUNÇIONADO   |                      |                 |                    |
| AliasFor:  | Local:2:I.Data.0(C)  |                 |                    |
| Base Tag:  | Local:2:I.Data.0(C)  |                 |                    |
| <i>FUNC_BA_FINOS - MainProgram/MainRoutine - 17(XIC), 20(XIO), 36(XIC)</i>                 |                      |                 |                    |
| <b>HORAS_BOMBA_FINOS</b>   | 0                    | DINT            | MainProgram        |
| <i>HORAS_BOMBA_FINOS - MainProgram/MainRoutine - *29(MOV)</i>                              |                      |                 |                    |
| <b>INTRUMENTOS</b>   | 0                    | INT             | Monografia_rodrigo |
| <i>INTRUMENTOS.0 - MainProgram/MainRoutine - *31(OTL)</i>                                  |                      |                 |                    |
| <i>INTRUMENTOS.1 - MainProgram/MainRoutine - *30(OTL)</i>                                  |                      |                 |                    |
| <i>INTRUMENTOS.10 - MainProgram/MainRoutine - *1(OTE)</i>                                  |                      |                 |                    |
| <i>INTRUMENTOS.11 - MainProgram/MainRoutine - *0(OTE)</i>                                  |                      |                 |                    |
| <i>INTRUMENTOS.12 - MainProgram/MainRoutine - *5(OTE)</i>                                  |                      |                 |                    |
| <i>INTRUMENTOS.13 - MainProgram/MainRoutine - *6(OTE), 15(XIO)</i>                         |                      |                 |                    |
| <i>INTRUMENTOS.2 - MainProgram/MainRoutine - *34(OTE)</i>                                  |                      |                 |                    |
| <i>INTRUMENTOS.3 - MainProgram/MainRoutine - *35(OTE)</i>                                  |                      |                 |                    |
| <i>INTRUMENTOS.4 - MainProgram/MainRoutine - *37(OTE)</i>                                  |                      |                 |                    |
| <b>Local:1:I</b>   |                      | AB:1769_IF8:I:0 | Monografia_rodrigo |
| <i>Local:1:I.Ch0OverRange - MainProgram/MainRoutine - 31(XIC), 32(XIO)</i>                 |                      |                 |                    |
| <i>Local:1:I.Ch0UnderRange - MainProgram/MainRoutine - 30(XIC), 32(XIO)</i>                |                      |                 |                    |
| <i>Local:1:I.Ch1OverRange - MainProgram/MainRoutine - 1(XIC), 2(XIO)</i>                   |                      |                 |                    |
| <i>Local:1:I.Ch1UnderRange - MainProgram/MainRoutine - 0(XIC), 2(XIO)</i>                  |                      |                 |                    |
| <b>NIVEL_FINOS</b>   | 0                    | INT             | MainProgram        |
| Entrada  |                      |                 |                    |
| AliasFor:  | Local:1:I.Ch1Data(C) |                 |                    |
| Base Tag:  | Local:1:I.Ch1Data(C) |                 |                    |
| <i>NIVEL_FINOS - MainProgram/MainRoutine - 2(MOV)</i>                                      |                      |                 |                    |
| <b>Nivel_finos_AUX</b>   | 0                    | DINT            | MainProgram        |
| <i>Nivel_finos_AUX - MainProgram/MainRoutine - *0(MOV), *1(MOV), *2(MOV), 3(CPT)</i>       |                      |                 |                    |
| <b>Nivel_finos_LAH</b>   | 0                    | INT             | MainProgram        |
| Alarme nível alto  |                      |                 |                    |
| <i>Nivel_finos_LAH - MainProgram/MainRoutine - 5(GRT)</i>                                  |                      |                 |                    |
| <b>Nivel_finos_LAL</b>   | 0                    | DINT            | MainProgram        |
| Alarme nível baixo   |                      |                 |                    |
| <i>Nivel_finos_LAL - MainProgram/MainRoutine - 6(LES)</i>                                  |                      |                 |                    |
| <b>Nivel_finos_superv</b>  | 0.0                  | REAL            | MainProgram        |
| Supervisão   |                      |                 |                    |
| <i>Nivel_finos_superv - MainProgram/MainRoutine - *4(CPT), 5(GRT), 6(LES)</i>              |                      |                 |                    |
| <b>PRESSAO_FINOS</b>   | 20000                | INT             | MainProgram        |
| Entrada  |                      |                 |                    |
| AliasFor:  | Local:1:I.Ch0Data(C) |                 |                    |
| Base Tag:  | Local:1:I.Ch0Data(C) |                 |                    |
| <i>PRESSAO_FINOS - MainProgram/MainRoutine - 32(MOV)</i>                                   |                      |                 |                    |
| <b>Pressao_finos_AUX</b>   | 0                    | DINT            | MainProgram        |
| <i>Pressao_finos_AUX - MainProgram/MainRoutine - *30(MOV), *31(MOV), *32(MOV), 33(CPT)</i> |                      |                 |                    |
| <b>Pressao_finos_PAH</b>   | 0                    | DINT            | MainProgram        |
| Alarme de pressão alta   |                      |                 |                    |
| <i>Pressao_finos_PAH - MainProgram/MainRoutine - 34(GRT)</i>                               |                      |                 |                    |
| <b>Pressao_finos_PAL</b>   | 0                    | DINT            | MainProgram        |
| Alarme de pressão baixa  |                      |                 |                    |
| <i>Pressao_finos_PAL - MainProgram/MainRoutine - 35(LES)</i>                               |                      |                 |                    |

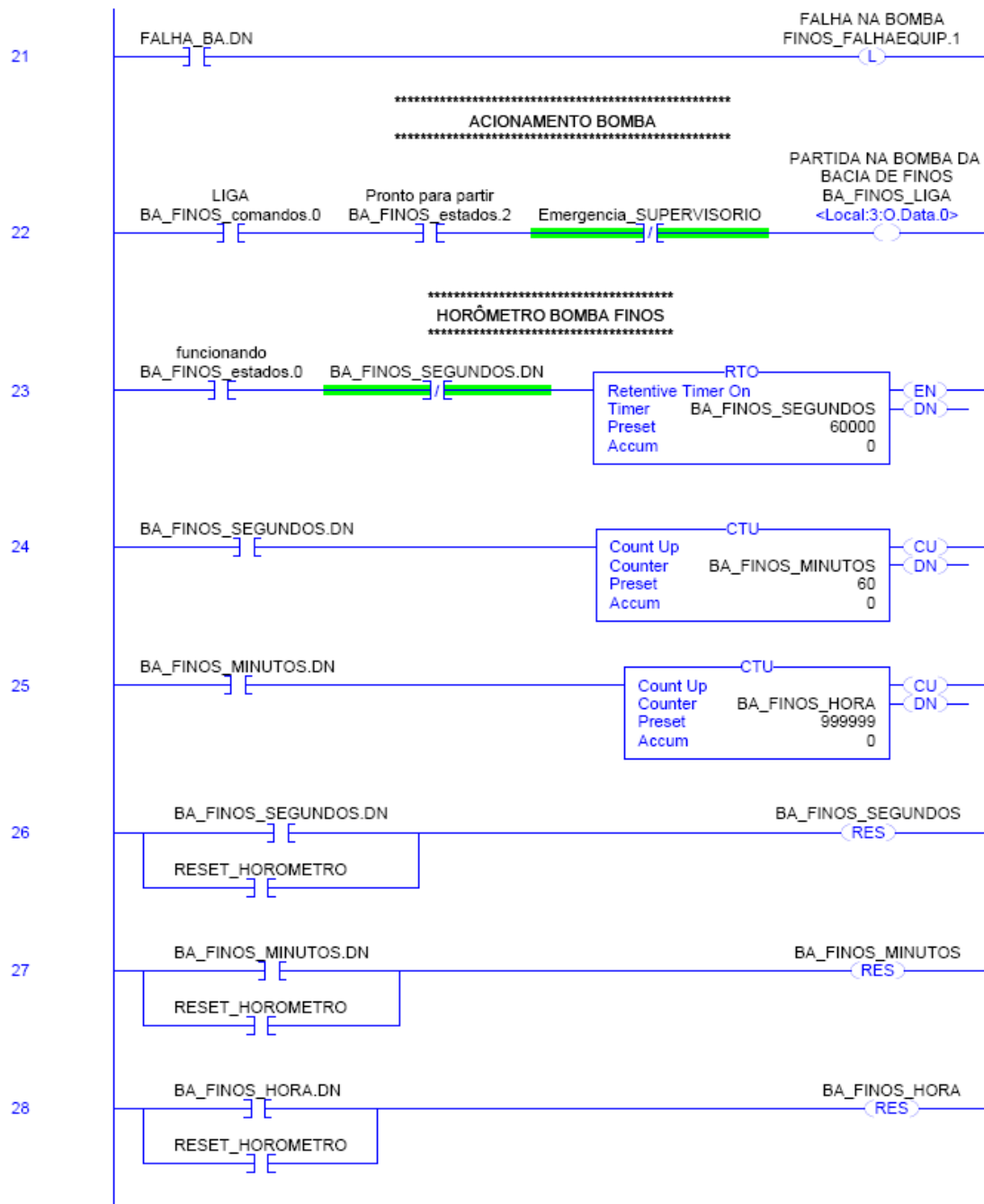
|  |                     |       |                    |
|--|---------------------|-------|--------------------|
| <b>Pressao_finos_superv</b>  | 7.187023            | REAL  | MainProgram        |
| Supervisório   |                     |       |                    |
| <i>Pressao_finos_superv - MainProgram/MainRoutine - *33(CPT), 34(GRT), 35(LES)</i>   |                     |       |                    |
| <b>REARME</b>  | 0                   | BOOL  | MainProgram        |
| Rearme   |                     |       |                    |
| <i>REARME - MainProgram/MainRoutine - 14(XIC)</i>                                    |                     |       |                    |
| <b>RESET_HOROMETRO</b>   | 0                   | BOOL  | MainProgram        |
| <i>RESET_HOROMETRO - MainProgram/MainRoutine - 26(XIC), 27(XIC), 28(XIC)</i>         |                     |       |                    |
| <b>VA_FINOS_comandos</b>   | 0                   | INT   | Monografia_rodrigo |
| <i>VA_FINOS_comandos.0 - MainProgram/MainRoutine - 15(XIC)</i>                       |                     |       |                    |
| <i>VA_FINOS_comandos.1 - MainProgram/MainRoutine - 15(XIO)</i>                       |                     |       |                    |
| <b>VA_FINOS_estados</b>  | 0                   | DINT  | Monografia_rodrigo |
| <i>VA_FINOS_estados.0 - MainProgram/MainRoutine - *7(OTE)</i>                        |                     |       |                    |
| <i>VA_FINOS_estados.1 - MainProgram/MainRoutine - *8(OTE)</i>                        |                     |       |                    |
| <i>VA_FINOS_estados.2 - MainProgram/MainRoutine - *9(OTE), 15(XIC)</i>               |                     |       |                    |
| <b>VA_FINOS_LIGA</b>   | 0                   | BOOL  | MainProgram        |
| Aciona   |                     |       |                    |
| AliasFor:  | Local:3:O.Data.1(C) |       |                    |
| Base Tag:  | Local:3:O.Data.1(C) |       |                    |
| <i>VA_FINOS_LIGA - MainProgram/MainRoutine - *15(OTE), 10(XIC), 12(XIO), 15(XIC)</i> |                     |       |                    |
| <b>VALVULA_ACIONA_BOMBA</b>  | 0                   | BOOL  | MainProgram        |
| AliasFor:  | Local:2:I.Data.3(C) |       |                    |
| Base Tag:  | Local:2:I.Data.3(C) |       |                    |
| <i>VALVULA_ACIONA_BOMBA - MainProgram/MainRoutine - 10(XIO), 16(XIC), 7(XIC)</i>     |                     |       |                    |
| <b>Vazamento</b>   |                     | TIMER | MainProgram        |
| <i>Vazamento - MainProgram/MainRoutine - *36(TON)</i>                                |                     |       |                    |
| <i>Vazamento.DN - MainProgram/MainRoutine - 37(XIC)</i>                              |                     |       |                    |

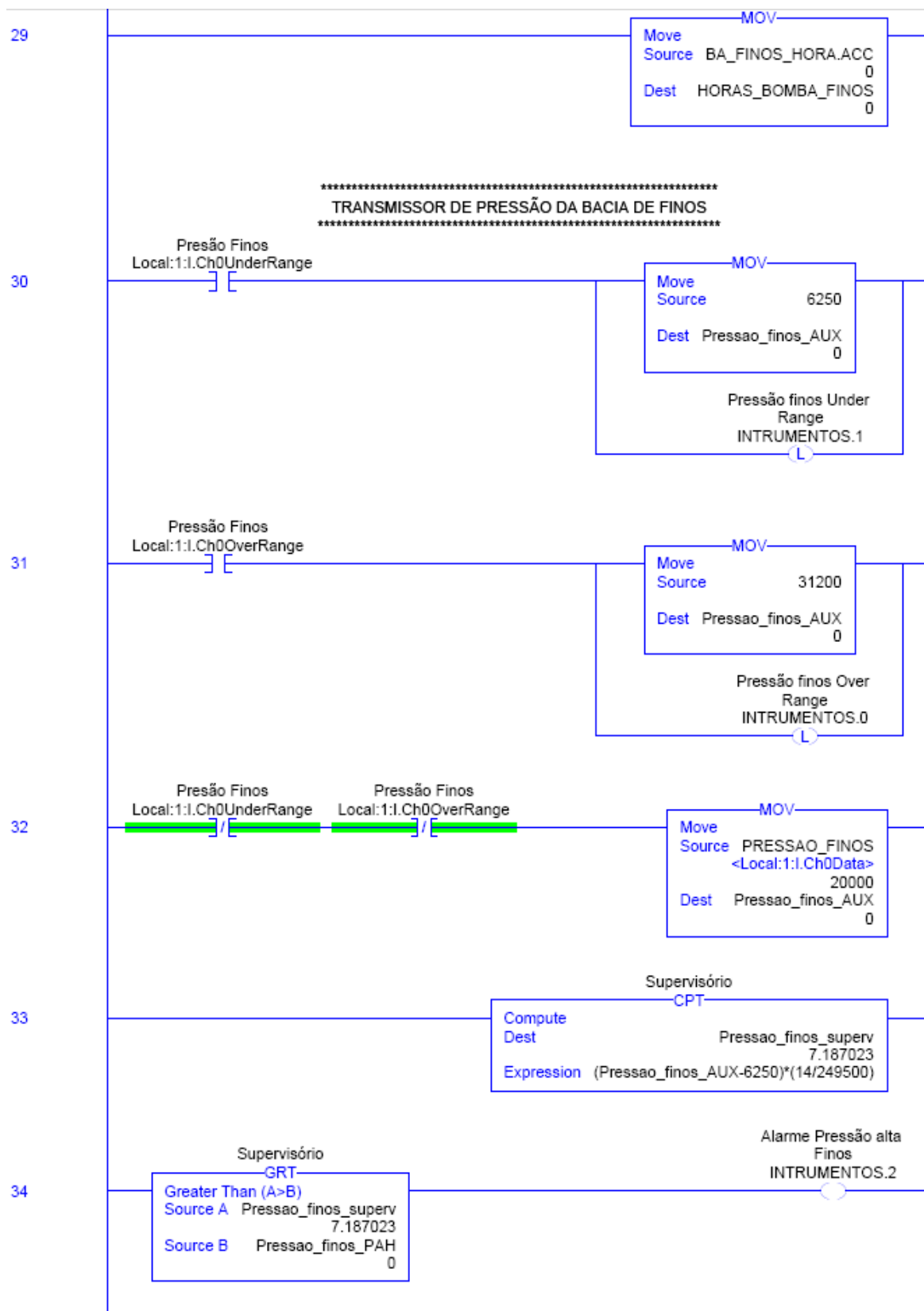
ANEXO B – PROGRAMAÇÃO EM LADDER



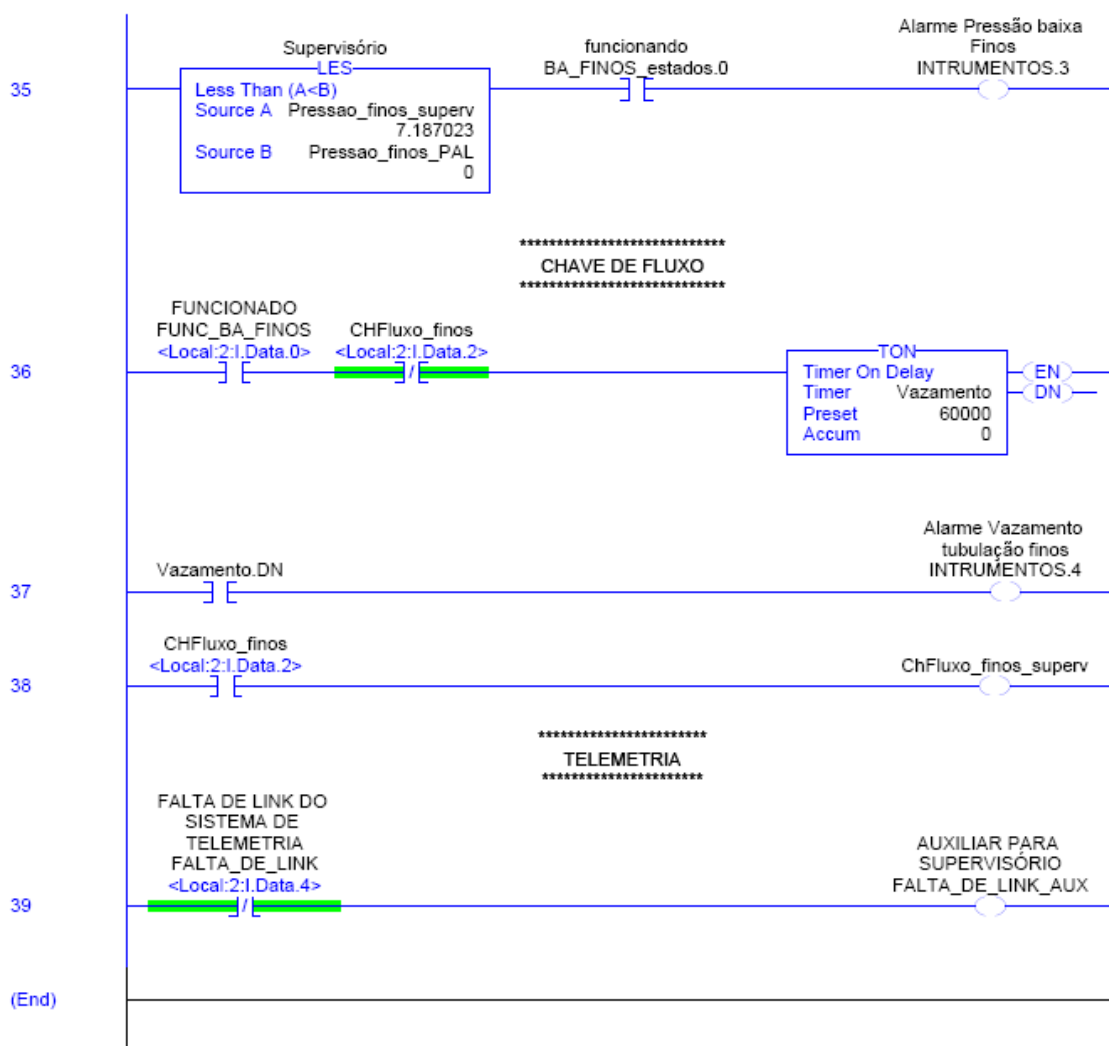












## ANEXO C – TAGS UTILIZADAS NO RSVIEW32

|    | Alm | Tag Name                | Type    | Description                                    |
|----|-----|-------------------------|---------|--|
| 1  | X   | mono\ALR_N_ALTO         | Digital | Alarme de Nível Alto                           |
| 2  | X   | mono\ALR_N_BAIXA        | Digital | Alarme de Nível baixo                          |
| 3  | X   | mono\ALR_N_OVER         | Digital | Alarme de Nível acima da escala                |
| 4  | X   | mono\ALR_N_UNDER        | Digital | Alarme de nível abaixo da escala               |
| 5  | X   | mono\ALR_P_ALTA         | Digital | Alarme Pressão Alta                            |
| 6  | X   | mono\ALR_P_BAIXA        | Digital | Alarme Pressão Baixa                           |
| 7  | X   | mono\ALR_P_OVER         | Digital | Alarme de Pressão acima da escala              |
| 8  | X   | mono\ALR_P_UNDER        | Digital | Alarme de Pressão abaixo da escala             |
| 9  | X   | mono\ALR_VAZ_TUB        | Digital | Alarme de vazamento da tubulação               |
| 10 |     | mono\BOMBA_DEFEITO      | Digital | Bomba com defeito                              |
| 11 |     | mono\BOMBA_FUNCIONANDO  | Digital | Bomba funcionando                              |
| 12 |     | mono\BOMBA_PPP          | Digital | Bomba pronta para partir                       |
| 13 |     | mono\CH_FLUXO           | Digital | Tag auxiliar para animação no supervisão       |
| 14 |     | mono\EMERGENCIA_SUPERV  | Digital | Botão de emergencia para desligar a bomba      |
| 15 | X   | mono\FALHA_DESL_VAL     | Digital | Falha ao desligar a válvula                    |
| 16 | X   | mono\FALHA_PART_BOMBA   | Digital | Falha na partida da bomba                      |
| 17 | X   | mono\FALHA_PART_VAL     | Digital | Falha na partida da válvula                    |
| 18 | X   | mono\FALTA_DE_LINK      | Digital | FALTA DE LINK NO SISTEMA DE TELEMETRIA         |
| 19 |     | mono\HOROMETRO          | Analog  | Totalizador do Horômetro                       |
| 20 |     | mono\LTFINOS            | Analog  | Nível da bacia de finos                        |
| 21 |     | mono\PT_FINOS           | Analog  | Pressão na linha de recalque da bacia de finos |
| 22 |     | mono\rearme             | Digital | Rearme da válvula e da bomba                   |
| 23 |     | mono\RESET_HOROMETRO    | Digital | Reset do Horômetro                             |
| 24 |     | mono\SET_LAH            | Analog  | Set do alarme de nível alto                    |
| 25 |     | mono\SET_LAL            | Analog  | Set do alarme de nível baixo                   |
| 26 |     | mono\SET_PAH            | Analog  | Set do alarme de pressão alta                  |
| 27 |     | mono\SET_PAL            | Analog  | Set do alarme de pressão baixa                 |
| 28 |     | mono\VAL_DESLIGA        | Digital | Comando para desligar a válvula                |
| 29 |     | mono\VAL_LIGA           | Digital | Comando para ligar válvula                     |
| 30 |     | mono\VALVULA_DEFEITO    | Digital | Valvula com defeito                            |
| 31 |     | mono\VALVULA_FUNCINANDO | Digital | Valvula funcionando                            |
| 32 |     | mono\VALVULA_PPL        | Digital | Valvula pronta para ligar                      |