



UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

Paulo Roberto Rodrigues dos Santos Júnior

TRABALHO DE CONCLUSÃO DE CURSO

Desenvolvimento de uma bancada didática e guia para aprendizado de Controladores Lógicos Programáveis e sua aplicação em uma esteira transportadora instrumentada.

Campina Grande - PB
Julho de 2019

PAULO ROBERTO RODRIGUES DOS SANTOS JÚNIOR

TRABALHO DE CONCLUSÃO DE CURSO

Desenvolvimento de uma bancada didática e guia para aprendizado de Controladores Lógicos Programáveis e sua aplicação em uma esteira transportadora instrumentada.

Trabalho de Conclusão de Curso submetido à Coordenação do Curso de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Controle e Automação

Campina Grande - PB
Julho de 2019

PAULO ROBERTO RODRIGUES DOS SANTOS JÚNIOR

TRABALHO DE CONCLUSÃO DE CURSO

Desenvolvimento de uma bancada didática e guia para aprendizado de Controladores Lógicos Programáveis e sua aplicação em uma esteira transportadora instrumentada.

Trabalho de Conclusão de Curso submetido à Coordenação do Curso de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Controle e Automação

Aprovado em 15/07/2019

Orientador: George Acioli Júnior D.Sc.
Universidade Federal de Campina Grande
UFCG

Professor Convidado: Rafael Bezerra Correia Lima D.Sc.
Universidade Federal de Campina Grande
UFCG

Campina Grande - PB
Julho de 2019

Dedico este trabalho aos meus pais, tias, e à minha avó, fontes de toda minha determinação e motivação. Além disso, dedico também a todos aqueles que de alguma forma contribuíram para eu concluir mais essa jornada da minha vida.

Agradecimentos

Agradeço primeiramente a Deus, que me proporcionou a vida e tem me dado saúde e inteligência para alcançar meus objetivos, estando Ele sempre presente nos momentos de maiores dificuldades.

Aos meus pais, Paulo e Cosma, que desde cedo batalharam para, dentro de suas possibilidades, oferecerem sempre o melhor para os seus filhos. Às minhas tias, Socorro e Margarida, e à minha avó Cecília, que sempre me deram total suporte, e ajudaram a manter vivo o sonho de um dia me formar. Obrigado por toda a paciência, dedicação e amor.

Agradeço também a toda minha família, que com todo apoio e carinho contribuíram para que eu chegasse até aqui.

Agradeço ao professor George Acioli por aceitar orientar este trabalho, pelos conhecimentos transmitidos, pelas orientações e contribuições.

Agradeço também aos professores Péricles Rezende e Rafael Bezerra por um dia me receberem no LIEC, me direcionando para a ênfase que escolhi justo em um momento do curso em que ainda havia muita indecisão. O conhecimento adquirido no LIEC foi fundamental para a minha formação.

Aos grandes amigos do LIEC, José Adeilmo, Lucas Oliveira e Ravi Helon, pessoas com quem muito aprendi, e passei com certeza, os melhores e os piores momentos do curso.

Além deles, sou grato pela amizade cada vez mais viva dos conterrâneos, Victor Hugo, Evelyne Alves, Lucas de Lima e Paulo César. Estes a quem considero desde o tempo do IFCE.

Finalmente, agradeço a todos que passaram pela minha vida e que contribuíram direta ou indiretamente para a construção de quem sou hoje, pois ainda que eu conseguisse dizer as mais belas palavras, não conseguiria expressar toda a minha gratidão.

“Estou longe de praticar o que entendo, mas o desejo que tenho de praticar é suficiente para me dar a paz”.

Santa Teresinha.

Resumo

O projeto desenvolvido no Laboratório de Automação do LIEC idealizou o desenvolvimento de uma bancada didática de caráter interdisciplinar a ser usado em aulas práticas para alunos de Engenharia Elétrica, para o controle de variáveis, baseada nos Controladores Lógicos Programáveis (CLPs) CompactLogix e ControlLogix da Rockwell Automation. A bancada desenvolvida permitirá aos alunos abordar tópicos de instrumentação e controle industrial através da operação de instrumentos utilizados em aplicações industriais e de malhas de controle. Além disso, o módulo pode ser usado para o aprendizado na área de Redes Industriais e para o desenvolvimento de aplicações de *Software* Supervisório. O uso de bancadas didáticas auxilia no processo de aprendizado e na evolução da educação técnica, estimulando a aquisição do conhecimento. Através das bancadas, os alunos podem experimentar a sensação de atuar sobre um processo real, mas em um ambiente distante deste e, embora seu principal uso seja como atividade didática complementar, acaba sendo uma experiência mais tangível e motivadora quando comparada, por exemplo, à realização de uma aula expositiva ou uma simulação. Assim, partindo dessa premissa é essencial que o engenheiro se familiarize com essa aproximação técnica, que é de fundamental importância para sua formação.

Palavras-chave: Bancada Didática, CLP, Malhas de Controle, Instrumentação.

Abstract

The project developed in the Automation Laboratory of LIEC devised a didactic bench of interdisciplinary character to be used in practical classes for students of Electrical Engineering, for the control of variables, based on the Programmable Logic Controllers (PLCs) CompactLogix and ControlLogix of Rockwell Automation. The developed bench will allow students to address topics of instrumentation and industrial control through the operation of instruments used in industrial applications and control loops. Moreover, the module can be used for learning in the area of Industrial Networks and for the development of Supervisory Software applications. The use of teaching aids stands in the learning process and the development of technical education, encouraging the acquisition of knowledge. Through the stands, students can experience the feeling of working on a real process, but in a distant this and although its main use is as a complementary didactic activity environment, ends up being a more tangible and motivating experience compared, for example, the realization a lecture or a simulation. Thus, based on this premise it is essential that the engineer is familiar with this technical approach, which is of fundamental importance for their formation.

Keywords: Didactic bench, CLP, Control Loop, Instrumentation.

Lista de Figuras

Figura 1	Modelos de CLPs da Allen-Bradley/Rockwell Automation	14
Figura 2	Rotina de execução para o funcionamento do CLP.	17
Figura 3	Partes que compõem o <i>hardware</i> do CLP.	18
Figura 4	Componentes básicos de um CLP.	18
Figura 5	Esquema de uma CPU e sua comunicação com módulos de E/S.	19
Figura 6	Operador como parte do <i>loop</i> de controle.	23
Figura 7	Bancada didática do laboratório de Automação do LIEC	26
Figura 8	Controlador CompactLogix e módulos de E/S 1769.	27
Figura 9	Módulo de Entrada Digital 1769-IQ16.	27
Figura 10	Módulo de Saída Digital 1769-OW8.	28
Figura 11	Módulo de Entrada Analógica 1769-IF4.	28
Figura 12	Módulo de Saída Analógica 1769-OF2.	29
Figura 13	Controlador ControlLogix e módulos de E/S 1756.	29
Figura 14	Módulo de CPU - <i>ControlLogix 5561</i>	30
Figura 15	Módulo de comunicação <i>Ethernet/IP</i>	30
Figura 16	Módulo de Entrada Analógica 1756-IF16.	30
Figura 17	Módulo de Saída Analógica 1756-OF8.	31
Figura 18	Módulo de Entrada Digital 1756-IB16I.	31
Figura 19	Módulo de Saída Digital 1756-OB16E.	32
Figura 20	Terminal <i>FactoryTalk View Machine Edition</i>	33
Figura 21	Terminal PanelView Plus 400.	33
Figura 22	Terminal PanelView Plus 700.	35
Figura 23	Vista frontal da estrutura da Bancada Didática.	36
Figura 24	Detalhes da estrutura da Bancada Didática.	36
Figura 25	Chaves de simulação.	37
Figura 26	Ligação das chaves de simulação e LEDs de supervisão às Entradas Digitais.	38
Figura 27	Detalhe da Interface das Chaves de Simulação	38
Figura 28	Detalhe da Interface de Entradas Digitais e LEDs de supervisão.	39
Figura 29	Detalhe da Interface de Entradas Digitais e LEDs de supervisão.	39
Figura 30	Retificador CA-CC para a fonte de tensão da bancada didática.	40
Figura 31	Fonte de tensão da bancada didática.	40
Figura 32	Interface de Saídas Digitais	41
Figura 33	Detalhe da Interface das Entradas Analógicas	41
Figura 34	Interface de Saídas Analógicas.	42
Figura 35	Interface de Saídas Analógicas.	43
Figura 36	Rede de Comunicação Ethernet entre as bancadas e os PCs.	45
Figura 37	Switch Mercusys - Modelo MS108.	45
Figura 38	Processo de laminação de Alumínio.	46
Figura 39	Desenho técnico com identificação dos dispositivos adicionados.	47
Figura 40	Esteira transportadora com dispositivos integrados.	48
Figura 41	Controle proposto para o módulo de Aquecimento.	49
Figura 42	Controle proposto para o módulo de Resfriamento.	50
Figura 43	Controle de posicionamento para o objeto antes do módulo de aquecimento.	50
Figura 44	Controle de posicionamento para o objeto entre os dois módulos.	51
Figura 45	Controle de posicionamento para o objeto depois do módulo de Resfriamento.	51
Figura 46	Controle <i>on-off</i> no Módulo de Resfriamento.	52

Figura 47	Controle <i>on-off</i> no Módulo de Aquecimento.	52
Figura 48	Visão Geral do Processo ao longo de 30 minutos.	53
Figura 49	Tela inicial do <i>VMware workstation 12 Player</i>	58
Figura 50	Áreas do ambiente de trabalho do RSLogix 5000.	59
Figura 51	Passos iniciais para a criação do projeto.	60
Figura 52	Janela inicial do RSLogix 5000	61
Figura 53	Seleção de um novo módulo.	62
Figura 54	Aba de adição de novos módulos.	62
Figura 55	Escolha e nomeação do novo módulo.	63
Figura 56	Formato do endereço de E/S.	63
Figura 57	Aba <i>Monitor Tags</i> da pasta <i>Controller Tags</i> do projeto.	64
Figura 58	Criação de uma <i>alias</i> para uma entrada digital.	65
Figura 59	Rotina Principal de programação em <i>ladder</i>	65
Figura 60	Janela <i>Who Active</i> , para a seleção do IP do processador do CLP desejado.	66
Figura 61	Chave de seleção de modo no painel do CompactLogix.	67
Figura 62	Diagrama lógico referente ao exercício 1.	68
Figura 63	Elementos <i>ladder</i> ainda sem definição de endereço.	68
Figura 64	Janela de endereçamento de um elemento <i>ladder</i>	68
Figura 65	Criando <i>Aliases</i>	70
Figura 66	<i>Examine if Closed</i>	71
Figura 67	<i>Examine if Open</i>	71
Figura 68	<i>Output Energize</i>	71
Figura 69	<i>Output Latched</i>	72
Figura 70	<i>Output Unlatched</i>	72
Figura 71	<i>One Shot Rising</i>	72
Figura 72	<i>One Shot Falling</i>	73
Figura 73	<i>Timer On Delay</i>	73
Figura 74	<i>Timer Off Delay</i>	74
Figura 75	<i>Count Up</i>	75
Figura 76	<i>Count Down</i>	76
Figura 77	<i>Reset</i>	76
Figura 78	<i>Compare</i>	77
Figura 79	Instruções de Comparação.	78
Figura 80	<i>Compute</i>	79
Figura 81	Instruções Matemáticas/Cálculo.	80
Figura 82	<i>Move</i>	80
Figura 83	<i>Jump to Subroutine</i>	81
Figura 84	Rearme do Controle Mestre.	81
Figura 85	Proporcional, Integral e Derivativo.	82

Lista de Tabelas

Tabela 1	Descrição das teclas para o PanelView Plus 400.	34
Tabela 2	Descrição das teclas para o PanelView Plus 700.	35
Tabela 3	Instrumentação da esteira no cartão de E/S do CLP ControlLogix.	48
Tabela 4	Módulos de E/S para o CompactLogix	61
Tabela 5	Módulos de E/S para o ControlLogix	62
Tabela 6	Informações de E/S apresentadas no conjunto de <i>tags</i>	64
Tabela 7	Proposta de descrições para as <i>tags</i> de entrada.	69
Tabela 8	Proposta de descrições para as <i>tags</i> de saída.	69
Tabela 9	Estrutura do Temporizador TON.	74
Tabela 10	Estrutura do Temporizador TOF.	75
Tabela 11	Estrutura do Contador CTU.	75
Tabela 12	Estrutura do Contador CTD.	76

Sumário

1	Introdução	14
1.1	Objetivos	15
1.2	Metodologia	15
2	Fundamentação Teórica	16
2.1	CLP - Controlador Lógico Programável	16
2.1.1	<i>Backplane</i>	18
2.1.2	CPU	19
2.1.3	Memória	19
2.1.4	Comunicações	20
2.1.5	Entradas/Saídas	20
2.1.5.1	Entradas Digitais e Saídas Digitais	21
2.1.5.2	Entradas e Saídas Analógicas	21
2.2	IHM - Interface Humano-Máquina	22
2.2.1	Sistemas Supervisórios SCADA	23
2.3	Programação de CLPs	24
3	Bancada Didática	25
3.1	Componentes da Bancada Didática	25
3.1.1	CLPs da Rockwell Automation	25
3.1.1.1	<i>CompactLogix</i> e módulos de E/S 1769	25
3.1.1.2	<i>ControlLogix</i> e módulos de E/S 1756	28
3.1.2	Terminais PanelView Plus	32
3.1.2.1	PanelView Plus 400	33
3.1.2.2	PanelView Plus 700	34
3.2	Detalhes Construtivos da Bancada	34
3.3	Máquinas Virtuais	42
3.4	Rede Ethernet de Comunicação	44
4	Esteira Transportadora Instrumentada	46
4.1	Projeto <i>ladder</i> da aplicação	48
4.2	Resultados da Aplicação	50
5	Conclusões	55
6	Referências	56
7	Apêndices	58

A	Guia de programação em <i>ladder</i> no ambiente da Rockwell	58
A.1	Iniciando o RSLogix 5000	58
A.2	Criação de Projetos no RSLogix5000	59
A.3	Adicionando os Módulos de E/S	61
A.4	Ler os dados de E/S	63
A.5	Criar <i>Alias</i> para uma <i>tag</i>	64
A.6	Rotina Principal	65
A.7	Fazer <i>Download/Upload</i> do Projeto	66
A.8	Exercícios	67
A.9	Manual de Referência Básico para Conjunto de Instruções do RSLogix5000	71
A.9.1	Instruções de Bit (XIC, XIO, OTE, OTL, OTU, OSR, OSF)	71
A.9.2	Instruções do Temporizador e do Contador (TON, TOF, CTU, CTD, RES)	73
A.9.3	Instruções de Comparação (CMP, EQU, GEQ, GRT, LEQ, LES, LIM, NEQ)	77
A.9.4	Instruções Matemáticas/Cálculo (CPT, ADD, SUB, MUL, DIV)	79
A.9.5	Instrução de Movimentação (MOV)	80
A.9.6	Instruções de Controle de Programa (JSR, MCR)	81
A.9.7	Instruções Especiais (PID)	82

1 Introdução

Dentre os diversos dispositivos usados na grande área da automação industrial, o controlador lógico programável (CLP) é um dos que recebem maior destaque. A Associação Brasileira de Normas Técnicas (ABNT) o define como um “equipamento eletrônico digital com *hardware* e *software* compatíveis com aplicações industriais” (PAREDE I. M.; GOMES, 2011). A *National Electrical Manufacturers Association* (NEMA) é mais completa e diz que um 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 entrada e saída vários tipos de máquinas e processos” (BRYAN; BRYAN, 1997).



Figura 1: Modelos de CLPs da Allen-Bradley/Rockwell Automation

Fonte: World Wide Web

De maneira a contribuir no processo de aprendizado, na melhoria da educação técnica, e instigar a aquisição do conhecimento através das bancadas didáticas, os alunos poderão experimentar a sensação de atuar sobre um processo real, mas em um ambiente distante deste e, embora seu principal uso seja como atividade didática complementar, acaba por se tornar uma experiência mais tangível e motivadora quando comparada, por exemplo, à realização de uma aula expositiva ou uma simulação (SILVA et al., 2015). Plantas didáticas, assim como plantas pilotos, são plataformas tecnológicas e constituem alternativas práticas no ensino de controle de processos.

1.1 Objetivos

O objetivo principal do trabalho consiste em familiarizar o aluno com a plataforma didática e a programação em *ladder* utilizada pelo *software* RSLogix 5000 para os CLPs *CompactLogix* e *ControlLogix* do fabricante *Allen-Bradley* disponíveis no laboratório de Automação do LIEC, a criação de uma documentação em formato de guia experimental referente ao uso da bancada, e realizar uma aplicação da bancada desenvolvida em uma esteira transportadora instrumentada.

Dos objetivos específicos, busca-se:

- Aprofundar os estudos no âmbito da Automação Industrial.
- Desenvolver habilidades na linguagem de programação *ladder*.
- Desenvolver atividades voltadas ao controle de processos industriais utilizando CLPs.
- Desenvolver um guia para que os futuros alunos possam utilizar os CLPs da *Allen-Bradley* no desenvolvimento de projetos na disciplina de Sistemas de Automação Industrial.

1.2 Metodologia

Inicialmente, foi realizada uma revisão bibliográfica aprofundada, referente ao tema pesquisado, mediante livros, artigos científicos, dissertações, teses, normas regulamentadoras e manuais de instruções.

Posteriormente, foi realizada a implementação do processo da esteira transportadora a ser controlado pelo CLP na bancada didática, o desenvolvimento das rotinas de controle através da programação em *ladder*, e a escrita do guia para o uso da bancada didática.

2 Fundamentação Teórica

2.1 CLP - Controlador Lógico Programável

Como dito anteriormente, a uma definição completa para 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 entrada e saída vários tipos de máquinas e processos” (BRYAN; BRYAN, 1997).

Em outras palavras, é um computador industrial que possui a capacidade de ser programado para desempenhar diversas funções de controle. Seu surgimento, no fim dos anos 60, significou um grande avanço, pois nessa época, o controle era executado por meio de relés com lógica fixa (ou *hardwired*) (PAREDE I. M.; GOMES, 2011). Era comum grandes armários de relés eletromecânicos nas indústrias, que resultava em circuitos complicados, extensas fiações e dor de cabeça na hora da manutenção.

O funcionamento do CLP baseia-se em um conjunto de instruções conhecido como programa, desenvolvido para atender às rotinas do usuário. Os sinais dos sensores ou transdutores acoplados à máquina ou equipamento são aplicados às entradas do controlador. Em cada ciclo, denominado *scan* ou varredura, há três etapas:

- **Etapa de entrada** – Todos os sinais vindos do meio externo e aplicados às entradas são lidos e transferidos para uma área da unidade de memória interna do CLP, chamada tabela imagem da entrada.
- **Etapa de programa** – O programa é executado, usando os sinais adquiridos durante a etapa de entrada e decidindo quais serão os valores dos sinais de saída digitais e analógicos. Esses valores são guardados em uma região de memória denominada imagem de saída. Durante a execução do programa, os sinais de saída não são alterados, a menos que sejam ativadas entradas de elevada prioridade. Tais entradas são utilizadas para emergências, que não podem esperar o término do *scan*. Nesse caso, é interrompido o trabalho do CLP, executando um programa que atende à interrupção. Terminado o atendimento à interrupção, é retomado o programa principal.

- **Etapa de saída** – Todos os valores de saída estabelecidos pelo programa e armazenados na imagem de saída são aplicados aos módulos de saída.

Terminadas as três etapas, reinicia-se o ciclo de *scan*, com a etapa de entrada (operação cíclica). Em algumas situações, por exemplo, ao utilizar controladores PID, torna-se necessário fazer as aquisições de sinais (etapa de entrada) em intervalos fixos. Nesse caso, ao terminar a etapa de saída, o CLP espera o instante predefinido para reiniciar o *scan* (operação periódica). O processo de execução segue aproximadamente a rotina apresentada na figura 2.

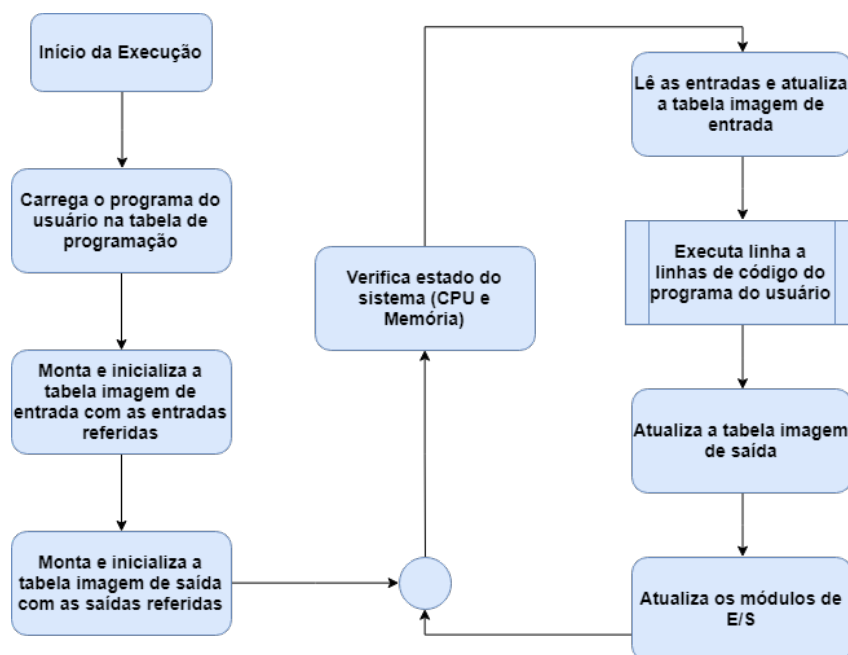


Figura 2: Rotina de execução para o funcionamento do CLP.

Fonte: Adaptado de (PAREDE I. M.; GOMES, 2011)

A principal diferença entre a grande maioria dos computadores e os CLPs é sua maior tolerância a condições mais severas (como poeira, umidade, calor, resfriamento, etc.) e sua quantidade de entradas/saídas para conexão, monitoramento e controle de sensores e atuadores. Além disso, os controladores lógicos programáveis disponíveis no mercado apresentam-se em diferentes formatos, que vão desde pequenos a grandes *racks* capazes de agregar partes distintas e compor sistemas complexos com algumas dezenas de portas de entradas/saídas que podem ser conectadas a outro CLP compondo sistemas de controle supervisão e aquisição de dados (SCADA).

Na figura 3, são apresentados os módulos constituintes de um CLP capazes de agregar partes distintas e compor um sistema.

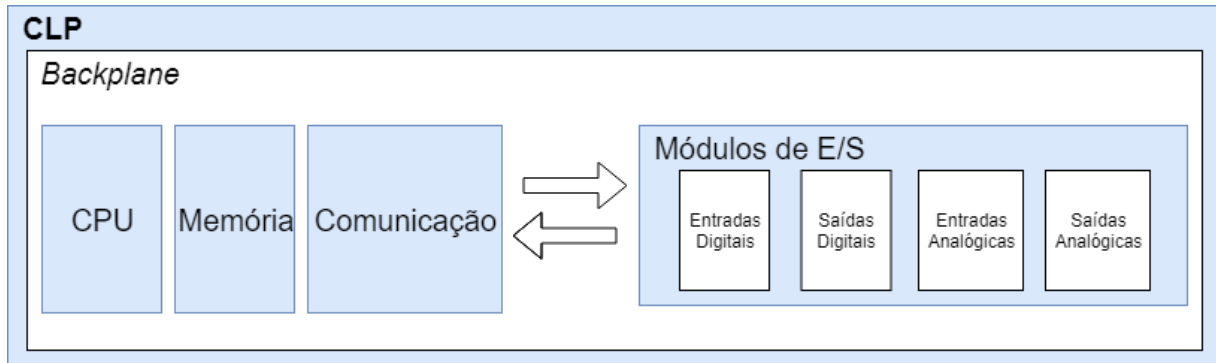
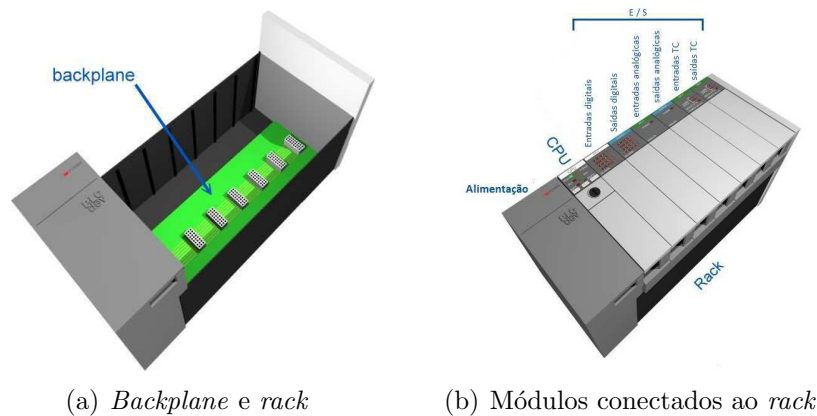


Figura 3: Partes que compõem o *hardware* do CLP.

Fonte: Autoria Própria.

2.1.1 *Backplane*

O *backplane* permite que os módulos sejam conectados e energizados, possibilitando a devida comunicação com a CPU. Sua composição se divide em duas unidades principais: o *rack* e a fonte de alimentação. A última tem a função de fornecer alimentação em corrente contínua (geralmente 24 V com entradas alternadas de 110V ou 220V) para todos os módulos do *rack*.



(a) *Backplane* e *rack*

(b) Módulos conectados ao *rack*

Figura 4: Componentes básicos de um CLP.

Fonte: World Wide Web

O *rack* é a “espinha dorsal” do sistema, com a função de prover o condicionamento físico e elétrico dos barramentos entre os módulos, permitindo que todos os módulos se comuniquem com a CPU. Após a conexão da CPU no *rack*, são adicionados os módulos de comunicações e E/S de modo a compor o sistema necessário para a aplicação desejada. Dependendo da necessidade, *racks* de expansão podem ser conectados ao principal a fim de aumentar o número de *slots*.

2.1.2 CPU

Se o *rack* pode ser visto como a “espinha dorsal” do sistema, a CPU seria o “cérebro” do CLP, pois é nela que está contido o programa que coordena os módulos de modo a executar devidamente as funções. Ela é composta de inúmeros componentes como memórias, microprocessadores, interfaces de E/S e demais *chips* que monitoram a lógica e a comunicação.



Figura 5: Esquema de uma CPU e sua comunicação com módulos de E/S.

Fonte: World Wide Web

A CPU possui 4 modos de operação: Programação, *Run*, *Stop* e *Reset*. No modo Programação é possível inserir uma lógica de programa na memória do dispositivo através de uma conexão com um computador. Após o *upload* de um código válido, o modo *Run* pode ser selecionado para rodar a rotina programada, realizando leituras de entradas, processamento e acionamento de saídas. Também é possível, neste modo, comunicar-se com outros CLPs ou interfaces humano-máquina através de protocolos como RS232, profibus, SCADA, etc.

O modo *Stop*, por sua vez, para a CPU desligando todas as suas saídas e permitindo a checagem das entradas.

No modo *Reset*, o CLP volta às condições iniciais preestabelecidas no programa em sua memória. Este *reset* pode ser feito de duas maneiras: apagando os dados dos E/S, dados e registradores da memória (*Cold Reset*) ou mantendo os dados na memória (*Warm Reset*).

2.1.3 Memória

Existem dois tipos de memórias utilizadas na arquitetura do CLP: memórias de carregamento e de execução. A primeira é responsável por armazenar o programa escrito pelo usuário, esta pode ser RAM, ROM ou EEPROM. A memória de execução consiste em

uma RAM integrada usada para alocar partes do programa e dados durante a execução do mesmo. Normalmente, quaisquer blocos de memória não necessários para a inicialização serão armazenados na memória de carregamento.

2.1.4 Comunicações

Conforme supracitado, os CLPs em geral requerem métodos de comunicação com outros dispositivos ou estações de programação. Em alguns casos, além de uma porta de programação da CPU, também podem ser instaladas placas de rede.

2.1.5 Entradas/Saídas

Um dispositivo conectado a uma porta de entrada do CLP pode prover dados de modo a influenciar o programa provocando uma saída em resposta. As estradas podem ser do tipo digital, analógica, chaveamento, sensores, dispositivos inteligentes e módulos de comunicação.

Geralmente nas entradas são conectados sensores analógicos ou digitais e comandos de outros dispositivos, já nas saídas são ligados os atuadores. Por exemplo, em um sistema de controle de fluxo, transdutores conectados às entradas do CLP enviam medições em tempo real do sistema a ser controlado. De acordo com os dados coletados, as saídas do CLP atuam sobre os motores de modo a controlar o processo de acordo com os parâmetros estabelecidos no modelo.

Os módulos de entrada de sinais discretos abrangem informações de natureza binária do tipo verdadeiro/falso que trabalham em faixas constantes de tensão, como chaves, botões de comando, etc. Estas entradas podem indicar o estado de algum contato externo, ou se determinado dispositivo está alimentado ou não.

Consoante aos módulos de entradas digitais, os módulos de entrada de sinais analógicos informam a amplitude de uma entrada variante no tempo, que, por sua vez, pode ser a tensão lida por um sensor, em uma malha de controle de um determinado processo. Os sinais analógicos dos transdutores, de acordo com os limites de corrente e tensão do dispositivo (usualmente 0-10 V e 4-20 mA), são digitalizados através de um conversor A/D, de acordo com a faixa de variação dos níveis de tensão em questão: cada sinal, seja unipolar ou bipolar, são recebidos por módulos unipolares ou bipolares específicos. Por exemplo, um sinal que varia de -10 a 10 V (isto é, bipolar) deverá ser lido

por um módulo de entrada analógico também bipolar.

Os módulos de saída são os caminhos pelos quais o controlador envia informações ao processo. Informações estas que, assim como as estradas, podem ser digitais ou analógicas e externas ou internas.

Saídas externas são por onde chegam os comandos aos atuadores, isto é, o controlador envia sinais elétricos a equipamentos externos a ele enquanto que as saídas internas enviam dados para dispositivos internos, como um temporizador, por exemplo.

As portas digitais de entrada e saída do CLP, como todo circuito elétrico, possuem limitações de tensão, corrente e frequência de acionamento. Observar esses limites, bem como suas corretas ligações elétricas, é um passo fundamental no projeto ou na manutenção de qualquer sistema automatizado (FREITAS, 2014).

2.1.5.1 Entradas Digitais e Saídas Digitais

Além da frequência, a principal limitação das entradas digitais de um CLP está na tensão de operação. Estes sinais são aqueles que fornecem apenas dois valores, 0 (falso, inativo, desligado) ou 1 (verdadeiro, ativo, ligado). Os sinais digitais são provenientes de diversos equipamentos no processo, tais como: pressostatos, termostatos, chaves de nível, botoeiras, sensores de proximidade e chaves fim de curso.

As saídas digitais são utilizadas para acionar atuadores como lâmpadas, contadoras, válvulas solenoide, motores, entre outros dispositivos, e podem ser encontradas basicamente em três tipos:

- Saída por relé;
- Saída por transistor;
- Saída por tiristor.

2.1.5.2 Entradas e Saídas Analógicas

Muitos dispositivos na indústria trabalham com sinais contínuos e por isso requerem sinais analógicos que sejam compatíveis com sua aplicação. São dispositivos como: transmissores, válvulas proporcionais, sensores, inversores de frequência e analisadores Industriais.

Utilizam sinais analógicos para transmitir ou receber informações de controle ou medição em variáveis como pressão, vazão, nível, temperatura, etc.

O sinal padrão de transmissão eletrônico é o de 4 a 20 mA dc (24Vdc), recomendado pela *International Electromechanical Commission* (IEC), em maio de 1975 (FREITAS, 2014).

Os módulos de entradas analógicas permitem a medição de sinais analógicos em tensão e corrente, para tanto é possível fazer a seleção de dois modos:

- Através de *jumper* no módulo;
- Através de terminais dedicados a cada tipo de sinal, tensão ou corrente.

As saídas analógicas são utilizadas para enviar sinais contínuos para atuadores como:

- Válvulas proporcionais;
- Inversores de frequência;
- Indicadores;
- Conversores I/P(corrente/pressão).

O Inversor de frequência é capaz de controlar diversas variáveis do motor elétrico como torque e velocidade, possibilitando comandar um motor através de um sinal analógico proveniente do CLP.

Na válvula de globo com posicionador, por exemplo, o CLP envia o sinal analógico para o posicionador e este, por sua vez, manipula a posição da válvula de 0 a 100% de acordo com a amplitude do sinal recebido (FREITAS, 2014).

2.2 IHM - Interface Humano-Máquina

Uma IHM é o dispositivo ou o lugar onde os operadores humanos vão manipular a infraestrutura através do ICS (*Industrial Control System*). Uma IHM podem ser telas de vídeo controladas com botões, a toque ou terminais de computador, botoeiras, *feedback* auditivo, luzes intermitentes ou gráficos que exibem e permitem a visualização por telemetria ou *logs*.

A IHM permite também que o operador não apenas visualize a infraestrutura de operação, mas também manipule-a através de funções do tipo DCS (*Distributed Control Systems*). Ferramentas e recursos de visualização de uma IHM podem consistir em

diagramas topológicos, gráficos, mostradores e qualquer outro símbolo de engenharia ou convenção para representar os elementos do processo (MACAULAY; SINGER, 2011).

O estudo e o projeto dessas interfaces são importantes para as características de ergonomia e garantia de que os operadores possam executar suas tarefas com eficiência, conforto e com o mínimo de erros. O aspecto mais importante é provavelmente o imediato espaço de trabalho e ambiente. Desempenho confiável livre de erros não é razoavelmente esperado de um operador que tenha uma dor de cabeça ou dor nas costas dentro de uma hora após o início do trabalho. Fatores como ruído, poeira, cheiro, vibração, umidade, temperatura (e mudanças de temperatura), níveis de iluminação (e ofuscamento) contribuem para a capacidade de concentração do trabalhador (PARR, 2003).

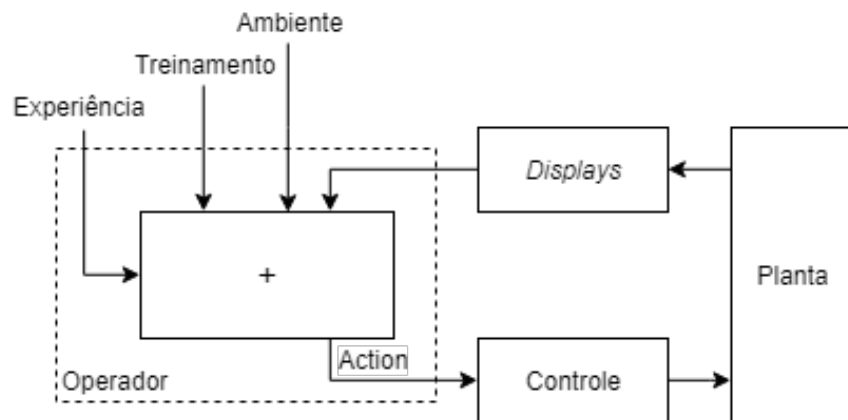


Figura 6: Operador como parte do *loop* de controle.

Fonte: Adaptado de (PARR, 2003)

2.2.1 Sistemas Supervisórios SCADA

Os PCs têm uma capacidade gráfica muito melhor do que o de um CLP médio, portanto, estão cada vez mais sendo usados como o elo entre os sistemas humanos e os CLPs. Um sistema como este, é denominado SCADA (*Supervisory Control and Data Acquisition*). Como o nome indica, eles agem como um supervisor de nível superior e são comumente usados para determinar as configurações dos processos da fábrica e exibir o *status* da planta em telas. Eles também fornecem armazenamento, permitindo que os problemas sejam investigados após o evento (PARR, 2003).

Outras características comuns são os *trendings* (em português, tendências, responsáveis pela produção de gráficos e históricos baseados no tempo de dados da planta) e anúncio de alarme.

2.3 Programação de CLPs

A linguagem *Ladder* foi a primeira linguagem de programação desenvolvida para os CLPs e, como a criação destes foi uma necessidade de substituição do controle de sistemas com relés lógicos, nada mais natural que o *Ladder* fosse similar aos diagramas utilizados para documentar a lógica por relés. Utilizando esta abordagem, os engenheiros e técnicos responsáveis pela programação dos CLPs não precisariam de treinamentos extensos para entender ou desenvolver um programa (BERTULUCCI, 2016).

No entanto a IEC 61131-3 levando em conta aspectos de **modularização**, **estruturação** e **reutilização**, define cinco linguagens de programação padrão para CLPs (TIEGELKAMP; JOHN, 1995):

1. **Lista de Instruções:** Consiste de uma sequência de comandos padronizados correspondentes a funções. Assemelha-se a linguagem *Assembler*.
2. **Texto Estruturado:** É uma linguagem de alto nível muito poderosa, com raízes em Pascal e C. É dita de como uma linguagem de alto nível, porque não usa operadores orientados à máquina em baixo nível, mas oferece uma grande variedade de instruções abstratas que descrevem funcionalidades complexas de maneira muito compacta. Contém todos os elementos essenciais de uma linguagem de programação moderna, incluindo condicionais (*IF-THEN-ELSE* e *CASE OF*) e iterações (*FOR*, *WHILE* e *REPEAT*).
3. **Diagramas em Bloco de Funções:** O diagrama funcional é uma forma gráfica de representação de instruções ou comandos que devem ser executados. Vem originalmente do campo de processamento de sinais, em que valores inteiros e/ou ponto flutuante são importantes. Nesse meio tempo, ela se estabeleceu como uma linguagem universalmente utilizável no campo dos controladores industriais. É baseada em blocos funcionais, por exemplo, uma porta *AND*. Estes blocos são em geral utilizados dentro de lógicas *ladder*.
4. **Ladder:** A linguagem *ladder* é, conforme mencionado anteriormente, a linguagem de programação de CLPs mais comum e a mais difundida, é também conhecida como lógica de diagrama de contatos, pois se assemelha à tradicional notação de diagramas elétricos e de painéis de controle a relés.

5. **Diagrama de Fluxo:** O tipo de estruturação dessa linguagem é uma rede. Os elementos desta rede são chamados de posições e transições. Uma posição pode se encontrar ativa ou inativa. Quando está ativa, um conjunto associado de instruções é executado repetidamente até que a posição se torne inativa. A decisão de alterar o status de uma etapa é definida por uma transição, que é o próximo elemento imediatamente abaixo da etapa. Uma transição é programada por uma condição de transição, que é uma expressão booleana. Quando a expressão se torna *TRUE*, o etapa está pronto para ser desativado. Uma ou mais setas abaixo da transição descrevem qual etapa ou etapas são ativadas quando a etapa atual se torna inativa.

3 Bancada Didática

Bancadas didáticas são ferramentas de auxílio para a realização de experimentos que possibilitam ao operador montar diversos sistemas variando seus parâmetros, ao familiarizar com os componentes e ao mesmo tempo verificar na prática a teoria vista em aula. Com subsídios teóricos citados na fundamentação teórica e os conhecimentos adquiridos ao longo do curso de Engenharia Elétrica, foram definidas como principais características da bancada desenvolvida o controle de variáveis com seus respectivos sistemas supervisórios, conforme as figuras 7(a) e 7(b).

3.1 Componentes da Bancada Didática

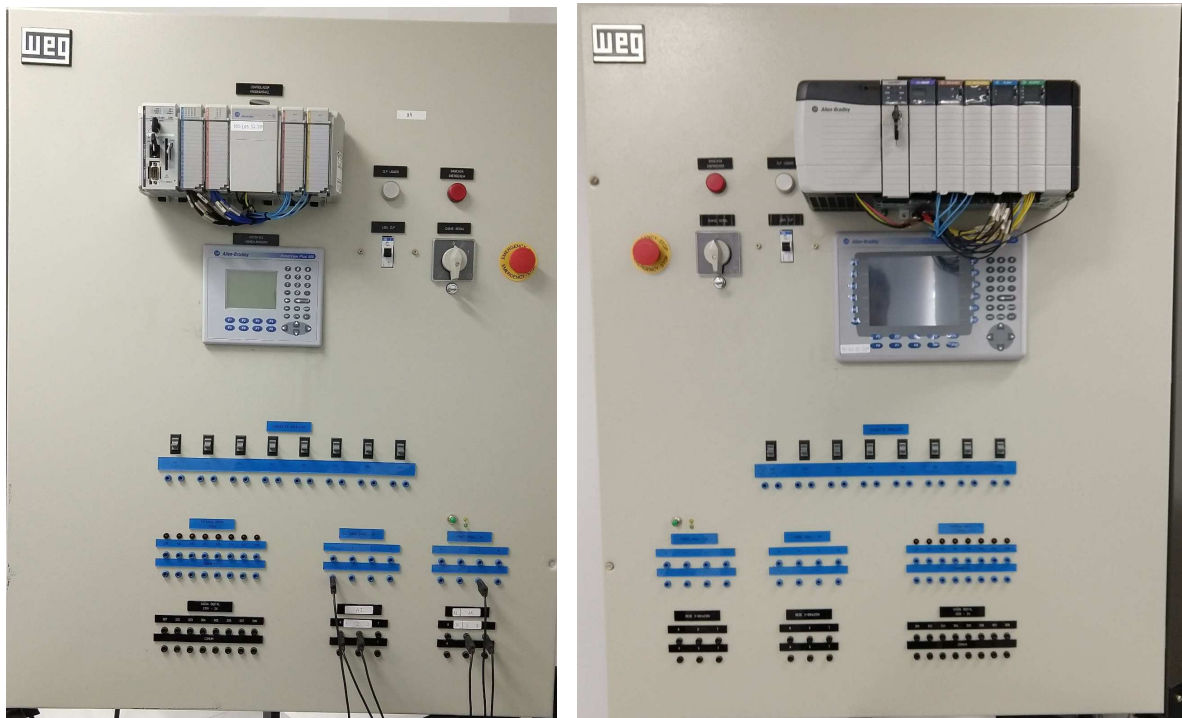
3.1.1 CLPs da Rockwell Automation

A fim de propiciar o aprendizado da programação e uso de controladores lógicos programáveis, em especial os CLPs da *Rockwell Automation*, o laboratório de Automação do LIEC conta com três bancadas didáticas com CLPs da linha *CompactLogix* e uma bancada com um CLP da linha *ControlLogix*.

A título de contextualização, a *Rockwell Automation* é um fornecedor global de soluções em automação industrial, energia, controle e informação. Suas marcas em automação industrial incluem *Allen-Bradley* e *Rockwell Software*.

3.1.1.1 *CompactLogix* e módulos de E/S 1769

O controlador *CompactLogix* é um CLP modular, oferece controle, comunicação e elementos



(a) Controlador CompactLogix e módulos de E/S (b) Controlador ControlLogix e módulos de E/S 1756.1769.

Figura 7: Bancada didática do laboratório de Automação do LIEC

Fonte: Autoria Própria

de E/S de tecnologia de ponta, em um pacote de controle distribuído. Faz parte da família *Logix* de controladores, que oferece um sistema pequeno, poderoso, de custo otimizado que consiste nos seguintes aspectos (ROCKWELL, 2013):

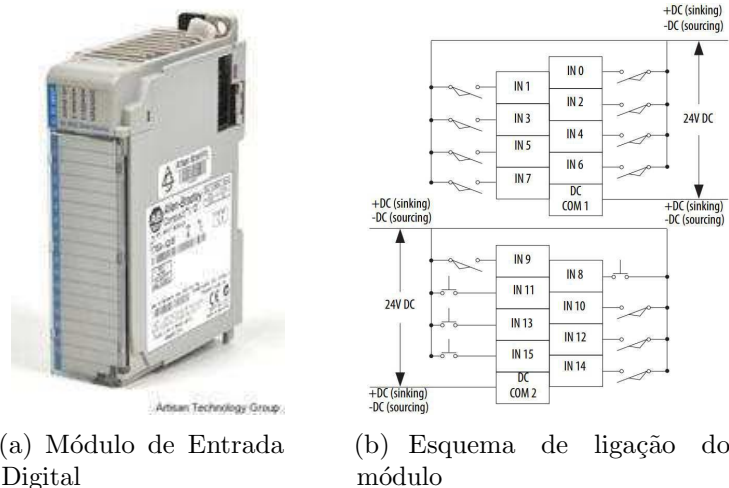
- Software de programação RSLogix 5000;
- Portas de comunicação integradas para redes EtherNet/IP e ControlNet;
- Módulos E/S que oferecem um sistema de E/S compacto, montado em painel ou em trilho DIN;
- Uma porta serial incorporada em todos os controladores CompactLogix.

Os CLPs *CompactLogix* do LIEC possuem quatro módulos de E/S instalados, sendo eles (ROCKWELL, 09/2015):

- 1769-IQ16 DC *Input*: disponibiliza 16 entradas digitais agrupadas em dois grupos de 8 entradas e possui uma alimentação de $24V_{DC}$ *sink/source*. A figura 9 ilustra como é realizado o esquema de ligação do módulo.



Figura 8: Controlador CompactLogix e módulos de E/S 1769.



(a) Módulo de Entrada Digital

(b) Esquema de ligação do módulo

Figura 9: Módulo de Entrada Digital 1769-IQ16.

Fonte: (ROCKWELL, 09/2015)

- *1769-OW8 AC/DC Relay Output*: é um módulo que disponibiliza 8 saídas digitais agrupados em dois grupos de 4 saídas normalmente abertas em que atuação é dada por meio de relés. Possui uma alimentação de 5 a 265 V_{AC} ou 5 a 125 V_{DC} . A figura 10 ilustra como é realizado o esquema de ligação do módulo.
- *1769-IF4 Analog Input*: este módulo analógico possui 4 entradas diferenciais analógicas isoladas ou entradas únicas de sinal, que podem ser de corrente ou tensão, nos seguintes intervalos: $\pm 10V$, 0-10V, 0-5V, 1-5V, 0-20mA, 4-20 mA. A figura 11 ilustra como é realizado o esquema de ligação do módulo.
- *1769-OF2 Analog Output*: por fim, este módulo que possui 2 saídas *single-ended* (saída em um único ponto) analógicas de tensão/corrente, nos seguintes intervalos: $\pm 10V$, 0-10V, 0-5V, 1-5V, 0-20mA, 4-20 mA. A figura 12 ilustra como é realizado o

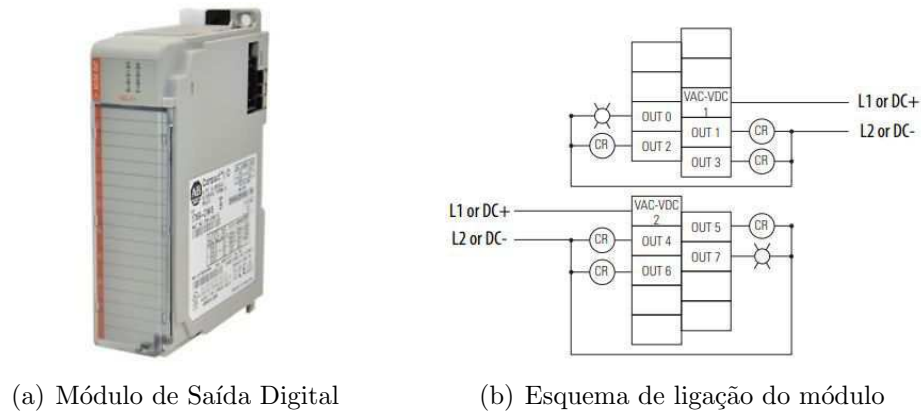


Figura 10: Módulo de Saída Digital 1769-OW8.

Fonte: (ROCKWELL, 09/2015)

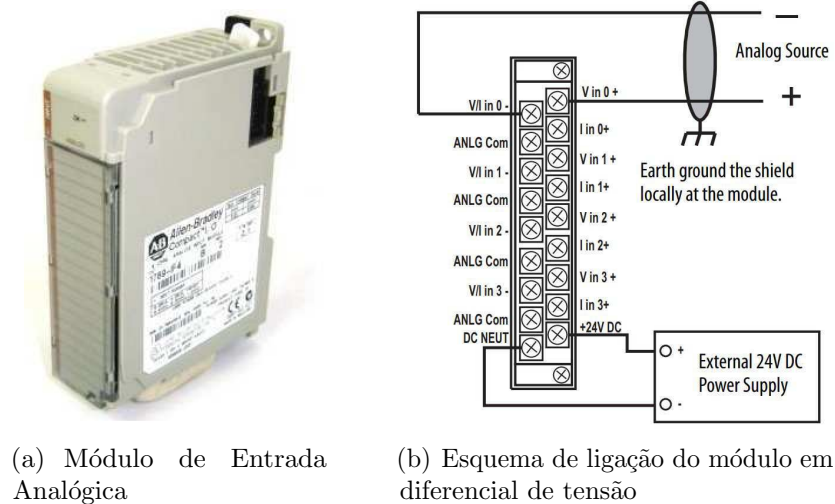


Figura 11: Módulo de Entrada Analógica 1769-IF4.

Fonte: (ROCKWELL, 09/2015)

esquema de ligação do módulo.

Para atingir um sistema com a flexibilidade desejada, estes CLPs possibilitam a instalação de vários controladores em um só *rack*, permite vários controladores conectados por rede, bem como E/S em várias plataformas distribuídas em muitos locais e conectadas por meio de vários links de E/S (ROCKWELL, 2013).

3.1.1.2 *ControlLogix* e módulos de E/S 1756

Diferentemente dos controladores programáveis convencionais, o sistema *ControlLogix* foi projetado em torno da comunicação, suportando redes abertas como Ethernet, ControlNet e DeviceNet, assim como redes *Data Highway Plus* e E/S Remotas (RIO) da Allen-Bradley.

A arquitetura *ControlLogix* combina controle sequencial, de processo e de posição

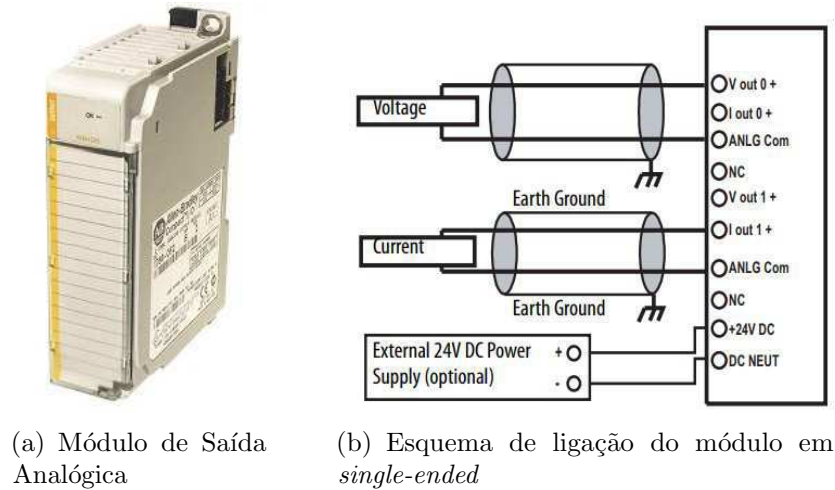


Figura 12: Módulo de Saída Analógica 1769-OF2.

Fonte: (ROCKWELL, 09/2015)

junto com comunicações e E/S de última geração. A flexibilidade oferecida por este sistema permite que ele seja utilizado em uma grande variedade de aplicações em controle.



Figura 13: Controlador ControlLogix e módulos de E/S 1756.

O *ControlLogix* é um CLP cujos módulos são instalados em uma estrutura de *chassi*, que consistem nos *slots*, onde para o caso do LIEC, cinco módulos estão instalados, sendo eles:

- 1756-L61 *ControlLogix5561 Controller*: Corresponde à CPU do controlador, parte responsável pelo controle, comunicação e integridade dos módulos.
- 1756-ENBT *EtherNet/IP Bridge Module*: Módulo responsável pela comunicação do CLP com computadores e outros dispositivos.
- 1756-IF16 *Voltage/Current*: estes são os módulos de entrada de tensão/corrente analógica não isolados *ControlLogix*, que convertem um sinal analógico de volts,



Figura 14: Módulo de CPU - *ControlLogix 5561*.

Fonte: World Wide Web



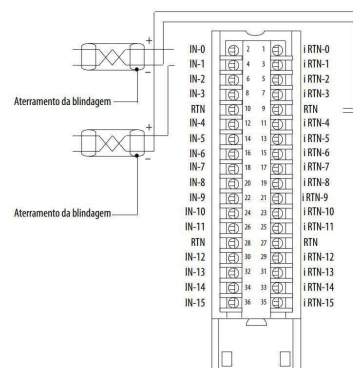
Figura 15: Módulo de comunicação *Ethernet/IP*.

Fonte: World Wide Web

milivolts, miliamperes ou ohms que está conectado aos terminais de parafuso do módulo para um valor digital. O valor digital que representa a magnitude do sinal analógico é então transmitido no *backplane* para um controlador ou para outras entidades de controle. A figura 16 ilustra como é realizado o esquema de ligação do módulo.



(a) Módulo de Entrada Analógica



(b) Esquema de ligação do módulo *single-ended* de tensão

Figura 16: Módulo de Entrada Analógica 1756-IF16.

Fonte: (ROCKWELL, 03/2015)

- 1756-OF8 *Analog Output*: estes são os módulos não isolados de saída analógica do *ControlLogix*, que convertem um valor digital que é entregue ao módulo através do *backplane* em um sinal analógico de -10,5 a 10,5V ou 0 a 21mA. O valor digital

representa a magnitude do sinal analógico desejado. O módulo converte o valor digital em um sinal analógico e fornece esse sinal aos terminais de parafuso do módulo. A figura 17 ilustra como é realizado o esquema de ligação do módulo.

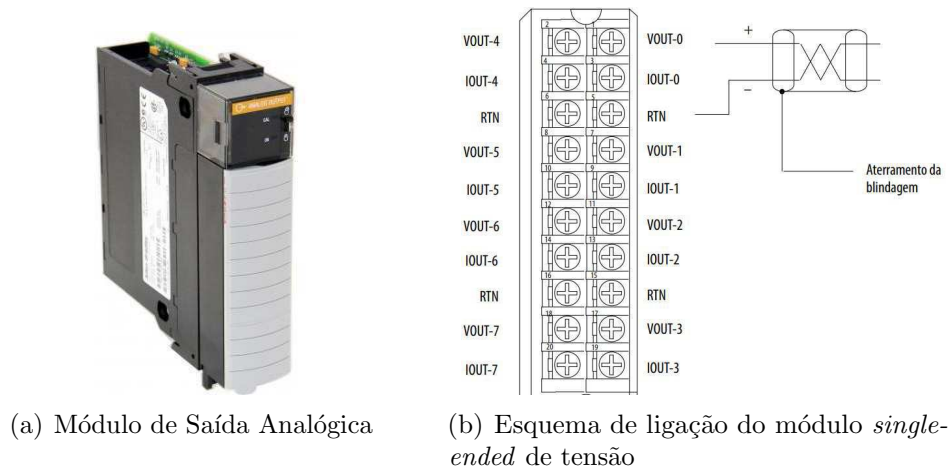


Figura 17: Módulo de Saída Analógica 1756-OF8.

Fonte: (ROCKWELL, 03/2015)

- 1756-IB16I *Isolated DC Input Module*: corresponde ao módulo de entrada isolada de 16 pontos de 10 a 30Vcc e retorna somente dados gerais de entrada e de falha. A figura 18 ilustra como é realizado o esquema de ligação do módulo.

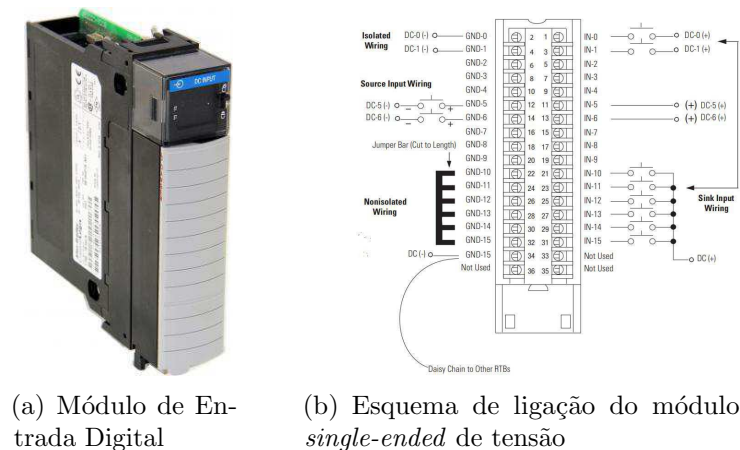


Figura 18: Módulo de Entrada Digital 1756-IB16I.

Fonte: (ROCKWELL, 05/2015)

- 1756-OB16E *Electronically Fused DC Output Module*: corresponde ao módulo de saída com fusível eletrônico, possui 16 pontos de 10 a 31,2Vcc e envia os dados de saída somente do módulo. O módulo retorna o status de fusível queimado com o valor do relógio de sistema (do *rack* local) quando o fusível está queimado ou resetado. O

fusível eletrônico impede que um excesso de corrente flua pelo módulo, danificando o mesmo. A figura 19 ilustra como é realizado o esquema de ligação do módulo.

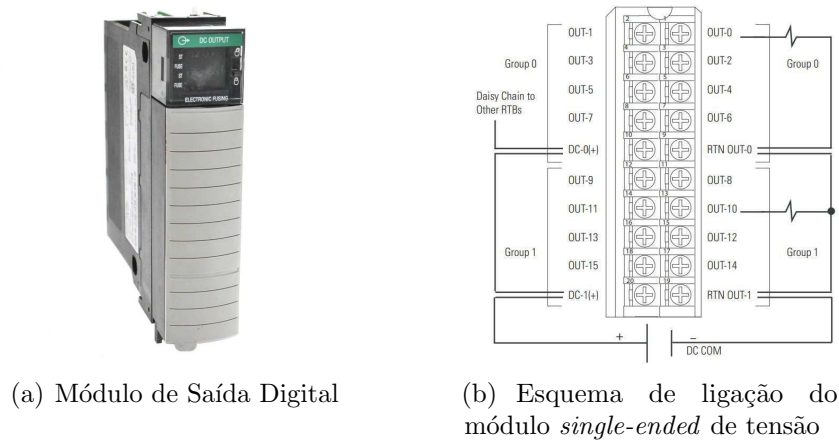


Figura 19: Módulo de Saída Digital 1756-OB16E.

Fonte: (ROCKWELL, 05/2015)

Essa arquitetura flexível permite que processadores, redes e E/S sejam combinados sem restrições em um *chassi*. E à medida que seu sistema cresce, a *ControlNet* fornece o vínculo para distribuir o controle em *chassis* adicionais.

Por serem controladores com módulos bem mais complexos, principalmente no que diz respeito às faixas de tensão/corrente das entradas e saídas analógicas, recomenda-se a leitura dos manuais de instrução e instalação destes, uma vez que contemplam grande parte das informações não apresentadas neste tópico.

3.1.2 Terminais PanelView Plus

Como dito na subseção 2.2, uma IHM podem ser **telas de vídeo** controladas com botões, a toque ou terminais de computador, botoeiras, *feedback* auditivo, luzes intermitentes ou gráficos que exibem e permitem a visualização por telemetria ou *logs*.

Então a família de terminais PanelView Plus, embora tenha sua produção descontinuada, são úteis para o desenvolvimento de IHMs para sistemas supervisórios de processos industriais.

Cada terminal PanelView Plus é pré-carregado com *software* de configuração de tempo de execução e terminal *FactoryTalk View Machine Edition*, que diz respeito ao sistema operacional interno para configurações básicas do terminal como configurações do terminal, carregamento de aplicações, *logs* de diagnóstico entre outros.

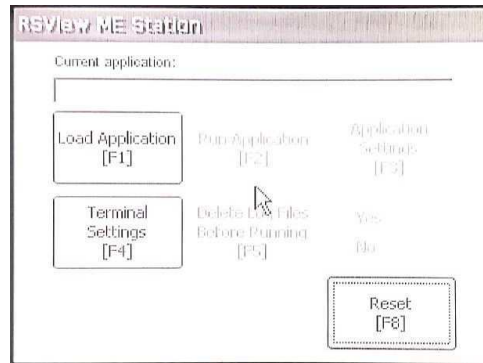


Figura 20: Terminal *FactoryTalk View Machine Edition*.

Fonte: Autoria Própria.

3.1.2.1 PanelView Plus 400

Os terminais PanelView Plus 400 oferecem (ROCKWELL, 2009):

- unidades de configuração base;
- módulos de comunicação;
 - RS-232, interface de *Ethernet* e comunicações modulares.
- fonte de alimentação, CA ou CC;
- visores em tons de cinza e em cores;
- porta USB para conectar mouse, teclado, impressora, *scanner* de código de barras e outros dispositivos.



Figura 21: Terminal PanelView Plus 400.

Fonte: (ROCKWELL, 2009)

Tabela 1: Descrição das teclas para o PanelView Plus 400.

Teclas	Descrição
F1 a F8	Teclas programáveis que iniciam funções no visor do terminal.
Teclado Numérico	Teclas 0 a 9, ., -, Backspace, Enter, Tab, esquerda e direita, Shift
Teclas de navegação	Utilize as teclas de seta para a navegação.

Fonte: (ROCKWELL, 2009)

3.1.2.2 PanelView Plus 700

Os terminais PanelView Plus 700 são dispositivos IHM que oferecem os seguintes recursos (ROCKWELL, 2009):

- módulos de visor gráfico em cores com suporte a teclado, *touchscreen* ou teclado e *touchscreen*;
- *touchscreen* analógica resistiva;
- comunicações de *Ethernet* e serial;
- interface de comunicação modular para facilitar recursos complementares;
- módulos de expansão de memória para upgrades de campo para RAM de 256 MB e CompactFlash de 512 MB;
- Alimentação CA (85 a 264 Vca) ou CC (18 a 32 Vcc);
- o slot para cartão CompactFlash suporta cartões CompactFlash tipo 1;
- as portas USB oferecem conexão a teclado, mouse e impressora;
- molduras substituíveis em campo;
- módulo de lógica padrão.

3.2 Detalhes Construtivos da Bancada

O desenvolvimento dessa bancada didática baseada em CPLs da Rockwell foi uma adaptação da bancada didática a qual estavam instalados CLPs da linha TP-02 da WEG.

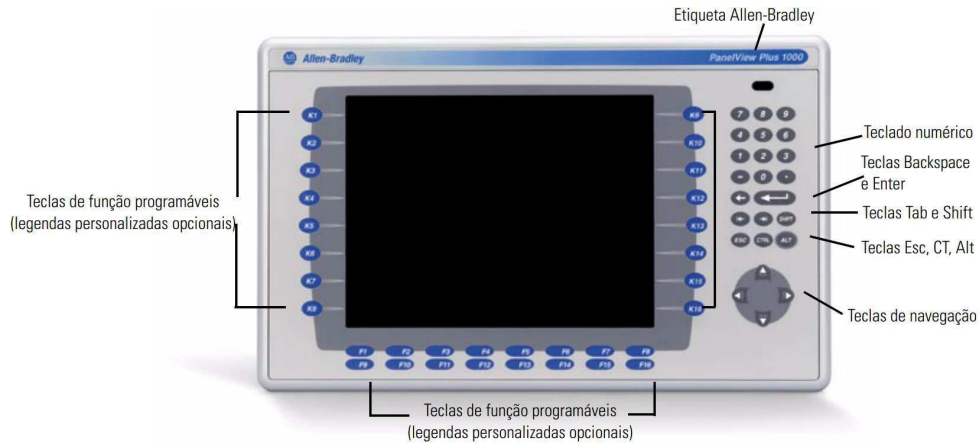


Figura 22: Terminal PanelView Plus 700.

Fonte: (ROCKWELL, 2009)

Tabela 2: Descrição das teclas para o PanelView Plus 700.

Teclas	Descrição
F1 a F10, K1 a K12	Teclas programáveis que iniciam funções no visor do terminal. Legendas substituíveis estão disponíveis para os terminais, permitindo etiquetas personalizadas para teclas de função.
Teclado Numérico	Teclas 0 a 9, ., -, Backspace, Enter, Tab, esquerda e direita, Shift, Esc, Ctrl, Alt.
Teclas de navegação	Utilize as teclas de seta para mover o cursor nas listas e selecionar objetos. Alt+tecla de seta ativa funções de início, final, página acima, página abaixo.

Fonte: (ROCKWELL, 2009)

A motivação por trás da adaptação destas bancadas, além da disponibilidade dos CLPs da Rockwell no LIEC, foi a necessidade de atualização das tecnologias de controle de processos industriais, pois os CPLs da linha TP-02 bem como o seu software de programação PC12 estavam bastante ultrapassados.

Foram então reestruturados os suportes da bancada, de modo que esta se tornasse mais compacta para que a sala da automação comportasse duas estruturas, esta estrutura é mostrada na figura 23.

O tipo de alimentação da bancada que era em tensão trifásica de linha (no caso, $220V_{AC}$ entre fases da saída de um transformador) foi adaptada para a alimentação monofásica ($220V_{AC}$ entre fase e neutro da própria instalação elétrica da sala).

Os controladores *CompactLogix* foram fixados em trilhos DIN instalados sobre a parte frontal da bancada, enquanto o *ControlLogix* foi parafusado na própria bancada.

De modo a detalhar cada componente da parte frontal da bancada, foram enumera-

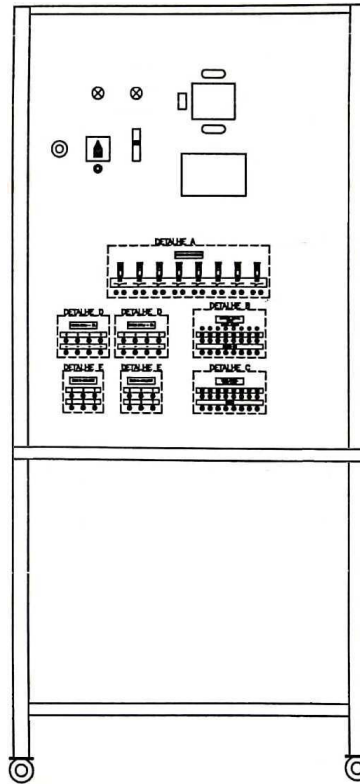


Figura 23: Vista frontal da estrutura da Bancada Didática.

Fonte: Adaptado de (WEG, 2002)

dos cada recurso da bancada, tal como é apresentado na figura 24.

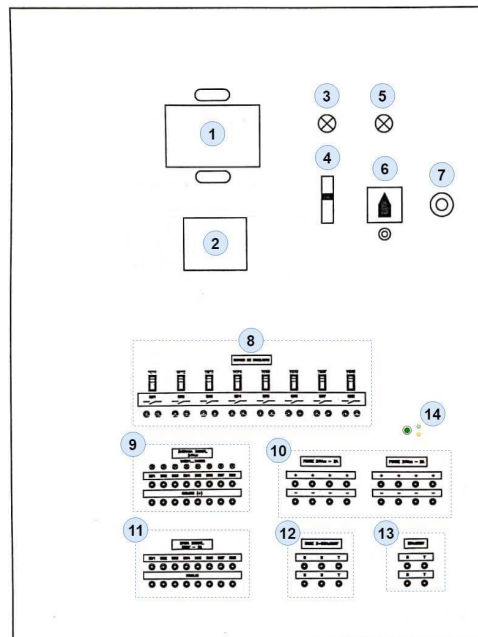


Figura 24: Detalhes da estrutura da Bancada Didática.

Fonte: Adaptado de (WEG, 2002)

Onde se tem:

1. Controlador Lógico Programável (*CompactLogix* ou *ControlLogix*);
2. Terminais PanelView Plus (400 ou 700);
3. Sinaleiro indicativo de CLP Energizado;
4. Disjuntor de 2A para proteção do CLP;
5. Sinaleiro indicativo de Bancada Energizada;
6. Chave Geral Liga(1)/Desliga(0) da Bancada;
7. Botão de Emergência com trava;
8. Chaves de Simulação;
9. Interface de Entradas Digitais com LEDs de indicação de estado;
10. Fontes de Tensão (24V - 2A);
11. Interface de Saídas Digitais;
12. Interface de Entradas Analógicas;
13. Interface de Saídas Analógicas;
14. Sistema de Fusível Eletrônico.

As chaves de simulação possuem três estados: desligado, ligado e pulsante, tal como é apresentado na figura 25.

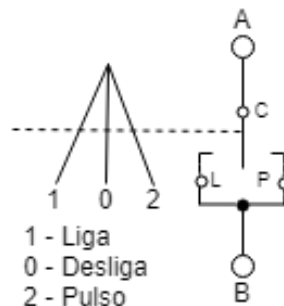


Figura 25: Chaves de simulação.

Fonte: Adaptado de (WEG, 2002)

A sua utilização na bancada é simular as entradas digitais, seja de forma contínua (posição 1), ou apenas para promover um pulsos (posição 2), a depender da lógica

desenvolvida no código, a posição mais adequada deve ser a escolhida. Por exemplo, para usar no comando de ligação de motores elétricos usando contato de selo, uma chave pulsante deve ser a escolhida.

A instalação das oito chaves de simulação às entradas digitais foi feita seguindo esquema apresentado na figura 26:

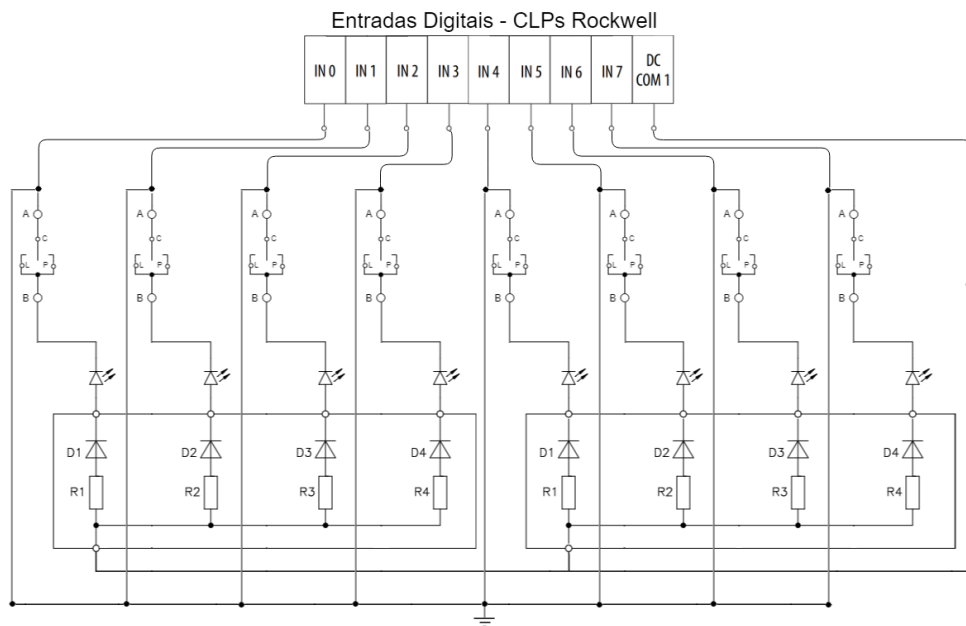


Figura 26: Ligação das chaves de simulação e LEDs de supervisão às Entradas Digitais.

Fonte: Autoria Própria.

Em detalhes, a interface das chaves de simulação pode ser mostrada na figura 27:

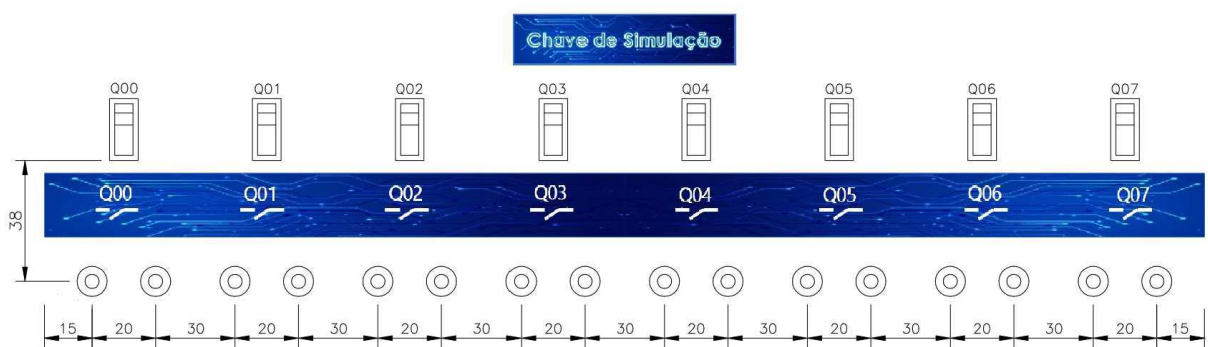


Figura 27: Detalhe da Interface das Chaves de Simulação

Fonte: Adaptado de (WEG, 2002)

A interface de Entradas Digitais via 16 bornes tipo banana para entradas externas com LEDs de supervisão pode ser vista na figura 28:

É importante atentar ao detalhe de que nessas entradas digitais o comum é positivo, no caso do uso destas chaves para simular entradas digitais, o esquema de montagem na

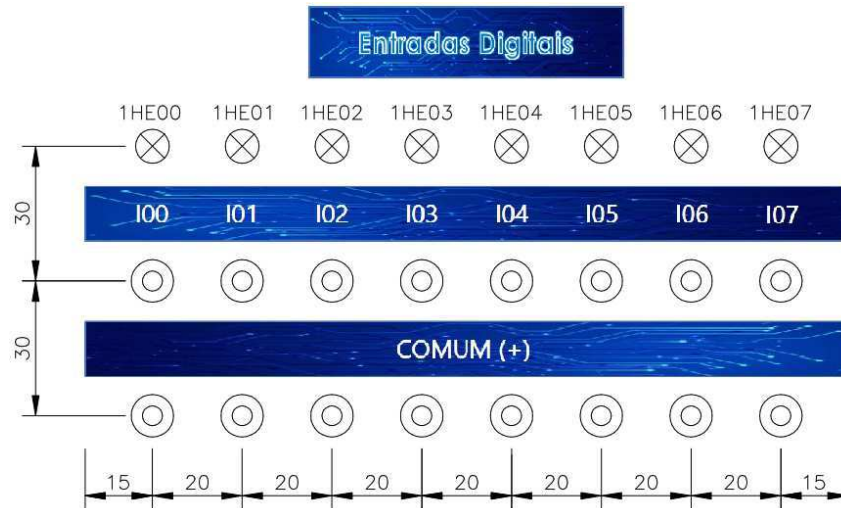


Figura 28: Detalhe da Interface de Entradas Digitais e LEDs de supervisão.

Fonte: Adaptado de (WEG, 2002)

bancada deverá seguir o passo a passo da figura 29:

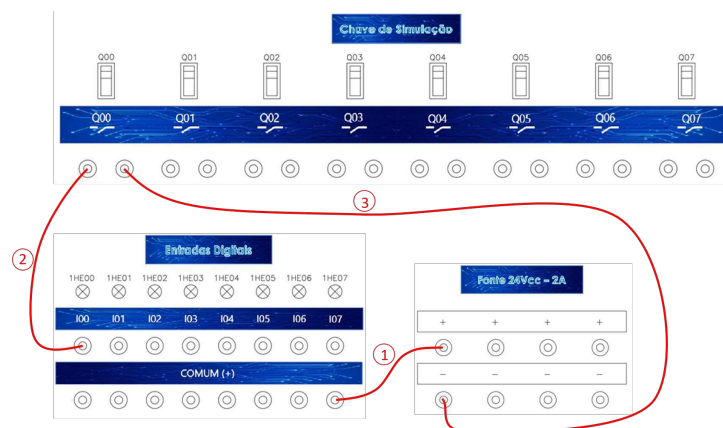


Figura 29: Detalhe da Interface de Entradas Digitais e LEDs de supervisão.

Fonte: Autoria Própria

1. Conectar o positivo da fonte ao Comum(+)
2. Conectar a entrada digital desejada na chave de simulação;
3. Fechar o circuito ligando a chave de simulação ao negativo da fonte.

Seguindo esse passo a passo, é respeitada a montagem segundo o circuito da figura 26, caso não seja ligado desta forma, o cartão de entradas digitais do CLP para este caso não reconhecerá a entrada como válida.

A bancada conta também com uma fonte de tensão de $24V_{DC}$ e 2A, retificada na própria parte interna da bancada segundo o circuito apresentado na figura 30, são oito bornes tipo banana para entradas externas para o terminal positivo e oito para o comum.

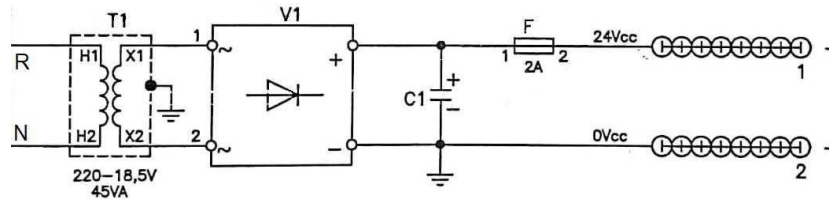


Figura 30: Retificador CA-CC para a fonte de tensão da bancada didática.

Fonte: Adaptado de (WEG, 2002)

A interface para as fontes de tensão pode ser vista na figura 31:

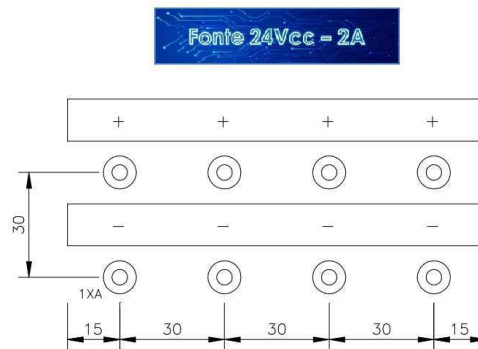


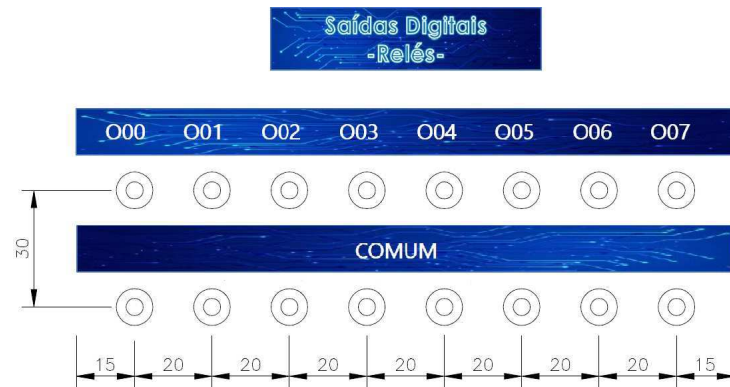
Figura 31: Fonte de tensão da bancada didática.

Fonte: Adaptado de (WEG, 2002)

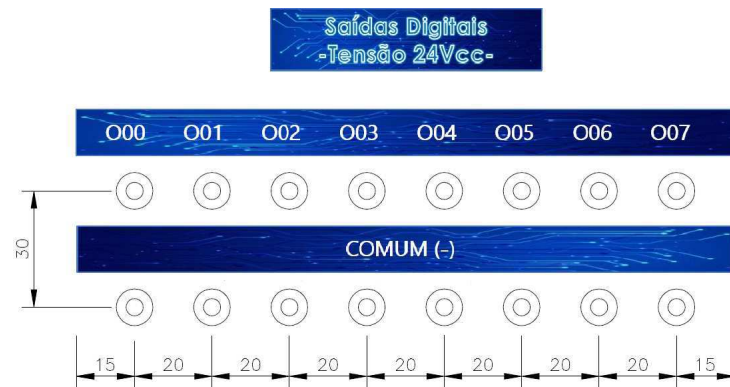
Conforme apresentado na subseção 3.1.1, as saídas digitais dos CLPs *ControlLogix* e *CompactLogix* são de características diferentes, no caso o *ControlLogix* possui saídas digitais em tensão, enquanto o *CompactLogix* possui saídas do tipo relé, desse modo são apresentadas na figura 32 as interfaces para cada caso:

A interface das Entradas Analógicas pode ser mostrada na figura 33, composta por um conjunto de quatro entradas analógicas que medem diferenças de tensão, pois para as aplicações do LIEC em sua maioria os sinais analógicos adotados são em tensão. Vê-se então quatro bornes tipo banana para entradas externas analógicas e quatro bornes para o terminal de referência.

Quanto às saídas analógicas, para os CLPs *CompactLogix* foram instaladas dois bornes tipo banana para as saídas externas em sinal analógico de tensão, pois era o que o cartão 1769-OF2 podia oferecer, e para o comum da saída analógica também foram instalados dois bornes. Já quando se tratou do CLP *ControlLogix*, por possuir maior



(a) Interface de Saídas Digitais tipo Relé



(b) Interface de Saídas Digitais em Tensão

Figura 32: Interface de Saídas Digitas

Fonte: Adaptado de (WEG, 2002)

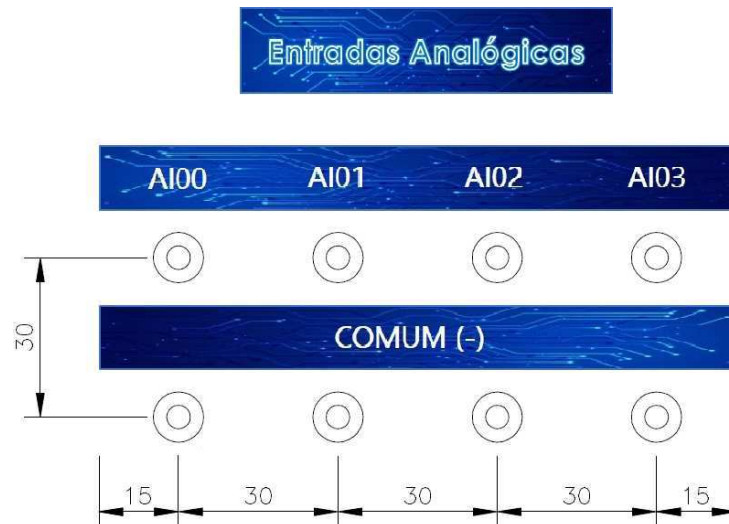


Figura 33: Detalhe da Interface das Entradas Analógicas

Fonte: Adaptado de (WEG, 2002)

robustez para aplicações industriais o seu cartão de saídas analógicas apresenta bem mais saída, por isto, foram instalados três bornes para saída externa em sinal analógico de tensão e três para os seus terminais de referência. A interface para as saídas analógicas

pode ser vista na figura 35.

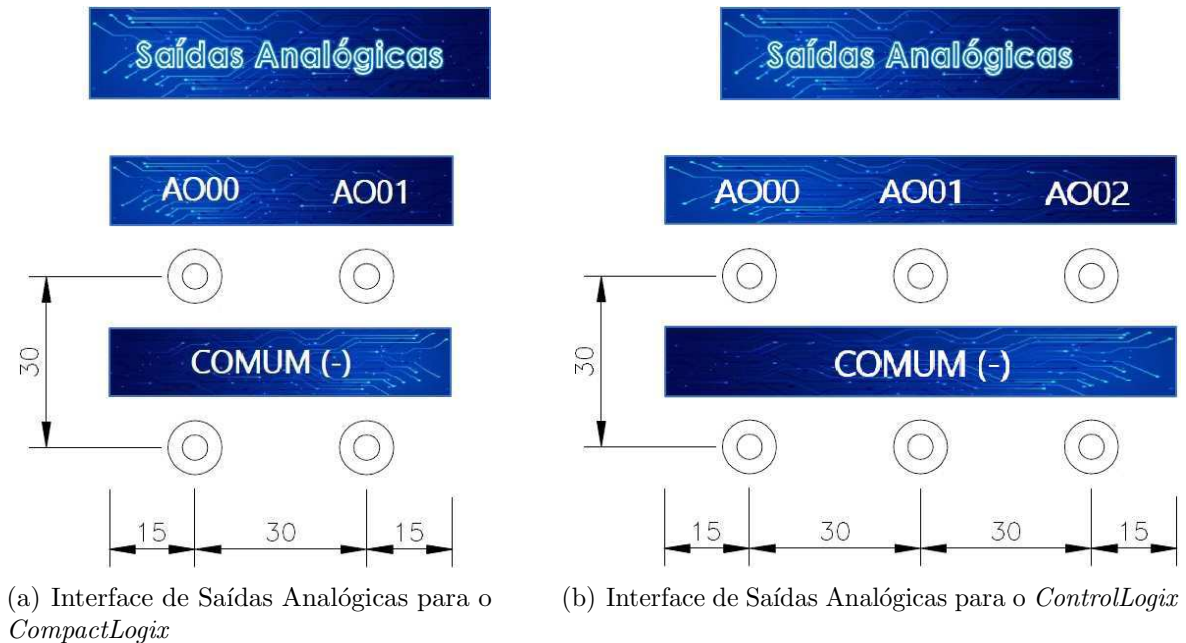


Figura 34: Interface de Saídas Analógicas.

Fonte: Adaptado de (WEG, 2002)

Quanto ao sistema de fusível eletrônico dado pela numeração 14, este consiste em um circuito que detecta possíveis curto-circuitos que possam acontecer na fonte de tensão da bancada, indicando sua atuação com o acendimento do LED amarelo e sinalizando com um alarme sonoro de um *buzzer*. Caso seja pressionado o botão deste sistema é forçada a sua atuação, isto é uma boa prática principalmente quando a bancada se encontra desenergizada, pois o capacitor do retificador apresentado na figura 30 é descarregado garantindo assim tensão nula nos terminais da fonte quando a bancada está desenergizada.

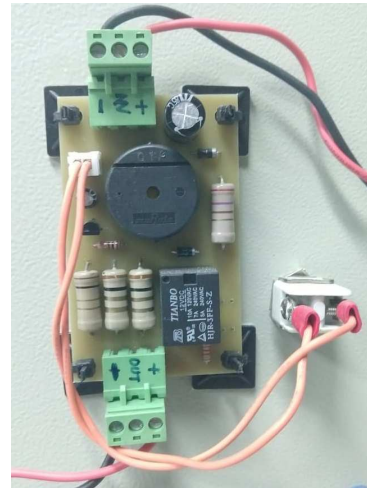
É apresentado na figura 35(a) a parte externa do sistema, com os LEDs de *status* (Verde, funcionamento normal; Amarelo, fusível atuando) e o botão de atuação, e na figura 35(b) o circuito que está na parte interna da bancada, com destaque para o *buzzer* para alarma em status de atuação.

3.3 Máquinas Virtuais

Uma máquina virtual é um arquivo de computador (normalmente chamado de imagem) que se comporta como um computador de verdade. Em outras palavras, é a criação de um computador dentro de um computador. Ela é executada em uma janela, assim como outros programas, proporcionando ao usuário final a mesma experiência na máquina virtual que



(a) Botão de Atuação e LEDs de *status* do Fusível Eletrônico



(b) Circuito do Fusível Eletrônico

Figura 35: Interface de Saídas Analógicas.

Fonte: Autoria Própria.

ele teria no sistema operacional *host*. A máquina virtual fica localizada e uma área restrita, separadamente do resto do sistema, o que significa que o software dentro de uma máquina virtual não pode escapar ou manipular o próprio computador (AZURE, 2019).

Cada máquina virtual conta com seu próprio hardware, incluindo CPUs, memória, discos rígidos, adaptadores de rede e outros dispositivos. Então, o hardware virtual é mapeado para o hardware real no computador físico, o que economiza custos ao reduzir a necessidade de sistemas de hardware físico, juntamente com os custos de manutenção associados, além de reduzir a demanda de energia e refrigeração.

Uma problemática que surgiu na área da automação industrial foi que quando se programa em diversos *softwares* de variadas empresas ou marcas em um mesmo computador real, são utilizados adaptadores de rede e outros recursos que podem levar o computador real a sofrer conflitos de *hardware*. Então a solução adotada para no meio da automação foi instalar todos os recursos de uma plataforma de programação de CLPs em uma máquina virtual diferente dentro da máquina real, por que além de se contornar os problemas de conflito de *hardware*, uma máquina virtual é armazenável em HDs o que facilita na hora da instalação dos *softwares* da plataforma principalmente no que diz respeito ao controle das versões e atualizações, bem como instalar somente os recursos de uma determinada plataforma torna a máquina mais leve para o desenvolvimento das aplicações.

Para o caso da aplicação da bancada didática desenvolvida, em computadores da sala da automação do LIEC foram instaladas e configuradas quatro máquinas virtuais com

o sistema operacional Windows XP, contando com todo o pacote de *software* disponível pela Rockwell Automation, dentre estes, podemos destacar:

- **RSLogix 5000:** faz parte da família RSLogix de pacotes de programação lógica *ladder* compatíveis com a norma IEC-1131. Ajuda a maximizar o desempenho, economizar tempo de desenvolvimento do projeto e melhorar a produtividade. Esta família de produtos foi desenvolvida para operar em sistemas operacionais Microsoft Windows e oferece:
 - Recursos de edição flexíveis e fáceis de usar;
 - Ferramentas de diagnósticos e solução de problemas;
 - Funcionalidades poderosas que poupam tempo.

- **RSLinx:** O RSLinx Classic para redes e dispositivos da Rockwell Automation é uma solução abrangente para gerenciamento de comunicações em fábricas. Ele proporciona acesso de controlador programável Allen-Bradley a uma ampla variedade de aplicativos da Rockwell Software. Isso inclui desde aplicativos de programação e configuração de dispositivos, como o RSLogix e o RSNetWorx, até aplicativos de IHM (Interface Humano-Máquina) (ROCKWELL, 2008).

3.4 Rede Ethernet de Comunicação

Quando se trata de automação, uma das coisas mais importantes ao realizar uma manutenção ou desenvolvimento de programas em plantas industriais é a **comunicação** entre os elementos do processo. Se não houver comunicação entre os dispositivos que compõe o processo, é muito difícil gerenciar e diagnosticar os problemas que surgem.

Daí a necessidade de se estabelecer uma rede confiável de comunicação. As redes de comunicação são muito importantes para que as empresas consigam processar dados, principalmente devido a imensa carga de informações que são criadas e compartilhadas a todo momento. A troca de informações ocorrem, principalmente, por meio de computadores e sensores.

Com base nisso, para que a bancada se comunicasse com os computadores e fosse possível gerenciar esta comunicação com o RSLinx, foi instalada uma rede de comunicação *ethernet* na sala de Automação.

A rede utilizada para a comunicação dos CLPs e IHMs das bancadas para os computadores é bastante simples, baseia-se numa LAN (*Local Area Network*) que utiliza o protocolo *Ethernet* TCP/IP. Faz-se uso de dois *switches* acoplados nas bancadas, estes estando ligados ao *switch* principal da rede da sala de Automação do LIEC. A figura 36 ilustra como a rede está montada.

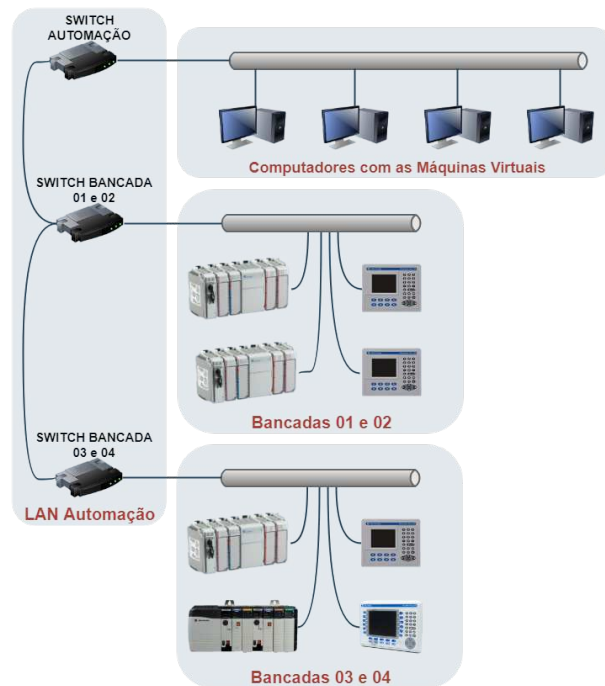


Figura 36: Rede de Comunicação Ethernet entre as bancadas e os PCs.

Fonte: Autoria Própria.

Os *switches* utilizados para a rede da figura 36, são da Mercusys - Modelo MS108 (figura 37) que possui oito portas 10/100Mbps, auto-negociação, Auto-MDI/MDIX, ótimo para a aplicação pois não exige configuração para instalação, bastando apenas plugar e usar.



Figura 37: Switch Mercusys - Modelo MS108.

Fonte: World Wide Web.

4 Esteira Transportadora Instrumentada

Tão importante quanto os CLPs para a maioria dos processos industriais são as esteiras transportadoras, por possibilitarem uma maior agilidade no transporte e manufatura de produtos. Desde a época da Revolução Industrial as esteiras transportadoras possibilitam, por exemplo, determinar o ritmo da produção, característica que muito agradou os empresários do setor. Com isso as esteiras transportadoras garantiram, até os dias de hoje a sua participação massiva no ambiente industrial.

A esteira transportadora continua presente em muitos processos industriais, agregando não apenas a função de transporte dos produtos manufaturados através de suas diversas células de produção, mas também a própria manufatura dos mesmos.

Consideremos por exemplo o processo de laminação do alumínio, a variação da temperatura ocorre durante as várias etapas do processo: num primeiro passo, o bloco a ser laminado é pré-aquecido a 200 °C. Entrando no laminador, o bloco de alumínio tem uma temperatura de cerca de 450 °C. Ao sair do laminador, a temperatura cai para menos de 100 °C.



Figura 38: Processo de laminação de Alumínio.

Fonte: World Wide Web.

Pode-se extrair daí dois importantes aspectos: o controle da temperatura e o controle de posicionamento. Controle de temperatura é evidenciado pelo fato de que para que o alumínio seja moldável, ele deve estar numa temperatura tal que ele se torne maleável ao mesmo tempo em que ele não esteja no estado de fusão. No caso do controle de posicionamento, trata de como será transportado o alumínio entre os estágios de laminação.

Para isto, estudar estratégias de controle que tratem destes aspectos é legítimo.

A esteira transportadora instrumentada é uma das plantas presentes no LIEC, possui dois módulos construídos para compor um processo simples de manufatura. Estes módulos são: o módulo de aquecimento e módulo de resfriamento, estes foram escolhidos para compor um processo simples, devido a sua grande usabilidade. O controle destes módulos será realizado pelo mesmo dispositivo de controle utilizado no controle da posição da esteira: a bancada didática de CLPs da plataforma Rockwell.

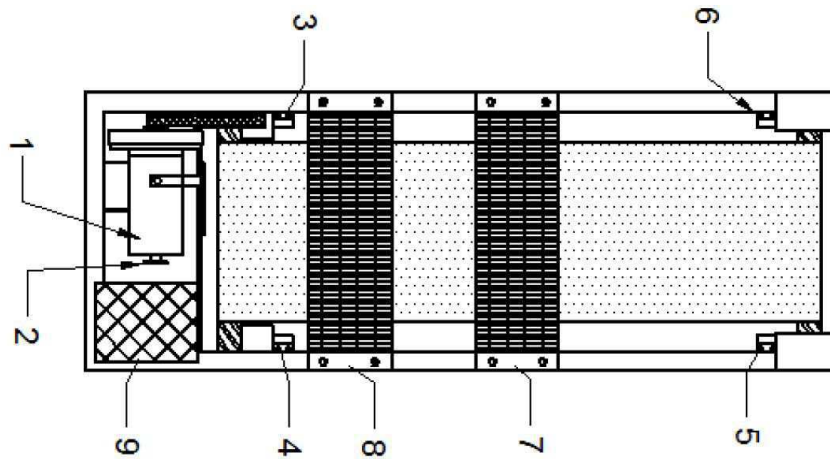


Figura 39: Desenho técnico com identificação dos dispositivos adicionados.

Fonte: (BATISTA, 2015)

Onde:

- 1- Motor DC;
- 2- *Encoder*;
- 3, 4, 5 e 6 - Sensores de barreira óptica;
- 7- Módulo de Resfriamento;
- 8- Módulo de Aquecimento;
- 9- Suportes para fixação de dispositivos conversores e fontes de alimentação.

Então foi realizada uma aplicação desta esteira para verificação da funcionalidade dos principais recursos da bancada didática, de modo que se buscou fazer uso da maior parte da instrumentação da esteira para a realização da aplicação.



Figura 40: Esteira transportadora com dispositivos integrados.

Fonte: Autoria Própria.

4.1 Projeto *ladder* da aplicação

Para o desenvolvimento dos códigos *ladder* da aplicação, inicialmente buscou-se associar a instrumentação da esteira aos módulos de E/S do CLP ControlLogix, segundo mostrado na tabela 3:

Tabela 3: Instrumentação da esteira no cartão de E/S do CLP ControlLogix.

Ligação aos Módulos de E/S	Instrumentação
Entradas Digitais 1756-IB16I	IN-2: Sensor Óptico de Barreira S1 IN-3: Sensor Óptico de Barreira S2 IN-4: Sensor Óptico de Barreira S3 IN-5: Sensor Óptico de Barreira S4 IN-6: Canal A do <i>Encoder</i> IN-7: Canal B do <i>Encoder</i>
Saídas Digitais 1756-OB16E	OUT-0: Alimentação do Módulo de Resfriamento OUT-1: Alimentação do Módulo de Aquecimento
Entradas Analógicas 1756-IF16	IN-0: Sensor LM35 para o Módulo de Resfriamento IN-1: Sensor LM35 para o Módulo de Aquecimento
Saídas Analógicas 1756-OF8	VOUT-0: Controle do PWM do Motor

Com isso, foram desenvolvidas as rotinas em *ladder* com o propósito de:

1. Dada a passagem de um objeto pelo sensor óptico de barreira S1, a esteira desloca o objeto a uma velocidade definida até o módulo de aquecimento. O deslocamento cessa ao interromper o fluxo luminoso do sensor óptico de barreira S2, e iniciando o

controle *on-off* para a temperatura do módulo de aquecimento.

2. Dado o tempo necessário para a realização do controle de temperatura do módulo de aquecimento, uma *flag* indica que a esteira pode prosseguir com o deslocamento do objeto para o módulo de resfriamento. O deslocamento cessa ao interromper o fluxo luminoso do sensor óptico de barreira S3, e iniciando o controle *on-off* para a temperatura do módulo de resfriamento.
3. Dado o tempo necessário para a realização do controle de temperatura do módulo de resfriamento, outra *flag* indica que a esteira pode prosseguir com o deslocamento do objeto para o final do percurso. O deslocamento cessa ao interromper o fluxo luminoso do sensor óptico de barreira S4, e dando fim ao ciclo de operação proposto para a aplicação.

Para o controle de temperatura do módulo de aquecimento, foi proposta a solução em *ladder* da figura 41:

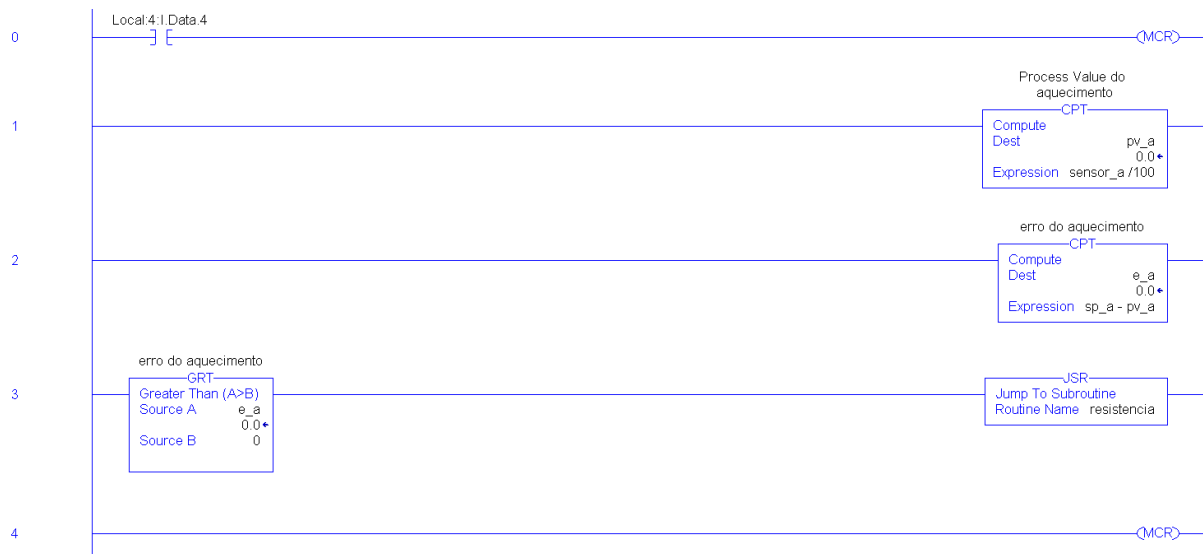


Figura 41: Controle proposto para o módulo de Aquecimento.

Fonte: Autoria Própria.

Para o controle de temperatura do módulo de resfriamento, foi proposta a solução em *ladder* da figura 42:

Para o controle de posicionamento de objetos, foi proposta a solução das figuras 43, 44 e 45:

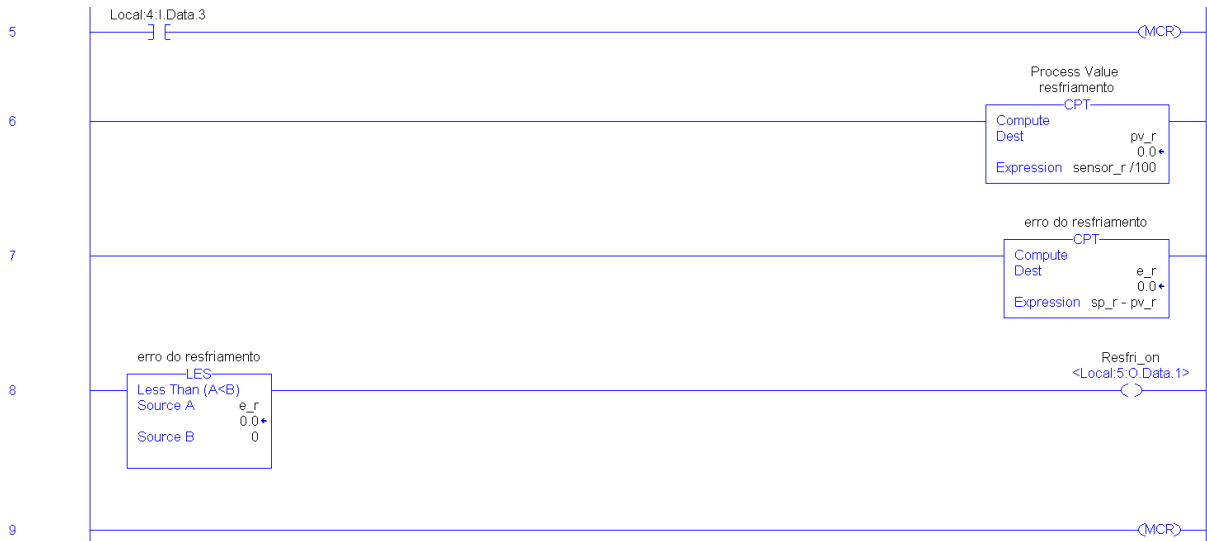


Figura 42: Controle proposto para o módulo de Resfriamento.

Fonte: Autoria Própria.

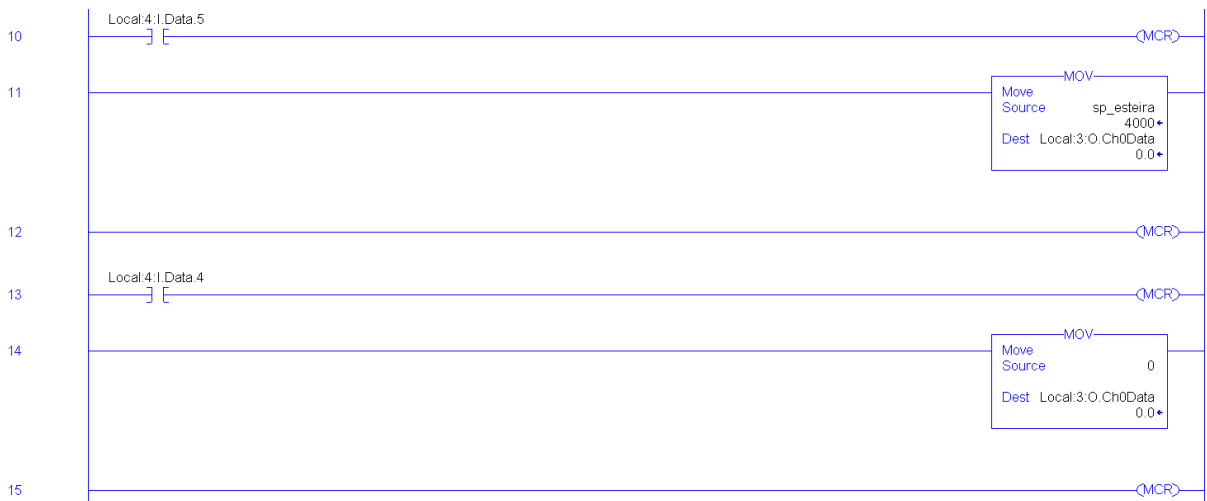


Figura 43: Controle de posicionamento para o objeto antes do módulo de aquecimento.

Fonte: Autoria Própria.

4.2 Resultados da Aplicação

O programa em *ladder* proposto na subseção anterior foi capaz de contemplar as propostas, a execução do controle de posicionamento da esteira se mostrou satisfatório.

Já no que diz respeito ao controle da temperatura nos módulos, os resultados foram obtidos devem ser reavaliados de a melhorar a forma com que o controle é efetuado.

Utilizando a ferramenta *trending* do RSLogix5000, o controle para o resfriamento pode ser visto na figura 46:

Nota-se que a temperatura na saída no módulo de resfriamento tende a seguir a referência, no entanto, no momento em que a temperatura é igual ao valor do *setpoint*, a

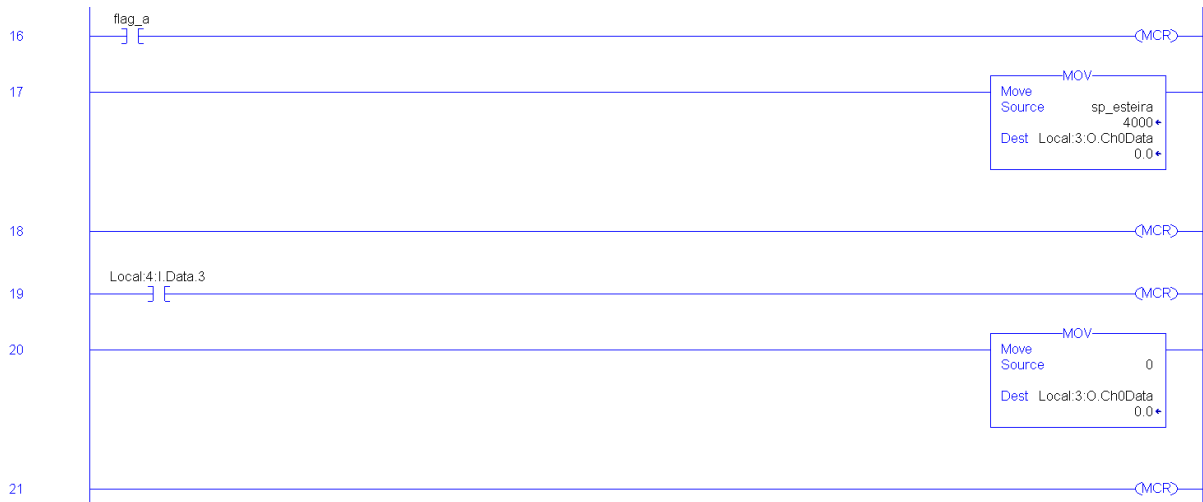


Figura 44: Controle de posicionamento para o objeto entre os dois módulos.

Fonte: Autoria Própria.



Figura 45: Controle de posicionamento para o objeto depois do módulo de Resfriamento.

Fonte: Autoria Própria.

rotina energiza os atuadores do módulo (*coolers*) muitas vezes seguidas, o que explica esses picos que surgem no sinal da temperatura. Isto seria corrigido no entanto ao controlar a alimentação dos *coolers* por meio de sinal PWM.

Da mesma forma, utilizando o *trending*, foi verificado o controle para o módulo de aquecimento, tal como é ilustrado na figura 47:

Este perfil oscilatório que se nota para temperaturas menores que a do *setpoint* de aquecimento é devido a solução adotada para que a subida da temperatura seja mais suave, visto que não havia um PWM para controlar com mais precisão a resistência. Neste caso, a resistência tem sua alimentação interrompida a cada 1 segundo, isso associado ao *cooler* destinado a fazer o que o calor circule dentro do módulo, fez com que a cada

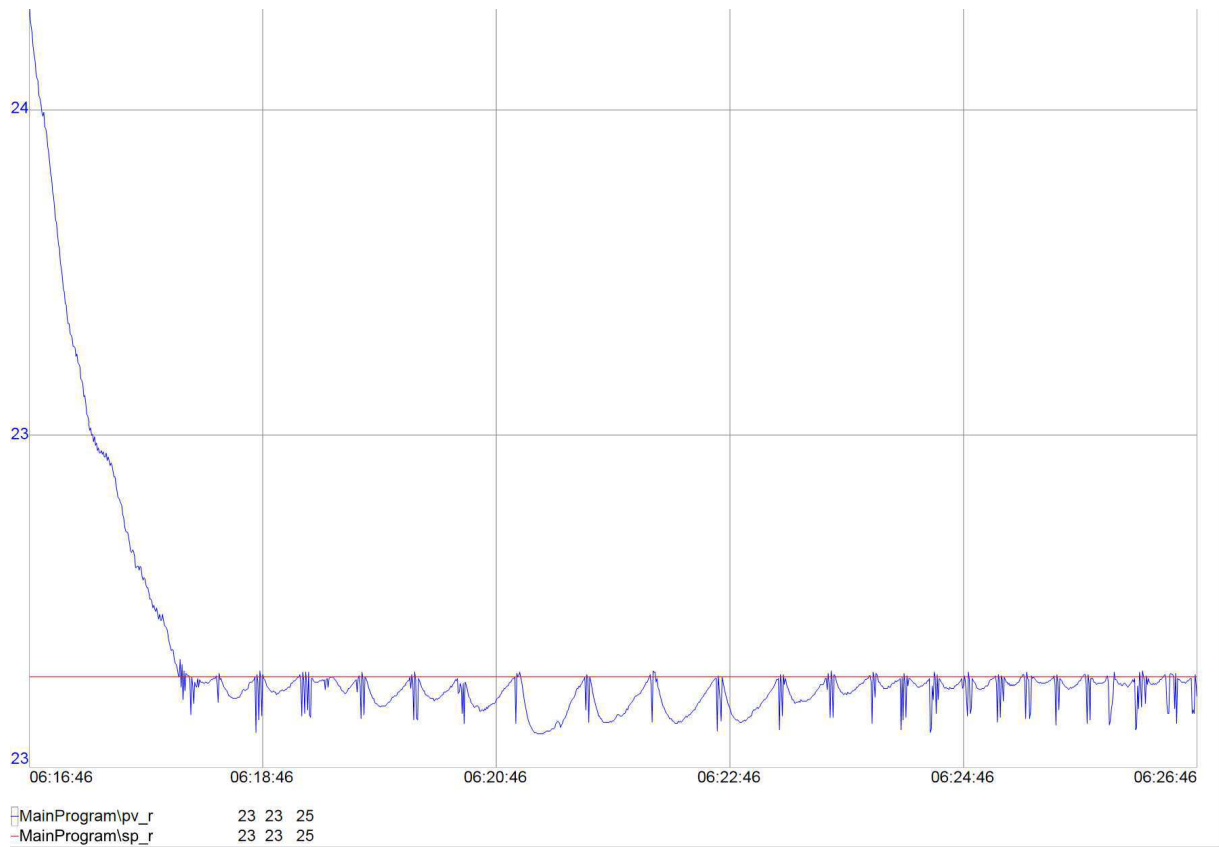


Figura 46: Controle *on-off* no Módulo de Resfriamento.

Fonte: Autoria Própria.

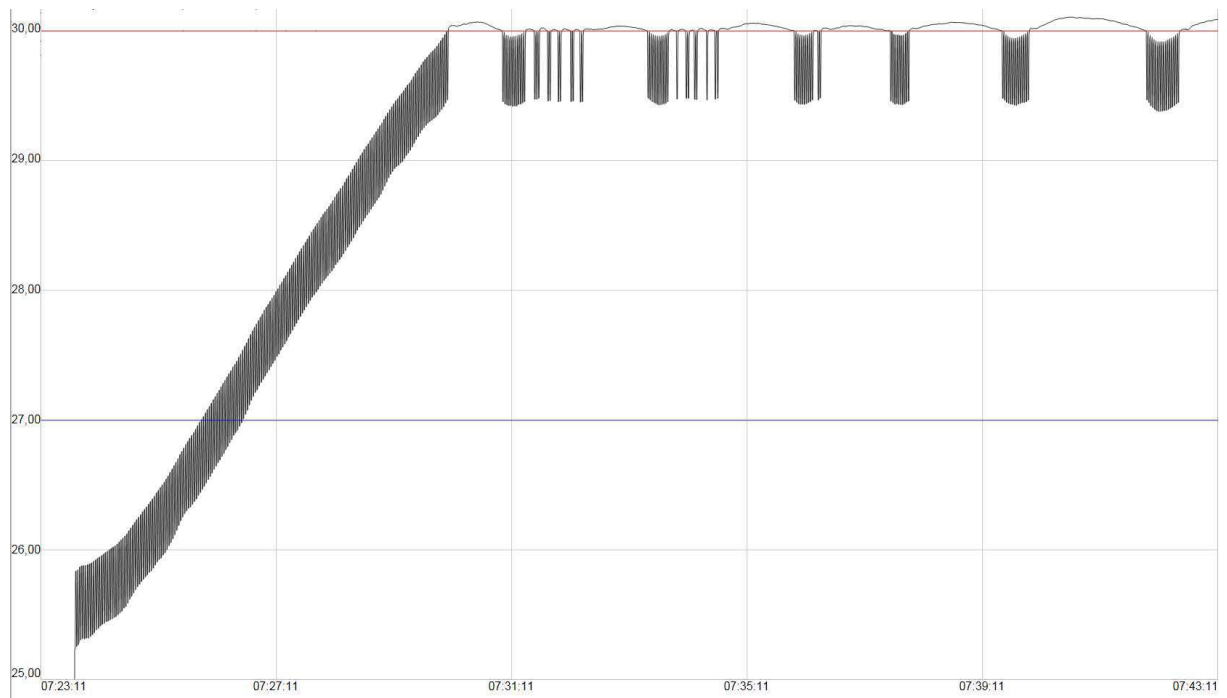


Figura 47: Controle *on-off* no Módulo de Aquecimento.

Fonte: Autoria Própria.

relição da resistência, o sensor lesse uma queda brusca na temperatura.

Avaliando o processo como um todo, ao fazer com que o objeto permanecesse por 20 minutos no módulo de aquecimento e por 10 minutos no módulo de resfriamento, supervisionando o processo pelo *trending*, podemos tirar algumas conclusões da análise da figura 48:

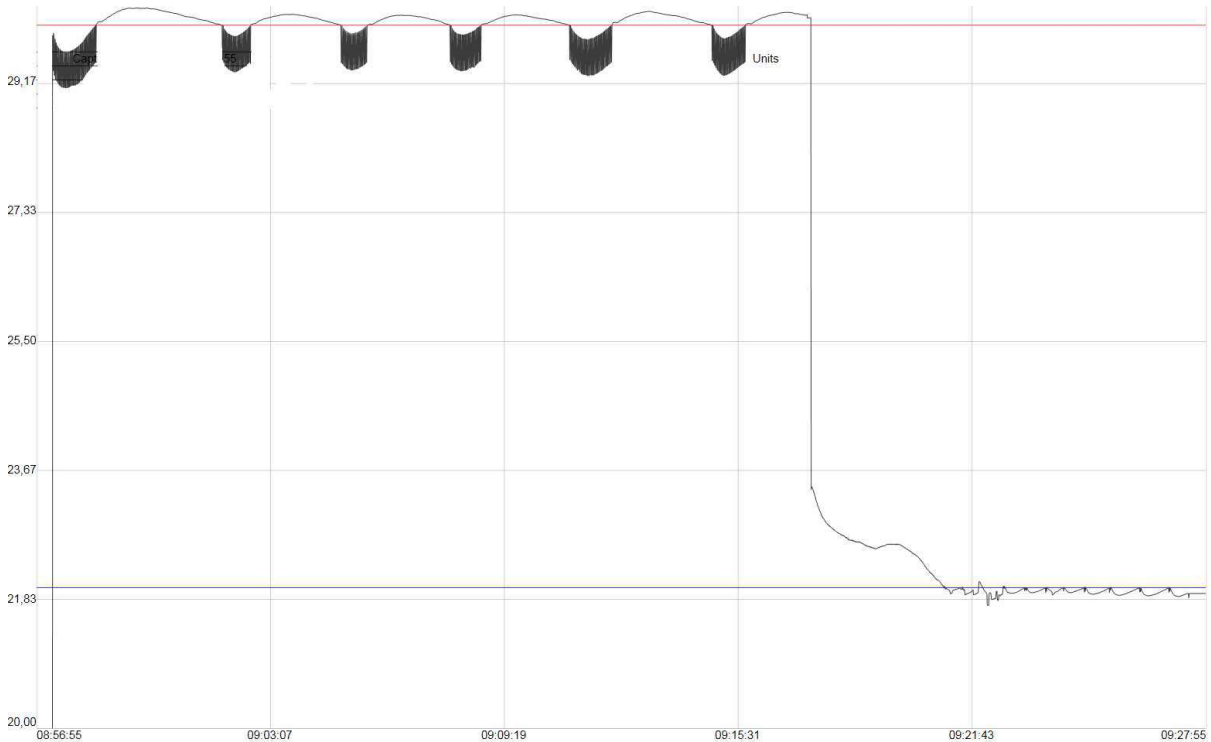


Figura 48: Visão Geral do Processo ao longo de 30 minutos.

Fonte: Autoria Própria.

1. O objeto de alumínio não foi aquecido à temperatura de referência do módulo de aquecimento, isto devido às limitações do próprio módulo, o atuador (Resistência + *Cooler*), não são capazes de promover a excitação suficiente para que o objeto fosse aquecido como esperado. Isto é evidenciado na queda abrupta de temperatura observada após os 20 primeiros minutos.
2. Por não haver um atuador mais eficiente no módulo de resfriamento (dois *coolers*, um levando o ar do ambiente externo para o módulo e o outro retirando), este está limitado a resfriar a temperaturas apenas próximas à temperatura ambiente.

No entanto, à título de aplicação da bancada didática desenvolvida, percebeu-se que esta cumpriu os requisitos de oferecer o suporte ao controle de processo, seja no controle

de forma remota das variáveis, nas ferramentas gráficas de supervisão, na capacidade de diagnóstico apresentado e entre vários outros aspectos importantes para o controle do processo da esteira instrumentada.

5 Conclusões

No contexto acadêmico alunos e professores de modo direto estão vinculados a diferentes formas de conhecimento, o aprendizado é de fundamental importância, a atividade prática como ferramenta é essencial para despertar a curiosidade e o interesse das pessoas por adquirirem novas habilidades ou conhecimentos a fim de melhorar seu desempenho.

O uso de bancadas didáticas auxilia no processo de aprendizado e na evolução da educação técnica, estimulando a aquisição do conhecimento, permitindo que o aluno em formação se familiarize com práticas relativas aos assuntos literários vistos em sala de aula e nos laboratórios, que é de fundamental importância por possibilitarem a criação de ambientes e situações controladas, permitindo, simultaneamente com a fundamentação teórica o desenvolvimento de soluções para problemas reais existentes na indústria.

A partir dos resultados demonstrados ao longo do desenvolvimento do projeto da bancada didática, pode-se concluir que se atingiu o objetivo de, baseado nos CLPs CompactLogix e ControlLogix da Rockwell Automation, propiciar uma plataforma consistente com o intuito de melhorar e aperfeiçoar a forma de aprendizado dos acadêmicos da UFCG, em especial aqueles que devem cursar a disciplina de Sistemas de Automação Industrial, o que irá fomentar nestes alunos, o interesse no desenvolvimento de novos experimentos acadêmicos, principalmente no que tange essa importante área de concentração que é a automação industrial.

6 Referências

- AZURE. *O que é uma máquina virtual?* 2019. Disponível em: <https://azure.microsoft.com/pt-br/overview/what-is-a-virtual-machine/>. Citado na página 43.
- BATISTA, E. D. S. *Trabalho de Conclusão de Curso: INSTRUMENTAÇÃO E CONTROLE DE ESTEIRA TRANSPORTADORA UTILIZANDO CLP*. [S.l.: s.n.], 2015. Citado na página 47.
- BERTULUCCI, C. *Como Funciona a Linguagem LADDER*. 2016. Disponível em: <https://www.citisystems.com.br/linguagem-ladder/>. Citado na página 24.
- BRYAN, L. A.; BRYAN, E. A. *Programmable controllers: theory and implementation*. [S.l.]: Industrial Text Company, 1997. Citado 2 vezes nas páginas 14 e 16.
- FREITAS, C. M. *CLP - Controlador Lógico Programável*. 2014. Disponível em: <https://www.embarcados.com.br/controlador-logico-programavel-clp-parte-2/>. Citado 2 vezes nas páginas 21 e 22.
- MACAULAY, T.; SINGER, B. L. *Cybersecurity for industrial control systems: SCADA, DCS, PLC, HMI, and SIS*. [S.l.]: Auerbach Publications, 2011. Citado na página 23.
- PAREDE I. M.; GOMES, L. E. L. H. E. *Eletrônica: Automação Industrial*. 6. ed.. ed. [S.l.]: São Paulo: Fundação Padre Anchieta., 2011. ISBN 978-85-8028-050-0. Citado 3 vezes nas páginas 14, 16 e 17.
- PARR, E. A. *Programmable controllers: an engineer's guide*. [S.l.]: Newnes, 2003. Citado na página 23.
- ROCKWELL. *Manual do usuário - Módulos de E/S analógica ControlLogix*. [S.l.]: Publicação 1756-UM009D-PT-P, 03/2015. Citado 2 vezes nas páginas 30 e 31.
- ROCKWELL. *Manual do usuário - Módulos de E/S digital ControlLogix*. [S.l.]: Publicação 1756-UM058H-PT-P, 05/2015. Citado 2 vezes nas páginas 31 e 32.
- ROCKWELL. *Technical Data - 1769 Compact I/O Modules Specifications*. [S.l.]: Publication 1769-TD006F-EN-P, 09/2015. Citado 4 vezes nas páginas 26, 27, 28 e 29.
- ROCKWELL. *RSLinx Classic: Obtendo resultados com o Guia*. [S.l.]: Publication Linx-GR001A-PT-E, 2008. Citado na página 44.
- ROCKWELL. *Manual do Usuário: Terminais PanelView Plus - terminais 400, 600, 700, 1000, 1250, 1500*. [S.l.]: Publicação 2711P-UM001J-PT-P, 2009. Citado 3 vezes nas páginas 33, 34 e 35.
- ROCKWELL. *Manual do usuário dos controladores CompactLogix 1769*. [S.l.]: Publicação 1769-UM011I-PT-P, 2013. Citado 2 vezes nas páginas 26 e 28.
- SILVA, A. M. da et al. Bancada didática baseada em clp compactlogix da rockwell para uso nas disciplinas de automação e instrumentação industrial. *TECNOLOGIA & INFORMAÇÃO-ISSN 2318-9622*, v. 2, n. 3, p. 28–43, 2015. Citado na página 14.

TIEGELKAMP, M.; JOHN, K.-H. *IEC 61131-3: Programming industrial automation systems*. [S.l.]: Springer, 1995. Citado na página 24.

WEG. *Manual do Instrutor: Bancada didática CPW TP-02 Automação c/ Controladores Programáveis. Publicação 0899.1804-MN03*. [S.l.]: WEG Industrias S/A - Divisão Automação, 2002. Citado 7 vezes nas páginas 36, 37, 38, 39, 40, 41 e 42.

7 Apêndices

A Guia de programação em *ladder* no ambiente da Rockwell

O *software* RSLogix 5000 é utilizado para programar os CLPs CompactLogix e ControlLogix das bancada didáticas da Automação do LIEC. Nele, é possível escrever um programa em *ladder* ou nas outras linguagens da norma IEC 61131-3 (exceto “Lista de Instruções”).

Os tópicos seguintes correspondem a um tutorial que o guiará na criação de um projeto, na edição de um programa simples em *ladder*, no carregamento deste programa no CLP e na monitoração das variáveis do programa.

A.1 Iniciando o RSLogix 5000

A criação de um projeto deve ser o primeiro passo ao se começar a trabalhar com os controladores da família Logix5000. Para a criação de um projeto será utilizado o *software* RSLogix 5000 instalado na máquina virtual.

Para utilizar a plataforma virtual, deve-se abrir o programa *VMware Workstation 12 Player*, selecionar a opção **Rockwell** → **Play virtual machine**.

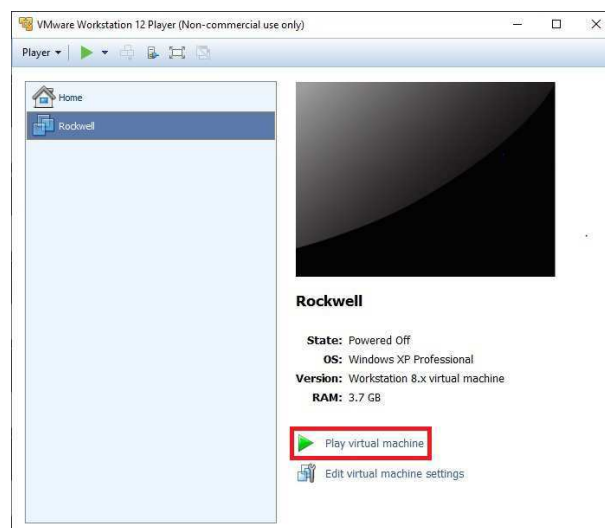


Figura 49: Tela inicial do *VMware workstation 12 Player*.

Em seguida será inicializado o Windows XP, e nele estarão contidas todas as aplicações de *software* relativas ao pacote RSLogix.

Estando a máquina virtual iniciada, para abrir o programa, selecione *Menu Iniciar* → *All Programs* → *Rockwell Software* → *RSLogix 5000 Enterprise Series* → *RSLogix 5000*. A tela que será apresentada estará vazia, pelo fato de não ter sido ainda criado ou carregado nenhum projeto. Contudo, quando um projeto estiver carregado, a tela do RSLogix 5000 apresentará os seguintes campos:

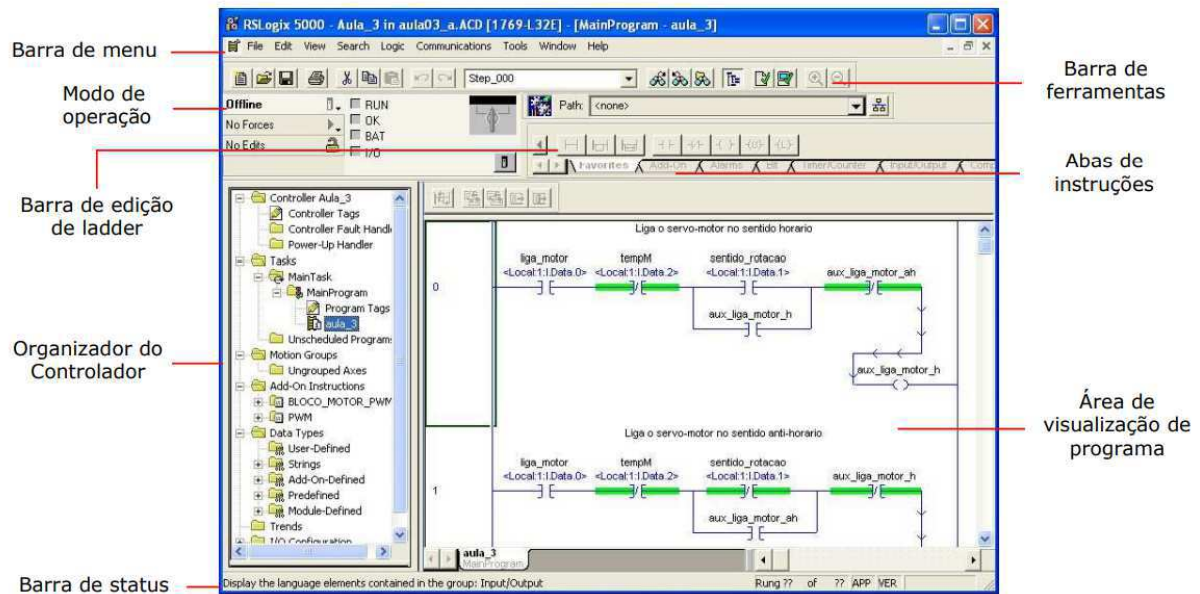


Figura 50: Áreas do ambiente de trabalho do RSLogix 5000.

A.2 Criação de Projetos no RSLogix5000

Para criar um novo projeto, na janela do RSLogix, selecione o menu *File* → *New* (ou use as teclas de atalho CTRL + N). Uma janela de nome *New Controller* será apresentada, siga então o passo a passo da figura 51.

Inicialmente, crie um diretório para o armazenamento dos seus projetos. Em seguida, na janela *New Controller*, selecione o tipo do controlador (*CompactLogix 1769-L32E* ou *ControlLogix 1756-L61 ControlLogix5561 Controller*), a revisão do software como sendo 15, atribua um nome para o projeto, escreva uma descrição (opcional) e defina a pasta onde o projeto será criado como sendo o diretório anteriormente criado.

Atente ao detalhe que para a criação de projetos no ControlLogix, deve-se selecionar o tipo do *chassi*, e o *slot* no qual se encontra o módulo de CPU. Para o caso do ControlLogix do LIEC, o *chassi* possui 7 *slots*, neste caso selecione A7.

Observações:

- Ao dar nomes para o controlador, *tags*, rotinas, módulos de E/S etc, deve-se usar apenas letras, números e *underscore* (“-”), sendo que o primeiro caractere não pode ser um número;
- Não há distinção entre letras maiúsculas e minúsculas.
- devem começar com uma letra ou sublinha;
- não ultrapassar 40 caracteres;
- sem sublinhas consecutivas ou delimitadoras;

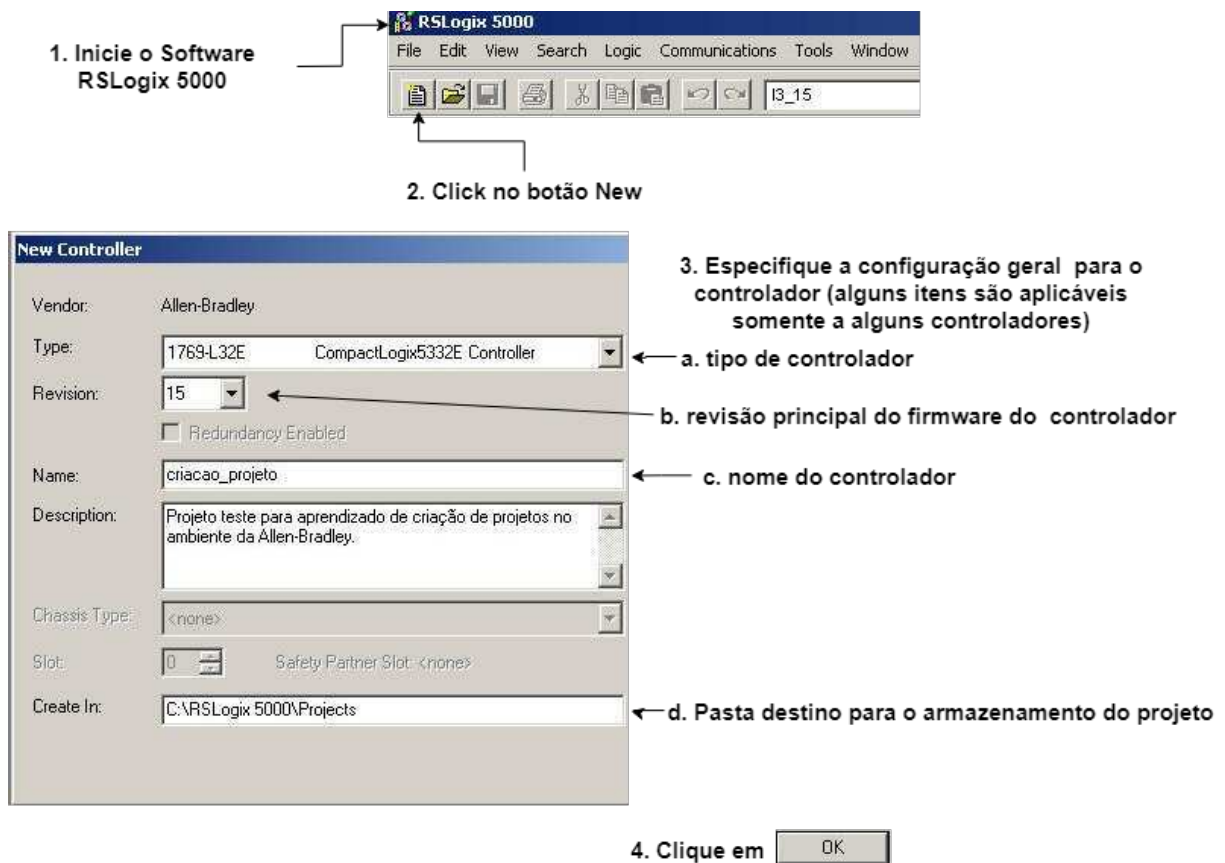


Figura 51: Passos iniciais para a criação do projeto.

Após a criação de um projeto, a tela do RSLogix estará semelhante à figura 52. Nessa figura, do lado esquerdo, tem-se o “organizador do controlador”. Ele exibe as características gerais do projeto e deve ser utilizado para navegar entre os vários componentes do projeto. Essa janela pode ser mostrada ou ocultada pelo menu *View* → *Controller Organizer* (ou usando as teclas de atalho ALT + 0).

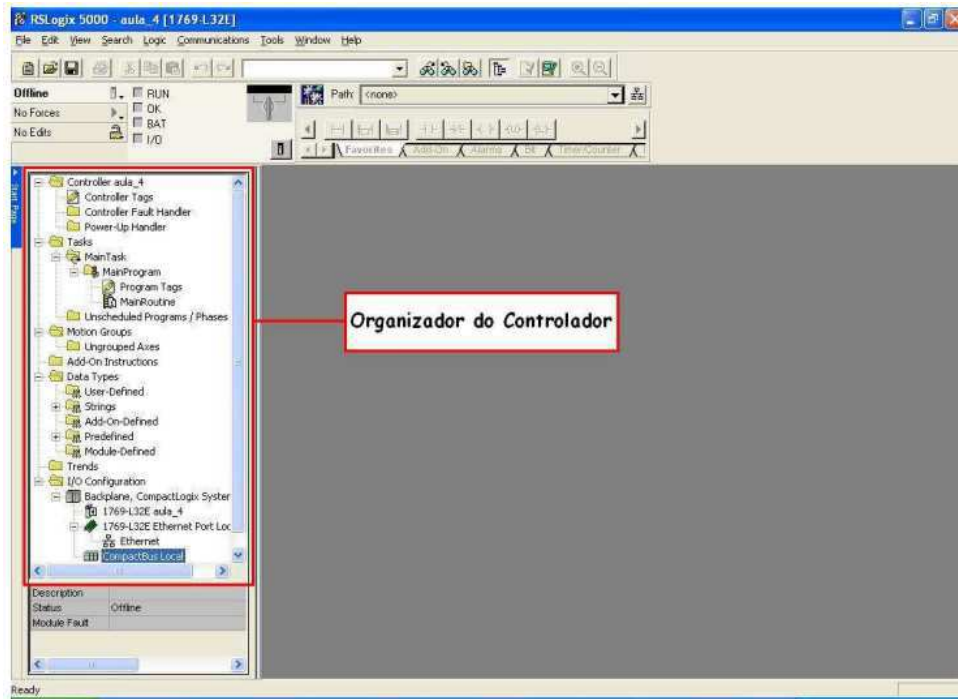


Figura 52: Janela inicial do RSLogix 5000 .

A.3 Adicionando os Módulos de E/S

Após criado um projeto, é necessário criar também uma configuração de controlador, adicionando-se à mesma os cartões de entrada e saída que estão conectados ao CLP.

Para poder comunicar-se com qualquer módulo de E/S de seu sistema, adicione os módulos à pasta de configurações de E/S do controlador. As propriedades que você selecionar para cada módulo definem o comportamento do módulo.

Para adicionar módulos ao controlador, selecione a opção *CompactBus Local* no organizador do controlador (como mostrado na figura 53) e clique sobre a mesma com o botão direito do mouse, selecionando a seguir a opção *New Module* e, em seguida, adicionando os módulos a serem inseridos (figura 54). Os módulos do modelo CompactLogix 1769-L32 são, da esquerda para a direita:

Tabela 4: Módulos de E/S para o CompactLogix

Código	Descrição	Slot
1769-IQ16	16 Entradas Digitais 24V	1
1769-OW8	8 Saídas Digitais AC/DC tipo Relé	2
1769-IF4	4 Entradas Analógicas	3
1769-OF2	2 Saídas Analógicas	4

Já para o ControlLogix:

Tabela 5: Módulos de E/S para o ControlLogix

Código	Descrição	Slot
1756-ENBT	Módulo de Comunicação Ethernet	1
1756-IF16	Entrada de tensão/corrente analógica não isolados	2
1756-OF8	Módulos não isolados de saída analógica	3
1756-IB16I	Módulo de Entrada Digital isolada de 16 pontos	4
1756-OB16E	Módulo de Saída Digital a fusível eletrônico de 16 pontos	5

Observações:

- É importante atentar que a adição dos módulos devem ser feita nos respectivos *slots* em que estão montados fisicamente

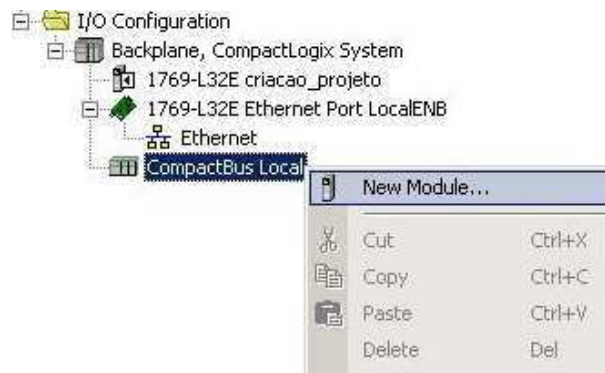


Figura 53: Seleção de um novo módulo.

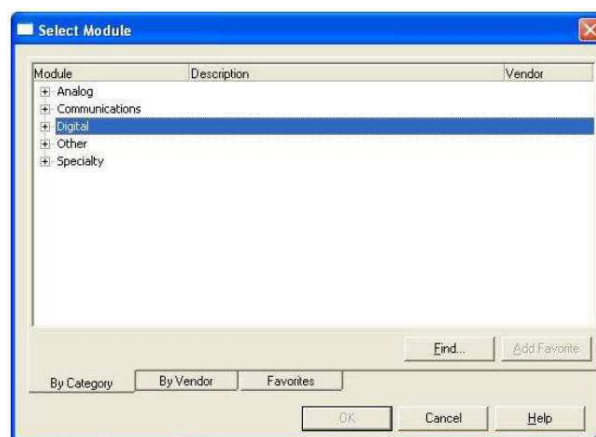


Figura 54: Aba de adição de novos módulos.

Uma nova janela para selecionar o módulo deverá aparecer, basta então procurar e selecionar o módulo desejado. Logo após, deve-se escolher um nome para o módulo e o seu respectivo *slot*, como ilustra a figura 55.

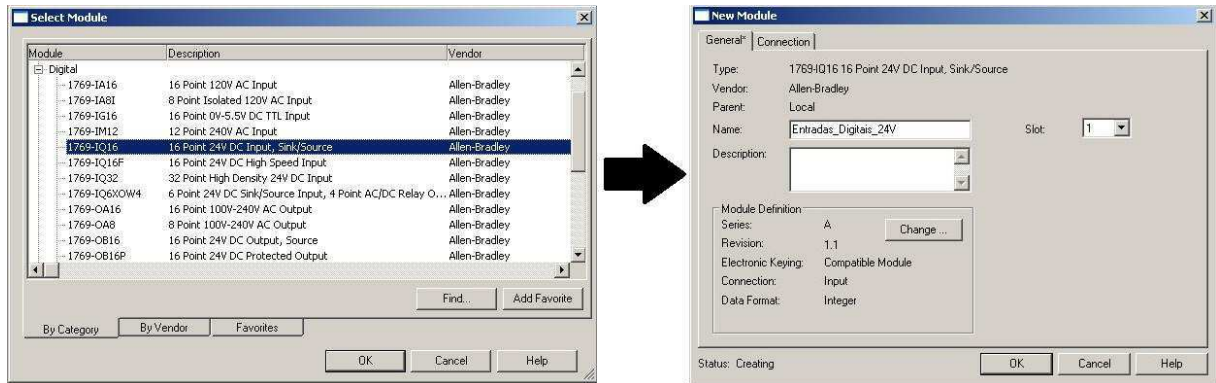


Figura 55: Escolha e nomeação do novo módulo.

Este processo deverá ser repetido para todos os outros módulos que estejam instalados. Lembre-se de salvar este projeto para as atividades futuras, de modo a evitar a necessidade de criar um novo projeto toda vez que for utilizar o CLP.

A.4 Ler os dados de E/S

As informações de E/S estão dispostas em *tags*. As *tags* são nomes baseado em textos para uma área da memória do controlador em que os dados são armazenados.

- Os *tags* são o mecanismo básico para alocar a memória, referenciar dados da lógica e monitorar dados.
- A alocação mínima da memória para um *tag* é quatro *bytes*.
- Quando é criado um *tag* que armazena os dados que precisam de mais de quatro *bytes*, o controlador aloca quatro *bytes*, mas os dados preenche apenas a parte que eles precisam.

Cada *tag* possui uma estrutura de dados, esta estrutura vai depender do respectivo módulo a qual a *tag* pertence. O nome da *tag* é baseado na localização do módulo E/S no sistema. Um endereço E/S segue o seguinte formato:

Localização:Slot:Tipo.Membro.SubMembro.Bit

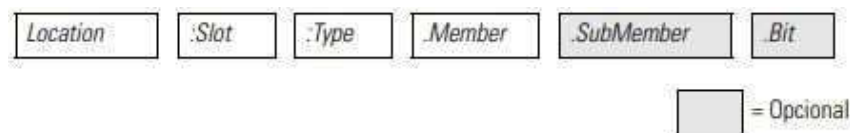


Figura 56: Formato do endereço de E/S.

Tabela 6: Informações de E/S apresentadas no conjunto de *tags*.

Onde	Corresponde a:
<i>Local</i>	Indica a localização do módulo de E/S. Pode ser: LOCAL = mesmo <i>rack</i> ou trilho DIN do controlador NOME_DO_ADAPTADOR = módulo adaptador de comunicação remota ou o módulo ponte
<i>Slot</i>	Número do slot do módulo de E/S em seu <i>rack</i> ou trilho DIN
<i>Tipo</i>	I = entrada (<i>input</i>) O = saída (<i>output</i>) C = configuração
<i>Membro</i>	Para um módulo de E/S discretas, um membro do tipo Data normalmente armazena os valores dos bits de entrada ou saída
<i>Submembro</i>	Dados específicos relacionados a um membro
<i>Bit</i>	Ponto específico em um módulo de E/S discretas. Depende do tamanho do módulo de E/S

Ao se adicionar um módulo novo o *software* automaticamente cria as respectivas *tags* para os dados disponíveis para aquele módulo. Estas *tags* estão disponíveis em **Controller Tags**, como ilustra a Figura 57.

Figura 57: Aba *Monitor Tags* da pasta *Controller Tags* do projeto.

A.5 Criar *Alias* para uma *tag*

De modo a aumentar a abstração do programa muitas vezes se deseja atribuir nomes que representem a função de cada variável. Para fazer isto, pode-se definir um *alias* para uma das *tags* de um módulo de E/S.

Para definir um alias deve-se ir na aba **Edit Tags** da pasta **Controller Tags** do projeto, escolher um nome e depois indicar qual o endereço da *tag* para a qual este alias se refere. A Figura 58 ilustra este procedimento de chamar a entrada digital 01 do módulo 1769-IQ16 DC *Input* de DIGITAL_IN_01.

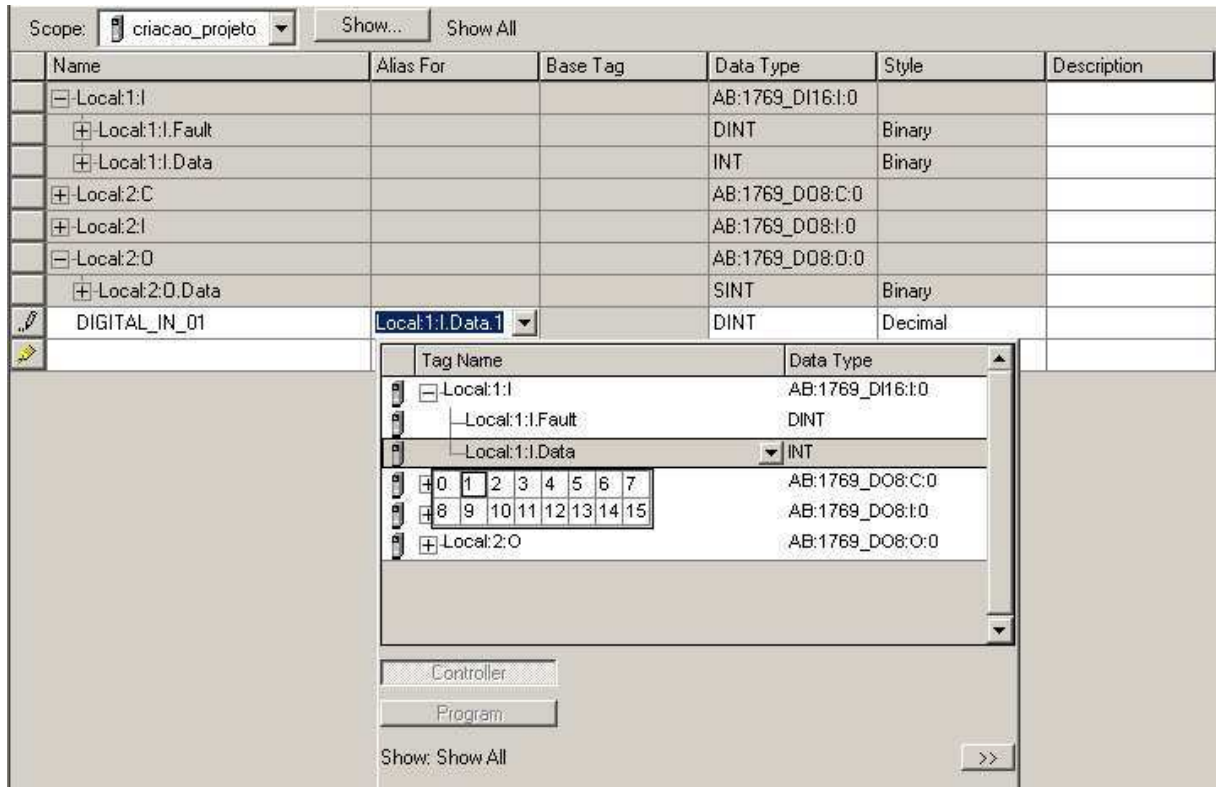


Figura 58: Criação de uma *alias* para uma entrada digital.

A.6 Rotina Principal

É chegado o momento em que de fato se inicia a programação em *ladder* no ambiente da *Rockwell*, pois quando é criado um novo projeto o *software* gera automaticamente uma rotina em *ladder*. Esta rotina está localizada na pasta do organizador chamada *MainProgram* e é nomeada por *MainRoutine*, como ilustrado na figura 59.

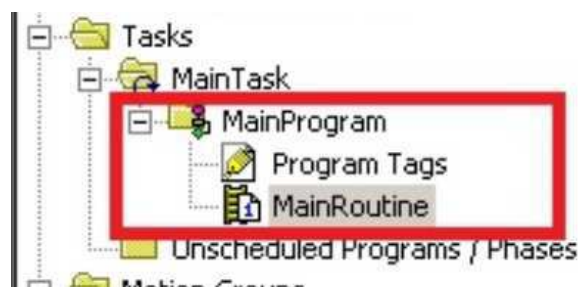


Figura 59: Rotina Principal de programação em *ladder*.

Para inserir uma instrução na linha, deve-se clicar na linha para selecioná-la e então clicar no ícone correspondente da instrução desejada. Se a instrução desejada não está visível, como por exemplo um *timer*, deve-se selecioná-la na aba de grupos da função apropriada.

A.7 Fazer *Download/Upload* do Projeto

Para estabelecer uma conexão com o CLP deve-se abrir a janela *Who Active*, depois selecionar a aba ETHIP_CLP01, *Ethernet* em que o seu CLP esteja localizado. Por fim, deve-se selecionar o IP do Processador do CLP e então clicar em *Go Online*. A Figura 60 ilustra este procedimento.

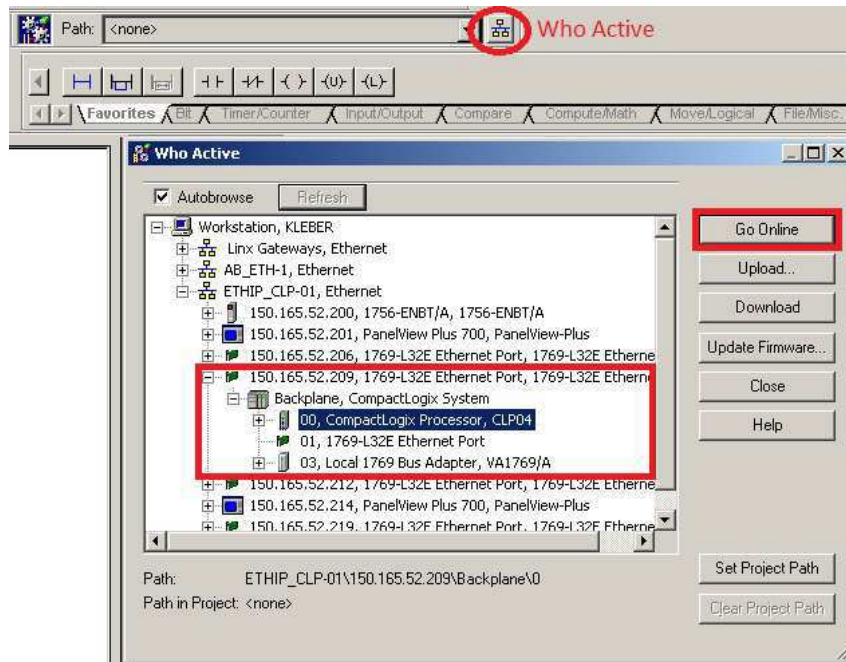


Figura 60: Janela *Who Active*, para a seleção do IP do processador do CLP desejado.

Ao ficar online com o CLP, deverá se optar entre fazer *Upload* do programa em *ladder* que está escrito na memória no CLP para o computador, ou fazer *Download* do programa que está atualmente no RSLogix5000 para o CLP.

Para executar ou interromper a execução lógica em um controlador, altere o modo de operação do controlador, alterando o estado da chave seletora no módulo CPU entre as posições *RUN*, *REM*, e *PROG*.

A chave frontal presente no CLP CompactLogix, apresentada na figura 61 seleciona a forma de operação desejada:

- **Modo RUN:** Corresponde ao modo final de execução do CLP nas indústrias, após sua programação e testes. Neste modo o programa carregado é executado e as saídas discretas e analógicas ficam habilitadas. Não é possível modificar o programa carregado.

- **Modo PROG:** Corresponde ao modo de programação do CLP. O usuário pode criar e alterar programas, tarefas e rotinas, mas estes não podem ser executados. As saídas discretas e analógicas ficam desabilitadas.
- **Modo REM:** Permite o chaveamento entre os modos REMOTE RUN, REMOTE PROG e REMOTE TEST, através do computador conectado ao CLP. Estes modos são descritos como segue:
 - **REMOTE RUN:** O programa carregado é executado, com as saídas habilitadas. Permite edição “*on-line*”.
 - **REMOTE PROG:** O usuário pode criar e alterar programas, tarefas e rotinas. Permite edição “*on-line*”. Não há execução do programa carregado e as saídas são mantidas desabilitadas.
 - **REMOTE TEST:** Semelhante ao REMOTE RUN, mas as saídas permanecem desabilitadas.

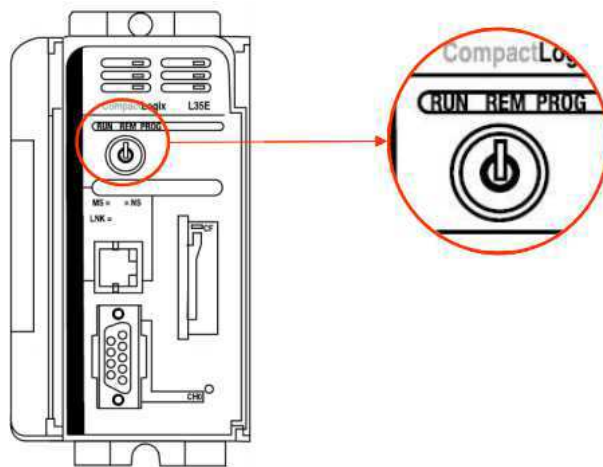


Figura 61: Chave de seleção de modo no painel do CompactLogix.

A.8 Exercícios

1 - Construindo um diagrama *ladder* simples

Utilizando as ferramentas de programação em *ladder*, construa o diagrama correspondente ao circuito lógico da figura 62, empregando contatos para representar as entradas e bobinas para representar as saídas:

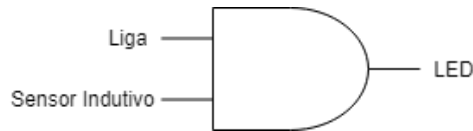


Figura 62: Diagrama lógico referente ao exercício 1.

Ao inserir um elemento no diagrama *ladder*, deve-se atribuir ao mesmo um endereço que corresponda a um ponto nos cartões de E/S do CLP ou a uma posição em sua memória interna. No RSLogix 5000, a atribuição de endereços a contatos ou bobinas pode ser feita de duas formas: executando-se o endereçamento direto ou empregando-se um “*alias*”. O endereçamento direto consiste na especificação completa do endereço, ao passo que o “*alias*” é um sinônimo de um elemento cujo endereço já havia sido especificado anteriormente.

Como dito anteriormente para os CLPs da família Logix 5000, um endereço de E/S segue o formato `Local:Slot:Tipo.Membro.Bit`.

Exemplo: o primeiro ponto do primeiro cartão de entradas digitais dos CLPs tem o endereço **Local:1:I.Data.0**, pois corresponde ao bit 0 do cartão instalado localmente em seu *slot* 1.

No endereçamento direto, basta fazer um duplo clique, com o botão esquerdo do mouse, no símbolo de interrogação que aparece logo acima do elemento figura 63:

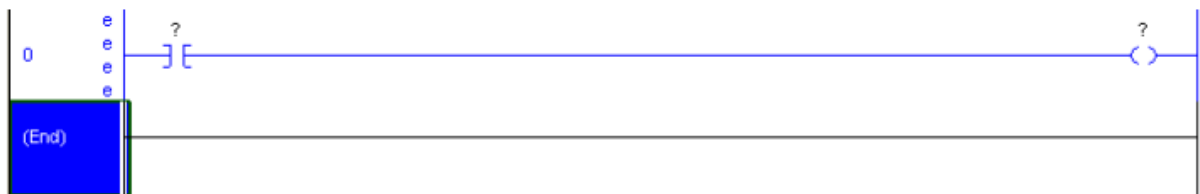


Figura 63: Elementos *ladder* ainda sem definição de endereço.

Dessa forma, o menu visto na figura 64 é apresentado para que se possa escolher o endereço do elemento.

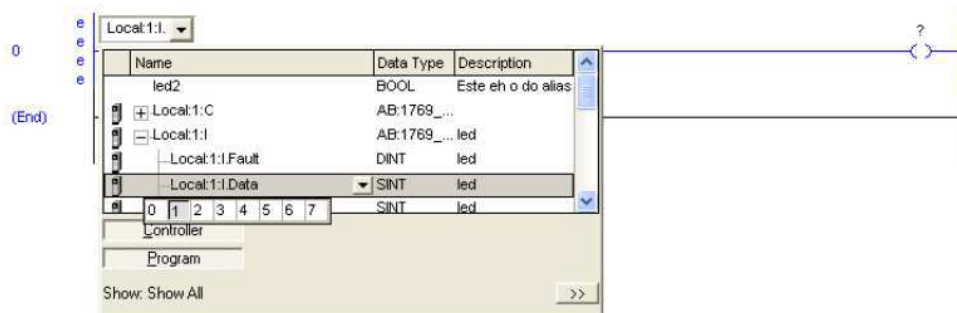


Figura 64: Janela de endereçamento de um elemento *ladder*.

Defina então os endereços para os elementos utilizados na lógica implementada e insira descrições para as mesmas de acordo com a tabela 7 e 8:

Tabela 7: Proposta de descrições para as *tags* de entrada.

	Variáveis de Entrada		
	Endereço	Descrição	ID na Bancada
CompactLogix	Local:1:I.Data.0	Liga/Desliga	Entradas Digitais/I00
	Local:1:I.Data.1	Sensor Indutivo	Entradas Digitais/I01
ControlLogix	Local:4:I.Data.0	Liga/Desliga	Entradas Digitais/I00
	Local:4:I.Data.1	Sensor Indutivo	Entradas Digitais/I01

Tabela 8: Proposta de descrições para as *tags* de saída.

	Variáveis de Saída		
	Endereço	Descrição	ID na Bancada
CompactLogix	Local:2:O.Data.0	LED	Saídas Digitais/O00
ControlLogix	Local:5:O.Data.0	LED	Saídas Digitais/O00

Para inserir a descrição de um elemento diretamente endereçado, basta clicar no endereço do mesmo com o botão direito, selecionar a opção *Monitor/Edit Force Value Of* e, na janela que se abrirá, inserir a descrição na coluna *Description* referente ao endereço desejado.

2 - Empregando “*aliases*”.

A segunda maneira de se endereçar elementos é criando-se um “alias”. Com o alias definido, pode-se reutilizar mais facilmente este mesmo elemento em outras partes do programa. Para testar essa funcionalidade, crie alias para o mesmo diagrama lógico da questão anterior. Para tal use este método alternativo ao apresentado na seção A.5 para endereçá-los.

Para cada elemento inserido, clique na interrogação logo acima do elemento com o botão direito do mouse e selecione a opção *New Tag*. Na janela *New Tag* apresentado na figura 65, preencha os campos *Name*, *Description*, *Type* e *Alias For* de acordo com a tabela a seguir. Observe que, quando for utilizar elementos já existentes, não é necessário criar um novo tag: basta fazer um duplo clique no símbolo “?” do elemento e digitar o *alias*.

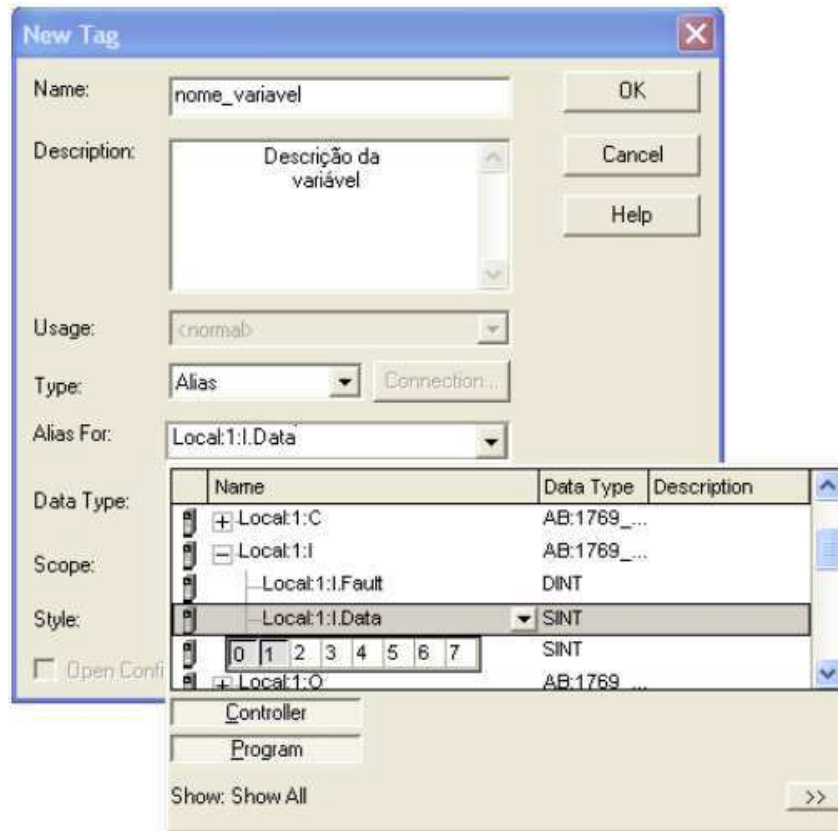


Figura 65: Criando *Aliases*.

A.9 Manual de Referência Básico para Conjunto de Instruções do RSLogix5000

A.9.1 Instruções de Bit (XIC, XIO, OTE, OTL, OTU, OSR, OSF)

1. **XIC - *Examine if Closed***: A instrução XIC examina os dados binários para ver se estão energizados. Para a lógica de contatos, corresponde ao contato Normalmente Aberto [NA].

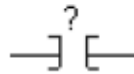


Figura 66: *Examine if Closed*.

Operando	Tipo	Formato	Descrição
Binário	BOOL	tag	bit para ser testado

2. **XIO - *Examine if Open***: A instrução XIO examina os dados binários para ver se foram desenergizados. Para a lógica de contatos, corresponde ao contato Normalmente Fechado [NF].

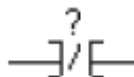


Figura 67: *Examine if Open*.

Operando	Tipo	Formato	Descrição
Binário	BOOL	tag	bit para ser testado

3. **OTE - *Output Energize***: A instrução OTE energiza ou desenergiza o dado binário. Para a lógica de contatos, corresponde às bobinas.



Figura 68: *Output Energize*.

Operando	Tipo	Formato	Descrição
Binário	BOOL	tag	bit para ser energizado ou desenergizado

4. **OTL - *Output Latched***: A instrução OTL energiza (com retenção) o dado binário. Para a lógica de contatos, corresponde à bobina com SET.

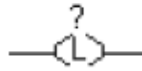


Figura 69: *Output Latched*.

Operando	Tipo	Formato	Descrição
Binário	BOOL	tag	bit para ser energizado

5. **OTU - *Output Unlatched***: A instrução OTU desenergiza (com retenção) o dado binário. Para a lógica de contatos, corresponde à bobina com RESET.

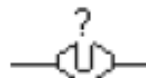


Figura 70: *Output Unlatched*.

Operando	Tipo	Formato	Descrição
Binário	BOOL	tag	bit para ser desenergizado

6. **OSR - *One Shot Rising***: A instrução OSR, também chamado de mono estável com borda de subida, energiza ou desenergiza o bit de saída, dependendo do status do bit de armazenamento. Para a lógica de contatos, corresponde à bobina com borda de subida.



Figura 71: *One Shot Rising*.

Operando	Tipo	Formato	Descrição
Bit de Armazenamento	BOOL	tag	bit de armazenamento interno armazena a entrada da condição da linha desde a última vez que a instrução foi executada
Bit de Saída	BOOL	tag	bit para ser energizado

7. **OSF - One Shot Falling:** A instrução OSF, também chamado de mono estável com borda de descida, energiza ou desenergiza o bit de saída, dependendo do status do bit de armazenamento. Para a lógica de contatos, corresponde à bobina com borda de descida.



Figura 72: *One Shot Falling*.

Operando	Tipo	Formato	Descrição
Bit de Armazenamento	BOOL	tag	bit de armazenamento interno armazena a entrada da condição da linha desde a última vez que a instrução foi executada
Bit de Saída	BOOL	tag	bit para ser energizado

A.9.2 Instruções do Temporizador e do Contador (TON, TOF, CTU, CTD, RES)

1. **TON - Timer On Delay:** A instrução TON é um temporizador não retentivo que acumula tempo quando a instrução é habilitada (entrada da condição da linha é verdadeira).

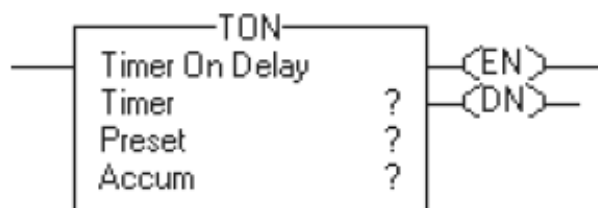


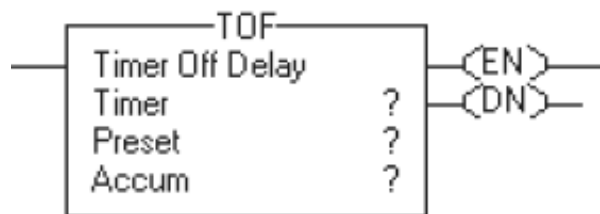
Figura 73: *Timer On Delay*.

Operando	Tipo	Formato	Descrição
Temporizador	TIMER	tag	estrutura do temporizador
Preset	DINT	imediato	Quanto tempo para retardar (tempo acumulado)
Accum	DINT	imediato	totalizar os milissegundos que o temporizador contou valor inicial é normalmente 0

Tabela 9: Estrutura do Temporizador TON.

Mnemônico	Tipo	Descrição
.EN	BOOL	O bit habilitado indica que a instrução TON está habilitada.
.TT	BOOL	O bit de temporização indica que uma operação de temporização está em andamento.
.DN	BOOL	O bit executado está definido quando $.ACC \geq .PRE$.
.PRE	DINT	O valor pré-selecionado especifica o valor (unidades de 1 ms) que o acumulador deve atingir antes da instrução energizar o bit .DN.
.ACC	DINT	O valor acumulado especifica a quantidade de milissegundos que transcorreram desde o momento em que a instrução TON foi habilitada.

2. **TOF - *Timer Off Delay***: A instrução TOF é um temporizador não retentivo que acumula tempo quando a instrução está habilitada (entrada da condição da linha é falsa).

Figura 74: *Timer Off Delay*.

Operando	Tipo	Formato	Descrição
Temporizador	TIMER	tag	estrutura do temporizador
Preset	DINT	imediato	Quanto tempo para retardar (tempo acumulado)
Accum	DINT	imediato	totalizar os milissegundos que o temporizador contou valor inicial é normalmente 0

Tabela 10: Estrutura do Temporizador TOF.

Mnemônico	Tipo	Descrição
.EN	BOOL	O bit habilitado indica que a instrução TOF está habilitada.
.TT	BOOL	O bit de temporização indica que uma operação de temporização está em andamento.
.DN	BOOL	O bit executado está definido quando $.ACC \geq .PRE$.
.PRE	DINT	O valor pré-selecionado especifica o valor (unidades de 1 ms) que o acumulador deve atingir antes da instrução energizar o bit .DN.
.ACC	DINT	O valor acumulado especifica a quantidade de milissegundos que transcorreram desde o momento em que a instrução TOF foi habilitada.

3. **CTU - Count Up**: A instrução CTU conta em ordem crescente.

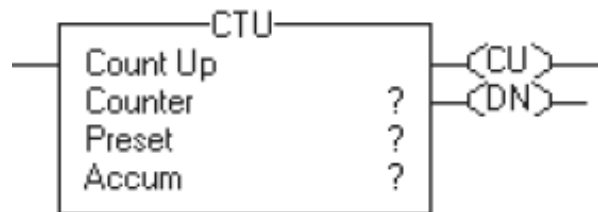


Figura 75: Count Up.

Operando	Tipo	Formato	Descrição
Contador	COUNTER	tag	estrutura do contador
Preset	DINT	imediatos	valor máximo de contagem
Accum	DINT	imediatos	quantidade de vezes que o contador contou valor inicial é normalmente 0

Tabela 11: Estrutura do Contador CTU.

Mnemônico	Tipo	Descrição
.CU	BOOL	O bit de habilitação do contador crescente indica que a instrução CTU está habilitada.
.DN	BOOL	O bit executado indica que $.ACC \geq .PRE$.
.OV	BOOL	O bit de overflow indica que o contador ultrapassou o limite superior de 2.147.483.647. O contador volta para -2.147.483.648 e inicia a contagem crescente novamente.
.UV	BOOL	O bit de underflow indica que o contador ultrapassou o limite inferior de -2.147.483.647. O contador volta para 2.147.483.647 e inicia a contagem decrescente novamente.
.PRE	DINT	O valor pré-programado especifica o valor que o acumulador deve atingir antes da instrução energizar o bit .DN.
.ACC	DINT	O valor acumulado especifica as transições contadas.

4. **CTD - *Count Down***: A instrução CTD conta no sentido decrescente.

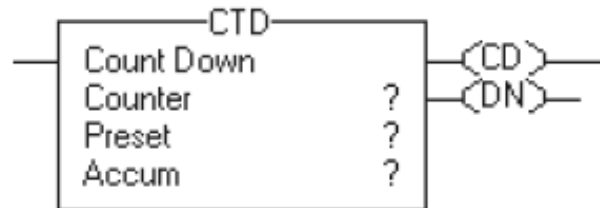


Figura 76: *Count Down*.

Operando	Tipo	Formato	Descrição
Contador	COUNTER	tag	estrutura do contador
Preset	DINT	imediatos	valor máximo de contagem
Accum	DINT	imediatos	quantidade de vezes que o contador contou valor inicial é normalmente 0

Tabela 12: Estrutura do Contador CTD.

Mnemônico	Tipo	Descrição
.CU	BOOL	O bit de habilitação do contador crescente indica que a instrução CTD está habilitada.
.DN	BOOL	O bit executado indica que $.ACC \geq .PRE$.
.OV	BOOL	O bit de overflow indica que o contador ultrapassou o limite superior de 2.147.483.647. O contador volta para -2.147.483.648 e inicia a contagem crescente novamente.
.UV	BOOL	O bit de underflow indica que o contador ultrapassou o limite inferior de -2.147.483.647. O contador volta para 2.147.483.647 e inicia a contagem decrescente novamente.
.PRE	DINT	O valor pré-programado especifica o valor que o acumulado deve atingir antes da instrução energizar o bit .DN.
.ACC	DINT	O valor acumulado especifica as transições contadas.

5. **RES - *Reset***: A instrução RES reseta uma estrutura de TEMPORIZADOR, CONTADOR ou CONTROLE.

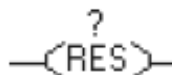


Figura 77: *Reset*.

Operando	Tipo	Formato	Descrição
Estrutura	TIMER COUNTER CONTROL	tag	estrutura para reset

A.9.3 Instruções de Comparação (CMP, EQU, GEQ, GRT, LEQ, LES, LIM, NEQ)

1. **CPM - *Compare***: A instrução CMP realiza uma comparação das operações aritméticas especificadas na expressão.

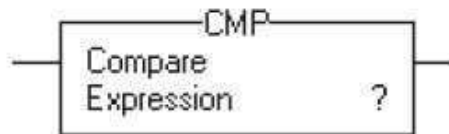


Figura 78: *Compare*.

Operando	Tipo	Formato	Descrição
Expressão	SINT INT DINT REAL	imediato tag	uma expressão composta por tags e/ou valores imediatos separados por operadores
Um tag SINT ou INT é convertido para um valor DINT pela extensão de sinal.			

2. **EQU - *Equal***: A instrução EQU testa se Source A é igual à Source B.
3. **GEQ - *Greater than or Equal***: A instrução GEQ testa se Source A é maior ou igual à Source B.
4. **GRT - *Greater than***: A instrução GRT testa se Source A é maior que Source B.
5. **LEQ - *Less than or Equal***: A instrução LEQ testa se Source A é menor ou igual à Source B.
6. **LES - *Less than***: A instrução LES testa se Source A é menor que Source B.
7. **LIM - *Limit test***: A instrução LIM testa se o valor de teste está dentro da faixa de Limite Inferior a Limite Superior.

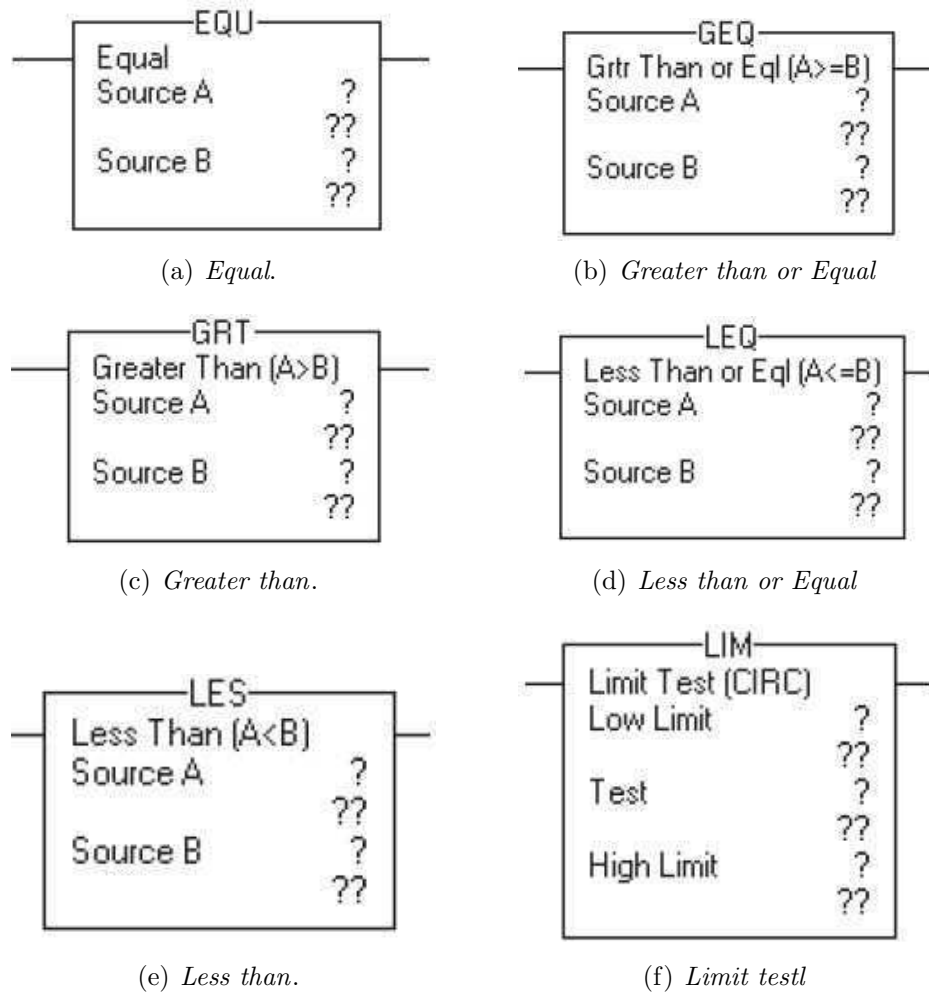


Figura 79: Instruções de Comparação.

Operando	Tipo	Formato	Descrição
Source A	SINT INT DINT REAL Tipo de String	imediato tag	valor a ser testado em relação Source B
Source B	SINT INT DINT REAL Tipo de String	imediato tag	valor a ser testado em relação Source A

Um tag SINT ou INT é convertido para um valor DINT pela extensão de sinal.

Nota:

- Valores REAL são raramente absolutamente iguais. Caso seja necessário determinar a igualdade de dois valores REAL, use a instrução LIM.

- Os tipos de string são:
 - * tipo de dados STRING padrão
 - * qualquer tipo de string novo que você criar
- Para testar os caracteres de um string, insira um tag tipo string para Source A e Source B.

A.9.4 Instruções Matemáticas/Cálculo (CPT, ADD, SUB, MUL, DIV)

CPT - *Compute*: A instrução CPT realiza as operações aritméticas definidas na expressão.

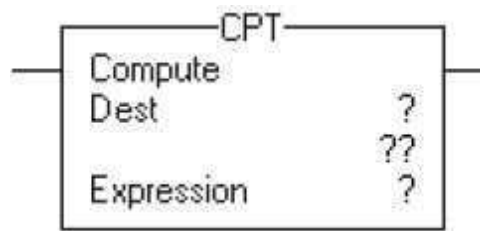


Figura 80: *Compute*.

Operando	Tipo	Formato	Descrição
Destino	SINT INT DINT REAL	tag	tag para armazenar o resultado
Expressão	SINT INT DINT REAL	imediate tag	uma expressão composta por tags e/ou valores imediatos separados por operadores

Um tag SINT ou INT é convertido para um valor DINT pela extensão de sinal.

ADD - *Add*: A instrução ADD soma Source A e Source B e coloca o resultado no Destino.

SUB - *Subtract*: A instrução SUB subtrai Source B de Source A e coloca o resultado no Destino.

MUL - *Multiply*: A instrução MUL multiplica Source A por Source B e coloca o resultado no Destino.

DIV - *Divide*: A instrução DIV divide Source A por Source B e coloca o resultado em Destination.

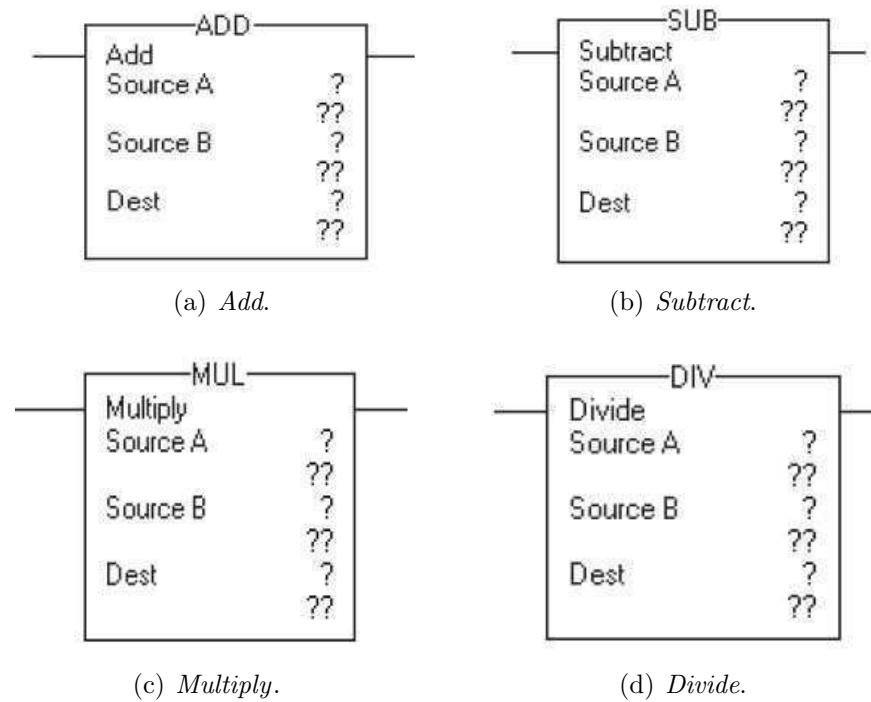


Figura 81: Instruções Matemáticas/Cálculo.

Operando	Tipo	Formato	Descrição
Source A	SINT INT DINT REAL	imediato tag	valor a ser operado com Source B
Source B	SINT INT DINT REAL	imediato tag	valor a ser operado com Source A
Destino	SINT INT DINT REAL	tag	tag para armazenar o resultado

Um tag SINT ou INT é convertido para um valor DINT pela extensão de sinal.

A.9.5 Instrução de Movimentação (MOV)

A instrução MOV copia Source em Destino. Source permanece inalterado.

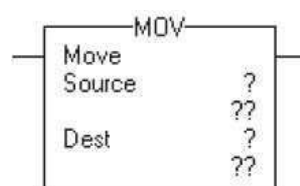


Figura 82: *Move.*

Operando	Tipo	Formato	Descrição
Source	SINT INT DINT REAL	imediatos tag	valor a ser movido (copiar)
Destino	SINT INT DINT REAL	tag	tag para armazenar o resultado

Um tag SINT ou INT é convertido para um valor DINT pela extensão de sinal.

A.9.6 Instruções de Controle de Programa (JSR, MCR)

1. **JSR - Jump to Subroutine:** A instrução JSR salta para uma rotina separada.

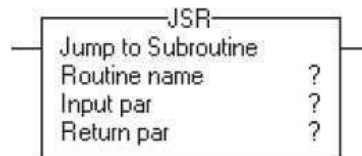


Figura 83: *Jump to Subroutine*.

Operando	Tipo:	Formato:	Descrição:
Instrução JSR			
Routine name	ROUTINE	nome	rotina a ser executada
Input par	SINT INT DINT REAL estrutura	imediatos tag tag da matriz	parâmetros (0-n) a serem passados para rotina
Return par	SINT INT DINT REAL estrutura	tag tag da matriz	parâmetros (0-n) a serem recebidos da rotina

2. **MCR - Rearme do Controle Mestre:** A instrução MCR, usada em pares, cria uma zona de programa que pode desabilitar todas as linhas dentro das instruções MCR. Não possui Operandos de Lógica *Ladder* a Relé.

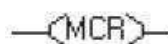


Figura 84: Rearme do Controle Mestre.

A.9.7 Instruções Especiais (PID)

1. Proporcional, Integral e Derivativo (PID): A instrução PID controla um tag de processo como, por exemplo, fluxo, pressão, temperatura ou nível.

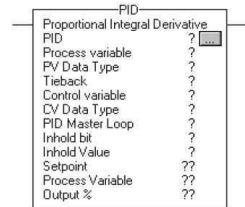


Figura 85: Proporcional, Integral e Derivativo.

Operando	Tipo:	Formato:	Descrição:
PID	PID	estrutura	estrutura PID
Process variable	SINT INT DINT REAL	tag	valor a ser controlado
Tieback	SINT INT DINT REAL	imediate tag	<i>(opcional)</i> saída de uma estação manual/automática de hardware que está realizando bypass da saída do controlador insira 0 se você não quiser usar esse parâmetro
Control variable	SINT INT DINT REAL	tag	valor que vai para o dispositivo de controle final (válvula, amortecedor etc.) se você estiver usando a zona morta, Control variable deve ser do tipo REAL ou o mesmo será forçado em 0 quando houver um erro dentro da zona morta
PID master loop	PID	estrutura	opcional tag PID para o PID mestre Se você estiver realizando o controle em cascata e esse PID for uma malha escrava, insira o nome do PID mestre. insira 0 se você não quiser usar esse parâmetro
Inhold bit	BOOL	tag	opcional status atual do bit inhold de um canal de saída analógico 1756 para suportar uma reinicialização ininterrupta insira 0 se você não quiser usar esse parâmetro
Inhold value	SINT INT DINT REAL	tag	opcional valor de nova leitura de dados de uma canal de saída analógica 1756 para suportar uma reinicialização ininterrupta. insira 0 se você não quiser usar esse parâmetro
Setpoint			somente display valor atual do setpoint.
Process variable			somente display valor atual de Process Variable convertido em escala
Output %			somente display valor de porcentagem da saída da corrente