

JOSUÉ GOMES AZOIA FILHO

DIGITALIZAÇÃO DE PROCESSOS INDUSTRIAIS ATRAVÉS DE
CONTROLADORES LÓGICOS PROGRAMÁVEIS ALTUS E SISTEMAS
DE SUPERVISÃO E CONTROLE ELIPSE

Assis
2009

DIGITALIZAÇÃO DE PROCESSOS INDUSTRIAIS ATRAVÉS DE
CONTROLADORES LÓGICOS PROGRAMÁVEIS ALTUS E SISTEMAS
DE SUPERVISÃO E CONTROLE ELIPSE

JOSUÉ GOMES AZOIA FILHO

Trabalho de Conclusão de Curso apresentado ao
Instituto Municipal de Ensino Superior de Assis,
como requisito do Curso de Graduação, analisado
pela seguinte comissão examinadora:

Orientador: REGINA FUMIE ETO

Analizador (1): FÁBIO EDER CARDOSO

Analizador (2): FELIPE ALEXANDRE
CARDOSO PAZINATTO

Assis
2009

JOSUÉ GOMES AZOIA FILHO

DIGITALIZAÇÃO DE PROCESSOS INDUSTRIAIS ATRAVÉS DE
CONTROLADORES LÓGICOS PROGRAMÁVEIS ALTUS E SISTEMAS
DE SUPERVISÃO E CONTROLE ELIPSE

Trabalho de Conclusão de Curso apresentado ao
Instituto Municipal de Ensino Superior de Assis,
como requisito do Curso de Graduação, analisado
pela seguinte comissão examinadora:

Orientador: REGINA FUMIE ETO

Área de Concentração: Automação
Industrial

Assis
2009

AGRADECIMENTOS

Em primeiro lugar a Deus que iluminou o meu caminho durante esta caminhada.

A minha família, minha esposa Vera, meu filho Daniel, pelo apoio, compreensão e carinho.

Aos meus pais Josué e Natalina por terem me educado com esforço e dignidade e cuidado para que me tornasse uma pessoa digna e justa.

À orientadora professora Regina pelo apoio e dedicação que me guiou para que este trabalho atingisse seus objetivos.

Aos professores da coordenadoria de informática da FEMA que durante o curso souberam transferir seus conhecimentos aos alunos, com dedicação e perseverança.

A Duke Energy, empresa que trabalho, por ter me incentivado a crescer através de uma formação superior.

RESUMO

Com o desenvolvimento da informática após os anos 90, a automação industrial deixou de ser um ramo de técnicos e engenheiros eletrônicos e se tornou um ramo de atividade para programadores, engenheiros e profissionais da informática.

Trabalhar com automação industrial é sinônimo de trabalhar com CLP (Controlador Lógico Programável) e SSC (Sistema de Supervisão e Controle), necessitando de um grande conhecimento em programação, lógicas, algoritmos, redes, eletrônica analógica e digital e sistemas numéricos.

A CLP trata os dados digitais e analógicos recebidos de diversos dispositivos espalhados pela usina e o SSC é a interface onde estes dados são disponibilizados para os operadores para controle e supervisão da instalação.

A integração de painéis elétricos, dispositivos eletromecânicos e a informática esta cada vez mais apurada, sendo necessário o desenvolvimento de profissionais multifuncionais que possam integrar hardware e software.

Este profissional procurado no mercado pode ser um técnico em eletrônica ou eletrotécnica que tem que se desenvolver em informática, como pode ser um profissional da área de informática que desenvolve conhecimentos na área de automação.

Palavras Chaves: Automação industrial, Lógica, Eletrônica Digital.

ABSTRACT

With the development of information after years 90, industrial automation is no longer a bunch of technicians and electronic engineers and became a hive of activity for programmers, engineers and computer professionals.

Working with industrial automation is synonymous with working with PLC (Programmable Logic Controller) and SSC (System Supervision and Control), requiring a strong knowledge of programming logic, algorithms, networks, analog electronics and digital number systems.

The PLC is the digital and analog data received from various devices throughout the plant and the SSC is the interface where data are available for operators to control and supervise the installation.

The integration of electrical panels, electromechanical devices and computers that increasingly refined, requiring the development of multi-professionals who would be hardware and software.

This professional looking marketing can be an electronics technician or electrical engineering who has to develop computers, how to be a professional computer science that develops expertise in automation

Keywords: Industrial Automation, Logic, Digital Electronics

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama de execução.....	08
Figura 2 - Processamento cíclico.....	11
Figura 3 - Processamento por interrupção.....	12
Figura 4 - Diagrama de contatos.....	14
Figura 5 - Diagrama de blocos lógicos.....	15
Figura 6 - Formato de uma lógica.....	18
Figura 7 - Configuração do CLP.....	20
Figura 8 - Configuração dos barramentos.....	21
Figura 9 - Configuração do barramento 0.....	21
Figura 10 - Configuração do barramento 2.....	22
Figura 11 - Configuração do barramento 3.....	23
Figura 12 - Tela inicial do MASTERTOOL.....	24
Figura 13 - Barra de ferramentas de comando.....	25
Figura 14 - Barra de ferramentas de relatórios.....	25
Figura 15 - Barra de ferramentas de instruções.....	25
Figura 16 - Caixa de diálogo para criação de um novo módulo.....	26
Figura 17 - Caixa de diálogo para editar projeto.....	27
Figura 18 - Barramento Ethernet.....	28
Figura 19 - Configuração do módulo Ethernet.....	28
Figura 20 - Estado da memória do CLP da UG-01 CNO I.....	30
Figura 21 - Formato de uma lógica.....	31
Figura 22 - Ordem de processamento de uma lógica.....	31
Figura 23 - Formato de um operando simples.....	31
Figura 24 - Formato geral de um operando constante.....	33
Figura 25 - Formato geral de um operando tabela.....	34
Figura 26 - Na lógica 4 do módulo P-REGTEN.009.....	36
Figura 27 - Lógica 119 do módulo P-PART.005.....	37
Figura 28 - Lógica 012 do módulo P-ANALOG.012.....	38
Figura 29 - Lógica 059 do módulo P-PART.005.....	38
Figura 30 - Lógica 031 do módulo P-PART.005.....	40
Figura 31 - Lógica 062 do módulo P-PART.005.....	41

Figura 32 - Lógica 008,009 e 010 do módulo P-PART.005.....	42
Figura 33 - Caixa de diálogo para Abrir Projeto.....	43
Figura 34 - Caixa de diálogo para Editar Projeto.....	44
Figura 35 - Caixa de diálogo de endereço do CLP.....	45
Figura 36 - Caixa de diálogo Menu Comunicação Estado.....	46
Figura 37 - Caixa de diálogo Menu Comunicação Ler/Enviar Módulo.....	47
Figura 38 - Caixa de diálogo Menu Comunicação Diretório de Módulo.....	48
Figura 39 - Caixa de diálogo Menu Comunicação Monitorar.....	50
Figura 40 - Lista de monitoramento de um octeto.....	50
Figura 41 - Lista de monitoramento de entradas analógicas.....	50
Figura 42 - Monitoramento da Lógica 003 P-REGTEN.009.....	51
Figura 43 - Monitoramento da Lógica 004 P-REGTEN.009.....	52
Figura 44 - Switch Industrial RUGGEDCOM RS900G.....	53
Figura 45 - Cartão de comunicação Altus AL 3414.....	55
Figura 46 - Gráfico do frame NMEA e o sinal de PPS.....	58
Figura 47 - Configuração da rede de sincronismo Mestre.....	58
Figura 48 - Configuração da rede de sincronismo Escravo.....	59
Figura 49 - Tela de supervisão do sistema digital.....	60
Figura 50 - Tela do E3 Studio projeto Driver.....	64
Figura 51 - Tela do E3 Studio projeto BIB_OBJETOS.....	65
Figura 52 - Objeto DISJUNTOR_1 OU 2.....	66
Figura 53 - Tela do E3 Studio projeto BIB_TELA.....	67
Figura 54 - Tela genérica para Telecomando de equipamentos.....	68
Figura 55 - Tela do E3 Studio projeto PRINCIPAL.....	68
Figura 56 - <i>Design</i> da tela de controle no E3 Studio.....	69
Figura 57 - Estrutura da tela de controle no E3 Studio.....	70
Figura 58 - Associações da tela de controle no E3 Studio	70
Figura 59 - Tela de controle de Potência no E3 Viewer.....	71

LISTA DE TABELAS

Tabela 1 - Memórias de um CLP.....	13
Tabela 2 - Certificação das fibras ópticas.....	54
Tabela 3 - Endereço IP da rede Automação.....	56

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
CAG	Controle automático de Geração
CESP	Companhia Energética de São Paulo
CLP	Controlador Lógico Programável
CPU	Central Processing Unit
CRO	Centro Regional Operação
DIO	Distribuidor interno óptico
E/S	Entradas e Saídas
GPS	Sistema de Posicionamento Global
IHM	Interface Homem Máquina
NEMA	National Electrical Manufacturers Association
ONS	Operador Nacional do Sistema
PC	Computador Pessoal
PCSE	Processador de Comunicação Setor Elétrico
PID	Proporcional Integral Derivada
PLC	Programable Logic Control
PPS	Pulso Por Segundo
RAM	Random Access Memory
ROM	Read Only Memory
SAT	Seqüência Automática de Telecomando
SCADA	<i>Supervisory Control and Data Acquisition</i>
SENAI	Serviço Nacional de Aprendizagem Industrial
SSC	Sistema de Supervisão e Controle
SSCH	Sistema Supervisor de Controle Hierárquico
UAC	Unidade de Aquisição e Controle
UCP	Unidade Central de Processamento
WD	Watch Dog

SUMÁRIO

Introdução.....	01
1 HISTÓRICO DA AUTOMAÇÃO NO BRASIL.....	02
1.1 DESCRIÇÃO.....	02
1.2 HISTÓRICO.....	03
1.3 A EVOLUÇÃO DA AUTOMAÇÃO INDUSTRIAL NAS HIDRELETRICAS.....	03
1.4 EVOLUÇÃO DAS CLP'S AO LONGO DO TEMPO.....	05
2 UTILIZAÇÃO DE CLP'S.....	07
2.1 CARACTERÍSTICAS.....	07
2.2 CAPACIDADE DE UM CLP.....	07
2.3 ARQUITETURA.....	08
2.4 APLICAÇÕES.....	09
2.5 PRINCÍPIOS DE FUNCIONAMENTO DE UM CLP.....	10
2.6 FUNCIONAMENTO DA UNIDADE CENTRAL DE PROCESSAMENTO (UCP).....	10
2.7 ARQUITETURA DE MEMÓRIA DE UM CLP.....	12
2.8 CONCEITOS DE PROGRAMAÇÃO DE CLPS.....	14
2.8.1 Apresentação.....	14
2.8.2 Diagrama de contatos.....	14
2.8.3 Diagrama de blocos lógicos.....	14
2.8.4 Lista de instrução.....	15
2.8.5 Linguagem corrente.....	15
2.8.6 Documentação.....	15
2.8.7 Conjunto de Instruções.....	16
2.8.8 Normalização.....	16
2.9 Programando o CLP Altus.....	17
3 CONFIGURAÇÃO DO CLP DA UHE CANOAS I.....	19

3.1 LEVANTAMENTO E PADRONIZAÇÃO DOS DADOS DE CAMPO.....	19
3.2 CONFIGURAÇÃO DO CLP.....	20
3.3 SOFTWARE DE PROGRAMAÇÃO DO CLP ALTUS.....	23
3.3.1 Uma Visão Geral do MasterTool Programming.....	24
3.3.2 Utilizando um projeto.....	26
3.3.3 Configurando o endereço de rede.....	27
4 PROGRAMAÇÃO DO CLP ALTUS.....	29
4.1 ELEMENTOS DE PROGRAMAÇÃO MASTERTOOL.....	29
4.2 ORGANIZAÇÃO DE MEMÓRIA DO CLP ALTUS.....	29
4.3 FORMATAÇÃO DOS OPERANDOS DO MASTERTOOL.....	31
4.3.1 Operandos Simples.....	31
4.3.2 Operandos Constante.....	33
4.3.2 Operandos Tabela.....	34
4.4 PRINCIPAIS INSTRUÇÕES DO MASTERTOOL.....	34
4.4.1 Instrução < - Menor.....	34
4.4.2 Instrução > - Maior.....	35
4.4.3 Instrução = - Igual.....	36
4.4.4 Instrução A/D - Conversão Analógico/Digital.....	37
4.4.5 Instrução PLS - Relé de Pulso.....	38
4.4.6 RNA - Contato normalmente aberto.....	39
4.4.7 RNF - Contato Normalmente Fechado.....	39
4.4.8 TEE - Temporizador na Energização.....	40
4.4.9 Bobinas.....	41
4.4.9.1 BOB - Bobina Simples.....	41
4.4.9.2 BBL - Bobina Liga.....	41
4.4.9.2 BBD - Bobina Desliga.....	42
5 COMO ANALISAR UM PROGRAMA NO MASTERTOOL.....	43
5.1 PRINCIPAIS FERRAMENTAS MENU PROJETO.....	43
5.1.1 Menu Projeto Abrir.....	43
5.1.2 Menu Projeto Editar.....	43
5.2 PRINCIPAIS FERRAMENTAS MENU COMUNICAÇÃO.....	44

5.2.1 Menu Comunicação Endereço.....	44
5.2.2 Menu Comunicação Estado.....	45
5.2.3 Menu Comunicação Ler/Enviar Módulo.....	47
5.2.4 Menu Comunicação Módulos.....	48
5.2.5 Menu Comunicação Monitorar.....	49
5.2.6 Menu Comunicação Monitorar Programa.....	51
6 ESTRUTURA DE REDE DE INTERLIGAÇÃO DOS CLP'S.....	53
6.1 Rede de comunicação.....	53
6.1.1 Configuração do Anel óptico.....	53
6.1.2 Cartão de comunicação do CLP.....	55
6.1.3 Comunicação do SSC com a rede.....	56
6.1.4 Endereçamento IP da rede de automação de Canoas I.....	56
6.2 Rede de Sincronismo.....	56
6.2.1 Funcionamento da rede de sincronismo.....	57
6.3 Rede de comunicação externa.....	59
6.4 Supervisão do Sistema Digital no SSC.....	60
7 SISTEMA DE SUPERVISÃO E COTROLE.....	61
7.1 Substituição do SSC Alstom por Elipse.....	61
7.2 Características do Elipse E3.....	61
7.3 Redundância de operação e servidor.....	62
7.4 Banco de Dados em <i>Hot-StandBy</i>	62
7.5 Divisão dos processos no Elipse SCADA.....	63
7.6 Divisão Sistema de Supervisão Elipse E3.....	64
7.6.1 Projeto Driver.....	64
7.6.2 Projeto BIB_OBJETOS.....	65
7.6.3 Projeto BIB_TELA.....	67
7.6.4 Projeto PRINCIPAL.....	68
8 CONCLUSÕES.....	73
8.1 RESULTADOS ALCANÇADOS.....	73
8.2 TRABALHOS FUTUROS.....	73
REFERÊNCIAS BIBLIOGRÁFICAS.....	74

Introdução

A automação industrial é um ramo crescente no mercado nacional que necessita de profissionais capacitados, para desenvolvimento e análise de programas.

O mais comum hoje é as fabricas e empresas integradoras do ramo de automação industrial, contratar profissionais que fizeram cursos profissionalizantes ou engenheiros e treinarem para programação.

Neste trabalho estudantes e profissionais da área de informática poderão verificar que para programar e configurar CLP (Controlador Lógico Programável) e SSC (Sistema de Supervisão e Controle), necessitam de conhecimentos como programação, lógicas, algoritmos, redes, eletrônica analógica e digital e sistemas numéricos.

Poderemos ver que a CLP trata os dados digitais e analógicos recebidos de diversos dispositivos espalhados pela usina e o SSC é a interface onde estes dados são disponibilizados para os operadores para controle e supervisão da instalação.

A integração de painéis elétricos, dispositivos eletromecânicos e a informática esta cada vez mais apurada, sendo necessário o desenvolvimento de profissionais multifuncionais que possam integrar hardware e software.

No trabalho veremos que o hardware pode ser configurado e expandido conforme a necessidade ou aplicação.

Vamos ver que através de um software podemos fazer seqüenciamento de eventos, controle de processos e intertravamento de comandos e controles de processos industriais.

Capítulo 1: HISTÓRICO DA AUTOMAÇÃO NO BRASIL

1.1 DESCRIÇÃO

O Controlador Lógico Programável, ou simplesmente CLP, tem revolucionado os comandos e controles industriais desde seu surgimento no final da década de 60.

Antes do surgimento dos CLP's as tarefas de comando e controle de máquinas e processos indústrias eram feitos por relés eletromagnéticos, especialmente projetados para este fim.

O primeiro CLP surgiu na indústria automobilística, até então um usuário em potencial dos relés eletromagnéticos utilizados para controlar operações seqüenciadas e repetitivas numa linha de montagem. A primeira geração de CLP's utilizou componentes discretos como transistores e circuitos integrados com baixa escala de integração. [SENAI 2002]

Este equipamento foi batizado nos Estados Unidos como PLC (*Programmable Logic Control*), em português CLP (Controlador Lógico Programável) e este termo é registrado pela Allen Bradley (fabricante de CLPs). [SENAI 2002]

Segundo a definição dada pela ABNT (Associação Brasileira de Normas Técnicas), CLP é um equipamento eletrônico digital com hardware e software compatíveis com aplicações industriais. Na norma internacional segundo a NEMA (National Electrical Manufacturers Association), CLP é um aparelho eletrônico digital que utiliza uma memória programável para o armazenamento interno de instruções para implementações específicas, tais como lógica, sequenciamento, temporização, contagem e aritmética, para controlar, através de módulos de entradas e saídas, vários tipos de máquinas ou processos. [NATALE 2000]

Nesta monografia será adotada a definição da NEMA, pois descreveremos uma aplicação, onde esta definição é a mais adequada, para mostrar como tratar sinais de campo na indústria de energia elétrica e através de programação de CLP do fabricante brasileiro Altus pode automatizar e supervisionar uma hidrelétrica.

1.2 HISTÓRICO

O Controlador Lógico Programável – CLP – nasceu dentro da General Motors, em 1968, devido a grande dificuldade de mudar a lógica de controle dos painéis de comando a cada mudança na linha de montagem. Tais mudanças implicavam em altos gastos de tempo e dinheiro. [SENAI 2002]

Sob a liderança do engenheiro Richard Morley, foi preparada uma especificação que refletia as necessidades de muitos usuários de circuitos e relés, não só da indústria automobilística como de toda a indústria manufatureira. [SENAI 2002]

Nascia assim um equipamento bastante versátil e de fácil utilização, que vem se aprimorando constantemente, diversificando cada vez mais os setores industriais e suas aplicações, o que justifica hoje um mercado mundial estimado em 4 bilhões de dólares anuais.

1.3 A EVOLUÇÃO DA AUTOMAÇÃO INDUSTRIAL NAS HIDRELETRICAS.

A automação vem crescendo com o passar dos anos e cada projeto que nasce traz novas evoluções nesta tecnologia. No caso das instalações da empresa CESP a migração dos sistemas analógicos começou no final dos anos 80. Antes todos os processos eram localizados na instalação através de grandes painéis de controle onde se localizava, chaves de comando, relés, instrumentos de medição, onde só era possível controles e comandos locais.

Como este controle era local, os operadores tinham a obrigação de realizar leituras de 30 em 30 minutos todo o controle dos processos passava pela mão do ser humano. Leituras, controle de desempenho, partidas e paradas de máquinas, perturbações de sistema, tudo era anotado pelo operador em livros de registro.

A necessidade de otimização destes processos originou uma série de melhorias e com a aparição de novas tecnologias chegou-se ao estágio atual de automação e supervisão das instalações.

A primeira evolução dos processos se deu no final dos anos 80 com a necessidade de desassistir pequenas subestações de derivação. Surgiu assim um processo que chama-se Telealarme. Este processo ainda era totalmente analógico. Foi feito um estudo das sinalizações que eram relevantes para o funcionamento da

instalação e foi montado um circuito com estes eventos em paralelo e quando atuava, uma destas entradas um circuito eletrônico ligado a uma linha telefônica fazia com que sinalizasse este problema para um centro de operação próximo e o operador deste centro de operação acionava um técnico para ir até a instalação para verificar no local o que estava com problema. Se fosse algo simples este resolvia o problema e rearmava o sistema para uma próxima sinalização.

Na segunda fase desta evolução, havia a necessidade de se ter uma visualização do estado dos equipamentos e medição remota das principais grandezas elétricas de uma planta, originando o sistema denominado SSCH (Sistema de supervisão e controle hierárquico). Este processo, ainda analógico, envolveu as equipes de manutenção em conjunto com a de telecomunicações. Este processo levava, de forma separada, os dois estados dos equipamentos: Ligado/Desligado para disjuntores e Aberto/Fechado para seccionadoras, a medição de tensão, corrente e potência ativa e reativa de cada circuito. Estes sinais eram levados a um painel, tratados e transformados em sinais binários e transmitidos através de canais de microondas.

A terceira evolução destes sistemas foi a integração do SSCH a mais dois sistemas: O primeiro denominado S.A.T. (Seqüência Automática de Telecomando), onde se tornou possível telecomandar os disjuntores a distância e o segundo denominado C.A.G. (Controle Automático de Geração) através deste o CRO (Centro Regional Operação) pode passar a controlar a potência e a tensão dos geradores.

Em meados da década de 90 começa a surgir o controle digital através das CLP's e SSC locais. O processo era controlado por um CLP que recebia os sinais do campo através de entradas digitais e analógicas e o automatismo digital controlava os equipamentos através do SSC (sistema de supervisão e Controle). Era possível visualizar na tela de um micro computador o que acontece no campo, através dos IHM's (interface Homem/Máquina) também passa a ser possível partir e parar Unidades Geradoras e controlar uma planta através de comandos disponíveis na IHM.

Hoje na segunda geração destes sistemas tem-se os processos controlados por CLP's e sincronizados por GPS e integrados ao O.N.S., que tendo integração do

sistema elétrico podemos analisar com grande eficiência a seqüência de eventos e todos os processos.

1.4 EVOLUÇÃO DAS CLP'S AO LONGO DO TEMPO

Pode-se dividir os CLPs historicamente de acordo com o sistema de programação por ele utilizado:

1ª. Geração: Os CLPs de primeira geração se caracterizam pela programação intimamente ligada ao hardware do equipamento. A linguagem utilizada era o Assembly que variava de acordo com o processador utilizado no projeto do CLP, ou seja, para poder programar era necessário conhecer a eletrônica do projeto do CLP. Assim a tarefa de programação era desenvolvida por uma equipe técnica altamente qualificada, gravando - se o programa em memória EPROM, sendo realizada normalmente no laboratório junto com a construção do CLP. [NATALE 2000]

2ª. Geração : Aparecem as primeiras “Linguagens de Programação” não tão dependentes do hardware do equipamento, possíveis pela inclusão de um “Programa Monitor “ no CLP, o qual converte (no jargão técnico, Compila), as instruções do programa, verifica o estado das entradas, compara com as instruções do programa do usuário e altera o estados das saídas. Os Terminais de Programação (ou Maletas, como eram conhecidas) eram na verdade Programadores de Memória EPROM. As memórias depois de programadas eram colocadas no CLP para que o programa do usuário fosse executado. [NATALE 2000]

3ª. Geração : Os CLPs passam a ter uma Entrada de Programação, onde um Teclado ou Programador Portátil é conectado, podendo alterar, apagar, gravar o programa do usuário, além de realizar testes (Debug) no equipamento e no programa. A estrutura física também sofre alterações sendo a tendência para os Sistemas Modulares com Bastidores ou Racks. . [NATALE 2000]

4ª. Geração : Com a popularização e a diminuição dos preços dos micro - computadores (normalmente clones do IBM PC), os CLPs passaram a incluir uma entrada para a comunicação serial. Com o auxílio do microcomputadores a tarefa de programação passou a ser realizada nestes. As vantagens eram a utilização de várias representações das linguagens, possibilidade de simulações e testes,

treinamento e ajuda por parte do software de programação, possibilidade de armazenamento de vários programas no micro, etc. . [NATALE 2000]

5ª. Geração : Atualmente existe uma preocupação em padronizar protocolos de comunicação para os CLPs, de modo a proporcionar que o equipamento de um fabricante “converse” com o equipamento outro fabricante, não só CLPs , como Controladores de Processos, Sistemas Supervisórios, Redes Internas de Comunicação e etc., proporcionando uma integração afim de facilitar a automação, gerenciamento e desenvolvimento de plantas industriais mais flexíveis e normalizadas, fruto da chamada Globalização. Existe uma Fundação Mundial para o estabelecimento de normas e protocolos de comunicação. . [NATALE 2000]

Capítulo 2: UTILIZAÇÃO DE CLP'S

2.1 CARACTERÍSTICAS

Basicamente, um controlador programável apresenta as seguintes características:

- Hardware e/ou dispositivo de controle de fácil e rápida programação e reprogramação com a mínima interrupção da produção.
- Capacidade de operação em ambiente industrial.
- Sinalizadores de estado e módulos tipo plug-in de fácil manutenção e substituição.
- Hardware ocupando espaço reduzido e apresentando baixo consumo de energia.
- Possibilidade de monitoração do estado e operação do processo ou sistema, através da comunicação com computadores.
- Compatibilidade com diferentes tipos de sinais de entrada e saída.
- Capacidade de alimentar, de forma contínua ou chaveada, cargas que consomem correntes de até 2 A.
- Hardware de controle que permite a expansão dos diversos tipos de módulos, de acordo com a necessidade.
- Custo de compra e instalação competitivo em relação aos sistemas de controle convencionais.
- Possibilidade de expansão da capacidade de memória.
- Conexão com outros CLPs através de rede de comunicação.

2.2 CAPACIDADE DE UM CLP.

Podemos ressaltar que, com a popularização dos micros - controladores e a redução dos custos de desenvolvimento e produção houve uma avalanche no mercado de tipos e modelos de CLP's, os quais podemos dividir em:

Nano e Micro – CLP's: São CLP's de pouca capacidade de E/S (máximo 16 Entradas e 16 Saídas), normalmente só digitais, composto de um só módulo (ou placa), baixo custo e reduzida capacidade de memória (máximo 512 passos).

CLP's de Médio Porte: São CLP's com uma capacidade de Entrada e Saída de até 256 pontos, digitais e analógicas, podendo ser formado por um módulo básico, que pode ser expandido. Costumam permitir até 2048 passos de memória,

que poder interna ou externa (Módulos de Estados Sólidos, Soquetes de Memória, etc.), ou podem ser totalmente modulares.

CLP's de Grande Porte: Se caracterizam por uma construção modular, constituída por uma Fonte de alimentação, CPU principal, CPUs auxiliares, CPUs Dedicadas, Módulos de E/S digitais e Analógicos, Módulos de E/S especializados, Módulos de Redes Locais ou Remotas, etc, que são agrupados de acordo com a necessidade e complexidade da automação. Permitem a utilização de até 4096 pontos de E/S. São montados em um Bastidor ou Rack que permite um Cabeamento Estruturado.

2.3 ARQUITETURA

O controlador programável tem sua estrutura baseada no hardware de um computador, tendo, portanto uma unidade central de processamento (UCP), interfaces de entrada e saída e memórias.

As principais diferenças em relação a um computador comum estão relacionadas a qualidade da fonte de alimentação, que possui características ótimas de filtragem e estabilização, interfaces de E/S imune a ruídos e um invólucro específico para aplicações industriais.

O diagrama de blocos a seguir, ilustra a estrutura básica de um controlador programável:

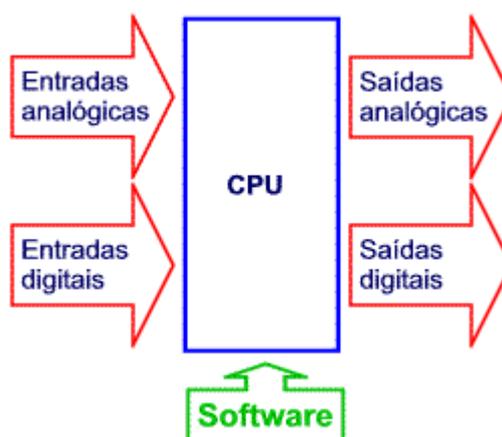


Figura 1 - Diagrama de execução [SENAI 2002]

CPU: os primeiros PLCs microprocessados usavam microcontroladores para esta função, devido ao menor custo e software relativamente simples. Atualmente, esta opção é restrita aos de pequeno porte e uma boa parte usa microprocessadores padrão IBM-PC.

Interfaces de entrada/saída (E/S): a troca de informações entre o equipamento e o PLC se dá através de seus dispositivos de entrada e saída, cujos principais são:

Entradas digitais: São usadas para elementos que apresentam somente dois estados: botões, pressostatos e termostatos, chaves de nível, de fim de curso, etc.

Entradas analógicas: Recebem sinais de dispositivo tipo sensores de pressão, temperatura, indicadores de posição, etc. No PLC, estas entradas são, normalmente, conversores analógico-digitais.

Saídas digitais: Similar às entradas digitais, aciona elementos de dois estados tais como: contadores, válvulas solenóide, lâmpadas, etc.

Saídas analógicas: No PLC são implementadas com conversores digitais / analógicos e usadas para acionar servomotores, posicionadores, instrumentos, etc.

2.4 APLICAÇÕES

O controlador programável existe para automatizar processos industriais, sejam de sequenciamento, intertravamento, controle de processos, etc.

Este equipamento tem seu uso tanto na área de automação da manufatura, de processos contínuos, elétrica, predial, entre outras.

Praticamente não existem ramos de aplicações industriais onde não se possa aplicar os CLPs, entre elas tem-se:

- Máquinas industriais (operatrizes injetoras de plástico, têxteis, calçados);
- Equipamentos industriais para processos (siderurgia, papel e celulose, petroquímica, química, alimentação, mineração, etc);
- Equipamentos para controle de energia (demanda, fator de carga);
- Controle de processos com realização de sinalização, intertravamento e controle PID;
- Aquisição de dados de supervisão em: fábricas, prédios inteligentes, etc;
- Bancadas de teste automático de componentes industriais;

Com a tendência dos CLPs terem baixo custo, muita inteligência, facilidade de uso e massificação das aplicações, a utilização deste equipamento não será apenas nos processos, mas também nos produtos. Pode se encontrar em produtos eletrodomésticos, eletrônicos, residências e veículos.

2.5 PRINCÍPIOS DE FUNCIONAMENTO DE UM CLP

Um controlador lógico programável tem seu funcionamento baseado num sistema de microcomputador onde se tem uma estrutura de software que realiza continuamente ciclos de varredura.

Basicamente a UCP de um controlador programável possui dois estados de operação :

Programação: Neste estado o UCP não executa programa, isto é, não assume nenhuma lógica de controle, ficando preparado para ser configurado ou receber novos programas ou até modificações de programas já instalados. Este tipo de programação é chamado off-line (fora de linha).

Execução: Estado em que o UCP assume a função de execução do programa do usuário. Neste estado, alguns controladores, podem sofrer modificações de programa. Este tipo de programação é chamado on-line (em linha).

A UCP pode assumir também o estado de erro, que aponta falha de operação e execução do programa.

2.6 FUNCIONAMENTO DA UNIDADE CENTRAL DE PROCESSAMENTO (UCP)

A Unidade Central de Processamento (UCP) é responsável pelo processamento do programa, isto é, coleta os dados dos cartões de entrada, efetua o processamento segundo o programa do usuário, armazenado na memória, e envia o sinal para os cartões de saída como resposta ao processamento.

Geralmente, cada CLP tem uma UCP, que pode controlar vários pontos de E/S (entradas e saídas) fisicamente compactadas a esta unidade - é a filosofia compacta de fabricação de CLPs, ou constituir uma unidade separada, conectada a

módulos onde se situam cartões de entrada e saída, - esta é a filosofia modular de fabricação de CLPs.

Este processamento poderá ter estruturas diferentes para a execução de um programa, tais como:

Processamento Cíclico: É a forma mais comum de execução que predomina em todas as UCPs conhecidas, e de onde vem o conceito de varredura, ou seja, as instruções de programa contidas na memória, são lidas uma após a outra seqüencialmente do início ao fim, daí retornando ao início ciclicamente.

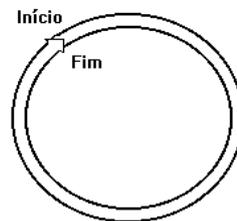


Figura 2 - Processamento cíclico [SENAI 2002]

Um dado importante de uma UCP é o seu tempo de ciclo, ou seja, o tempo gasto para a execução de uma varredura. Este tempo está relacionado com o tamanho do programa do usuário (em média 10 ms a cada 1.000 instruções).

Processamento por interrupção: Certas ocorrências no processo controlado não podem, algumas vezes, aguardar o ciclo completo de execução do programa. Neste caso, ao reconhecer uma ocorrência deste tipo, a UCP interrompe o ciclo normal de programa e executa um outro programa chamado de rotina de interrupção.

Esta interrupção pode ocorrer a qualquer instante da execução do ciclo de programa, podendo ser pré-programado ou quando um evento externo com prioridade é reconhecido pela UCP. Ao finalizar esta situação o programa voltará a ser executado do ponto onde ocorreu a interrupção.

Uma interrupção pode ser necessária, por exemplo, numa situação de emergência onde procedimentos referentes a esta situação devem ser adotados.

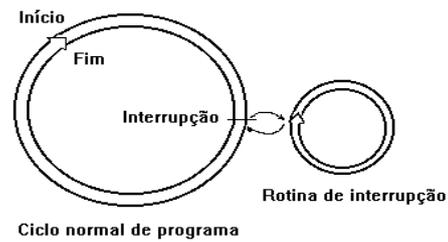


Figura 3 - Processamento por interrupção [SENAI 2002]

Processamento comandado por tempo: Da mesma forma que determinadas execuções não podem ser dependentes do ciclo normal de programa, algumas devem ser executados a certos intervalos de tempo, as vezes muito curto, na ordem de 10 ms.

Este tipo de processamento também pode ser encarado como um tipo de interrupção, porém ocorre a intervalos regulares de tempo dentro do ciclo normal de programa.

Processamento por evento: Este é processado em eventos específicos, tais como no retorno de energia, falha na bateria e estouro do tempo de supervisão do ciclo da UCP. Neste último, temos o chamado *Watch Dog Time* (WD), que normalmente ocorre como procedimento ao se detectar uma condição de estouro de tempo de ciclo da UCP, parando o processamento numa condição de falha e indicando ao operador através de sinal visual e às vezes sonoro.

2.7 ARQUITETURA DE MEMÓRIA DE UM CLP

A arquitetura de memória de um controlador programável pode ser constituída por diferentes tipos de memória.

A memória do computador é onde se armazenam os dados que devem ser manipulados pelo computador (chamada memória de dados) e também onde esta armazenado o programa do computador (memória de programa).

Aparentemente não existe uma diferença física entre as memórias de programa, apenas utilizam-se memórias fixas para armazenar dados fixos ou programas e memórias que podem ser alteradas pelo sistema para armazenar dados que podem variar de acordo com o programa. Existem diversos tipos de

memórias que podem ser utilizadas pelo computador: fita magnética, disco magnético e até memória de semicondutor em forma de circuito integrado.

As memórias a semicondutores podem ser divididas em dois grupos diferentes:

- Memória ROM (*Read Only Memory*) memória apenas de leitura.
- Memória RAM (*Random Access Memory*) memória de acesso aleatório.

A UCP efetuará a leitura das instruções contidas nesta área a fim de executar o programa do usuário, de acordo com os procedimentos predeterminados pelo sistema operacional.

As memórias destinadas ao usuário podem ser do tipo:

- ⇒ RAM
- ⇒ RAM/EPROM
- ⇒ RAM/EEPROM

<i>Tipo de Memória</i>	<i>Descrição</i>
RAM	A maioria do CLPs utiliza memórias RAM para armazenar o programa d usuário assim como os dados internos do sistema. Geralmente associada a baterias internas que evitarão a perda das informações em caso de queda da alimentação.
RAM/EPROM	O usuário desenvolve o programa e efetua testes em RAM. Uma vez checado o programa, este é transferido para EPROM.
RAM/EEPROM	Esta configuração de memória do usuário permite que, uma vez definido o programa, este seja copiado em EEPROM. Uma vez efetuada a cópia, o CLP poderá operar tanto em RAM como em EEPROM. Para qualquer modificação bastará um comando via software, e este tipo de memória será apagada e gravada eletricamente.

Tabela 1 - Memórias de um CLP [Altus 2004]

2.8 CONCEITOS DE PROGRAMAÇÃO DE CLPS.

2.8.1 Apresentação

Normalmente podemos programar um controlador através de um software que possibilita a sua apresentação ao usuário em quatro formas diferentes:

- Diagrama de contatos;
- Diagrama de blocos lógicos (lógica booleana);
- Lista de instruções;
- Linguagem corrente.

Alguns CLPs, possibilitam a apresentação do programa do usuário em uma ou mais formas.

2.8.2 Diagrama de contatos

Também conhecida como:

- Diagrama de relés;
- Diagrama escada;
- Diagrama "ladder".

Esta forma gráfica de apresentação está muito próxima a normalmente usada em diagramas elétricos. No exemplo abaixo para que aconteça a energização de S1, deve acontecer que E1 e E2 estejam em 1 ou E3 esteja em 1.



Figura 4 - Diagrama de contatos

2.8.3 Diagrama de blocos lógicos

Mesma linguagem utilizada em lógica digital, onde sua representação gráfica é feita através das chamadas portas lógicas.

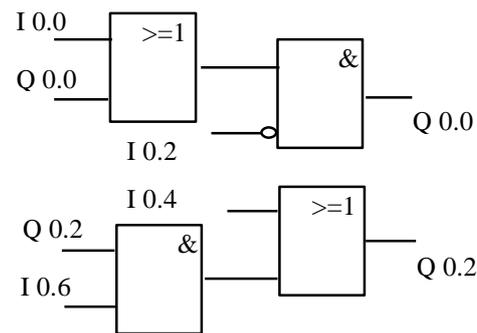


Figura 5 - Diagrama de blocos lógicos

2.8.4 Lista de instrução

Linguagem semelhante à utilizada na elaboração de programas para computadores.

Exemplo :

```
: A I 1.5
: A I 1.6
: O
: A I 1.4
: A I 1.3
:= Q 3.0
(I 1.5 . I 1.6) + (I 1.4 . I 1.3) = Q 3.0
```

2.8.5 Linguagem corrente

É semelhante ao basic, que é uma linguagem popular de programação, e uma linguagem de programação de alto nível. Comandos típicos podem ser "fechar válvula A" ou "desligar bomba B", "ligar motor", "desligar solenóide",

2.8.6 Documentação

A documentação é mais um recurso do editor de programa que de linguagem de programação. De qualquer forma, uma abordagem neste sentido torna-se cada vez mais importante, tendo em vista que um grande número de profissionais estão envolvidos no projeto de um sistema de automação que se utiliza de CLPs, desde sua concepção até a manutenção.

Quanto mais rica em comentários, melhor a documentação que normalmente se divide em vários níveis.

2.8.7 Conjunto de Instruções

É o conjunto de funções que definem o funcionamento e aplicações de um CLP.

Podem servir para mera substituição de comandos a relés:

- Funções Lógicas;
- Memorização;
- Temporização;
- Contagem.

Como também manipulação de variáveis analógicas:

- Movimentação de dados;
- Funções aritméticas.

Se funções complexas de algoritmos, comunicação de dados, interfaces homem-máquina, podem ser necessárias:

- Saltos controlados;
- Indexação de instruções;
- Conversão de dados;
- PID;
- sequenciadores;
- aritmética com ponto flutuante;
- etc.

2.8.8 Normalização

Existe a tendência de utilização de um padrão de linguagem de programação onde será possível a intercambiabilidade de programas entre modelos de CLPs e até de fabricantes diferentes.

Esta padronização está de acordo com a norma IEC 1131-3, na verdade este tipo de padronização é possível utilizando-se o conceito de linguagem de alto nível, onde através de um chamado compilador, pode-se adaptar um programa para a

linguagem de máquina de qualquer tipo de microprocessador, isto é, um programa padrão, pode servir tanto para o CLP de um fabricante A como de um fabricante B.

A norma IEC 1131-3 prevê três linguagens de programação e duas formas de apresentação. As linguagens são:

- *Ladder Diagram* - programação como esquemas de relés.
- *Boolean Blocks* - blocos lógico representando portas “E”, “OU”, “Negação”, “Ou exclusivo”, etc.
- *Structured Control Language (SCL)* - linguagem que vem substituir todas as linguagens declarativas tais como linguagem de instruções, BASIC estruturado e inglês estruturado. Esta linguagem é novidade no mercado internacional e é baseada no Pascal.

As formas de representação são :

- Programação convencional;
- *Sequential Function Chart (SFC)* - evolução do graphcet francês.

A grande vantagem de se ter o software normalizado é que em se conhecendo um, conhecem-se todos, economizando em treinamento e garantindo que, caso um fornecedor deixe o mercado, teremos condições de expandir ou repor equipamentos.

2.9 Programando o CLP Altus

Os CLP's Altus utilizam um software de programação chamado MasterTool composto de 4 elementos básicos: Módulos, Lógicas, Instruções e Operandos.

Um programa é composto de diversos módulos, permitindo uma melhor estruturação do programa de acordo com suas funções.

Um módulo de programa é dividido em lógicas de programação e o formato de uma lógica permite até 8 elementos em série e 4 em paralelo.

Instruções são utilizadas para executar determinadas tarefas no meio da leitura.

Os operandos identificam diversos tipos de variáveis e constantes utilizadas em um programa.

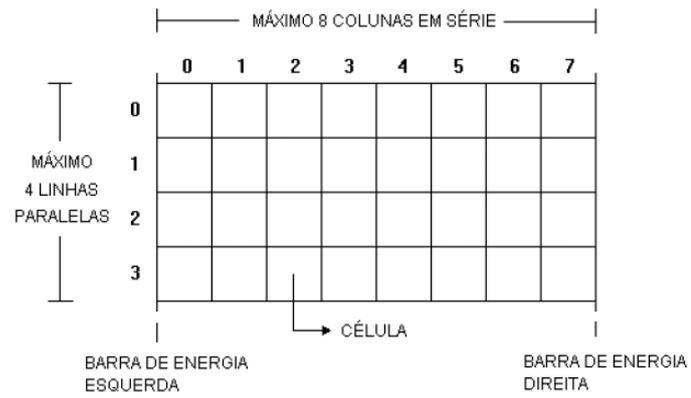


Figura 6 - Formato de uma lógica [Altus 2004]

Capítulo 3: CONFIGURAÇÃO DO CLP DA UHE CANOAS I

3.1 LEVANTAMENTO E PADRONIZAÇÃO DOS DADOS DE CAMPO

Quando se decide fazer um processo de digitalização de um processo industrial, existe a necessidade de se fazer um levantamento de dados para se saber quantos pontos serão supervisionados e quantos serão telecomandados, quais instrumentos você tem no campo, para que se possa padronizar tais sinais, estes sinais são divididos basicamente em entradas e saídas digitais e entradas e saídas analógicas, a quantidade e a forma de registro define as características de seu CLP.

Este levantamento apontou para a escolha do CLP Altus modelo Quark, por sua estrutura modular pode ser configurada de forma a atender a necessidade.

Decidimos dividir o processo em 6 partes da seguinte forma (UG-01, UG-02, UG-03, SA, SE, PCSE) e cada uma destas partes recebeu um CLP que foram integrados entre si através de uma rede de comunicação ao sistema de supervisão e controle.

Cada CLP foi integrado a um painel que denominamos de UAC (Unidade de Aquisição e Controle), neste painel foi montada a CLP na configuração adequada para a supervisão e controle dos equipamentos a ela integrada.

Nas UAC 01, 02, 03 que supervisionam os geradores 1, 2 e 3 respectivamente a configuração é a mesma devido a semelhança dos equipamentos, na UAC 04 que supervisiona o serviço auxiliar e vertedouros da usina e a UAC 05 que supervisiona a subestação de 88KV tem configurações de hardware diferentes em relação a quantidade de entradas e saídas analógicas e digitais, UAC PCSE utilizada como gateway entre a usina e o COG Duke em Chavantes e ONS no Rio de Janeiro, UAC PCSE também é responsável pelos cálculos que integram os dados dos CLP para o calculo hidráulico é a única que tem apenas cartões de comunicação e CPU's uma vez que ela aquisita os dados das outras UAC's através da rede de comunicação.

3.2 CONFIGURAÇÃO DO CLP

Conforme a figura 7 abaixo, para configuração do CLP é definido a CPU para utilização, estamos utilizando a AL 2004, após é alocado a quantidade de operandos, tabelas, tempo máximo de execução do programa, tempo de execução, barramento e configuração de rede.

C-CAN1G1.000

Modelo de CP: **AL-2004**

Operandos

Total de operandos		Operandos Retentivos	
Memória:	8064 %M0000 a %M8063	0	Não Declarado
Decimal:	64 %D0000 a %D0063	0	Não Declarado
Real:	0 Não Declarado	0	Não Declarado
Inteiro:	0 Não Declarado	0	Não Declarado
Saída:	16 %S0056 a %S0071	0	Não Declarado
Auxiliar:	512 %A0000 a %A0511	0	Não Declarado

Tabelas

Memórias	65	posições...	Reais	0	posições...
Decimais	4	posições...	Inteiro	0	posições...

Bytes livres: 28900

Tempo Máximo de Execução de Programa: 800 ms

Período acionamento E018: 50 ms

Barramento...
Parâmetros...

Redes

COM 1... ALNET II...
Ethernet... Sincronismo...
COM 2... COM 3...
PROFIBUS...

Figura 7 - Configuração do CLP [Altus 2007]

Na figura 8, a configuração do barramento pode ser configurada a troca a quente dos cartões também é declarado o octeto da primeira saída digital. Ao selecionar o botão do Barramento 0, é exibida a tabela Figura 9 para configuração do barramento 0. Cada linha da tabela corresponde a uma posição no barramento onde são declarados o tipo do cartão e seu endereçamento.

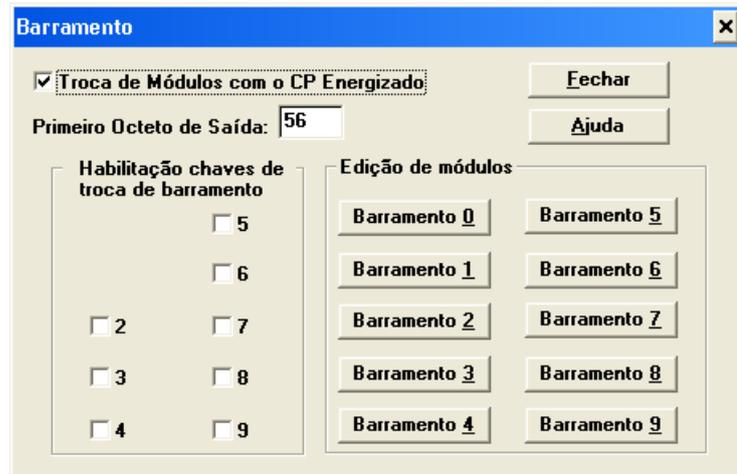


Figura 8 - Configuração dos barramentos [Altus 2007]

Na Figura 9, está declarada na posição 00 um cartão AL-3411 responsável pela comunicação entre os barramentos da CLP. Em seguida na posição 01, está o cartão AL-3414 responsável pela comunicação da rede de CLP's e o SSC. Nas posições 02, 03, 04, 09 e 10 estão os cartões AL-3131 utilizado para registro das entradas digitais de evento rápido chamado de SOE. Nas posições 05, 06, 07 e 08 foram alocados os cartões AL-3203 de saída digital.

Posição	Módulo	Entradas	Saídas	Endereço
00	AL-3411			%R0000
01	AL-3414			%R0008
02	AL-3131	%E0000..%E0003		%R0016
03	AL-3131	%E0004..%E0007		%R0024
04	AL-3131	%E0008..%E0011		%R0032
05	AL-3203		%S0056..%S0059	%R0040
06	AL-3203		%S0060..%S0063	%R0048
07	AL-3203		%S0064..%S0067	%R0056
08	AL-3203		%S0068..%S0071	%R0064
09	AL-3131	%E0012..%E0015		%R0072
10	AL-3131	%E0016..%E0019		%R0080
11				%R0088
12				%R0096
13				%R0104
14				%R0112
15				%R0120

Primeiro octeto de saída: 56

<< Anterior Próximo >> Adicionar... Remover Fechar

Figura 9 - Configuração do barramento 0 [Altus 2007]

Na Figura 10 aparece a configuração do barramento 2. Está declarada na posição 00, 01, 02 e 03 4 cartões AL-1116, responsável pela leitura dos sinais analógicos do campo padronizados através de transdutores com valor de referência de corrente entre 4 a 20 mA. Nas posições 04 a 11 estão declarados cartões do tipo AL-1117 responsável pela leitura dos valores de temperatura diretamente aos sensores de temperatura PT 100.

Posição	PA	Módulo	Entradas	Saídas	Endereço
00	00	AL-1116			%R0256
01	01	AL-1116			%R0258
02	02	AL-1116			%R0260
03	03	AL-1116			%R0262
04	04	AL-1117			%R0264
05	05	AL-1117			%R0266
06	06	AL-1117			%R0268
07	07	AL-1117			%R0270
08	10	AL-1117			%R0272
09	11	AL-1117			%R0274
10	12	AL-1117			%R0276
11	13	AL-1117			%R0278
12	14				%R0280
13	15				%R0282
14	16				%R0284
15	17				%R0286

Primeiro octeto de saída: 56

<< Anterior Próximo >> Adicionar... Remover Fechar

Figura 10 - Configuração do barramento 2 [Altus 2007]

Na Figura 11 aparece a configuração do barramento 3. Na posição 00 a 08 estão declarados 9 cartões AL-1131 responsável pela leitura dos sinais digitais do campo. Nas posições 11 e 12 estão declarados cartões do tipo AL-1117 responsável pela leitura dos valores de temperatura diretamente aos sensores de temperatura PT 100. Na posição 13 está declarado um cartão AL-1222 responsável pelos 4 sinais de saída analógica de 4 a 20mA.

Posição	PA	Módulo	Entradas	Saídas	Endereço
00	00	AL-1131	%E0020..%E0023		%R0288
01	01	AL-1131	%E0024..%E0027		%R0290
02	02	AL-1131	%E0028..%E0031		%R0292
03	03	AL-1131	%E0032..%E0035		%R0294
04	04	AL-1131	%E0036..%E0039		%R0296
05	05	AL-1131	%E0040..%E0043		%R0298
06	06	AL-1131	%E0044..%E0047		%R0300
07	07	AL-1131	%E0048..%E0051		%R0302
08	10	AL-1131	%E0052..%E0055		%R0304
09	11	AL-1117			%R0306
10	12	AL-1117			%R0308
11	13	AL-1222			%R0310
12	14				%R0312
13	15				%R0314
14	16				%R0316
15	17				%R0318

Primeiro octeto de saída: 56

<< Anterior Próximo >> Adicionar... Remover Fechar

Figura 11 - Configuração do barramento 3 [Altus 2007]

3.3 SOFTWARE DE PROGRAMAÇÃO DO CLP ALTUS

O CLP Altus é programado através do software MasterTool. O controlador programável é um equipamento que realiza controle sob o comando de um programa aplicativo escrito em linguagem de relés e blocos. Compõe-se de uma unidade central de processamento (UCP), fonte de alimentação e estrutura de E/S.

MasterTool é utilizado para realizar a edição de programas para os CPs ALTUS, verificar programas já enviados, modificar programas prontos ou para examinar o estado dinâmico das variáveis do sistema de controle. É possível acompanhar todos os passos do programa aplicativo em tempo real, forçando a ocorrência de ações específicas. Se desejável, o microcomputador pode operar permanentemente conectado ao controlador programável.

3.3.1 Uma Visão Geral do MasterTool Programming

O MasterTool Programming é uma aplicação executada no ambiente operacional Windows possuindo todas as facilidades e padronizações oferecidas por este ambiente.

Sua operação, como as demais aplicações Windows, é orientada a comandos de menus e caixas de diálogos que permitem a realização das tarefas e escolha das opções.

Com o intuito de facilitar os procedimentos mais utilizados no MasterTool Programming, existem diversos atalhos onde o pressionamento de duas ou mais teclas simultaneamente realizam um comando, bem como uma Barra de Ferramentas onde basta clicar no botão desejado para a rápida realização do comando.

O MasterTool Programming possui ainda janelas de edição específicas para Módulo de Configuração, Módulo de Configuração de Rede e Módulos de Programa. Estas janelas permitem uma edição eficiente e uma visualização adequada dos diversos componentes de cada módulo específico.

Quando o MasterTool Programming é inicializado, exibe a tela inicial mostrada na figura 12 permitindo que sejam executados todos os comandos disponíveis.

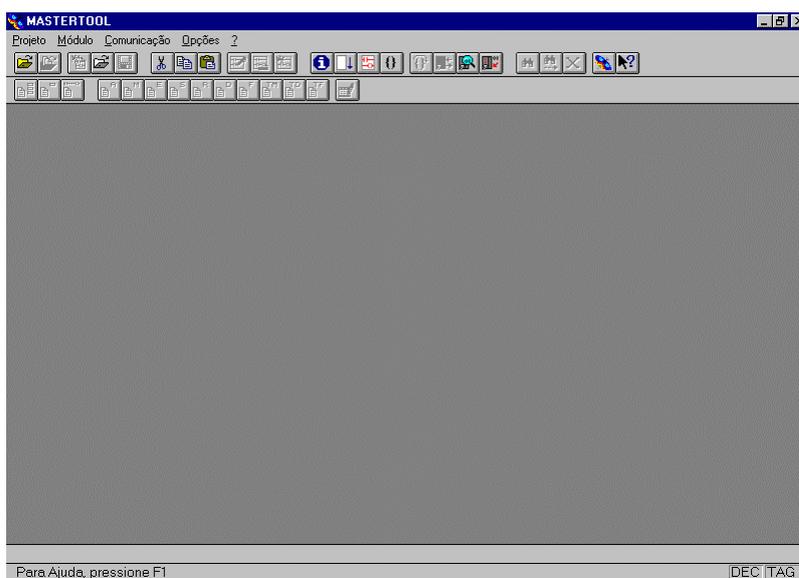


Figura 12 - Tela inicial do MASTERTOOL [Altus 2007]

As instruções ficam habilitadas ou desabilitadas na barra de ferramentas de acordo com as restrições da célula onde está posicionado o cursor de lógica.

3.3.2 Utilizando um projeto

Um projeto é um conjunto de módulos que são utilizados para a realização das tarefas de um CP. O nome do arquivo de projeto pode possuir até 6 caracteres de comprimento e é finalizado com a extensão .MTL.

Um projeto de programação é utilizado para reunir todos os módulos necessários à execução das tarefas de um CP.

Neste tipo de projeto é obrigatória a existência de um módulo de configuração e de um módulo de execução E001. Quando se cria um novo projeto de programação, o módulo de configuração é automaticamente criado e visualizado na janela de edição de módulo C.

O nome do módulo de configuração é formado pelo tipo do módulo (C-), pelo nome do projeto e pela extensão .000.

O nome do módulo de execução principal é formado pelo tipo do módulo (E-), por um nome de até 6 caracteres (sugere-se o nome do projeto) e pela extensão .001. Este módulo é responsável pela chamada dos módulos de procedimentos

É muito importante que se divida a programação em módulos de procedimentos. Cada módulo deve ser responsável por uma funcionalidade dentro do programa. Além dos módulos de configuração, será configurado a disposição dos cartões e o endereçamento das funções e o também o módulo de partida com extensão .000, que só roda ao ligar o CLP para aquisição dos sinais de campo.



Figura 16 - Caixa de diálogo para criação de um novo módulo [Altus 2007]

Para a utilização de um módulo do projeto é necessário abrir o módulo, através do menu projeto editar. Após este comando será aberta a tela abaixo representada pela figura 17. Ela traz o nome do projeto, o diretório onde está armazenado o programa, os módulos existentes em ordem alfabética, a quantidade de módulos existentes. Ao selecionar o módulo isto habilita as funções de abrir ou retirar módulo.

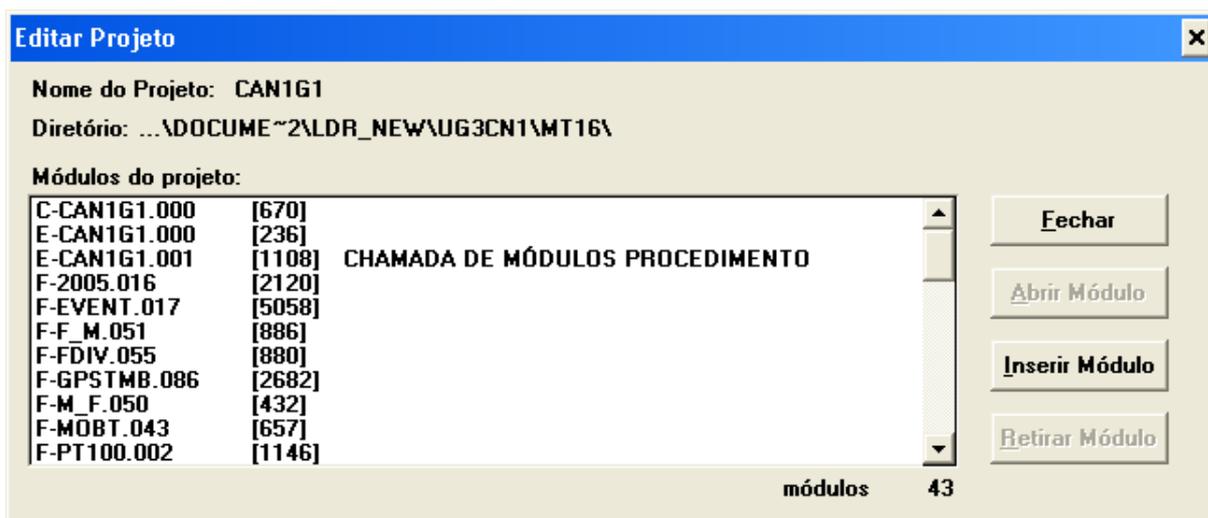


Figura 17 - Caixa de diálogo para editar projeto [Altus 2007]

3.3.3 Configurando o endereço de rede

No módulo C pode-se configurar a comunicação ethernet do CLP para a comunicação entre os CLP's e sistema de supervisão e após acessar a tela que aparece na figura 18, deve-se selecionar o botão da posição 1 do cartão de comunicação AL-3414. o qual abrirá a caixa de diálogo de configuração de módulo ethernet, onde será colocado o endereço IP, a máscara da sub-rede e gateway default, conforme aparece na figura 19.

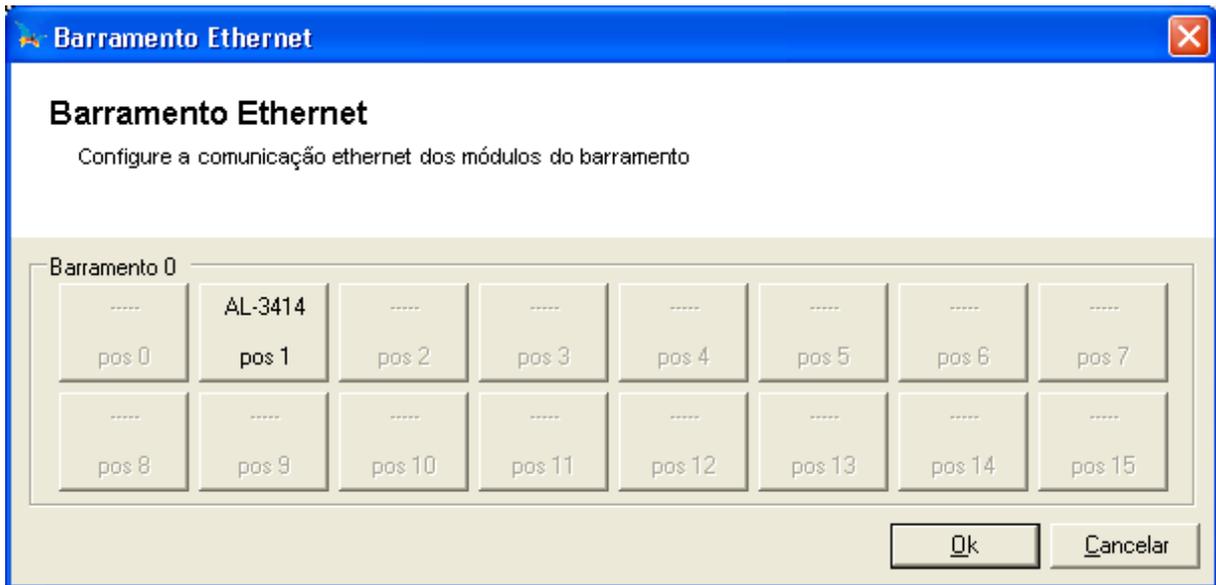


Figura 18 - Barramento Ethernet [Altus 2007]

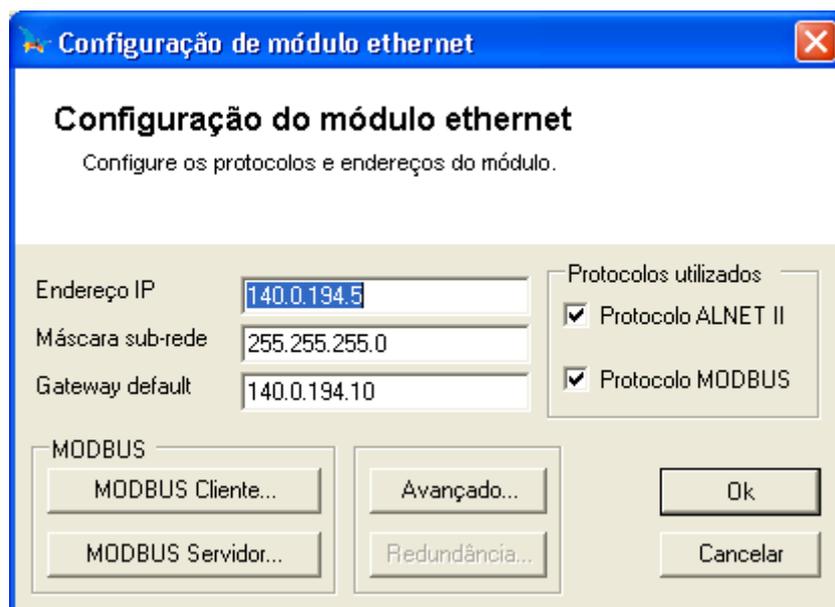


Figura 19 – Configuração do módulo Ethernet [Altus 2007]

Capítulo 4: PROGRAMAÇÃO DO CLP ALTUS

4.1 ELEMENTOS DE PROGRAMAÇÃO MASTERTOOL

Os controladores programáveis surgiram para substituir painéis de controle a relés. O Mastertool é uma linguagem de relés, bobinas e blocos, portanto quando se pode programar em uma linguagem que se assemelha aos antigos esquemas elétricos isto facilita a compreensão dos técnicos e engenheiros que analisam o programa.

O Mastertool é apresentado como uma linguagem de diagrama de relés, e o programa aplicativo é composto por 4 elementos básicos:

- Módulos
- Lógicas
- Instruções
- Operandos

Um programa aplicativo é composto por diversos módulos, permitindo uma melhor estruturação das rotinas de acordo com as suas funções. Os módulos são programados em linguagem de relés, seguindo a tendência mundial de normalização nesta área.

Um módulo de programa aplicativo é dividido em lógicas de programação. O formato de uma lógica de programa aplicativo utilizado nos CPs Altus permite até oito elementos em série e até quatro caminhos em paralelo.

As instruções são utilizadas para executar determinadas tarefas por meio de leituras e/ou alterações do valor dos operandos.

Os operandos identificam diversos tipos de variáveis e constantes utilizadas na elaboração de um programa aplicativo, podendo ter seu valor modificado de acordo com a programação realizada. Como exemplo de variáveis pode-se citar pontos de E/S e memórias contadoras.

4.2 ORGANIZAÇÃO DE MEMÓRIA DO CLP ALTUS

O programa aplicativo é armazenado no controlador em uma área de memória dividida em bancos. Podem existir um ou mais bancos de memória RAM e EPROM, o modelo do CPU utilizado em Canoas I AL 2004 possui 16 bancos de 64

Kbytes de memória Flash e 128 Kbytes de memória RAM. O nome EPROM refere-se indistintamente à memória para gravação permanente do programa aplicativo utilizada no CLP, seja do tipo cartucho de EPROM ou Flash EPROM

Na janela do diretório de módulos do CP (opções Comunicação, Módulos) é possível visualizar a quantidade de memória livre em cada banco, para cada tipo existente no controlador. Abaixo segue o exemplo do CLP da UG de Canoas I.

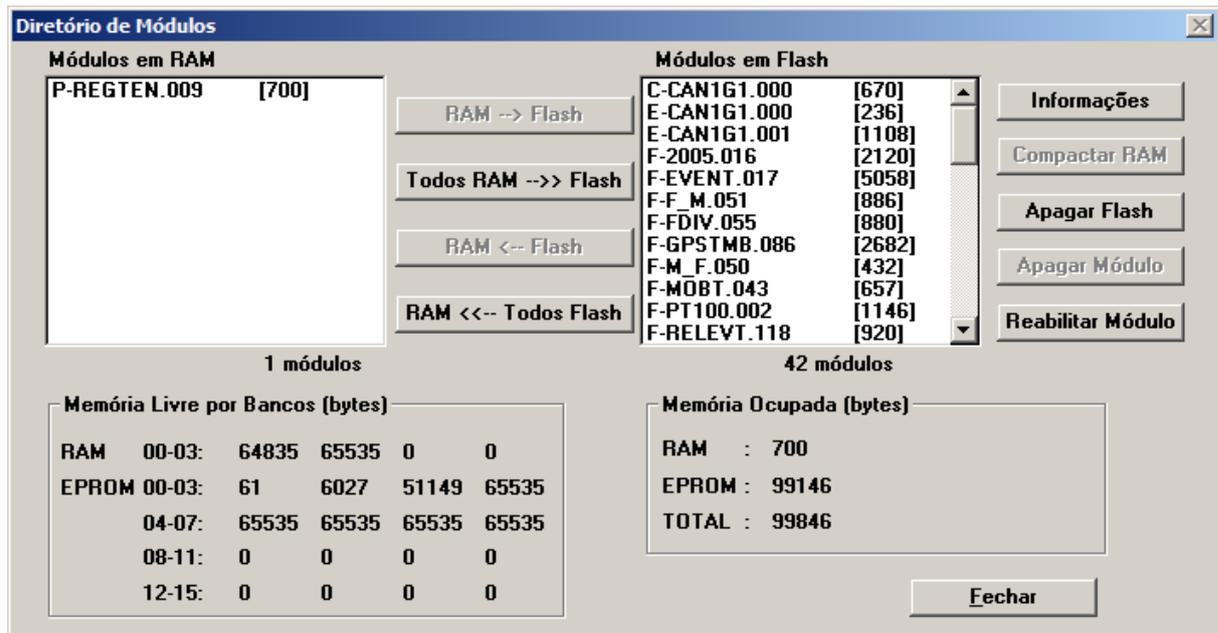


Figura 20 - Estado da memória do CLP da UG-01 CNO I [Altus 2007]

Lógicas chama-se de lógica a matriz de programação formada por 32 células (elementos da matriz) dispostas em 4 linhas (0 a 3) e 8 colunas (0 a 7). Em cada uma das células podem ser colocadas instruções, podendo-se programar até 32 instruções em uma mesma lógica.

Cada lógica presente no programa simula um pequeno trecho de um diagrama de relés real.

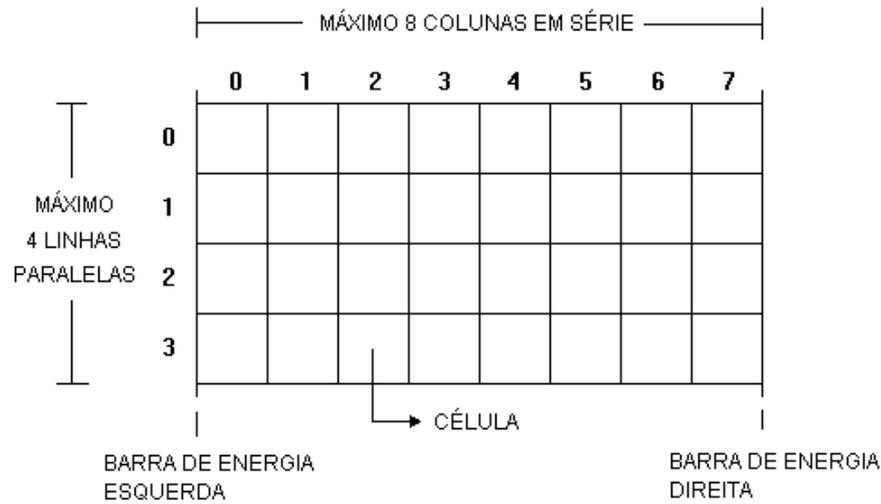


Figura 21 - Formato de uma lógica [Altus 2004]

	0	1	2	3	4	5	6	7
0	1	5	9	13	17	21	25	29
1	2	6	10	14	18	22	26	30
2	3	7	11	15	19	23	27	31
3	4	8	12	16	20	24	28	32

Figura 22 - Ordem de processamento de uma lógica [Altus 2004]

4.3 FORMATAÇÃO DOS OPERANDOS DO MASTERTOOL

4.3.1 Operandos Simples

Os operandos simples são utilizados como variáveis de armazenamento de valores no programa aplicativo. Conforme a instrução que os utilizam, eles podem ser referenciados na sua totalidade ou em uma subdivisão (uma parte do operando). As subdivisões de operandos podem ser palavra, octeto, nibble ou ponto.

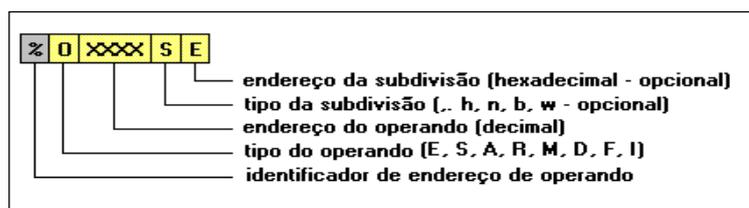


Figura 23 - Formato de um operando simples [Altus 2004]

Tipos dos operandos:

%E - entrada

%S - saída

%A - auxiliar

%R - endereço no barramento

%M - memória

%D - decimal

%F - real

%I - inteiro

Tipo da subdivisão:

. - ponto da palavra baixa (1 ponto)

h - ponto da palavra alta (1 ponto)

n - nibble (4 pontos)

b - octeto (8 pontos)

w - palavra (16 pontos)

Exemplos:

%E0002.3 - ponto 3 do operando de entrada 2

%S0004.7 - ponto 7 do operando de saída 4

%A0039n1 - nibble 1 do operando auxiliar 39

%A0045 - octeto auxiliar 45

%M0205 - operando memória 205

%M0205b0 - octeto 0 da memória 205

%D0029 - operando decimal 29

%D0034w1 - palavra 1 do decimal 34

%F0001 - operando real 1
 %I0002 - operando inteiro 2

4.3.2 Operandos Constante

Os operandos constantes são utilizados para a definição de valores fixos durante a edição do programa aplicativo.

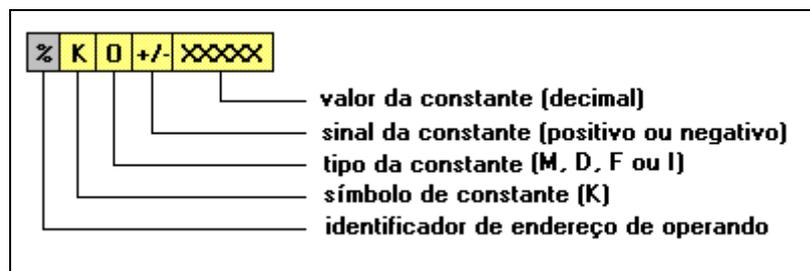


Figura 24 - Formato geral de um operando constante [Altus 2004]

Tipo da constante:

%M memória

%D decimal

%F real

%I inteiro

Exemplos:

%KM+05172 constante memória positiva

%KD-0974231 constante decimal negativa

%KF+0153.78 constante real positiva

%KI+06000 constante inteiro positiva

4.3.2 Operandos Tabela

Tabelas de operandos são conjuntos de operandos simples, constituindo arranjos unidimensionais. São utilizados índices para determinar a posição da tabela que se deseja ler ou alterar. São possíveis tabelas de operandos memória ou decimal.

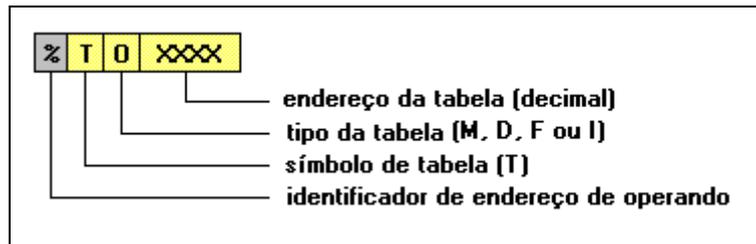


Figura 25 - Formato geral de um operando tabela [Altus 2004]

Tipo da tabela:

%TM memória

%TD decimal

%TF real

%TI inteiro

Exemplos:

%TM0026 tabela memória 26

%TD0015 tabela decimal 15

%TF0069 tabela real 69

%TI0010 tabela inteiro 10

4.4 PRINCIPAIS INSTRUÇÕES DO MASTERTOOL

4.4.1 Instrução < - Menor

A instrução menor realiza a comparação do operando especificado com o valor previamente carregado no registrador interno com a instrução CAR (Carrega

Operando), fornecendo o resultado da comparação em suas saídas. Caso algum acesso indireto seja inválido, a saída é desacionada.

A instrução menor energiza a sua saída se o valor contido no registrador interno previamente carregado com a instrução CAR for menor que o valor contido no operando OPER fornecido nesta instrução.

Se os operandos a serem comparados são do mesmo tipo, são comparados conforme o seu formato de armazenamento (considerando o seu sinal). Se não são do mesmo tipo, são comparados ponto a ponto (como valores binários sem sinal). Se algum dos operandos diferentes for do tipo real, o operando de menor precisão é convertido para real e após é efetuada a mesma comparação ponto a ponto.

Não pode ser feita a comparação entre operandos decimais e operandos flutuante.

4.4.2 Instrução > - Maior

A instrução maior realiza a comparação do operando especificado com o valor previamente carregado no registrador interno com a instrução CAR (Carrega Operando), fornecendo o resultado da comparação em suas saídas. Caso algum acesso indireto seja inválido, a saída é desacionada.

A instrução maior energiza a sua saída se o valor contido no registrador interno previamente carregado com a instrução CAR for maior que o valor contido no operando OPER fornecido nesta instrução.

Se os operandos a serem comparados são do mesmo tipo, são comparados conforme o seu formato de armazenamento (considerando o seu sinal). Se não são do mesmo tipo, são comparados ponto a ponto (como valores binários sem sinal). Se algum dos operandos diferentes for do tipo real, o operando de menor precisão é convertido para real e após é efetuada a mesma comparação ponto a ponto.

Não pode ser feita a comparação entre operandos decimais e operandos flutuante.

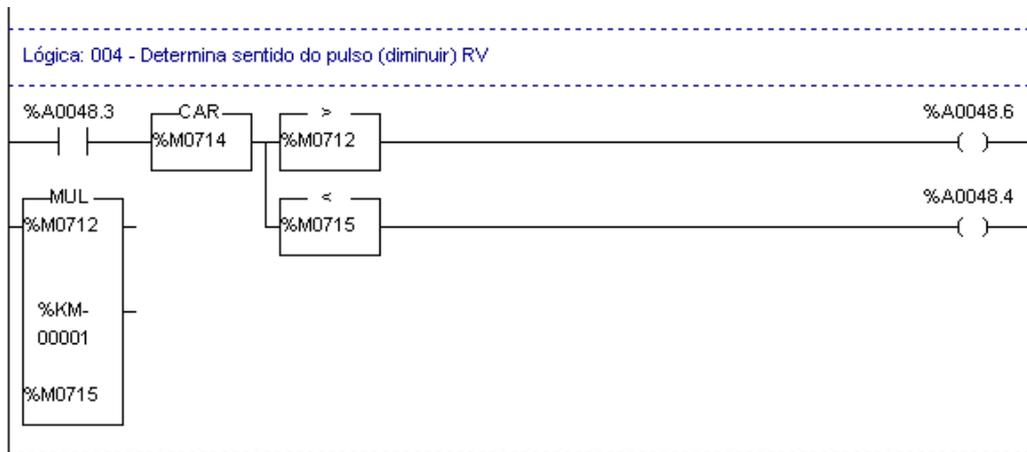


Figura 26 - Na lógica 4 do módulo P-REGTEN.009 [Altus 2007]

Na lógica 4 do módulo P-REGTEN.009 o valor carregado na %M0714 através de um comando do SSC é comparado com as memórias %M0712 e %M0715 se maior executa o comando %A0048.6 se menor executa o comando %A0048.4.

4.4.3 Instrução = - Igual

A instrução igual realiza a comparação do operando especificado com o valor previamente carregado no registrador interno com a instrução CAR (Carrega Operando), fornecendo o resultado da comparação em suas saídas. Caso algum acesso indireto seja inválido, a saída é desacionada.

A instrução igual energiza a sua saída se o valor do seu operando for igual ao valor do operando presente na última instrução CAR ativa.

Se os operandos a serem comparados são do mesmo tipo, são comparados conforme o seu formato de armazenamento (considerando o seu sinal). Se não são do mesmo tipo, são comparados ponto a ponto (como valores binários sem sinal). Se algum dos operandos diferentes for do tipo real, o operando de menor precisão é convertido para real e após é efetuada a mesma comparação ponto a ponto.

Não pode ser feita a comparação entre operandos decimais e operandos flutuante.

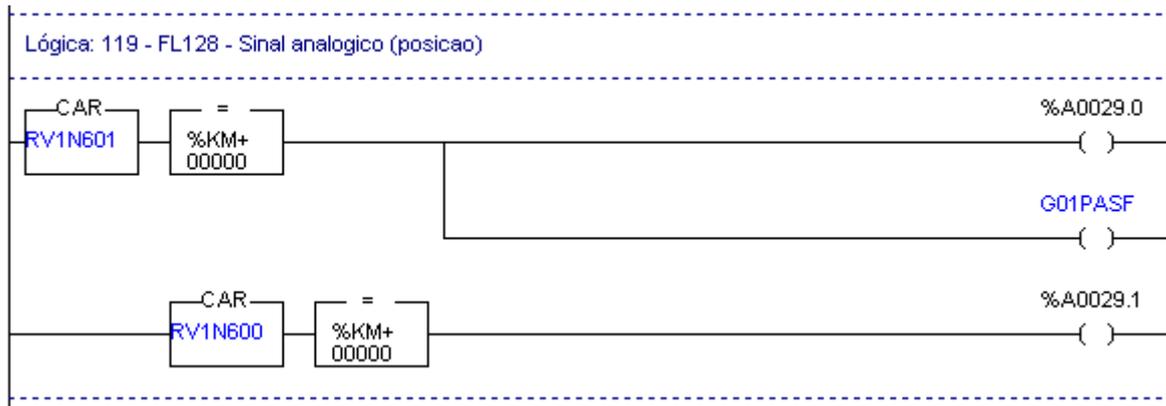


Figura 27 - Lógica 119 do módulo P-PART.005. [Altus 2007]

Na lógica 119 do módulo P-PART.005 na primeira parte o valor analógico recebido do transdutor de posição da roda é comparado com o valor “0” para sinalização de pás da roda fechada e na parte inferior da lógica o valor de abertura do distribuidor é comparado com “0” para fazer sinalização a sinalização de distribuidor fechado.

4.4.4 Instrução A/D - Conversão Analógico/Digital

Esta instrução converte os valores lidos de um módulo de entrada analógica para valores numéricos armazenados em operandos.

É possível efetuar a leitura de 1 ou 8 canais alterando-se apenas a especificação do primeiro operando, o qual indica o endereço no barramento ocupado pelo módulo A/D. Este módulo deve estar especificado na declaração do barramento.

O endereço a ser programado em OPER1 pode ser obtido diretamente através da coluna Endereço na Declaração dos Barramentos. Os valores convertidos são colocados em operandos do tipo memória, definidos em OPER2.

A conversão é realizada apenas se a entrada habilita estiver energizada. NA UAC utilizadas em Canoas I o cartão é o AL-1116 (12 bits) os sinais de entrada são padronizados em corrente de 4 a 20 mA e os valores de saída são convertidos de 0000 a 4095 bits.

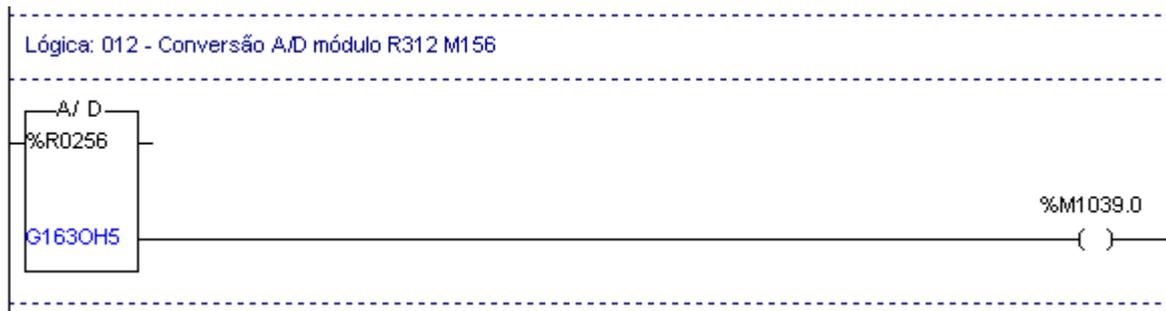


Figura 28 - Lógica 012 do módulo P-ANALOG.012. [Altus 2007]

Na lógica 012 do módulo P-ANALOG.012 o cartão de leitura de entradas analógicas é endereçado %R0256 e a primeira posição de leitura o TAG **G1630H5** abre uma sequência de 8 leituras de %M0120 a %M0127, conforme exemplo apresentado na figura 41, caso ocorra um erro na conversão a memória %M1039.0 é ativada sinalizando erro de conversão analógica digital.

4.4.5 Instrução PLS - Relé de Pulso

A instrução relé de pulso gera um pulso de uma varredura em sua saída, ou seja, permanece energizada durante uma varredura do programa aplicativo quando o estado da sua entrada passar de desenergizado para energizado.

O relé auxiliar declarado serve como memorizador, evitando limitações quanto ao número de instruções de pulso presentes no programa aplicativo.

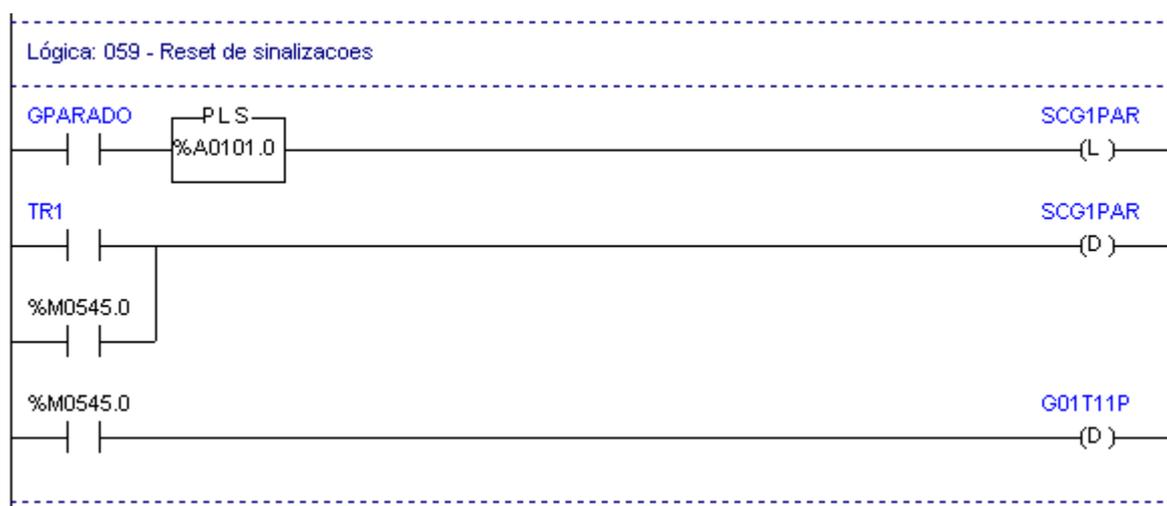


Figura 29 - Lógica 059 do módulo P-PART.005. [Altus 2007]

Na lógica 059 do módulo P-PART.005 será feito um reset das sinalizações, na primeira linha da lógica após a sinalização de UG parada através do TAG **GPARADO** um pulso será gerado para ligar a bobina de liga **SCG1PAR** que sinaliza SAC “Sequência automática de comando” de parada concluída, esta bobina será desativado apenas quando receber um telecomando de reset de SAC através da %M0545.0 ou a durante uma partida da UG quando será ativada a Transição 1 “TR1”.

4.4.6 RNA - Contato normalmente aberto

Esta instrução reflete, logicamente, o comportamento real de um contato elétrico de um relé no programa aplicativo.

O contato normalmente aberto fecha conforme o estado do seu operando associado. Caso o ponto do operando esteja no estado lógico 1 ou 0, o contato normalmente aberto está fechado ou aberto, respectivamente.

Quando um contato está fechado, a instrução transmite o estado lógico da sua entrada para a sua saída. Se estiver aberto, o valor da entrada não é colocado na saída.

4.4.7 RNF - Contato Normalmente Fechado

Esta instrução reflete, logicamente, o comportamento real de um contato elétrico de um relé no programa aplicativo.

O contato normalmente fechado possui comportamento oposto ao normalmente aberto. Caso o ponto do operando associado esteja no estado lógico 1 ou 0, o contato normalmente fechado está aberto ou fechado, respectivamente.

Quando um contato está fechado, a instrução transmite o estado lógico da sua entrada para a sua saída. Se estiver aberto, o valor da entrada não é colocado na saída.

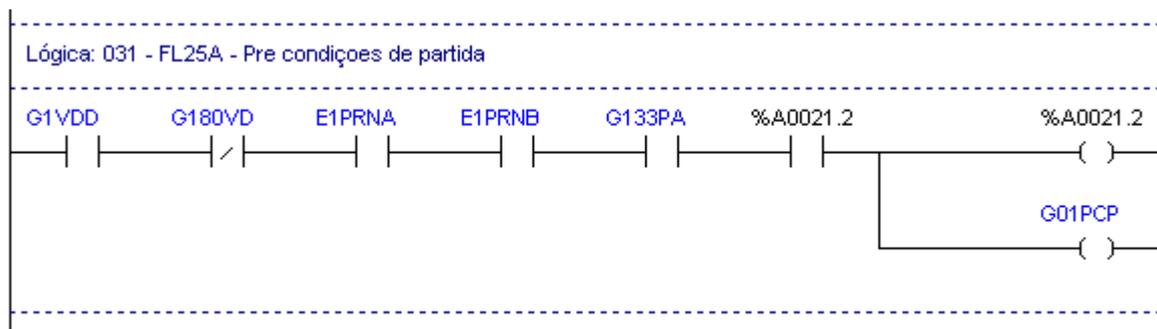


Figura 30 - Lógica 031 do módulo P-PART.005. [Altus 2007]

Na lógica 031 do módulo P-PART.005 representa a ultima lógica de uma série de 11 lógicas que faz a verificação das pré condições de partida da UG a cada lógica que é lida pelo programa a bobina **%A0021.2** é ativada e nesta lógica mostra a utilização dos relés RNA e RNF nesta lógica é verificada através do **G1VDD** se a vedação de parada esta desaplicada deve estar em “1”, se a vazão de água de vedação não esta baixa através do **G180VD** deve ser “0” e em seguida se a proteção sistema A e B esta em ordem através dos contatos **E1PRNA** e **E1PRNB** e se a escotilha do gerador esta fechada através de **G133PA** e se todas as pré condições anteriores estão satisfeitas o auxiliar **%A0021.2** estará em “1” será ativado o TAG **G01PCP** assim será permitido a partida da UG.

4.4.8 TEE - Temporizador na Energização

Esta instrução realiza contagens de tempo com a energização das suas entradas de acionamento.

A instrução TEE possui dois operandos. O primeiro OPER1 especifica a memória acumuladora da contagem de tempo. O segundo operando OPER2 indica o tempo máximo a ser acumulado. A contagem de tempo é realizada em décimos de segundos, ou seja, cada unidade incrementada em OPER1 corresponde a 0,1 segundo.

Enquanto as entradas libera e ativa estiverem simultaneamente energizadas, o operando OPER1 é incrementado a cada décimo de segundo. Quando OPER1 for maior ou igual a OPER2, a saída Q é energizada e -Q desenergizada, permanecendo OPER1 com o mesmo valor de OPER2.

Desacionando-se a entrada libera, há a interrupção na contagem do tempo, permanecendo OPER1 com o mesmo valor. Desacionando-se a entrada ativa, o valor em OPER1 é zerado.

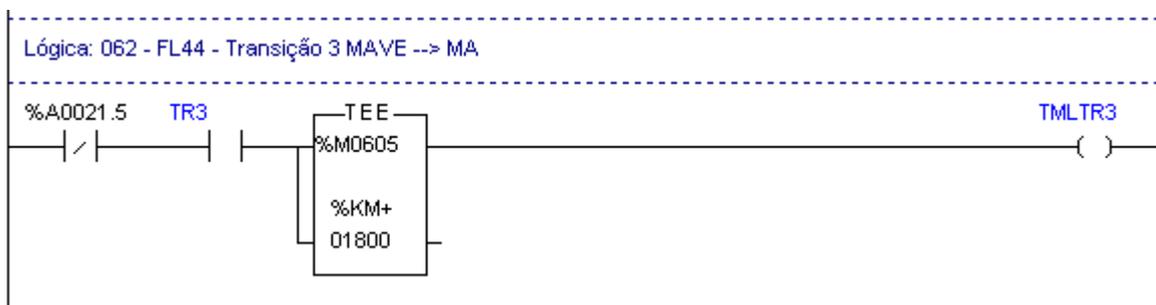


Figura 31 - Lógica 062 do módulo P-PART.005. [Altus 2007]

Na lógica 62 do módulo P-PART.005 após pedido de sincronismo ativado TR3 a instrução TEE será ativada para uma contagem 180 segundos, caso a TR3 não seja desativada nestes 180 segundos a bobina TMLTR3 será acionada fazendo a sinalização de tempo excedido para sincronismo.

4.4.9 Bobinas

As instruções bobinas simples , liga e desliga modificam o estado lógico do operando na memória imagem do controlador programável, conforme o estado da linha de acionamento das mesmas.

Estas instruções somente pode ser posicionada na coluna 7 da lógica.

4.4.9.1 BOB - Bobina Simples

A instrução bobina simples liga ou desliga o ponto do operando conforme a linha de acionamento.

Se a linha de acionamento estiver energizada o ponto do operando associado é ligado, caso contrário o ponto do operando associado é desligado.

4.4.9.2 BBL - Bobina Liga

A instrução bobina liga, liga o ponto do operando associado quando a linha está energizada ("set") e o mantém ligado mesmo após a desenergização da linha.

4.4.9.2 BBD - Bobina Desliga

A instrução bobina desliga modifica o estado lógico do operando na memória imagem do controlador programável, conforme o estado da sua linha de acionamento.

A instrução bobina desliga, desliga o ponto do operando associado quando a linha está energizada ("reset").

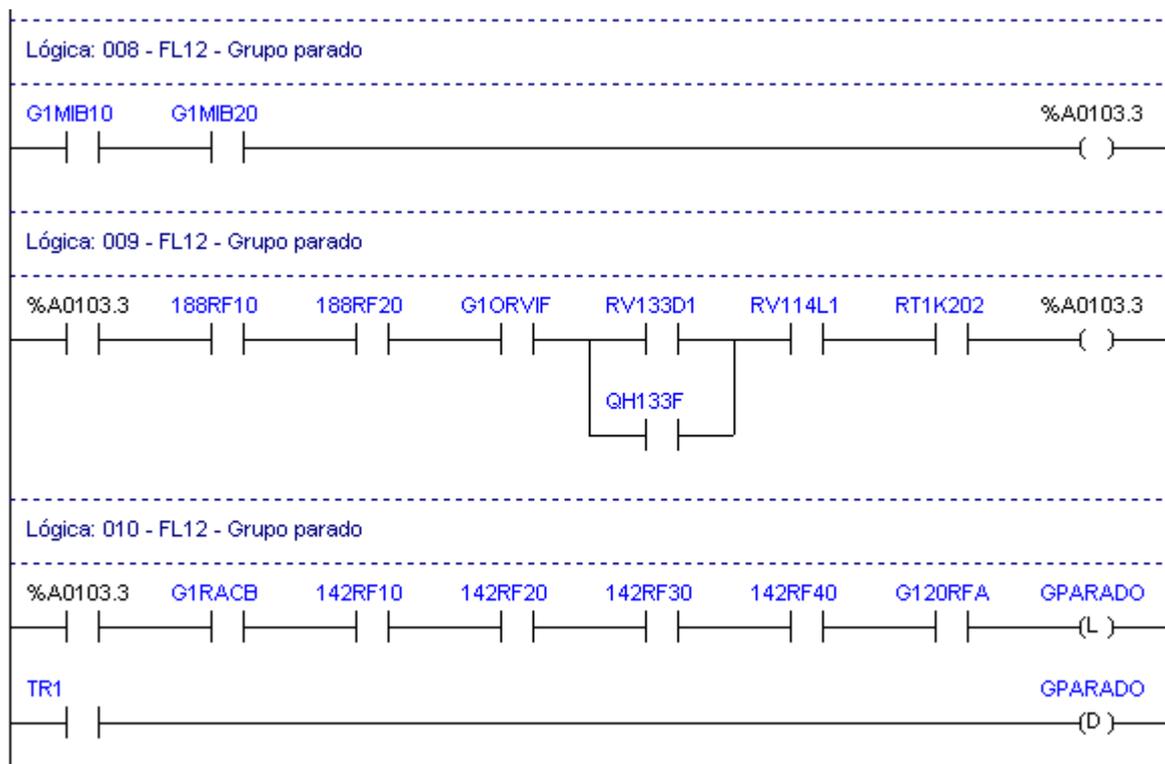


Figura 32 - Lógica 008,009 e 010 do módulo P-PART.005. [Altus 2007]

Na figura 32 temos exemplo de utilização das bobinas simples na série que verifica se o estado da UG é o estado **GPARADO** até que na lógica 10 a bobina de liga é acionada na linha 1 e a memória **%M0350.0** que esta representada pelo TAG **GPARADO** é energizado permanecendo em "1" até ser desligado quando da habilitação da **TR1** durante uma nova partida da UG.

Capítulo 5: COMO ANALISAR UM PROGRAMA NO MASTERTOOL

5.1 PRINCIPAIS FERRAMENTAS MENU PROJETO

Neste capítulo será descrito as funcionalidades de acesso, comunicação e configuração para acesso a CLP através do MASTERTOOL e um computador pode se acessar os módulos do programa, supervisionar o funcionamento da CLP, cartões, entradas e saídas, digitais ou analógicas.

O Menu Projeto é utilizado para acessar um novo projeto, abrir projetos gravados em dispositivo de memória, imprimir e configurar a impressora.

5.1.1 Menu Projeto Abrir

O botão do menu projeto abrir é utilizado para abrir um projeto já existente, como ao fechar o *Mastertool* este fica memorizado o último projeto aberto a caixa de diálogo a ser aberta terá aparência da figura 33 com o caminho do projeto aberto.

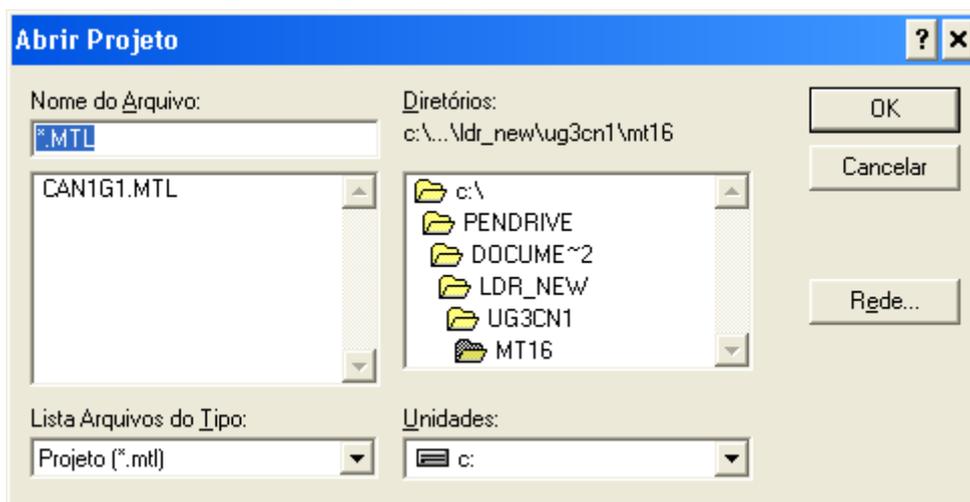


Figura 33 - Caixa de diálogo para Abrir Projeto. [Altus 2007]

5.1.2 Menu Projeto Editar

O comando Editar do Menu Projeto deve ser utilizado para abrir, incluir ou excluir módulos do projeto corrente, também pode ser utilizado para se verificar qual projeto está aberto. Após aberto terá a aparência da figura 34

onde é exibido o nome do projeto corrente, o diretório do projeto corrente, nesta caixa são exibidos todos os módulos integrantes do projeto e permite a seleção de um módulo para ser retirado ou aberto pelos botões "Retirar Módulo" e "Abrir Módulo". O botão "Abrir Módulo" é utilizado para abrir um módulo do projeto para análise.

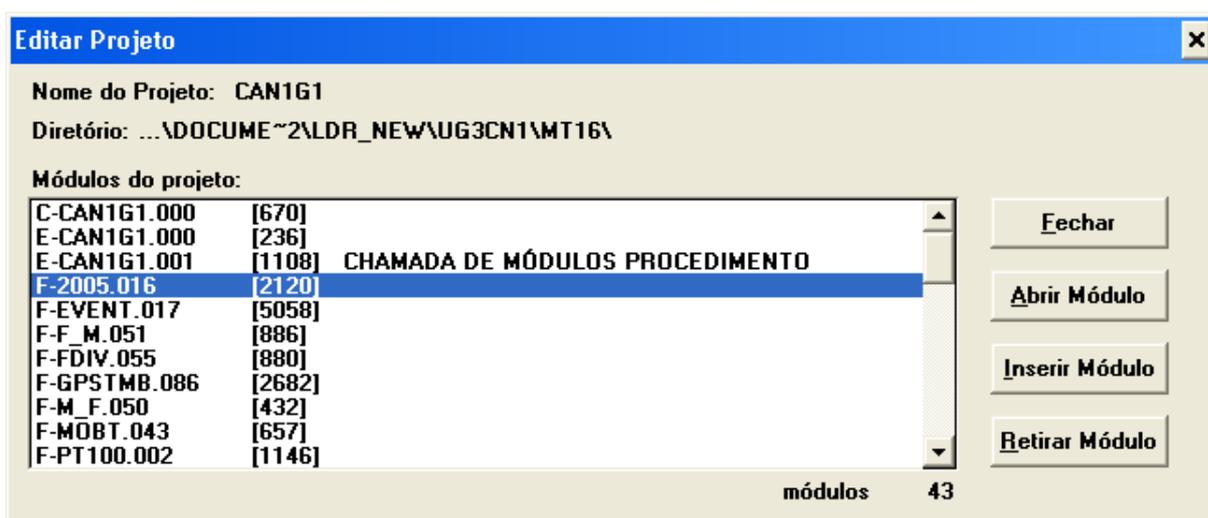


Figura 34 - Caixa de diálogo para Editar Projeto. [Altus 2007]

5.2 PRINCIPAIS FERRAMENTAS MENU COMUNICAÇÃO

O Menu Comunicação é a ferramenta para comunicação entre o programador em um computador e o CLP esta comunicação pode ser através da rede ou diretamente conectando o computador a CPU através de cabo serial .

5.2.1 Menu Comunicação Endereço

Ao utilizar o botão Endereço no Menu de comunicação será exibida a tela com o endereço IP da CLP com que esta aberto o projeto. Na figura 35 podemos visualizar a caixa de diálogo com o endereço da CLP da UG-01.

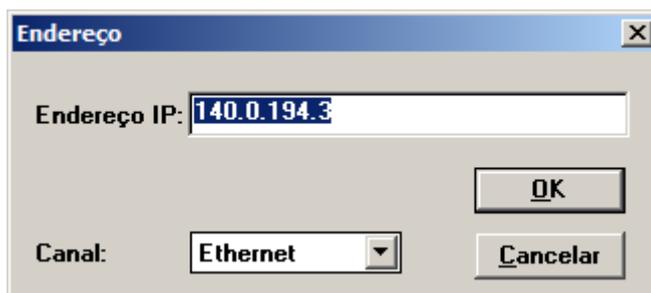


Figura 35 - Caixa de diálogo de endereço do CLP. [Altus 2007]

5.2.2 Menu Comunicação Estado

O comando Estado do Menu Comunicação deve ser utilizado para obter ou alterar o estado do CLP, obter o estado da rede ALNET II, dos operandos forçados e dos barramentos, alterar a senha e o nível de proteção, habilitar ou desabilitar as saídas e liberar operandos forçados. Opções da caixa de diálogo da figura 36 estão relacionados abaixo:

Saídas

Deve ser informado se as saídas devem estar habilitadas ou desabilitadas.

Estado

Deve ser informado qual estado de operação o CP ou roteador deve assumir.

Pontos Forçados

Exibe os operandos forçados e o valor dos forçamentos.

Liberar

Deve ser escolhido para liberar pontos de um operando.

Liberar Todos

Deve ser escolhido para liberar todos os forçamentos.

Salvar

Deve ser escolhido para salvar os valores e os operandos forçado em um arquivo .ini

Restaurar

Deve ser escolhido para recuperar de um arquivo .ini os valores e os operandos forçados.

Executa Ciclo

Deve ser escolhido para executar um ciclo de varredura no CLP, quando este estiver em modo ciclado.

Senha

Deve ser escolhido para alterar a senha do CLP.

Proteção

Deve ser escolhido para alterar o nível de proteção do CLP.

ALNET II

Deve ser escolhido para obter o estado da rede ALNET II. Este botão somente é habilitado se a CPU do projeto for AL-2000, AL-2002 ou QK2000.

Barramento

Deve ser escolhido para obter o estado dos barramentos. Este botão somente é habilitado quando a CPU do projeto for AL-2002.

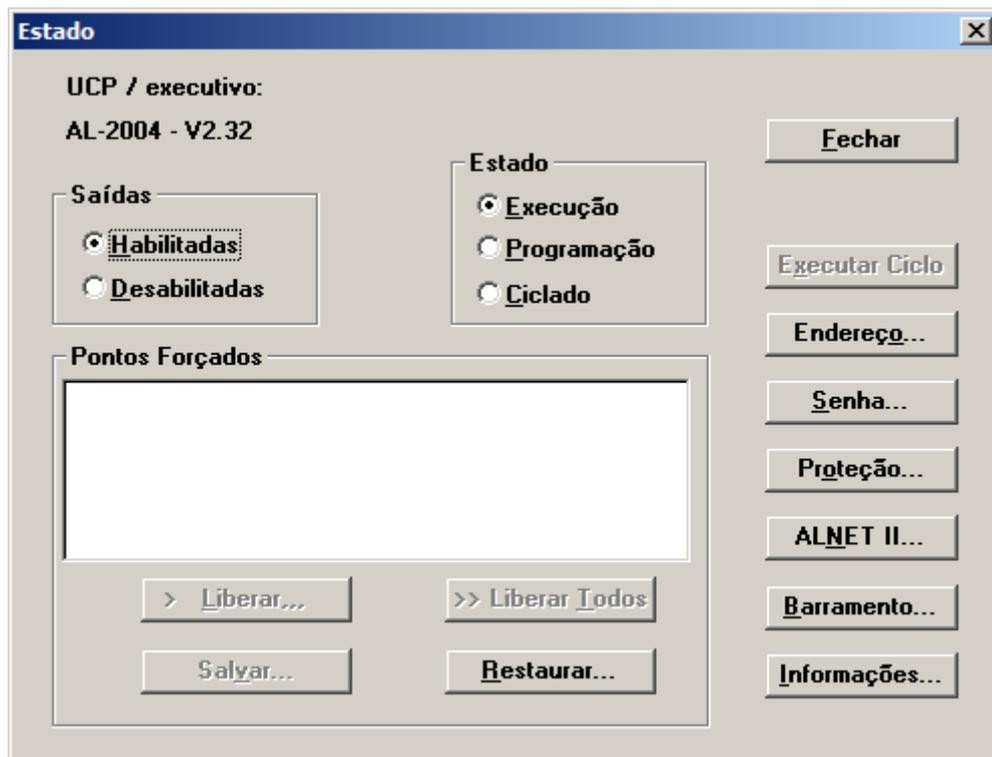


Figura 36 - Caixa de diálogo Menu Comunicação Estado [Altus 2007]

5.2.3 Menu Comunicação Ler/Enviar Módulo

O comando Ler/Enviar Módulo é utilizado para ler módulos do CLP enviar módulos para o CLP ou gravador de EPROM e compactar a memória RAM.

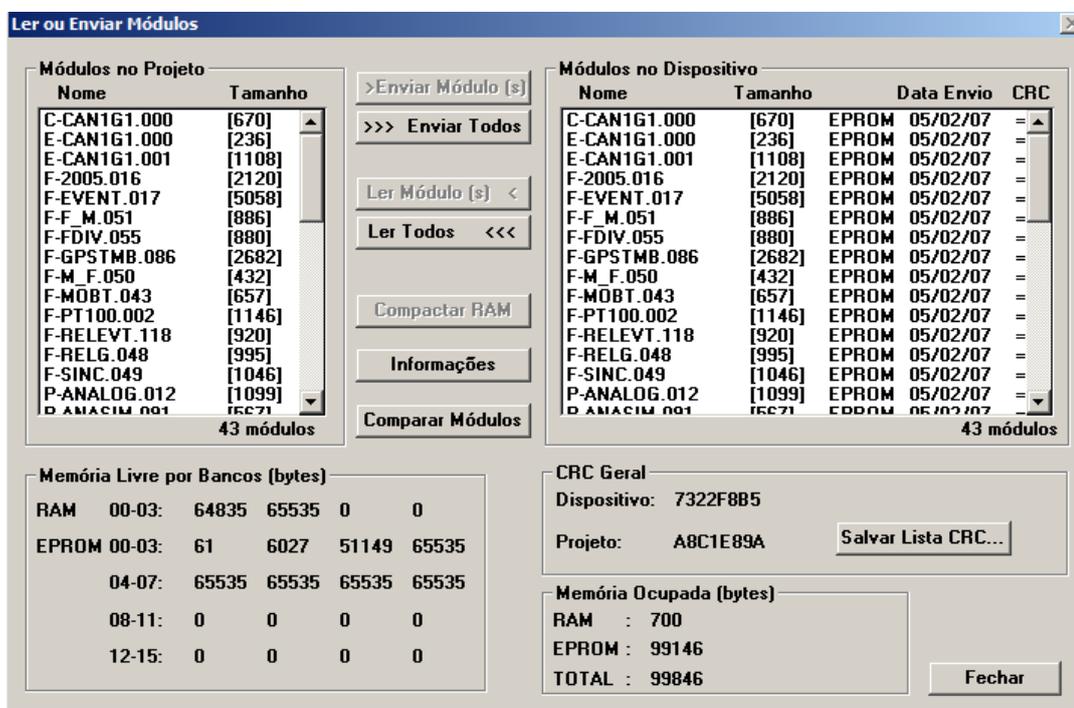


Figura 37 - Caixa de diálogo Menu Comunicação Ler/Enviar Módulo [Altus 2007]

Na figura 37 Ler/Enviar Módulo será possível fazer as seguintes tarefas:

Enviar Módulo

Deve ser escolhido para enviar os módulos selecionados na caixa "Módulos no Projeto", este comando é utilizado para atualizar um módulo alterado no projeto e enviar para a CLP.

Enviar Todos

Deve ser escolhido para enviar todos os módulos integrantes do projeto, este comando será realizado quando se deseja baixar todo o projeto no CLP, quando da troca de uma CPU ou atualizações do programa na CLP.

Ler Módulo

Deve ser escolhido para ler o módulo selecionado na caixa "Módulos no dispositivo", este comando é usado para ler o módulo da CLP para seu projeto.

Ler Todos

Deve ser escolhido para ler todos os módulos presentes no CLP.

Controlador, Gravador

Deve ser informado de/para onde os módulos serão lidos/enviados.

Compacta RAM

Deve ser escolhido para realizar a compactação da memória RAM do CLP.

5.2.4 Menu Comunicação Módulos

Este comando deve ser utilizado para obter o diretório do CP ou roteador, transferir módulos de EPROM para RAM e de RAM para Flash EPROM, apagar módulos, reabilitar módulos e apagar a memória Flash EPROM.

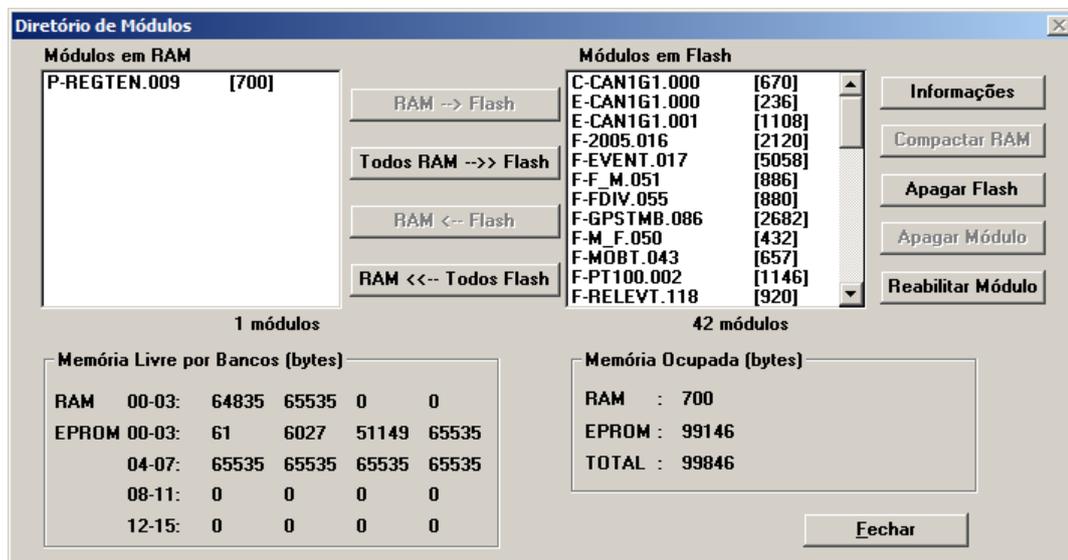


Figura 38 - Caixa de diálogo Menu Comunicação Diretório de Módulo [Altus 2007]

Na figura 38 Diretório de Módulo será possível realizar os seguintes comandos:

RAM --> Flash

Deve ser escolhido para realizar a transferência dos módulos selecionados na caixa "Módulos em RAM" para memória Flash EPROM.

Todos RAM --> Flash

Deve ser escolhido para realizar a transferência de todos os módulos que estão em RAM para Flash EPROM.

RAM <-- Flash

Deve ser escolhido para realizar a transferência dos módulos selecionados na caixa "Módulos em EPROM" para memória RAM.

RAM <<-- Flash

Deve ser escolhido para realizar a transferência de todos os módulos que estão em EPROM para memória RAM.

Apagar Módulo

Deve ser escolhido para realizar o apagamento dos módulos selecionados nas caixas "Módulo em RAM" ou "Módulos em Flash".

Apagar Flash

Deve ser escolhido para apagar o conteúdo de toda a memória Flash EPROM.

Compactar RAM

Deve ser escolhido para compactar a memória RAM.

Reabilitar Módulo

Deve ser escolhido para reabilitar um módulo em EPROM.

Memória Livre

Exibe a memória livre nos bancos de EPROM e RAM.

Memória Ocupada

Exibe a memória ocupada e o número de módulos.

5.2.5 Menu Comunicação Monitorar

Este comando permite a inserção de um operando na lista de monitoração ou a criação de uma janela de monitoração para um bloco de operandos.

Deve ser informado o endereço de nó do dispositivo que contém o operando a ser monitorado.

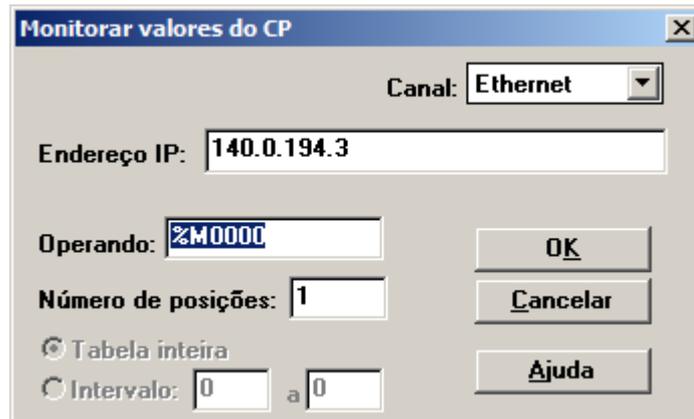


Figura 39 - Caixa de diálogo Menu Comunicação Monitorar [Altus 2007]

Após acessar, deve ser informado o endereço ou TAG do operando a ser monitorado, pode ser monitorado mais de uma posição indicando quantos pontos queremos monitorar, pode ser apenas uma entrada ou saída como o cartão inteiro.

Operando	Valor
%E0000.1	0
%E0000.2	0
%E0000.3	0
%E0000.4	0
%E0000.5	0
%E0000.6	1
%E0000.7	0

Figura 40 - Lista de monitoramento de um octeto. [Altus 2007]

Na figura 40 foi pedido para monitorar a entrada digital %E0000.1 e informado 8 posições para monitoramento, neste caso apenas a entrada %E0000.6 esta energizada as demais estão desenergizadas.

Operando	Valor
G1630H5	02547
%M0121	00000
RV1N601	03100
RV1N602	02041
RV1N606	02752
RV1N607	03858
RV1N608	02574
RV1N600	03290

Figura 41 - Lista de monitoramento de entradas analógicas. [Altus 2007]

Na figura 41 foi pedido para monitorar 8 entradas analógicas a partir da %M0120 que é a primeira memória reservada no CLP mais 7 posições. Através do valor verificado na %M0120 que tem o Tag G163OH5 que é a pressão do tanque Ar-Óleo do Sistema de Regulação tem o valor de 2547 bits. Podemos concluir que:
Escala do instrumento: 0 a 100 BAR.

Transdutor: 4 a 20 mA.

Escala cartão 12bits: 0 a 4096

Temos:

$X \rightarrow 2547$

$100 \rightarrow 4096$

$4096X = 2547 \times 100$

$X = 62,18 \text{ Bar}$

5.2.6 Menu Comunicação Monitorar Programa

Este comando deve ser utilizado para monitorar lógicas de módulos de programa. Funciona ligando ou desligando a monitoração de programa.

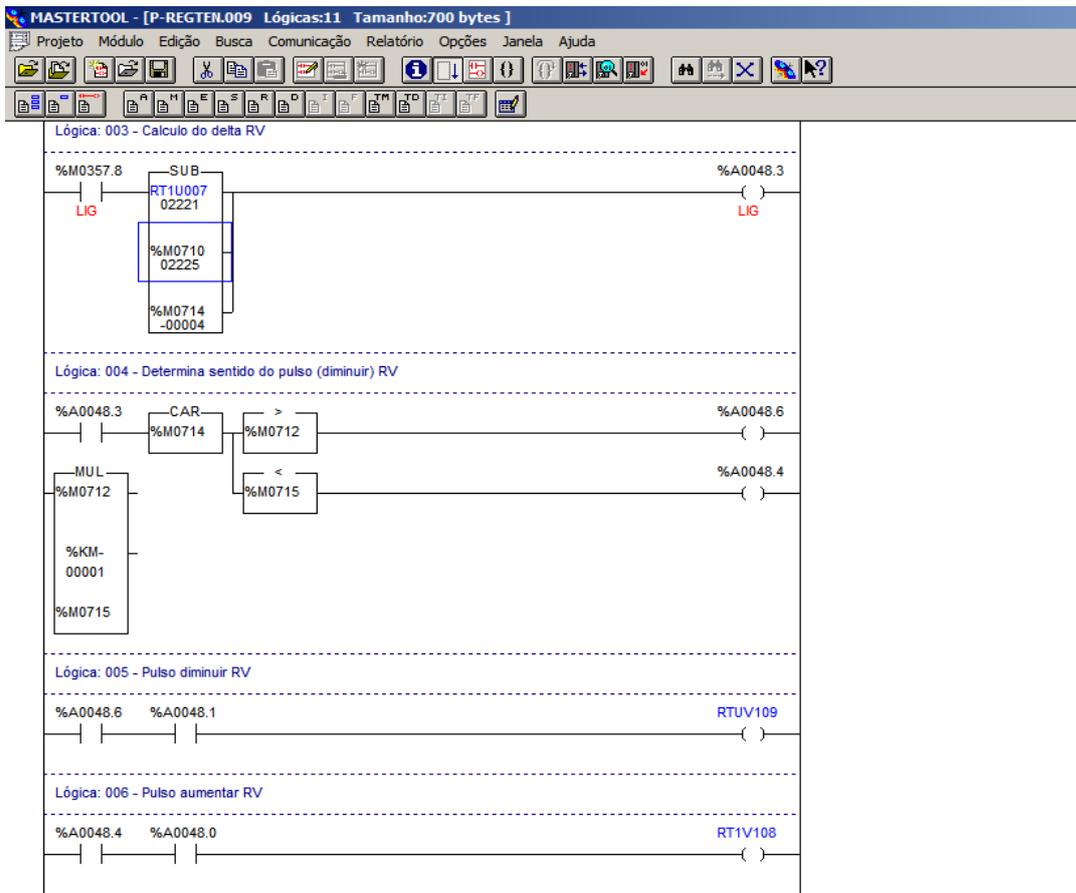


Figura 42 - Monitoramento da Lógica 003 P-REGTEN.009 [Altus 2007]

Na figura 42 o programa esta sendo monitorado, apenas a primeira lógica é monitorada, caso queira monitorar a lógica 4 esta deve ser selecionada e ser a primeira no topo da tela. Podemos analisar que a memória %M0357.8 esta ligada, o valor de referencia RT1U007 recebido do transdutor de tensão é 02221, o valor de referência recebido do SSC através da memória %M0710 é 02225 e o resultado da subtração feita na instrução e carregada na %M0714 é -00004 e a bobina simples %A0048.7 esta ligado.

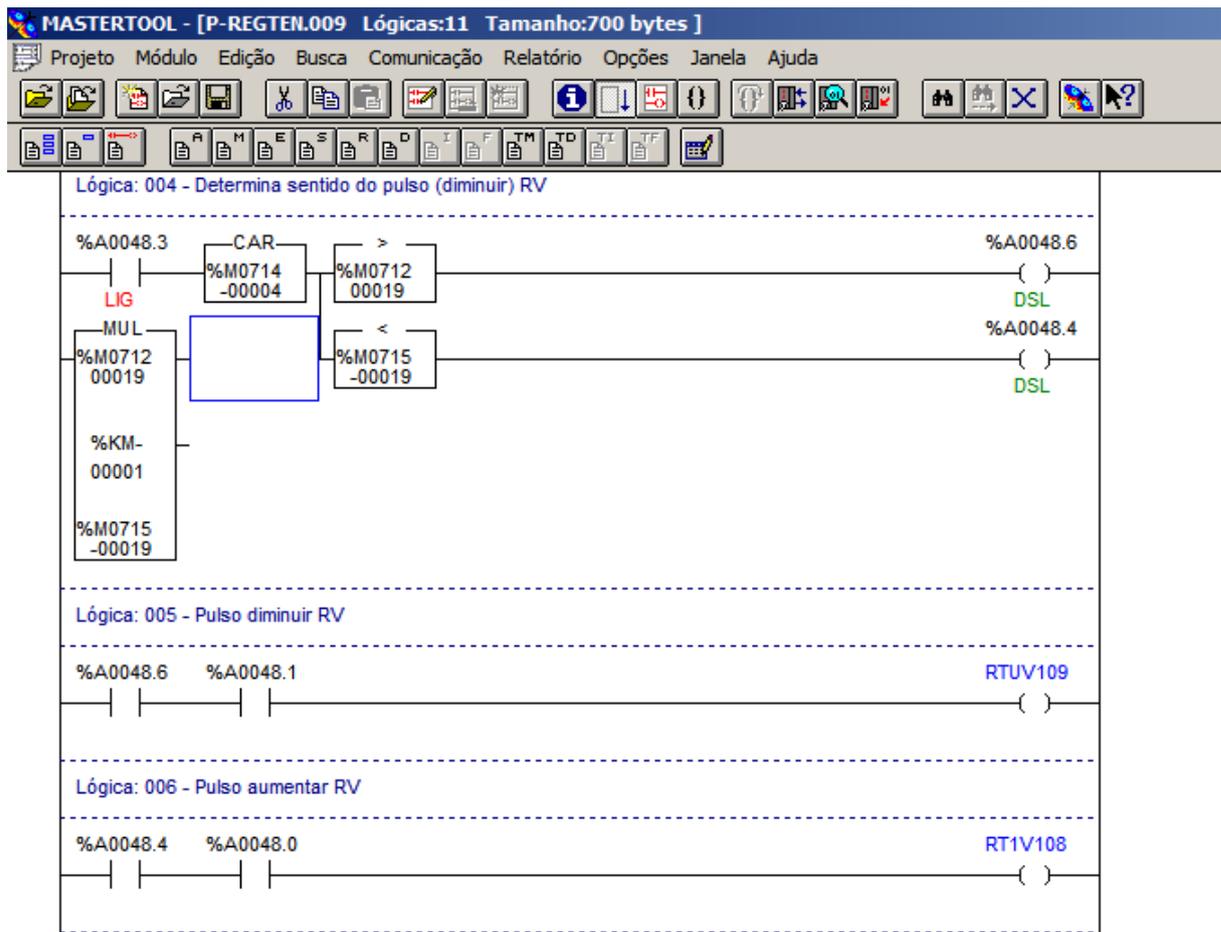


Figura 43 - Monitoramento da Lógica 004 P-REGTEN.009 [Altus 2007]

Na sequência da análise na figura 43, podemos verificar que o %A0048.3 auxiliar para comando do regulador esta ligado o valor da %M0714 erro do regulador é comparada com os valores de banda morta das %M0712 e %M 0715 como o valor do erro não é maior que 19 e não é menor que -19 não habilita as saídas para diminuir ou aumentar a tensão.

Capítulo 6: ESTRUTURA DE REDE DE INTERLIGAÇÃO DOS CLP'S

6.1 Rede de comunicação

A rede de comunicação é responsável pelo tráfego de informações entre os CLP's e o SSC interligando as 6 UAC's através de um anel óptico aos cartões de comunicação dos CLP's e no caso da UAC PCSE que faz a função de Gateway fazendo a aquisição dos dados da usina na rede interna e através de um isolamento feito através dos cartões de comunicação AL 3414 faz a comunicação com o COG Duke e o ONS através de link dedicado de comunicação via satélite.

6.1.1 Configuração do Anel óptico

A rede de comunicação implantada para CLP's da UHE Canoas I consiste em anel com 6 lances de fibra óptica interligando as UAC's através de 6 Switchs industrial RS900G fabricado pela *RuggedCom* do Canadá, alimentado em 125 VCC configurado com 2 portas para fibra óptica Gigabit Ethernet (1000BaseX) para criar o anel em fibra óptica com alta imunidade a ruídos e de longo curso conectividade e 8 portas Fast Ethernet (10/100BaseTX) para cabeamento estruturado em cobre.

Este equipamento foi desenvolvido para trabalhar em ambientes severos e oferece um alto nível de imunidade a interferências eletromagnéticas e fortes descargas elétricas típicas de ambientes encontrados em usinas e subestações de energia elétrica podendo suportar altas temperaturas.



Figura 44 - Switch Industrial RUGGEDCOM RS900G

Na tabela 2, é apresentado o resultado da certificação dos lances de fibra óptica, que interliga os painéis da UHE Canoas I, as fibras são do tipo multimodo e foram instaladas em D.I.O., instalados nos painéis das UAC. Como padrão foi adotado as cores da fibra verde e amarela para transmissão e recepção de dados entre os CLP's, a branca será responsável pela transmissão do tempo real do GPS.

Cabo	DIO	Fibra	Comprimento	Atenuação	Lance
F001	1	TX 1 - VD	540 mts.	< 1,2 dB	UAC 5 / PCSE
F002		RX 2 - AM	540 mts.	< 1,2 dB	UAC 5 / PCSE
F003		TX 3 - BR	540 mts.	< 1,2 dB	UAC 5 / PCSE
F004		RX 4 - AZ	540 mts.	< 1,2 dB	UAC 5 / PCSE
F001	2	TX 1 - VD	75 mts.	< 1,0 dB	PCSE / UAC 4
F002		RX 2 - AM	75 mts.	< 1,0 dB	PCSE / UAC 4
F003		TX 3 - BR	75 mts.	< 1,0 dB	PCSE / UAC 4
F004		RX 4 - AZ	75 mts.	< 1,0 dB	PCSE / UAC 4
F001	3	TX 1 - VD	24 mts.	< 1,0 dB	UAC 4 / UAC 1
F002		RX 2 - AM	24 mts.	< 1,0 dB	UAC 4 / UAC 1
F003		TX 3 - BR	24 mts.	< 1,0 dB	UAC 4 / UAC 1
F004		RX 4 - AZ	24 mts.	< 1,0 dB	UAC 4 / UAC 1
F001	4	TX 1 - VD	26 mts.	< 1,0 dB	UAC 1 / UAC 2
F002		RX 2 - AM	26 mts.	< 1,0 dB	UAC 1 / UAC 2
F003		TX 3 - BR	26 mts.	< 1,0 dB	UAC 1 / UAC 2
F004		RX 4 - AZ	26 mts.	< 1,0 dB	UAC 1 / UAC 2
F001	5	TX 1 - VD	26 mts.	< 1,0 dB	UAC 2 / UAC 3
F002		RX 2 - AM	26 mts.	< 1,0 dB	UAC 2 / UAC 3
F003		TX 3 - BR	26 mts.	< 1,0 dB	UAC 2 / UAC 3
F004		RX 4 - AZ	26 mts.	< 1,0 dB	UAC 2 / UAC 3
F001	6	TX 1 - VD	592 mts.	< 1,5 dB	UAC 3 / UAC 5
F002		RX 2 - AM	592 mts.	< 1,5 dB	UAC 3 / UAC 5
F003		TX 3 - BR	592 mts.	< 1,5 dB	UAC 3 / UAC 5
F004		RX 4 - AZ	592 mts.	< 1,5 dB	UAC 3 / UAC 5

Tabela 2 - Certificação das fibras ópticas

6.1.2 Cartão de comunicação do CLP

A Interface para rede Ethernet AL-3414, integrante da série AL-2000, permite a conexão de CLPs Altus a redes de comunicação abertas que seguem o padrão TCP/IP. Possibilita a comunicação entre CLPs Altus e destes com quaisquer outros equipamentos que se comuniquem através do protocolo de Ethernet TCP/IP com nível de aplicação ALNET II ou MODBUS. [Altus 2004]

O módulo possui interface elétrica 10/100Base-TX através de um conector RJ45 fêmea blindado padrão UTP (par trançado não blindado) de categoria 5, suportando também cabos blindados com a vantagem de melhorar a sua imunidade a ruído externo. [Altus 2004]

O cartão AL-3414 conectado ao bastidor na “posição 01” do “barramento 0” é responsável pela transmissão do estado de entradas e saídas digitais e analógicas para o SSC (Sistema de Supervisão e Controle), também é responsável pela entrada dos comandos recebidos do SSC.

Este cartão está ligado eletricamente ao Switch RS900G através de cabo estruturado de categoria 5.



Figura 45 - Cartão de comunicação Altus AL 3414 [Altus 2007]

6.1.3 Comunicação do SSC com a rede

O SSC (Sistema de Supervisão e Controle) é integrado de dois micros computadores ligados na rede em 2 pontos distintos da rede um micro está localizado na sala de controle ligada a rede no Switch RS900G da UAC PCSE denominado CANOASI_A e o outro instalado na sala de máquinas da usina ligado ao Switch da UAC-04 foi denominado CANOASI_B.

6.1.4 Endereçamento IP da rede de automação de Canoas I

Para que não exista conflito das informações proveniente de cada CLP, e seja possível a comunicação entre os CLP's e o SSC na tabela abaixo esta o endereço IP utilizado para na configuração de cada elemento da rede.

Elemento	Endereço IP	Máscara
UAC - 01	140.0.194.3	255.255.255.0
UAC - 02	140.0.194.4	255.255.255.0
UAC - 03	140.0.194.5	255.255.255.0
UAC - 04	140.0.194.7	255.255.255.0
UAC - 05	140.0.194.6	255.255.255.0
UAC - PCSE	140.0.194.8	255.255.255.0
Micro_A	140.0.194.20	255.255.255.0
Micro_B	140.0.194.21	255.255.255.0
Manutenção	140.0.194.38	255.255.255.0
Default Gateway	140.0.194.10	

Tabela 3 - Endereço IP da rede Automação

6.2 Rede de Sincronismo

A rede de sincronismo visa manter todos as CLP's sincronizadas com precisão de 1 ms, para sequenciamento de eventos e alarmes na unidade para análise do funcionamento dos equipamentos e análise de bloqueios em equipamentos, o sincronismo tempo também é uma exigência do ONS para quando

for necessário uma análise de blecaute regional ou sistêmico, com todos os agentes sincronizados na mesma base de tempo se analisar as causas e conseqüências.

6.2.1 Funcionamento da rede de sincronismo.

Um receptor de GPS AL-1480 integra em um único módulo a antena e a unidade eletrônica, foi instalado próximo a UAC-05 e se comunica com o módulo AL-1422 através de uma interface RS-422 com alta imunidade a ruído, via cabo AL-1725 de 25 metros. Esse primeiro AL-1422 será configurado como Mestre sincroniza a UAC-05 por meio de dois cabos AL-1719, pelas portas COM e SYNC da CPU AL-2004, através do protocolo NMEA/PPS e através da porta NET, conectada ao conversor RS-232/485 óptico, repete o sinal de sincronismo para as outras UAC's.

Em cada uma das UAC's o sinal chega através de um link de fibra óptica simples e sai por outro, assim na UAC-SE (a primeira) e na UAC-SA (a última) apenas tem um cabo de fibra para sincronismo.

O protocolo NMEA/PPS será utilizado para o sincronismo das UAC's com o AL-1480, consiste em dois cabos seriais contendo duas informações diferentes, frame NMEA-0183 e PPS (pulso por segundo), para garantia do sincronismo. Abaixo são mostradas as características e os sinais recebidos do GPS para o sincronismo da CLP. [Altus 2004]

O frame NMEA-0183 especifica data e hora do instante aproximado em que iniciou este frame. Por exemplo, se o primeiro frame NMEA-0183 da figura contém a data e hora "17/09/2001, 23:42:33.1", então a borda de subida anterior do PPS (cerca de 100 ms antes) corresponde a "17/09/2001, 23:42:33 ". [Altus 2004]

ASCII, 8 data-bits, paridade ímpar, 1 stop bit, 1200 bp

\$GPZDA,hhmmss.s,dd,mm,aaaa,,*hh<CR><LF>

hhmmss.s (hora, minuto, segundo e décimo de segundo)

dd,mm,aaaa (dia, mês, ano)

hh = checksum

<CR> = 0x0A

<LF> = 0x0D

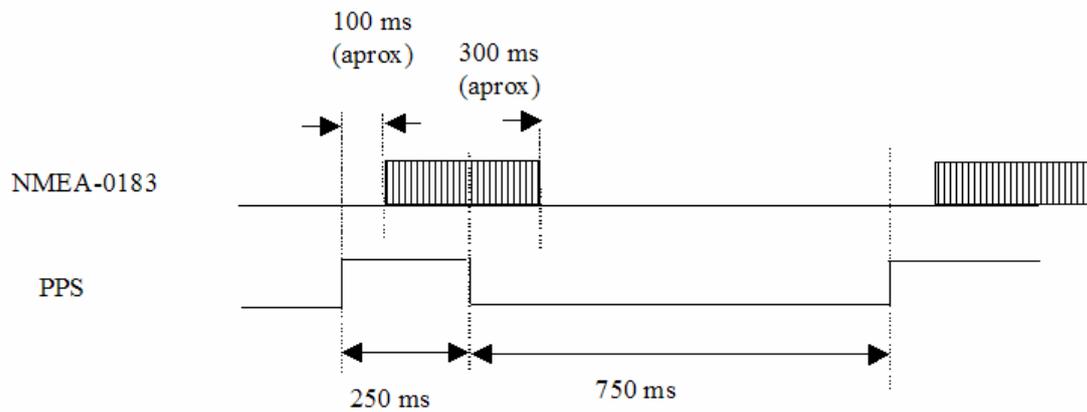


Figura 46 - Gráfico do frame NMEA e o sinal de PPS [Altus 2004]

Mestre de Rede Ótica (Chaves: config = 0000)

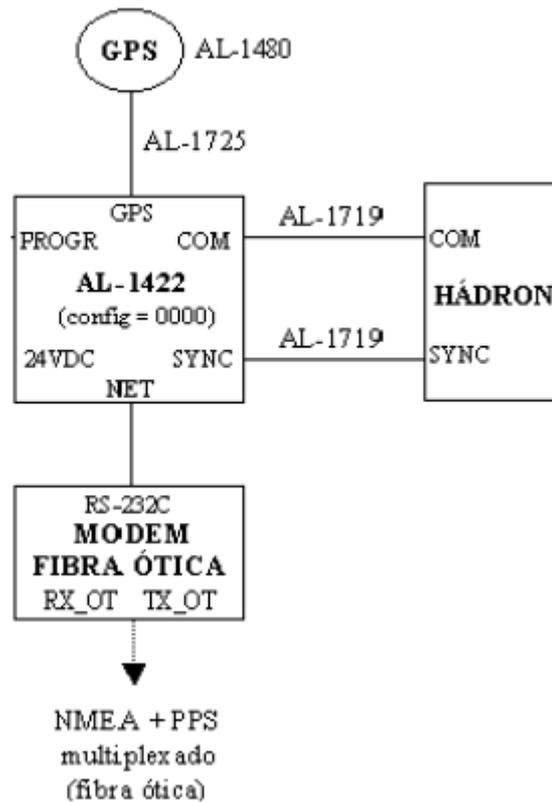


Figura 47 - Configuração da rede de sincronismo Mestre [Altus 2004]

Esclavo de Rede Ótica (Chaves: config = 0001)

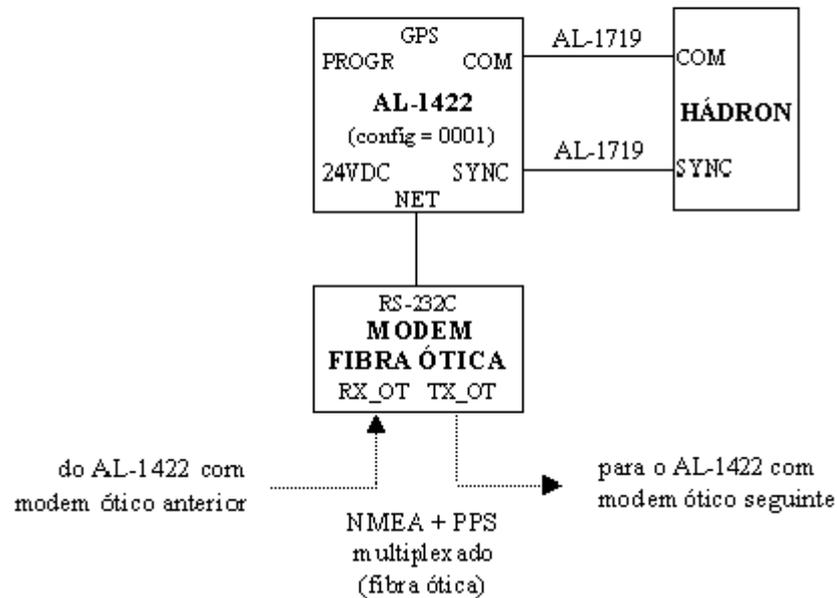


Figura 48 - Configuração da rede de sincronismo Escravo [Altus 2004]

6.3 Rede de comunicação externa.

No painel da UAC PCSE esta configurada uma rede de comunicação que interliga ao segundo cartão AL-3414 da UAC-PCSE ao roteador via satélite para enviar dados para o COG Duke em Chavantes e para ONS no Rio de Janeiro.

6.4 Supervisão do Sistema Digital no SSC.

No SSC (sistema de supervisão e controle) foi criada uma tela para supervisão da comunicação é possível visualizar um diagrama das redes de comunicação e sincronismo, também é possível visualizar o relógio interno de cada CLP, um campo para sinalização de falha na comunicação, se o comando dos equipamentos estão para o COG ou local e qual IHm esta ativa como servidor e qual é a reserva.

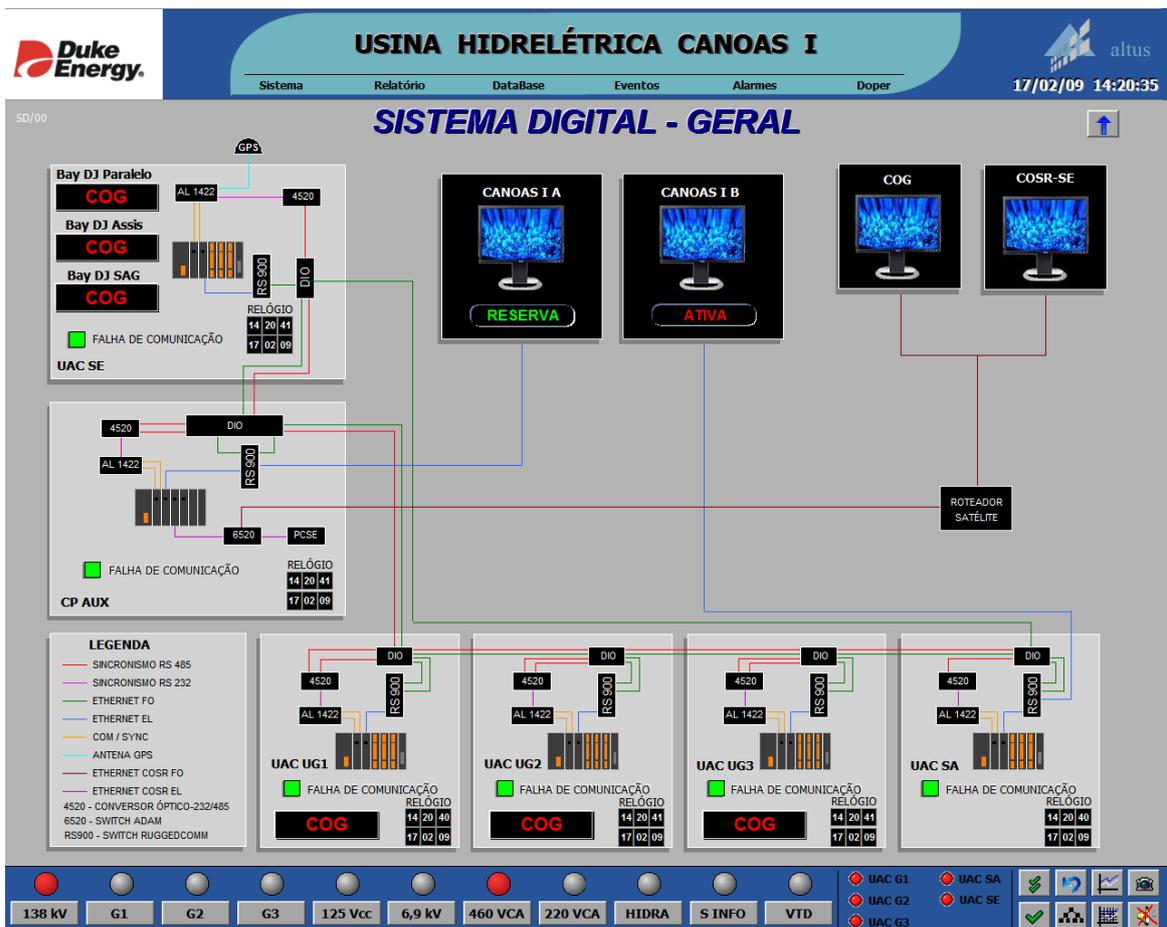


Figura 49 - Tela de supervisão do sistema digital [ELIPSE-2008]

Capítulo 7: SISTEMA DE SUPERVISÃO E COTROLE

O SSC (Sistema de supervisão e Controle) é um aplicativo desenvolvido para ser a interface entre o homem e a máquina, com referencia as entradas digitais e analógicas aquisitadas pela CLP através de um drive de comunicação, no SSC são desenvolvidas telas com referencia aos processos com atualização instantânea do estado na tela, esta mudança de estado é denominada de evento que é referenciada em uma lista de eventos e guardada em um banco de dados para pesquisa e análise, estes eventos podem ser configurados como alarmes que gera uma sinalização visual na tela e sonora.

7.1 Substituição do SSC Alstom por Elipse

Com a necessidade de adequação do sistema de Supervisão e Controle da UHE Canoas I da Duke Energy ao submódulo 10.19 ONS, nasceu uma necessidade de modernização dos equipamentos que estavam em operação na usina desde sua construção em 1998, esta modernização culminou na substituição do antigo SSC por uma nova plataforma.

O novo sistema tecnologicamente mais atual permitiu mais rapidez na comunicação entre os CLP's e SSC, possibilitou a instalação do GPS e a substituição dos computadores que trabalham como IHM.

Para minimizar as dificuldades dos operadores decidimos manter *layout* das telas, mantendo assim cerca de 85% das telas do SSC Alstom antigo. Eliminamos 14 telas que não utilizávamos e criamos 6 novas telas.

7.2 Características do Elipse E3

O novo sistema escolhido foi o Elipse E3 que é um sistema SCADA (*Supervisory Control and Data Acquisition*) que oferece um avançado modelo de objetos, uma poderosa interface gráfica.

O SSC Elipse aquisita os dados que possibilita criar várias maneiras para exibir, analisar, controlar, comandar, gravar ou divulgar as informações.

Telas que permitem criar uma IHM (*Interface Homem-Máquina*) localmente, pela rede local ou Internet, com o intuito de exibir o status atual ou passado dos dados em várias formas, com o auxílio de um editor gráfico e objetos específicos

Alarmes e Eventos monitoram o acontecimento de situações específicas.

Históricos armazenam os dados em bancos de dados relacionais.

Relatórios permitem visualizar e imprimir os dados, dentre outros módulos e possibilidades.

A interface gráfica com o usuário pode também ser executada através da Internet ou intranet via Internet Explorer.

7.3 Redundância de operação e servidor

O sistema será composto por duas estações de operação que também vão fazer a comunicação com as UACs. Os micros funcionam no modo *Hot-Standby*/Servidor-Cliente, onde apenas o servidor requisita e processa os dados das UACs o IHM servidor sinalizará ATIVA em vermelho é a estação CANOASI_B e a estação CANOASI_A que fica apenas como cliente sinaliza RESERVA em verde na tela conforme na figura 49.

A ferramenta de *Hot-StandBy* do E3 tem como objetivo permitir que um servidor fique em estado de espera (*standby*), aguardando uma possível falha do servidor. A troca de servidor ativo pode ser manual ou automática.

A troca manual é acionada através do menu do E3 Server do micro em *standby* (opção Servidor - Ativar).

A troca automática ocorre quando o servidor em *standby* detecta que o servidor principal não está mais rodando.

7.4 Banco de Dados em *Hot-StandBy*

No Supervisório desenvolvido temos dois servidores rodando separadamente numa mesma aplicação. Um deles fica ativo, enquanto que o outro fica em estado de espera (*standby*), aguardando uma possível falha.

O *Hot-Standby* garante a continuidade da aplicação, mas quando utilizamos Banco de Dados, também temos que garantir a continuidade do acesso à base de dados.

Assim da forma que foi desenvolvida a aplicação a gravação de dados é feita em bases diferentes. Neste caso, o temos os dois bancos de dados rodando localmente, em ambos servidores.

Para garantir que o acesso ao BD estará sempre disponível, pois a máquina que está rodando a aplicação também roda o Banco de Dados.

Nesta aplicação utilizamos a função do E3 Server de sincronizar as diferentes bases da aplicação principal e a secundária. Selecionado a propriedade *Enable Synchronization* dos bancos de dados utilizados na aplicação habilitou-se essa característica na aplicação. Quando ela está ativa, todos os dados gravados na base principal são atualizados na base em Standby quase que instantaneamente.

7.5 Divisão dos processos no Elipse SCADA

Um Domínio é um conjunto de objetos e configurações que definem um sistema de supervisão. Corresponde ao que se chamava de Aplicativo no Elipse SCADA. O E3 é um ambiente modular de desenvolvimento e execução de sistemas de supervisão, composto pelos seguintes módulos:

O **E3 Server** é o coordenador de todos os módulos do E3. Ele sempre roda como serviço, mesmo no modo Demo. Pode também ser iniciado quando um E3 Studio local for iniciado, ou através de um atalho para rodar o Domínio. O E3 Server é responsável por verificar e gerenciar as licenças dos diferentes módulos do E3. [ELIPSE 2008]

O **E3Run** é o processo responsável pela execução dos objetos do servidor como Tags, Alarmes, Banco de Dados, Históricos, etc.; ou seja, todos os objetos exceto as Telas e o objeto Viewer. [ELIPSE 2008]

O **E3 Admin** é o módulo responsável por monitorar e comandar o E3 Server

O **IOServer** é o processo responsável pela execução dos *Drivers* de Comunicação. Cada *driver* roda em um *IOServer* separado. Os *IOServers* são criados, mantidos e monitorados pelo processo **E3Run**.

O **E3 Studio** é o ambiente de desenvolvimento e configuração de aplicativos. Permite abrir ou editar arquivos .PRJ e .LIB. O E3 Studio é multi-usuário, isto é, vários E3 Studios podem modificar os mesmos arquivos ao mesmo tempo. O E3 Studio utiliza a conexão *REC* para se comunicar com o E3 Server.

O **E3 Viewer** é a interface com o usuário em tempo de execução, pois mostra as Telas em execução e permite a operação do sistema. O E3 Viewer pode ser

executado a partir de qualquer ponto da rede que tenha acesso ao E3 Server sem a necessidade de copiar o aplicativo para os outros E3 Viewers, pois as Telas e bitmaps são carregadas conforme a necessidade em tempo de execução. [ELIPSE 2008]

7.6 Divisão Sistema de Supervisão Elipse E3

O Sistema de Supervisão desenvolvido para UHE Canoas I esta dividido em 4 projetos.

7.6.1 Projeto Driver

Este projeto possui os drivers de comunicação com as UAC's 01, 02, 03, SA, SE e CPAUX.

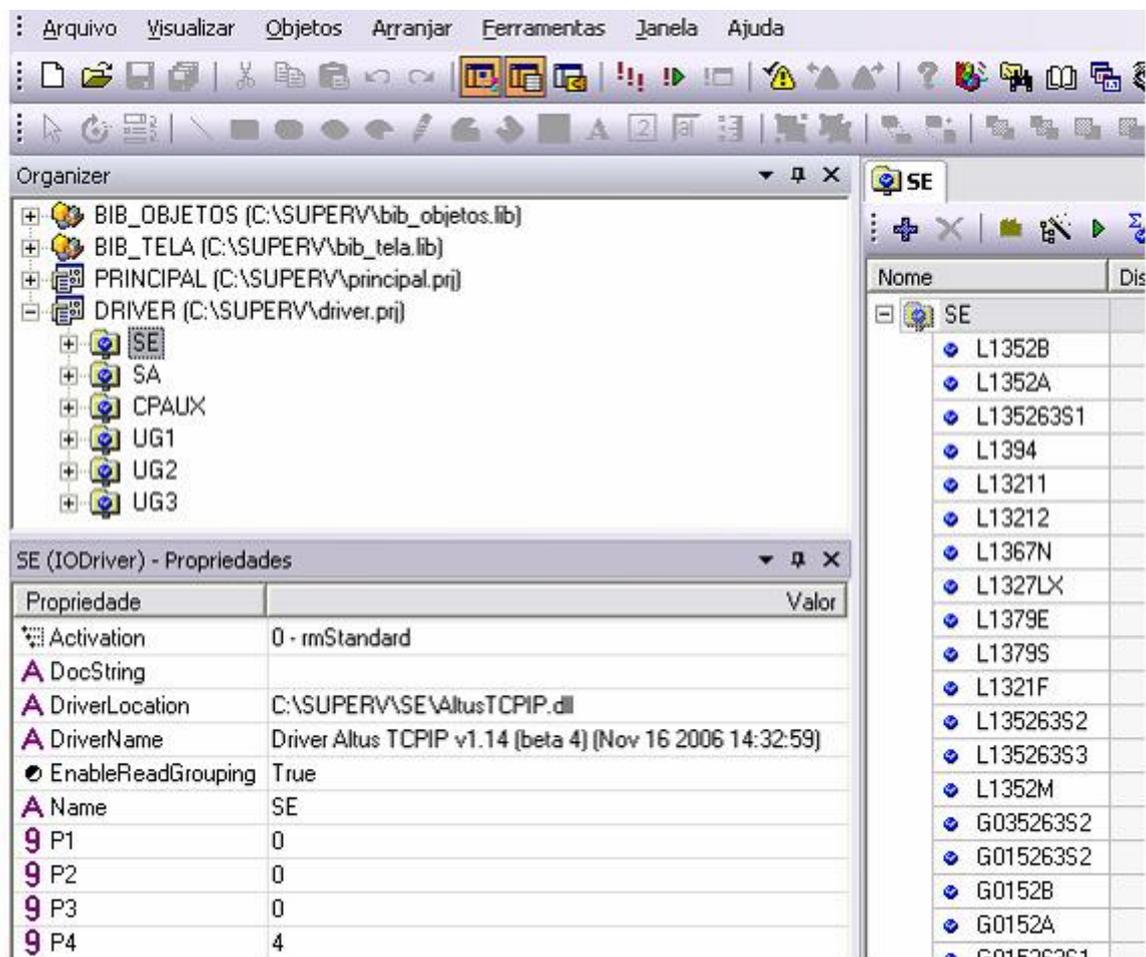


Figura 50 – Tela do E3 Studio projeto Driver. [ELIPSE-2008]

Neste projeto é configurado o protocolo de comunicação e os endereços de comunicação do SSC com as UAC's, Neste projeto deve ser criado um TAG de

comunicação para cada ponto correspondente no CLP, um cuidado especial deve ser tomado pois as memórias do CLP é numerada de 0 a 15 e no Driver 1 a 16.

7.6.2 Projeto BIB_OBJETOS

Este projeto do tipo biblioteca possui todos os objetos utilizados na aplicação, alguns destes objetos já fazem parte da galeria do Elipse outro foram criados para esta aplicação.

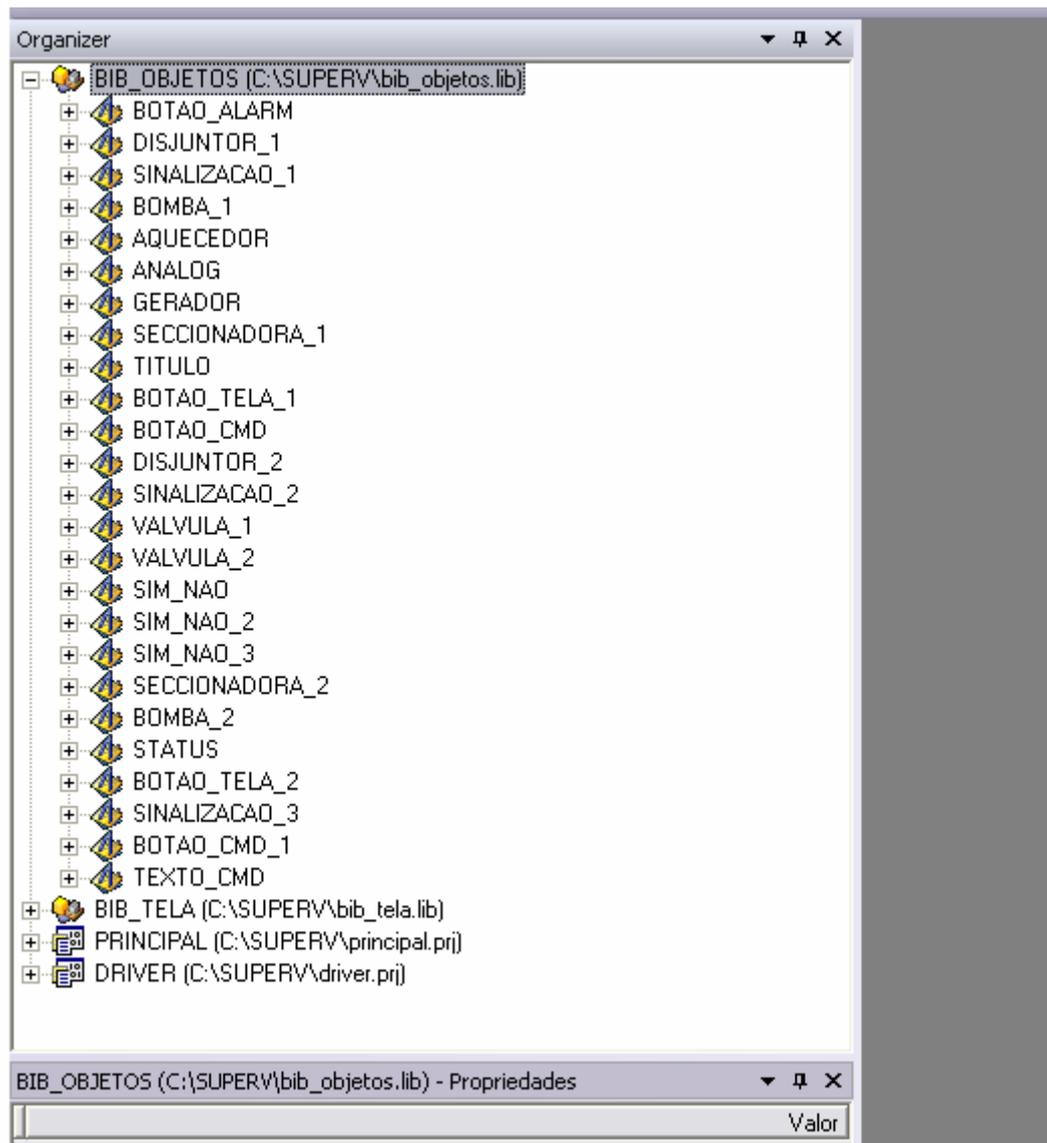


Figura 51 – Tela do E3 Studio projeto BIB_OBJETOS [ELIPSE-2008]

A seguir será mostrado como o exemplo o objeto **DISJUNTOR_1** OU **2** utilizado na aplicação

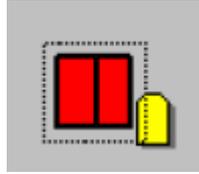


Figura 52 – Objeto DISJUNTOR_1 OU 2 [ELIPSE-2008]

Um evento “MouseUp” (na subida do botão do mouse) faz abrir o menu para seleção do comando.

Telecomando (abre a tela “Telecomando”)

Bloqueio (abre a tela “Bloq”)

Entrada de Dados (abre a tela entrada de dados tipo 2)

Lista de Alarme (abre a tela de alarmes ativos)

As propriedades deste objeto são:

A_TRANSICAO = quando verdadeiro, ativa objeto na cor laranja.

A_FECHADO = quando verdadeiro, ativa objeto na cor vermelho.

A_ABERTO = quando verdadeiro, ativa objeto na cor verde.

A_INVALIDO = quando verdadeiro, ativa objeto na cor roxa.

A_HABTELECMD = quando verdadeiro, habilita menu telecomando.

A_HABBLOQUEIO = quando verdadeiro, habilita menu bloqueio.

A_HABENTRADADADOS = quando verdadeiro, habilita menu entrada de dados.

A_HABLISTAALM = quando verdadeiro, habilita menu lista de alarme.

A_TAG = deve possuir o nome do tag do equipamento.

A_DESCRICAO = deve possuir a descrição do equipamento.

A_BLOQUEIO = deve possuir o caminho (path) do tag interno de bloqueio. Se “a_habbloqueio” não

for verdadeiro, o valor deve ficar vazio.

A_BLOQUEIO2 = deve possuir a propriedade Value do tag interno de bloqueio. Se “a_habbloqueio”

não for verdadeiro, o valor deve ficar em 0.

A_HABCMD2 = se verdadeiro, habilita o segundo botão de comando.

A_DESCRICAOCMD1 ou 2 = descrição dos botões de comando.

A_SELECAO = deve possuir o valor de seleção do objeto. Pode ser configurado com qualquer valor,

desde que não tenha outro objeto com o mesmo valor.

A_CMD1TM41POS0 a 7 = deve possuir os valores para o comando na UAC do botão de comando 1

(ver especificação do sistema).

A_CMD2TM41POS0 a 7 = deve possuir os valores para o comando na UAC do botão de comando 2

(ver especificação do sistema).

A_UAC = deve possuir o nome da UAC que envia as informações deste objeto.

A_ESTADO = interno do script, não utilizar.

A_PATHTELA = deve possuir o caminho da tela de alarmes ativos deste subsistema.

A_PATHCMD1 ou 2 = deve possuir o caminho do tag interno deste comando.

7.6.3 Projeto BIB_TELA

Este projeto do tipo biblioteca possui as telas genéricas utilizadas na aplicação.

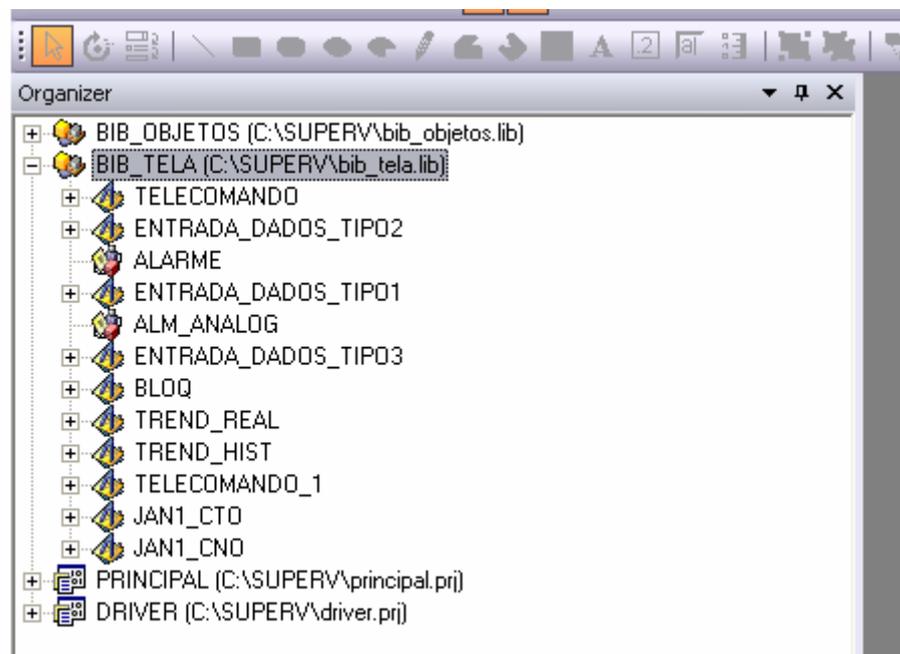


Figura 53 – Tela do E3 Studio projeto BIB_TELA [ELIPSE-2008]

Abaixo exemplo da tela de telecomando que é utilizada algumas vezes na aplicação principal.

Figura 54 – Tela genérica para Telecomando de equipamentos. [ELIPSE-2008]

7.6.4 Projeto PRINCIPAL

Este projeto possui telas, históricos e configurações de alarmes do sistema.

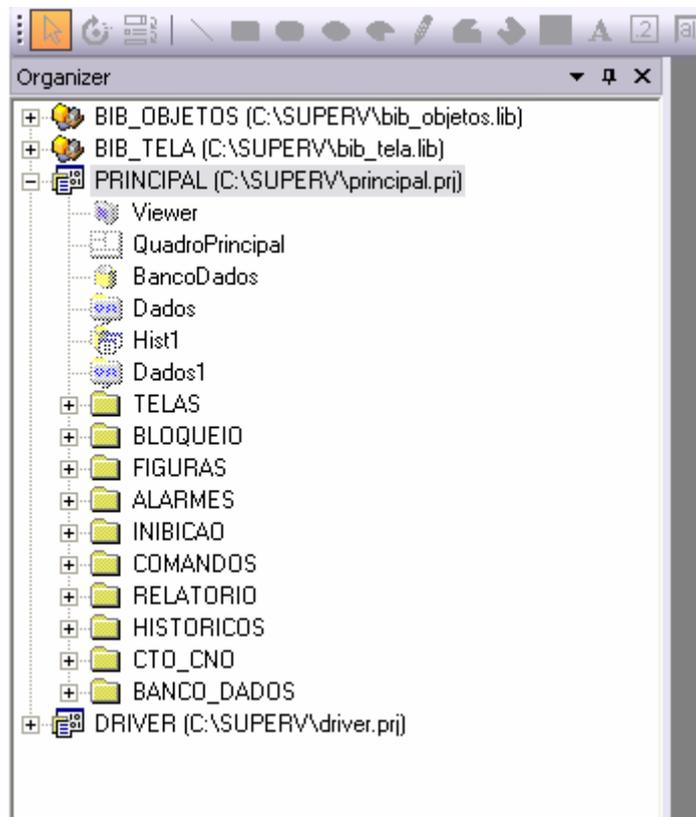


Figura 55 – Tela do E3 Studio projeto PRINCIPAL [ELIPSE-2008]

O projeto principal é o responsável por construir todas as funcionalidades do SSC da UHE Canoas I, nele esta as telas que desenvolvemos para supervisionar a instalação, as funções de comando, relatórios, históricos, gravação de eventos e alarmes em banco de dados,

Os comandos de inicialização da aplicação como o Viewer, Quadro Principal, a sincronização no modo *Hot-Standby*/Servidor-Cliente dos dois IHM's.

Para exemplificar o quanto é complexo o desenvolvimento do projeto principal utilizarei como exemplo o Telecomando para Potência Ativa do Gerador, na figura 56 mostrará o *Design* da tela para Controle de Potência Ativa do Gerador, na Figura 57 a estrutura da tela para Controle de Potência Ativa do Gerador, na Figura 58 está as associações para os campos dinâmicos da tela, na figura 59 já aparecerá a tela em funcionamento no E3

Controle de Potência Ativa do Gerador

Tag: G01_CPA UAC: []

Descrição: []

Valor Atual: 0 []

Referência: 0 []

Lim. Oper. Min.: 0,00 Lim. Oper. Max.: 27,5

Ok Cancelar

Figura 56 - *Design* da tela de controle no E3 Studio [ELIPSE-2008]

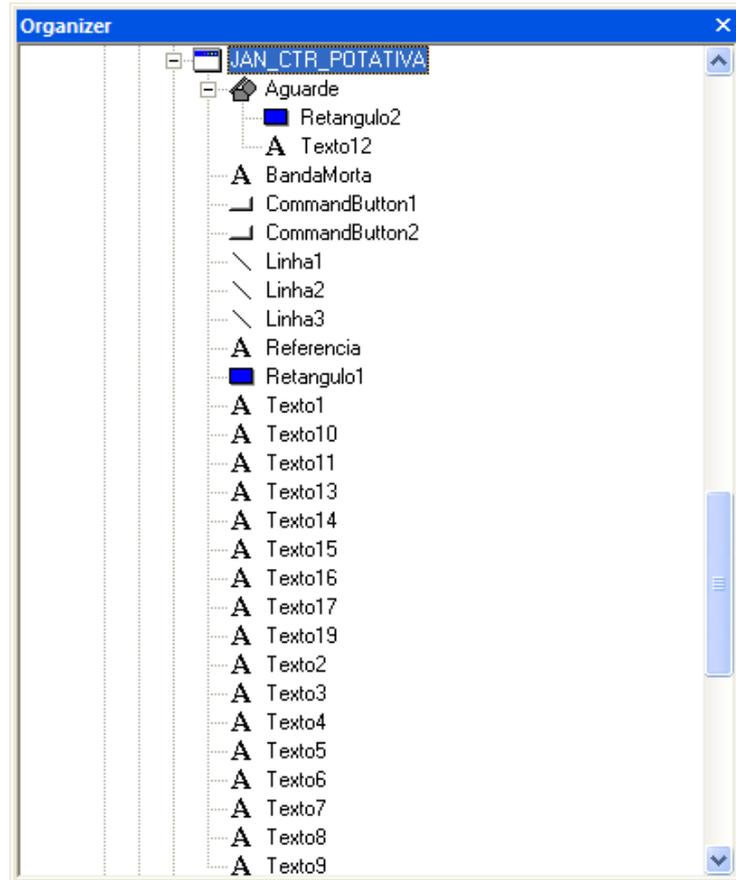


Figura 57 - Estrutura da tela de controle no E3 Studio [ELIPSE-2008]

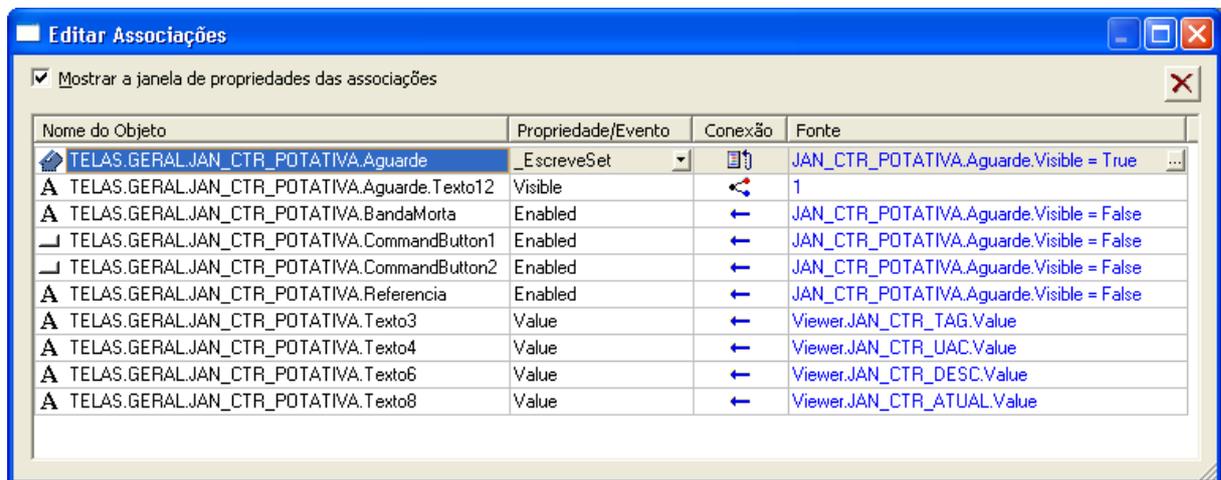


Figura 58 - Associações da tela de controle no E3 Studio [ELIPSE-2008]

Tag	UAC
G01_CPA	UAC UG1
Descrição	
G01 - Controle de Potência Ativa	
Valor Atual	Unidade
23,31623932	MW
Referência	Banda Morta
23,00	0,00
Lim. Oper. Min.	Lim. Oper. Max.
0,00	27,5

Figura 59 - Tela de controle de Potência no E3 Viewer [ELIPSE-2008]

Para que o comando acima seja realizado foi necessário escrever um Scripts que rodará o algoritmo a seguir assim que o operador apertar o botão OK.

```

Sub CommandButton1_Click()
If Application.Item("JAN_CTR_UAC").Value = "UAC UG1" then
    Valor = (Screen.Item("BandaMorta").Value * 4095) / 35
    Application.GetObject("UG1.TM41_00").WriteEx 0
    Application.GetObject("UG1.TM41_01").WriteEx 4096
    Application.GetObject("UG1.TM41_02").WriteEx Valor
    Application.GetObject("UG1.TM41_03").WriteEx 0
    Application.GetObject("UG1.TM41_04").WriteEx 0
    Application.GetObject("UG1.TM41_05").WriteEx 267
    Application.GetObject("UG1.TM41_06").WriteEx 0
    Application.GetObject("UG1.TM41_07").WriteEx 1
End If

```

```
If Application.Item("JAN_CTR_UAC").Value = "UAC UG2" then
    Valor = (Screen.Item("BandaMorta").Value * 4095) / 35
    Application.GetObject("UG2.TM41_00").WriteEx 0
    Application.GetObject("UG2.TM41_01").WriteEx 4096
    Application.GetObject("UG2.TM41_02").WriteEx Valor
    Application.GetObject("UG2.TM41_03").WriteEx 0
    Application.GetObject("UG2.TM41_04").WriteEx 0
    Application.GetObject("UG2.TM41_05").WriteEx 267
    Application.GetObject("UG2.TM41_06").WriteEx 0
    Application.GetObject("UG2.TM41_07").WriteEx 1
End If

If Application.Item("JAN_CTR_UAC").Value = "UAC UG3" then
    Valor = (Screen.Item("BandaMorta").Value * 4095) / 35
    Application.GetObject("UG3.TM41_00").WriteEx 0
    Application.GetObject("UG3.TM41_01").WriteEx 4096
    Application.GetObject("UG3.TM41_02").WriteEx Valor
    Application.GetObject("UG3.TM41_03").WriteEx 0
    Application.GetObject("UG3.TM41_04").WriteEx 0
    Application.GetObject("UG3.TM41_05").WriteEx 267
    Application.GetObject("UG3.TM41_06").WriteEx 0
    Application.GetObject("UG3.TM41_07").WriteEx 1
End If

Screen.Item("Aguarde").Visible = True
End Sub
```

8 CONCLUSÕES

8.1 RESULTADOS ALCANÇADOS

Acredito ter demonstrado neste trabalho a experiência na digitalização e supervisão de um processo produtivo que é gerar energia elétrica, demonstrando a integração de componentes elétricos à CLP (Controlador Lógico Programável) e à SSC (Sistema de Supervisão e Controle) foi descrito neste trabalho fases do processo e partes deste processo, como a programação do CLP Altus, a integração através de rede de comunicação e como o sequenciamento de eventos e alarmes e a criação de telas no SSC, podem determinar o estado de sua instalação através de transmissão destes dados até centros de operação a distância.

8.2 TRABALHOS FUTUROS

Fica a sugestão de trabalhos futuros:

- 1 Administração e manutenção de banco de dados para Sistemas de Supervisão e Controle.
- 2 Desenvolvimento de interfaces gráficas com o usuário executada através da Internet ou intranet via Internet Explorer.
- 3 Fazer um estudo de caso de uma aplicação orientada a objeto para SSC (Sistema de Supervisão e Controle).

REFERÊNCIAS BIBLIOGRÁFICAS

[NATALE-2000]: Natale, Ferdinando. **Automação industrial**, Editora Érica, 2000.

[ALTUS-2004]: Mastertool; Manual de Utilização - Altus – 2004.

[ALTUS-2004]: Mastertool; Manual de Programação - Altus – 2004.

[ELIPSE-2008]: Manual do usuário Software Elipse E3

[ELIPSE-2008]: Manual de referencia scripts Software Elipse E3

[SENAI-2002]: Curso CLP SENAI 2002