

**CURSO DE EXTENSÃO  
FAT-UERJ**

# **AUTOMAÇÃO AVANÇADO**



# SUMÁRIO

<b>1. CONTROLADORES LÓGICOS PROGRAMÁVEIS..</b>	1
1.1 Utilização dos CLPs.....	1
1.2 Arquitetura dos CLPs e princípio de funcionamento.....	2
1.3 Tipos de memória.....	5
1.4 Estrutura de Memória e Capacidade.....	7
1.4.1 Definições importantes.....	7
1.5 Modo de Execução.....	8
1.6 Linguagem de Programação.....	10
1.6.1 Norma IEC 61131-3.....	10
1.6.2 Tipos de Linguagem de Programação.....	11
1.7 Linguagem Ladder (LD).....	11
1.7.1 Instruções de Temporizador.....	12
1.7.2 Instruções de Contador.....	15
1.8 Programando o CLP Allen Bradley Micrologix 1000.....	17
1.8.1 Conexão do PC ao CLP.....	21
1.8.2 Programação do aplicativo Lader.....	22
1.8.3 Carregando o programa do PC no CLP.....	23
1.9 Criando Sistemas Supervisórios com RS View 32.....	24
1.9.1 Tipos de Tags.....	26
1.9.2 Tipos de Data Source.....	26
1.9.3 Criação de Displays Gráficos .....	26
<b>2. NORMAS DE SEGURANÇA EM ELETRICIDADE</b>	28
2.1 NR10.....	28
2.1.1 Fases que se aplica a NR 10.....	28
2.1.2 Prontuário de Instalações Elétricas.....	28
2.1.3 Medidas de Proteção Coletiva.....	29
2.1.4 Segurança em Projetos.....	30
2.1.5 Segurança em Instalações Desenergizadas.....	30
2.1.6 Segurança em instalações elétricas Energizadas.....	31
2.1.7 Responsabilidade os trabalhadores nos serviços em eletricidade.....	31
2.2 NBR-5410.....	31
2.2.1 NBR-5410 se aplica.....	32
2.2.2 NBR-5410 não se aplica.....	32
2.2.3 Importância do cumprimento da Norma.....	33
<b>3. BIBLIOGRAFIA.....</b>	33



# 1. CONTROLADORES LÓGICOS PROGRAMÁVEIS

Podemos considerar o CLP um computador projetado para trabalhar no ambiente industrial. Os transdutores e os atuadores são conectados a robustos cartões de interface. Comparados com um computador de escritório, os primeiros CLPs tinham um conjunto de instruções reduzido, normalmente apenas condições lógicas e não possuíam entradas analógicas, podendo manipular somente aplicações de controle digital (discreto).

Um Controlador Lógico Programável é definido pelo IEC (*International Electrotechnical Commission*) como:

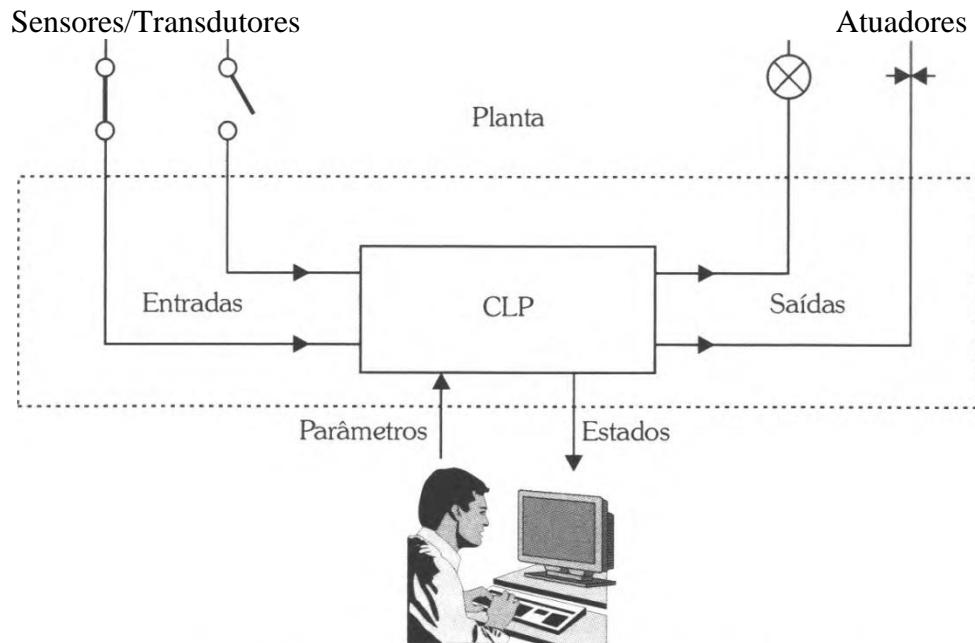
*“Sistema eletrônico operado digitalmente, projetado para uso em um ambiente industrial, que usa uma memória programável para a armazenagem internas de instruções orientadas para o usuário para implementar funções específicas, tais como lógica, sequencial, temporização, contagem e aritmética, para controlar, através de entradas e saídas digitais ou analógicas, vários tipos de máquinas e processos. O controlador programável e seus periféricos associados são projetados para serem facilmente integráveis em um sistema de controle industrial e facilmente usados em todas suas funções previstas.”*

## 1.1 Utilização dos CLPs

Toda planta industrial necessita de algum tipo de controlador para garantir uma operação segura e economicamente viável. Desde o nível mais simples, em que pode ser utilizado para controlar o motor elétrico de um ventilador para regular a temperatura de uma sala, até um grau de complexidade elevado, controlando a planta de um reator nuclear para produção de energia elétrica. Embora existam tamanhos e complexidades diferentes, todos os sistemas de controle podem ser divididos em três partes com funções bem definidas: os tradutores (sensores), os controladores e os atuadores.

- **Sensores/transdutores:** transdutores é um dispositivo que converte uma condição física do elemento sensor em um sinal elétrico para ser utilizado pelo CLP através da conexão às entradas do CLP. Um exemplo típico é um botão de pressão momentânea, em que um sinal elétrico é enviado do botão de pressão ao CLP, indicando sua condição atual (pressionando OU liberando).
- **Atuadores:** Sua função é converter o sinal elétrico oriundo do CLP em um condição física, normalmente ligando ou desligando algum elemento. Os atuadores são conectados às saídas do CLP. Um exemplo típico é fazer o controle do acionamento do motor através do CLP. Neste caso a saída do CLP vai ligar ou desligar a bobina do contato que o comanda.
- **Controladores:** de acordo com os estados das suas entradas, o controlador utiliza um programa de controle para calcular os estados das suas saídas. Os sinais elétricos das saídas são convertidos no processo através dos atuadores. Muitos atuadores geram movimentos, tais como válvulas, motores, bombas; outros utilizam energia elétrica ou pneumática. O operador pode interagir com o controlador por meio dos parâmetros de controle. Alguns controladores podem mostrar o estado do processo em uma tela ou em um *display*.

Um sistema de controle típico encontra-se na figura a seguir.



*Representação dos sistemas de controle.*

## 1.2 Arquitetura dos CLPs e princípio de funcionamento

O CLP é um equipamento de estado sólido que pode ser programado para executar instruções que controlam dispositivos, máquinas e operações de processos pela implementação de funções específicas, como lógica de controle, sequenciamento, controle de tempo, operações aritméticas, controles estatístico, controle de malha, transmissão de dados etc.

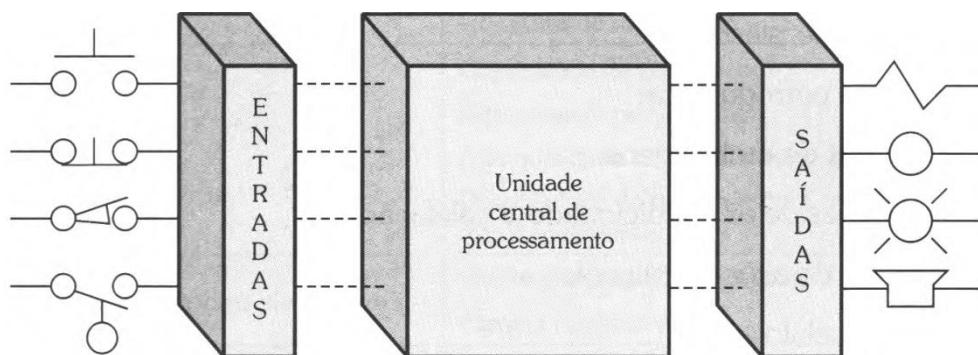
Os CLP são projetados e construídos para operarem em ambientes severos, portanto devem resistir a altas temperaturas, ruídos elétricos, poluição atmosférica, ambientes úmidos etc.

Sua capacidade quanto ao número de entradas e saídas, memória, conjunto de instruções, velocidade de processamento, conectividade, flexibilidade, IHM, etc, varia conforme o fabricante e modelo.

Os primeiros controladores lógicos programáveis tinham como função primordial somente substituir os relés utilizados na indústria. A sua função era somente realizar operações sequenciais que eram anteriormente implementadas com relés, como, por exemplo, controle liga/desliga de máquinas e processos que necessitavam de operações repetitivas. Em um curto tempo esses controladores tiveram muitas melhorias em relação aos relés, como o uso de menor espaço e energia, indicadores de diagnóstico e ao contrário dos relés, a sua lógica de operações poderia ser mudada sem a necessidade de alteração das conexões físicas dos elementos.

Um controlador lógico programável pode ser dividido em duas partes, conforme a figura a seguir:

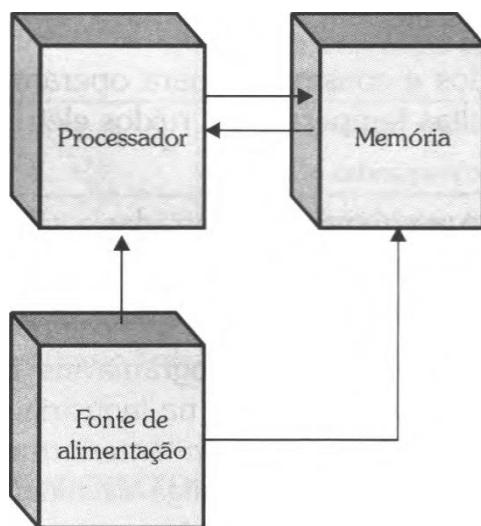
- Uma unidade central de processamento;
- Sistemas de interface de entrada/saída.



*Diagrama de blocos de controlador lógico programável.*

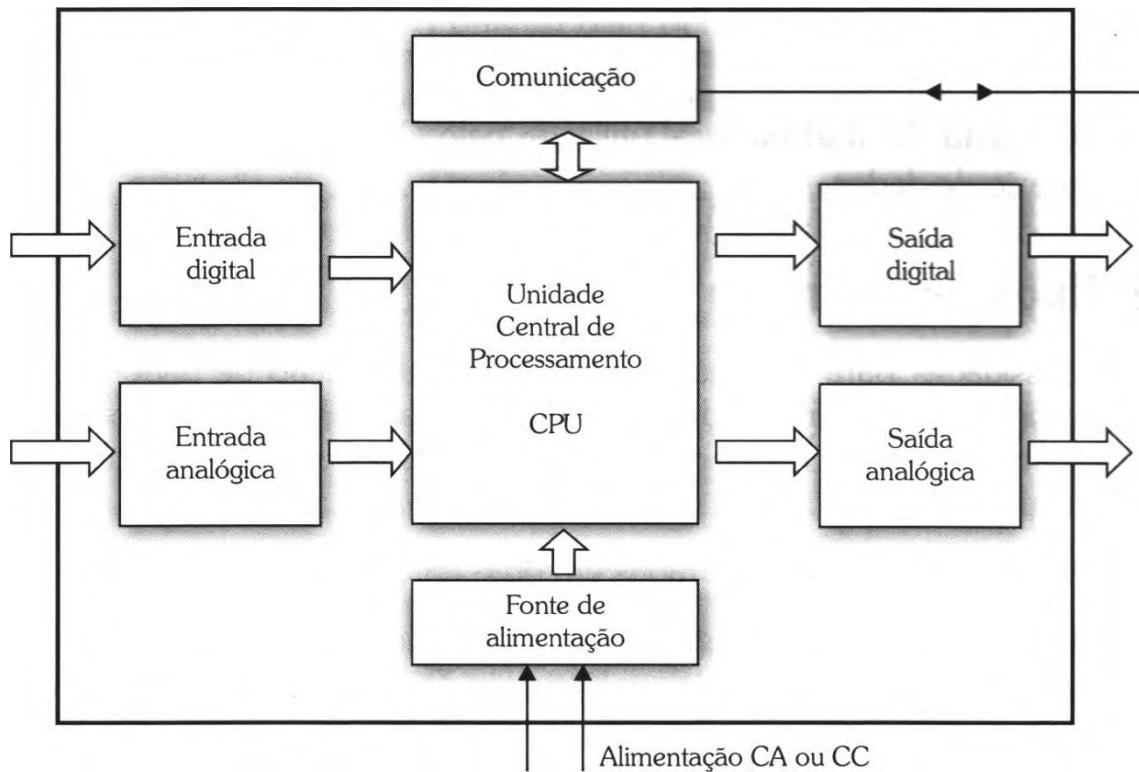
A Unidade Central de Processamento (UCP), mas conhecida pela sua sigla originária da língua inglesa CPU (*Central Processing Unit*), comanda todas as atividades do CLP, sendo formada pelos três elementos na figura a seguir:

- Processador;
- Sistema de Memórias;
- Fonte de alimentação.



*Diagrama de bloco dos principais componentes da CPU.*

Podemos ter um diagrama de blocos simples do CLP, como está ilustrado na Figura a cima. Juntamente com a interface de comunicação e as interfaces de entrada e saída, temos o controlador lógico programável, como exibe a figura a seguir.



*Diagrama de blocos do CLP.*

De acordo com a Figura a cima, um CLP pode ser dividido em cinco partes:

1. Fonte de alimentação
2. Entradas (analogicas e/ ou digitais)
3. Saídas (analogicas e/ou digitais)
4. Unidade Central de Processamento (CPU)
5. Unidade de comunicação

A fonte de alimentação é responsável pelo fornecimento da energia necessária para a alimentação da CPU e dos módulos de entrada e de saída. Fornece todos os níveis de tensão exigidos para as operações internas do CLP. Convém lembrar que, como geralmente os CLPs são modulares, existem casos em que uma segunda fonte é necessária devido ao aumento de consumo com a expansão dos módulos. Cada fabricante especifica as condições que tornam necessária a segunda fonte. Certos modelos de CLPs são projetados para operarem com uma tensão de alimentação de 220 V, outros trabalham com tensão de alimentação continua de 24 V.

### 1.3 Tipos de memória

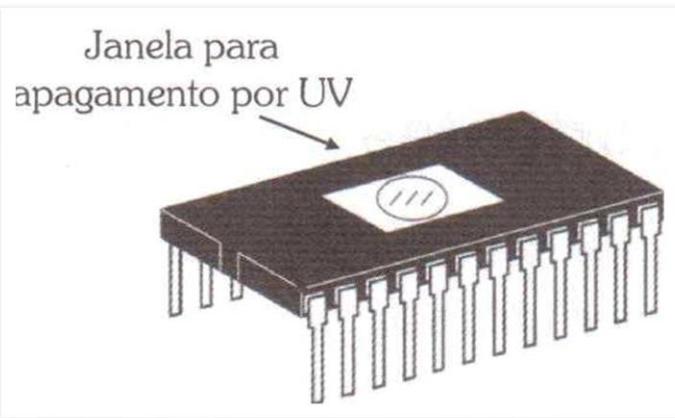
As necessidades para o armazenamento e a recuperação de dados para memória de programa e memória de dados não são as mesmas. Por exemplo, normalmente o conteúdo da memória de dados necessita ser alterado conforme os dados vão sendo coletados.

As memórias podem ser separadas em duas categorias: voláteis e não voláteis.

- **Memórias voláteis:** perdem seu conteúdo quando sua alimentação elétrica é removida. Memórias voláteis são facilmente alteradas e é recomendado para a grande maioria das aplicações que utilizem uma bateria que mantenha sua alimentação, mesmo na ausência de alimentação externa. As baterias são chamadas de “bateria de *backup*”
- **Memórias não voláteis:** retêm o conteúdo programado, mesmo durante uma completa falta de energia, sem necessidade de uma bateria de *backup*. Memórias não voláteis podem ser reprogramáveis ou fixas.

A seguir acompanhe a descrição dos seis principais tipos de memórias e suas características que afetam a maneira com as instruções programadas são alteradas ou armazenadas em um CLP.

- **Memória ROM (Read Only Memory):** projetada para armazenamento permanente de um determinado programa ou de dados. Após a gravação do seu conteúdo (normalmente feito na fábrica), somente pode ser lido e nunca mais alterado. Desta forma, por natureza, as memórias ROM são imunes a alterações por ruídos elétricos e perda de energia. Utilizada para o sistema operacional e dados fixos usado pela CPU. Outra aplicação da memória ROM é em micro CLPs construídos para uma função específica e fixa (função dedicada), em que não há necessidade de alteração do programa.
  - **Memória RAM (Random Access Memory):** desenvolvida para que a informação possa ser escrita ou lida em qualquer posição de memória com alta velocidade. Esse tipo de memória é volátil, ou seja, não retém a informação se a fonte de alimentação for desligada. Requer o uso de uma bateria para manter os dados em caso de falta de energia. Um grande número de CLPs usa memória RAM para armazenar o programa aplicativo junto com baterias de *backup*. É uma memória relativamente rápida em comparação com outros tipos. Uma grande desvantagem é que a bateria pode eventualmente falhar. Por isso, normalmente os CLPs que utilizam esse sistema possuem um dispositivo que constantemente monitora o estado da bateria e informa ao processador. Memória RAM suportadas por baterias têm tido excelentes resultados para a grande maioria das aplicações em CLPs.
- Memória EPROM (Erasable PROM):** é uma memória PROM que pode ser reprogramada depois de ser inteiramente apagada por uma fonte de luz ultravioleta. O apagamento completo do conteúdo do chip necessita que a janela do chip seja exposta a uma fonte de luz ultravioleta por aproximadamente 20 minutos.



*Janela para apagamento da memória EPROM por ultravioleta (UV)*

A memória EPROM pode ser considerada um dispositivo de armazenamento semipermanente, porque armazena um programa até que ele necessite ser alterado. A memória EPROM fornece um excelente meio de armazenamento para programas que não necessitam de volatilidade, entretanto ela não permite mudanças no programa e entradas de dados on-line. Muitos fabricantes usam controladores com memórias EPROM para prover armazenamento permanente de programas em equipamentos que não necessitam de alterações ou entrada de dados pelo usuário.

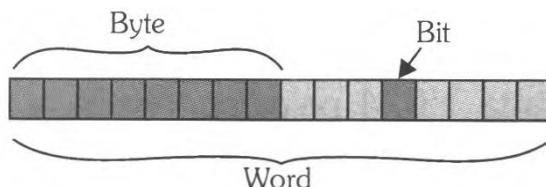
**Memória EEPROM (Electrically Erasable PROM):** é não volátil e oferece a mesma flexibilidade de programação que a RAM. A grande maioria dos controladores de médio e pequeno porte usa EEPROM como única memória do sistema. Ela fornece armazenamento permanente para o programa e pode ser facilmente alterado com o uso de um dispositivo de programação (por exemplo, PC) ou uma unidade de programação manual. Estas duas características ajudam a reduzir o tempo para alteração de programas. Uma das desvantagens da EEPROM é que um *byte* de memória só pode ser escrito depois que o conteúdo anterior tiver sido apagado, causando um atraso. Esse período de atraso é considerável quando mudanças *on-line* de programação forem feitas. Outras desvantagem de EEPROM é a limitação do número de vezes que pode ser executada a operação de escrever/apagar em único *byte* de memória (de 10.000 a 100.000 vezes). No entanto, essas desvantagens podem ser desprezadas, se compararmos com as notáveis vantagens que ela oferece.

- **Memória FLASH:** é um dos tipos mais recentes de memória. É utilizada pelas placas-mãe de computadores pessoais para armazenar o programa BIOS. Sua grande vantagem é a facilidade de atualização de firmware dos equipamentos através de software externos. Diversos fabricantes de CLPs já utilizam esse tipo de memória nos seus CLPs. A Siemens, por exemplo, as utiliza no modelo S7-300.

## 1.4 Estrutura de Memória e Capacidade

### 1.4.1 Definições importantes

- **Bit:** menor unidade de informação, pode ter apenas dois estados: ativos (1) ou inativo (0). Pode ser utilizado para armazenar variáveis lógicas (binárias). Também pode ser utilizado, combinados com outros bits, para formatar outros tipos de dados mais complexos.
- **Nibble ou quarteto:** agrupamento de quatro bits, utilizando principalmente para armazenamento de códigos BCD.
- **Byte ou octeto:** Agrupamento de oitos bits. Pode armazenar um caractere do tipo ASCII ou um número entre 0 e 255, dois números BCD ou oito indicadores de um bit.
- **Word ou palavra:** Uma palavra corresponde a uma certa quantidade de bits que pode variar de um processador para outro. No entanto, é comum considerar uma palavra como a composição de 16 bits.
- **Double word ou palavra dupla:** é a composição de duas palavras, ou seja, para os processadores de 16 bits correspondentes a um agrupamento de 32 bits.



Unidades básicas de memória de um CLP: bits, bytes e words.

A capacidade de armazenamento de uma unidade de memória é determinada pelo número de palavras (*words*) que ela pode armazenar.

O IEC (*International Electrotechnical Commission*) aprovou, em janeiro de 1999, uma norma internacional – IEC 60027-2 – para a designação de nomes e símbolos para prefixos de múltiplos de binários utilizados nos campos de processamento e transmissão de dados. Outra norma também foi publicada pelo IEEE (IEE 541) em 2005. Os prefixos são apresentados nas tabelas seguintes:

Múltiplo (SI)
Kilo: $(10^3)^1$
Mega: $(10^3)^2$
Giga: $(10^3)^3$
Tera: $(10^3)^4$

Fator	Nome	Símbolo	Referência	Fator
$2^{10}$	Kibi	Ki	Kilobinary : $(10^3)^1$	$2^{10}$
$2^{20}$	Mebi	Mi	Megabinary: $(10^{10})^2$	$2^{20}$
$2^{30}$	Gibi	Gi	Gibabinary: $(2^{10})^3$	$2^{30}$
$2^{40}$	Tebi	Ti	Terabinary: $(2^{10})^4$	$2^{40}$

Múltiplos SI

Multiplos IEC 60027-2

Portanto, deve ser observado que:

1 Kibibit	1 Kibit	$= 2^{10}$ bit	= 1024 bits
1 Kilobit	1 Kbit	$= 10^3$ bit	= 1000 bits
1 Mebibyte	1 MiB	$= 2^{10}$ B	= 1 048 576 B
1 Megabyte	1 MB	$= 10^6$ B	= 1 000 000 B

Sugere-se que a primeira sílaba do nome do múltiplo do binário seja pronunciada da mesma maneira que o prefixo correspondente no SI, e a segunda sílaba seja pronunciada como “bi”.

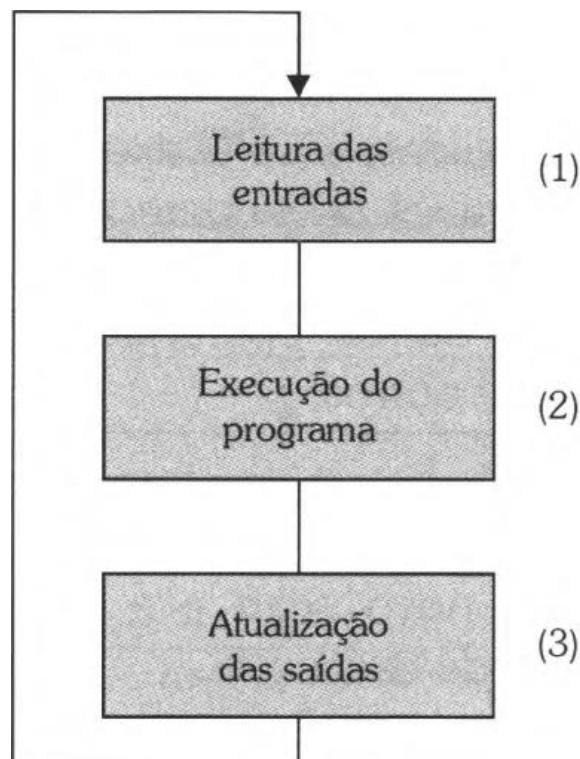
## 1.5 Modo de Execução

No modo de execução (*Run*), o CLP passa a executar o programa do usuário. CLPs de maior porte podem sofrer alterações de programa mesmo durante a execução. Esse tipo de programação é chamado de *on-line* (em operação).

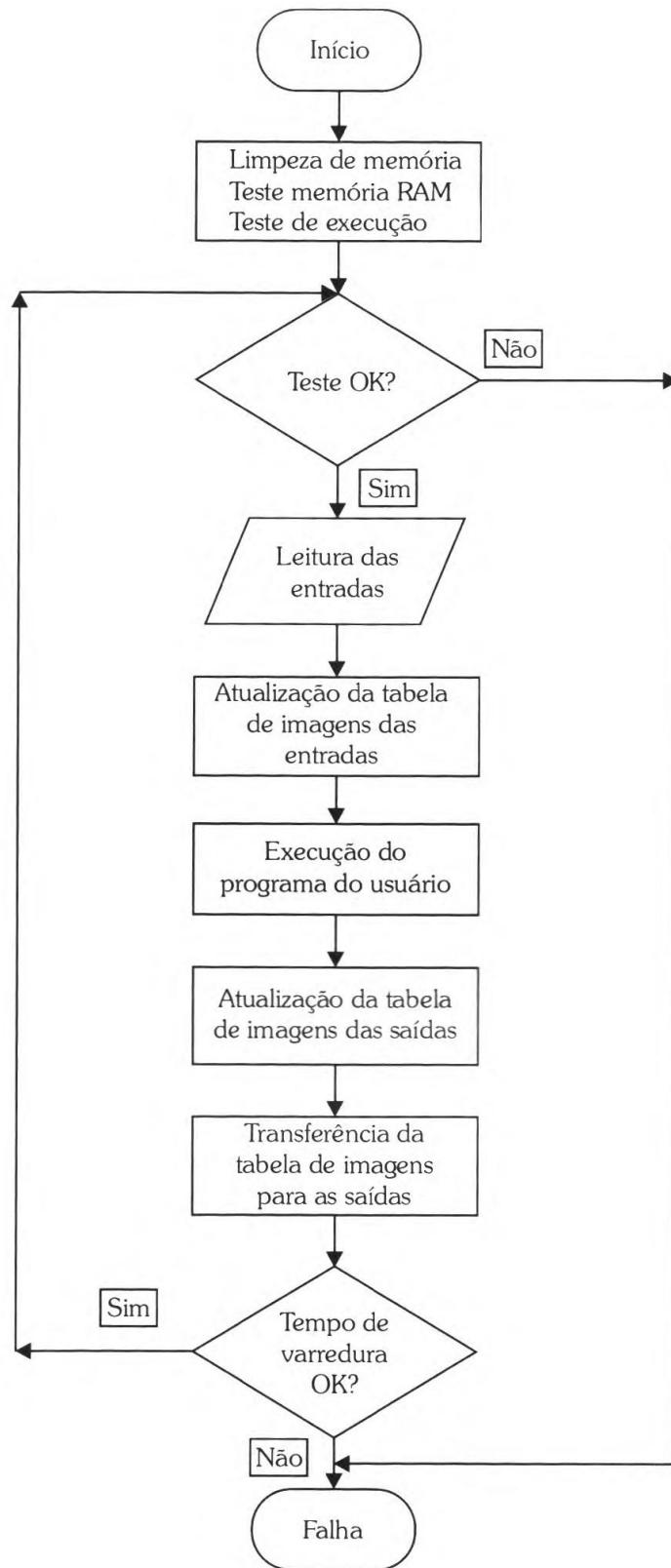
O funcionamento do CLP é baseado nem sistema microprocessado em que há uma estrutura de *software* que realiza continuamente ciclos de leitura, chamados de *scan*. O *scan* é constituído de três processos:

1. Efetua a leitura de dados através dos dispositivos via interface de entrada.
2. Executa o programa de controle armazenado na memória.
3. Escreve ou atualiza os dispositivos de saída via interface de saída.

A Figura a seguir mostra o processo ocorrido no ciclo de *scan* de um CLP.



*Ilustração do scan do CLP.*



*Fluxograma de funcionamento do CLP*

## 1.6 Linguagem de Programação

Imagine que um gerente deseja passar instruções a um operador de determinado processo. Se ambos falarem português, instruções típicas poderiam ser: ligue o motor, desligue o motor, some dois valores, subtraia dois valores, acenda a lâmpada, apague a lâmpada, ligue a sirene e assim por diante. Portanto, para que haja uma efetiva comunicação, é necessário utilizar uma linguagem que ambos entendam. Os efeitos seriam os mesmos se as instruções fossem dadas em japonês, desde que ambos entendam japonês.

Genericamente, linguagem é um meio de transmissão de informações entre dois ou mais elementos com capacidade de se comunicarem. Esses elementos não ficam restritos aos seres humanos, nem mesmo é exclusividade dos seres vivos, já que máquinas podem ser construídas com tal capacidade.

Na área da computação, define-se instrução como um comando que permite a um sistema com capacidade computacional realizar determinada operação.

Linguagem de programação é o conjunto padronizado de instruções que o sistema computacional é capaz de reconhecer. Programar significa fornecer uma série de instruções a um sistema com capacidade computacional, de maneira que este seja capaz de comportar-se determinadamente, executando de forma automática as decisões de controle em função do estado atual, das entradas e das saídas do sistema num dado instante.

O programador é responsável por prever as situações possíveis do sistema, planejar uma estratégia de controle e codificar as instruções em uma linguagem de programação padronizada para posteriormente serem passadas ao sistema computacional.

### 1.6.1 Norma IEC 61131-3

Nos últimos anos houve em enorme avanço nas técnicas e nas linguagens de programação. Vários métodos de modelagem foram desenvolvidas e poderosas linguagens criadas, visando atender aos mais diversos setores. Inegavelmente a programação dos controladores lógicos programáveis é atualmente mais simples e flexível do que antes, principalmente porque foram desenvolvidas ou aperfeiçoadas várias linguagens proprietárias, incluindo variações da linguagem *Ladder* e da linguagem de Lista de Instruções. A inexistência de normas em relação às linguagens de programação dos CLPs fez surgir inúmeras variantes destas ao longo do tempo, todas diferentes entre si. Do ponto de vista das empresas usuárias, é claramente um desperdício de dinheiro e de recursos humanos, já que as habilidades desenvolvidas por seus funcionários na utilização de um determinado tipo de CLP não pode ser reaproveitadas quando sua substituição por outro tipo de fabricante.

Assim, quando a empresa necessita trocar o CLP antigo devido ao aumento de demanda ou mesmo para incorporar novos recursos, investimentos devem ser feitos em aquisição e desenvolvimentos de *software*, *hardware* e em treinamento do pessoal envolvido na implantação e manutenção desses sistemas incluindo técnicos, projetistas de sistemas até mesmo os gerentes da planta.

Felizmente a comunidade industrial reconheceu que era necessário estabelecer um padrão aberto para os CLPs, visando à uniformização de procedimentos dos diversos fabricantes. Para tanto, foi criado um grupo de trabalho no IEC (*International Electrotechnical Commission*) para estabelecer normas a todo o ciclo de desenvolvimento dos CLPs, incluindo o projeto de *hardware*, instalação, testes, documentação, programação e comunicação.

No inicio da década de 1990, o IEC publicou várias partes da norma IEC 1131 que cobre o ciclo de vida completo dos CLPs. Essa norma é considerada, por alguns, um marco histórico para os CLPs. Alguns anos depois essa norma foi revisada e recebeu o número IEC 61131 cuja terceira parte – IEC61131-3 – trata das linguagens de programação. Com o objetivo de simplificar a sua análise, é usual dividi-la em três seções:

- Generalidades;
- Elementos comuns;
- Linguagem de programação.

### **1.6.2 Tipos de Linguagem de Programação**

Visando atender aos diversos segmentos da indústria, incluindo seus usuários, e uniformizar as várias metodologias de programação dos controladores industriais, a norma IEC 61131-3 definiu sintáticamente e semanticamente cinco linguagens de programação:

- Diagrama de Blocos de Funções (FBD – *Function Block Diagram*)
- Linguagem Ladder (LD – *Ladder Diagram*)
- Sequenciamento Gráfico de Funções (SFC – *System Function Chart*)
- Lista de Instruções (IL – *Instruction List*)
- Texto Estruturado (ST – *Structured Text*)

### **1.7 Linguagem Ladder (LD)**

É uma linguagem gráfica baseada na lógica de relés e contatos elétricos para a realização de circuitos de comandos de acionamentos. Por ser a primeira linguagem utilizada pelos fabricantes, é a mais difundida e encontrada em quase todos os CLPs da atual geração.

Bobinas e contatos são símbolos utilizados nessa linguagem. Os símbolos de contatos programados em uma linha representam as condições que serão avaliadas de acordo com a lógica. Como resultado determina o controle de uma saída, que normalmente é representado pelo símbolo de uma bobina.

Recebeu vários nomes desde sua criação, entre eles diagrama escada, diagrama de contatos e linguagem de contatos. Consideramos linguagem de *Ladder* pelos seguintes motivos: Primeiramente por ser o nome mais conhecido no meio industrial, em segundo lugar, pela tradução literal, a palavra mais próxima seria “diagrama do tipo escada”. No entanto, poderia gerar confusão, já que a pronúncia é a mesma da palavra SCADA, a

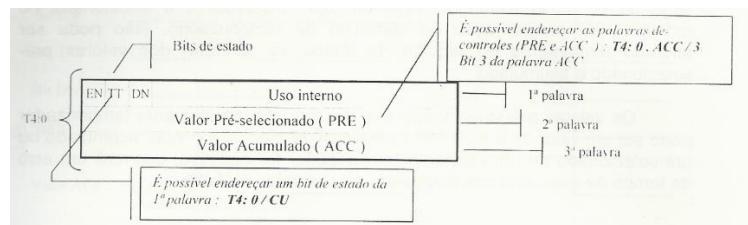
qual é comumente adotada no meio industrial para referir-se aos sistemas supervisórios. Por último, “diagrama de contatos” somente esclarece que é um tipo de diagrama e não é suficiente para caracterizar que este seja implementado em CLP, uma vez que um diagrama de contatos também pode ser implementado utilizando elementos físicos discretos como relés, contatores, temporizadores e outros, montados em painéis elétricos.

### 1.7.1 Instruções de Temporizador

As instruções de temporização são:

- Temporizador na Energização (TON)
- Temporizador na Desenergização (TOF)
- Temporizador Retentivo (RTO)

Cada elemento temporizador tem uma palavra para os bits de estado e a base de tempo, uma palavra para o valor pré-selecionado de temporização (PRE) e uma palavra para armazenar o valor acumulado (ACC).



Bit de estado são os bits de palavra de controle utilizados para indicar o estado atual do temporizador, são eles:

EN= Bit de habilitação do temporizador.

TT= Bit de Temporização do tempo de curso.

DN= Bit de Realização do temporizador.

É possível endereçar o bit de estado, para controlar uma outra instrução.

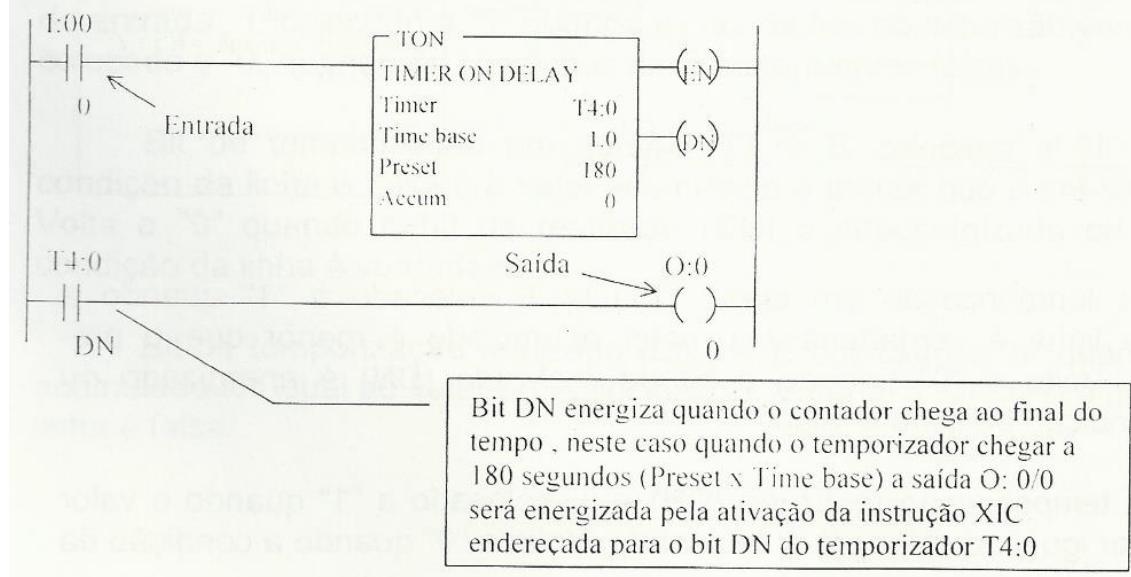
Exemplo de endereçamento dos bit de estado:

T4:0/EN

T4:0/TT

T4:0/DN

Exemplo:



#### Base de Tempo:

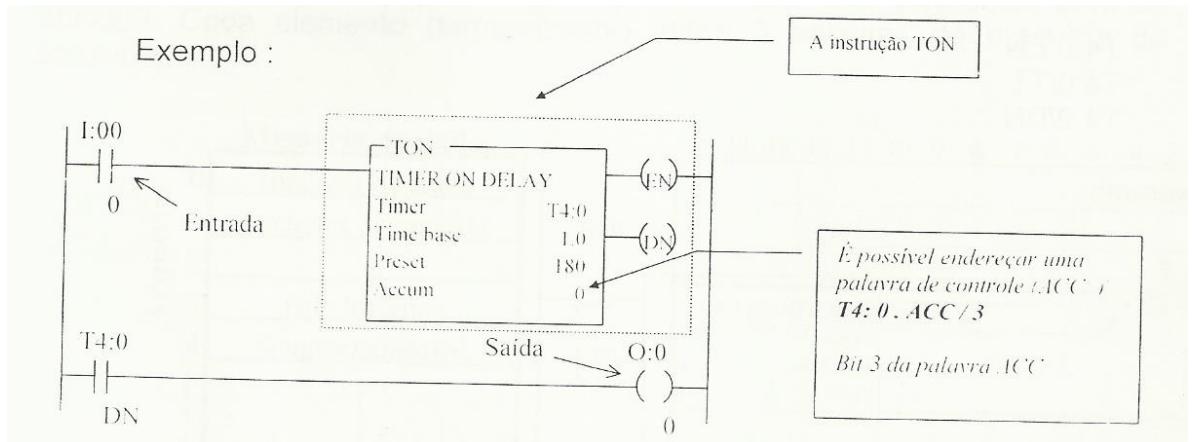
A base de tempo, que funciona como um clock para temporizador, define o tempo entre um incremento do valor acumulado, pode ser de 1 segundo ou 1 centésimo de segundo (0,01 segundo) e é selecionado no momento da programação da instrução de temporização. Não pode ser alterado diretamente nos arquivos de dados ao contrário dos valores pré-selecionados e acumulado.

Os valores pré-selecionados e acumulados de um elemento temporizador pode ser um valor de 0 a 32.767(palavra=16 bits). Se o valor acumulado ou pré-selecionado de um elemento temporizador for negativo, ocorrerá um erro de tempo de execução que colocará o processador em falha.

#### Instrução Temporizador a Energização (TON):

A instrução TON inicia a contagem dos intervalos da base de tempo quando as condições da linha de entrada tornam-se verdadeiras. Enquanto o as condições da linha de entrada permanecer verdadeira o temporizador incrementará seu valor acumulado (ACC) a cada varredura até atingir o valor pré-estabelecido (PRE). O valor acumulado é zerado quando as condições da linha tornam-se falsas, independente do temporizador ter atingido o valor pré-selecionado.

Bit de habilitação do temporizador (EN) acompanha o estado da linha de entrada. É colocado a “1” quando as condições da linha tornam-se verdadeiras, e a “0” quando as condições da linha tornam-se falsas.

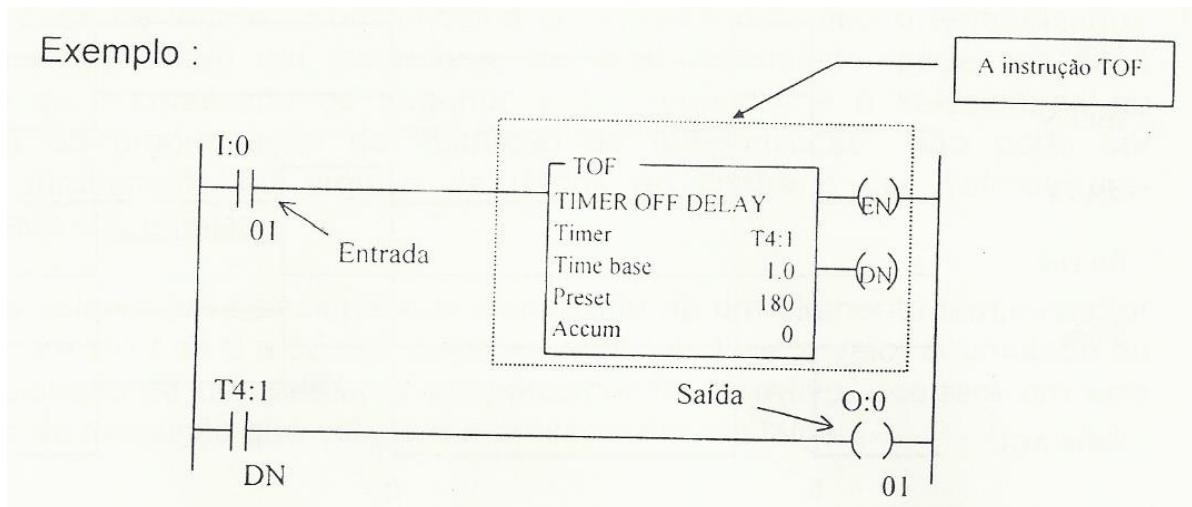


Bit de temporização em curso (TT) é colocado a “1” quando a condição da linha é verdadeira e o valor acumulado é menor que o pré-selecionado. Volta a “0” quando o bit de realizado (DN) é energizado ou quando a condição da linha é falsa. Bit de temporização realizado (DN) é colocado a “1” quando o valor acumulado for igual ao valor pré-estabelecido. Volta a “0” quando a condição da linha é falsa.

#### Instrução Temporizador na Desenergização (TOF):

A instrução TOF inicia a contagem dos intervalos da base de tempo quando as condições da linha de entrada tornam-se falsas. Enquanto as condições da linha permanecerem falsas o temporizador incrementará seu valor acumulado (ACC) a cada varredura até atingir o valor pré-selecionado (PRE). O valor acumulado é zerado quando as condições da linha tornam-se verdadeiras, independente do temporizador ter atingido o valor pré-selecionado.

Bit de validação do temporizador (EN) acompanha o estado da linha de entrada. É colocado a “1” quando as condições da linha são verdadeiras; é colocado a “0” quando as condições da linha tornam-se falsas. Bit de temporização em curso (TT), é colocado “1” quando a condição da linha é falsa e o valor acumulado é menor que o pré-selecionado. Volta a “0” quando o bit de realizado (DN) é desenergizado ou quando a condição da linha é verdadeira. Bit de temporização (DN) é colocado a “0” quando o valor acumulado for igual ao valor pré-selecionado. Volta a “1” quando as condições da linha são falsas.



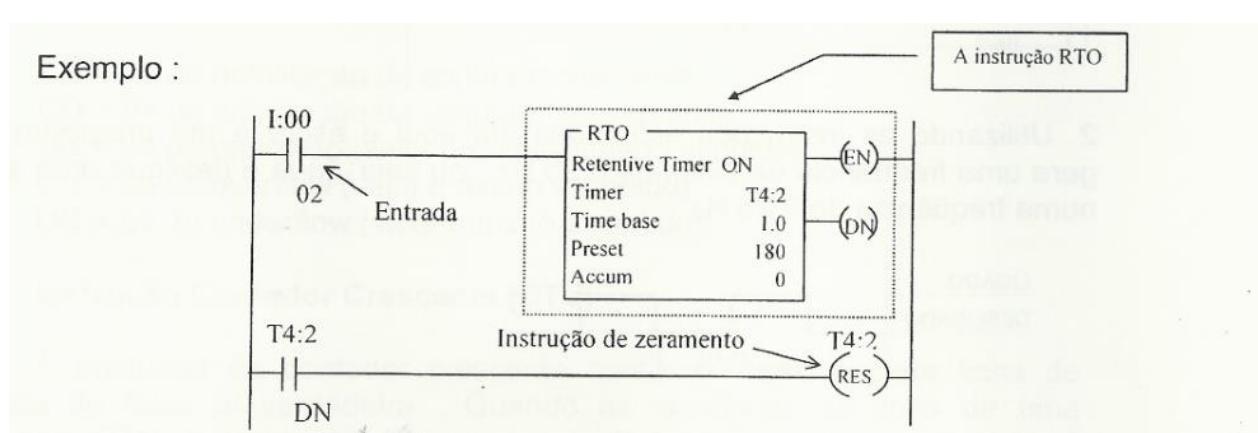
### Instrução Temporizador Retentivo (RTO):

A instrução RTO inicia a contagem dos intervalos da base de tempo quando as condições da linha de entrada tornam-se verdadeiras. Enquanto as condições da linha permanecerem verdadeiras o temporizador incrementará seu valor acumulado (ACC) a cada varredura até atingir o valor pré-selecionado (PRE). O valor acumulado é retido quando:

- A condição da linha se tornar falsa;
- O controlador é alterado de RUN ou TESTE para PROG;
- O controlado perde a alimentação (desde que seja mantida a bateria back up/ lítio em perfeitas condições);
- Ocorre uma falha;

Bit de validação do temporizador (EN) é colocado a “1” quando as condições da linha são verdadeiras; é colocado a “0” quando as condições da linha terminam-se falsas. Bit de temporização em curso (TT) é colocado a “1” quando a condição da linha é verdadeira e o valor acumulado é menor que o pré-selecionado. Volta o “0” quando o bit de realizado (DN) é energizado ou quando a condição da linha é falsa. Bit de temporização realizado (DN) é colocado a “1” quando o valor acumulado for igual ao valor pré-selecionado. No entanto, só volta a “0” quando a instrução RES é habilitada.

O valor acumulado é zerado pela instrução RES. Quando essa instrução com o mesmo endereço da instrução RTO for habilitada, o valor acumulado e os bits de controle serão desenergizados.



### 1.7.2 Instruções de Contador

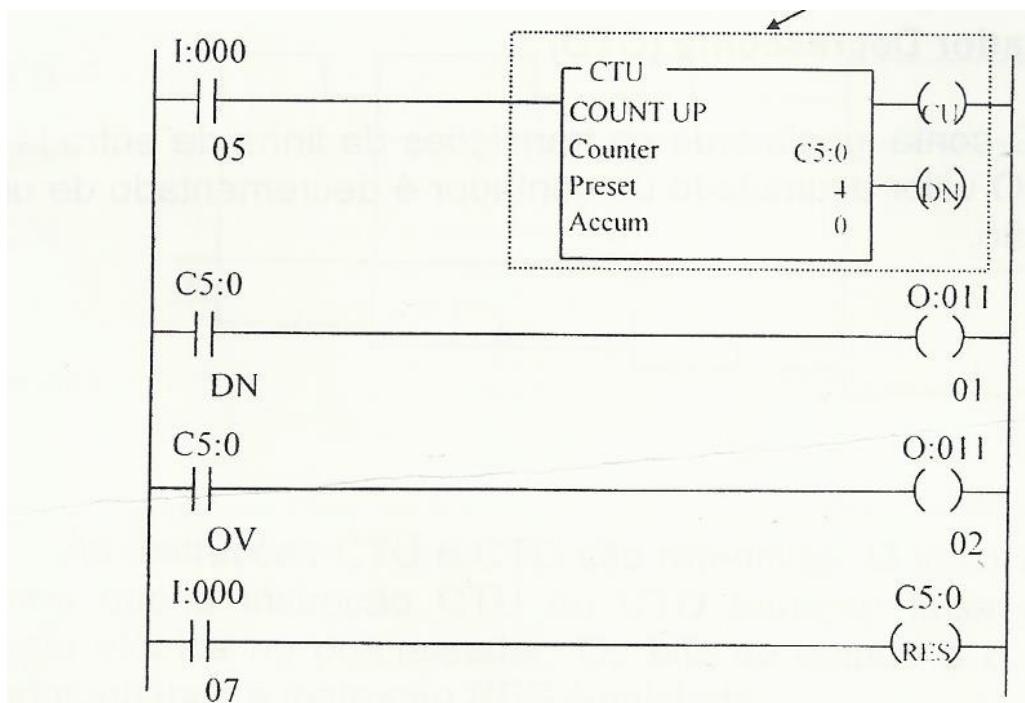
As instruções de Contador são:

- Contador Crescente (CTU)
- Contador Descrescente (CTD)

#### Instrução Contador Crescente (CTU):

A instrução de contador crescente conta as transições da linha de entrada de falsa para verdadeira. Quando as condições da linha de uma instrução (CTU) passam da falsa para verdadeira, o valor acumulado é incrementado de uma unidade.

Quando esta operação é sucessiva, de forma que o valor acumulado torne-se igual ao pré-selecionado, o bit DN é colocado a “1” e permanece quando é ultrapassado o valor pré-selecionado.



O bit de validação de contador crescente (CU) é colocado a ”1” quando as condições da linha tornam-na verdadeira, e ”0” quando as condições da linha tornam-na falsa ou uma instrução RES com o mesmo endereço da instrução CTU é verdadeira.

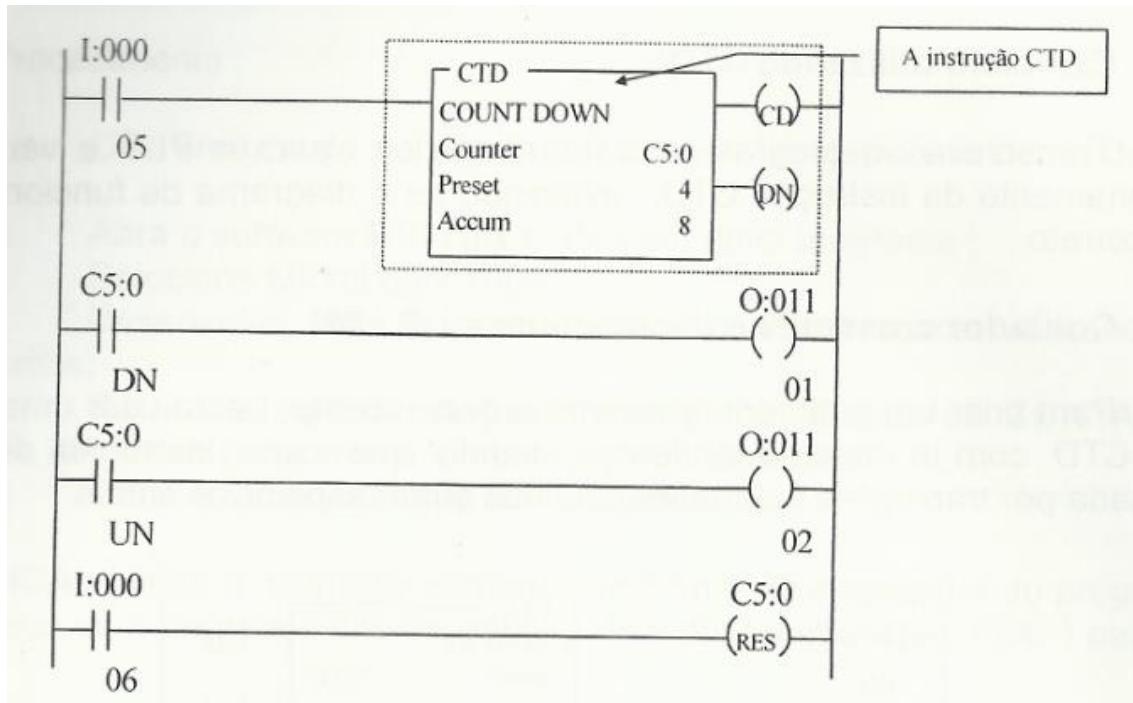
A instrução CTU pode contar além do valor pré-selecionado. Quando a contagem atinge o valor  $32767 + 1$ , ocorre uma condição overflow (capacidade máxima atingida). Isto é indicado pelo bit 12 (OV) que assume ”1”. Quando o bit OV é colocado a ”1”, o valor acumulado assume o valor -32768 e continua a contar a partir daí.

É possível recolocar o bit OV a ”0” habilitando a instrução RES com o mesmo endereço da CTU.

#### Instrução Contador Decrescente (CTD):

A instrução CTD conta igualmente as transições da linha de entrada de falsa para verdadeira. O valor acumulado do contador é decrementado de uma unidade a cada transição.

Quando esta operação é sucessiva de forma que o valor acumulado torna-se melhor que o pré-selecionado, o bit DN é colocado à zero.



O bit de validação do contador decrescente (CD) é colocado a “1” quando as condições da linha tornam-na verdadeira, o colocado a “0” quando as condições da linha tornam-na falsa ou quando a instrução RES com o mesmo endereço é validada.

Quando uma instrução CTD atinge o valor  $-32768 - 1$ , ocorre uma condição underflow (capacidade máxima negativa atingida). Isto é indicado pelo bit 12 (UN) que assume 1. Quando o bit UN é colocado a “1”, o valor acumulado assume o valor  $+32767$  e continua a descontar a partir daí.

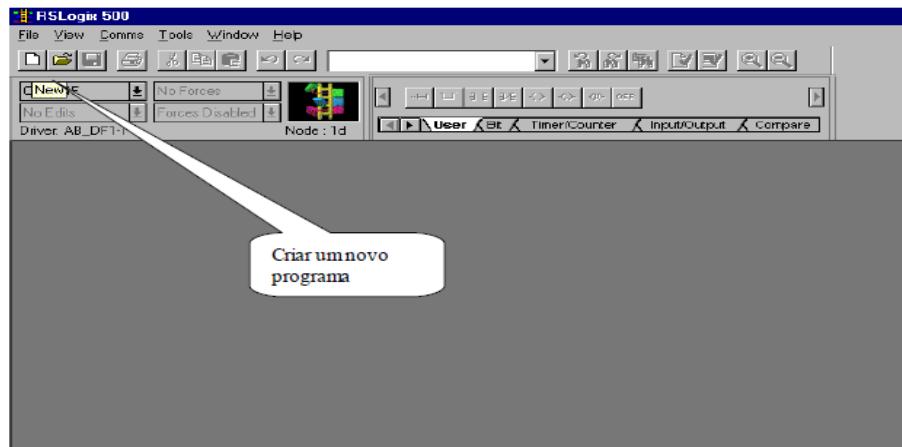
É possível recolocar o bit UN a “0” habilitando a instrução RES com o mesmo endereço da CTD.

As instruções CTU e CTD são retentivas. O valor acumulado é mantido mesmo que a instrução CTU ou CTD torne-se falsa ou mesmo que falte energia elétrica no processador. Os bits de estado e o valor acumulado são zerados quando a instrução RES é validada.

### 1.8 Programando o CLP Allen Bradley Micrologix 1000

A família dos CLPs MicroLogix da Allen-Bradley tem característica modular, e a versão do controlador disponível no laboratório é constituída por: chassi, fonte, módulo processador (CPU), módulo de entradas digitais, módulo de saídas digitais.

O software RSLogix500 é um programa desenvolvido com o objetivo de configurar, programar, monitorar e comandar os CLPs da série MicroLogix e SLC-500 da Allen Bradley. Após acessar o software vamos criar um novo programa. A comunicação do PC com o CLP MicroLogix (Ou família SLC - 500) se faz fisicamente por meio de um cabo de comunicação que atenda o padrão RS-232, e virtualmente por meio do software RSLinx.

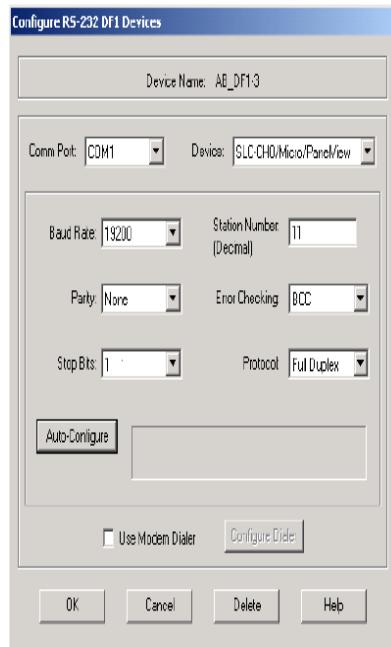


*Tela inicial do RSLogix500*

Por meio deste software, pode-se configurar o CLP, transferir programas elaborados no RSLogix para o CLP (Download), transferir programas existentes no CLP para o RSLogix (Upload), e ainda monitorar e comandar o CLP, em tempo real.

Primeiramente, deve-se criar um driver para comunicação do CLP com o PC. No software RSLinx, clique em Communications / Configure Drivers. Em Available Driver Types, selecione o driver desejado e configure-o. Neste caso o driver RS-232 DF1 device deve ser selecionado. Em Comm Port, selecione a porta de comunicação utilizada e em Device, selecione SLC-CH0/Micro/PannelView.

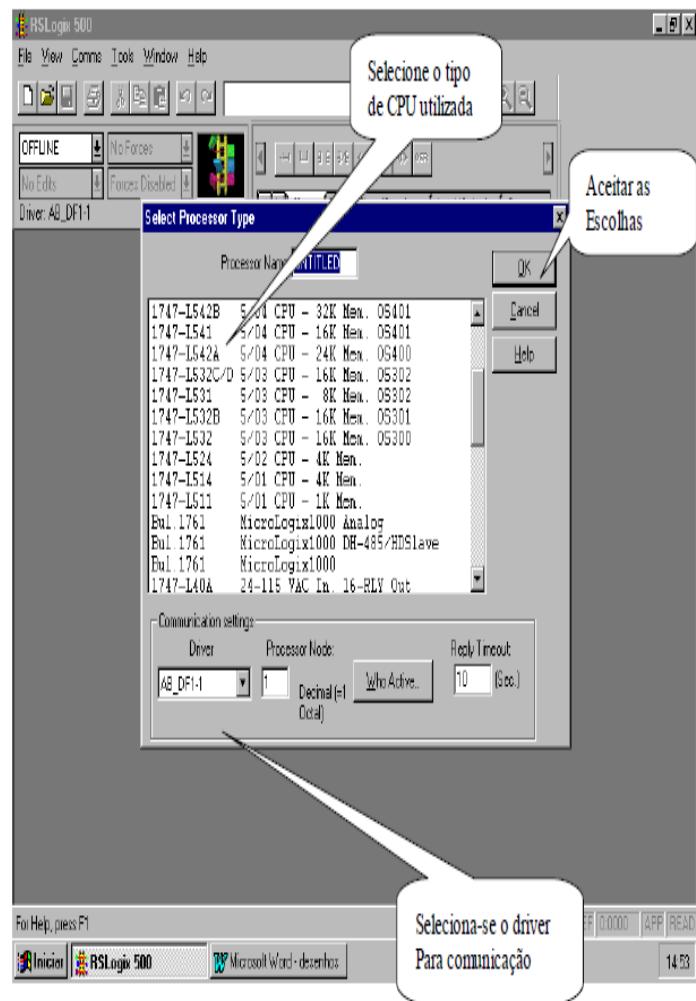
As outras informações necessárias para configuração do driver podem ser obtidas por meio do Auto-Configure. A Figura apresenta um exemplo de tela de configuração do Driver serial Linx.



*Tela do Drive RS-232 DF1*

Para criar um novo arquivo no RSLogix500, siga os seguintes passos:

- 1 - No RSLogix, ao selecionar o menu File – New, surgira uma lista com os processadores possíveis;
- 2 - Selecione a CPU Utilizada. Escolha “BUL. 1762-Micrologix 1000” e clicar em OK;
- 3 - Selecione o Drive para comunicçāo- menu- Comm, clicar em System Comms; isso abre o RSLinx e a janela Communications, semelhante à anterior. Procurar a estação correspondente ao CLP em uso, no canal DF-1. Caso exista um “X” vermelho no ícone do CLP, está ocorrendo uma falha na comunicação.
- 4 - No menu à esquerda, selecione IO Configuration, fazendo surgir a tela de configuração de entradas e saídas, onde é feita a configuração dos módulos que compõem as gavetas.



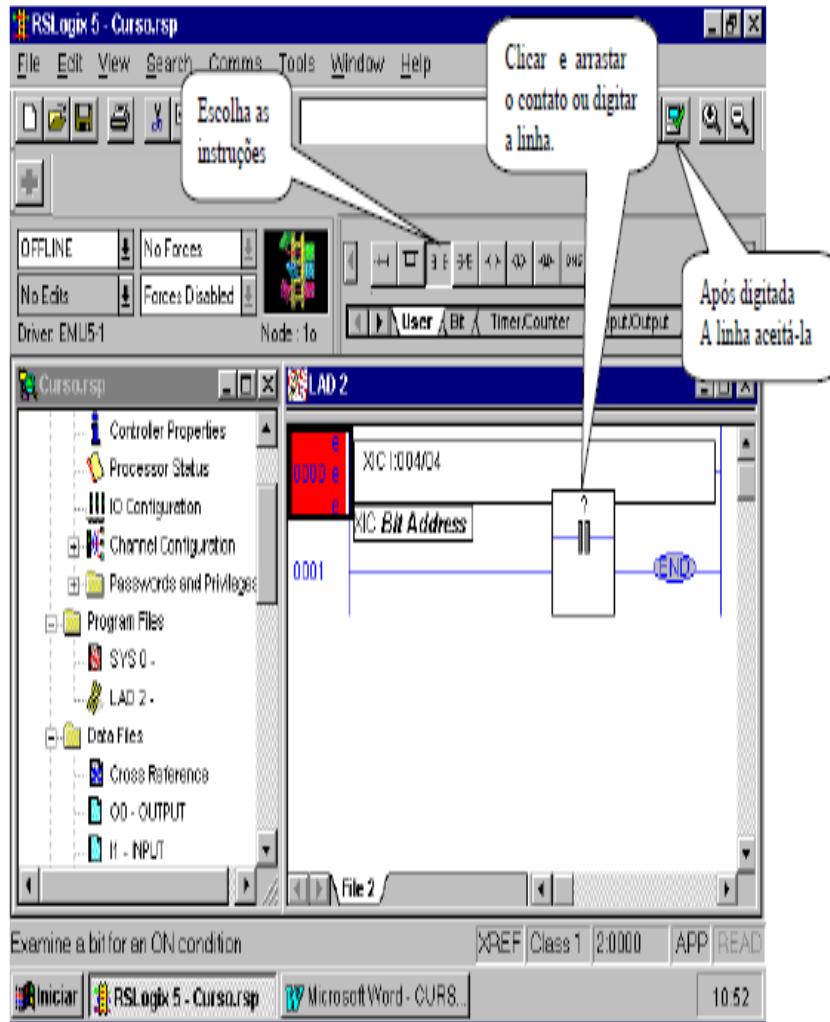
*Escolha do CLP*

Após observar que o módulo do processador (CPU) já está configurado para a gaveta 0, clique então em Read IO Config para que o programa leia os outros cartões acoplados ao CLP.

Os CLPs da família MicroLogix, são programados por meio de uma combinação entre a linguagem ladder com blocos de função.

As instruções podem ser inseridas no programa por meio de linhas de instruções. Para isto, basta dar um duplo-clique na linha desejada e em seguida digitar a instrução e o endereço correspondente.

Pode-se também programar por objetos gráficos, por meio da técnica de arrasta e solta dos símbolos do menu de instruções. É necessário que se faça o endereçamento da instrução após a colocação da figura na posição desejada. Como é mostrado na Figura 4. Após a programação de todo ladder estar finalizada, é recomendado fazer uma verificação lógica do programa, por meio do botão.



Programação Ladder

Havendo erros, eles serão enumerados e indicados na tela. Após finalizar um programa no software RSLogix500, é necessário que o mesmo seja transferido para o CLP; isto é denominado Download.

### 1.8.1 Conexão do PC ao CLP

- 1º passo: Ligar o PC e o CLP e verificar a conexão do cabo serial.
- 2º passo: Abrir o programa RS Linx Classic localizado em Menu Iniciar/Todos os programas/Rockwell Software/RS Linx.
- 3º passo: Na tela do RS Linx Classic Lite, clique em Communications/RSWho. Vide Figura A.2.
- 4º passo: Verifique a existência da conexão AB\_D F1-1, DH-485. Se ela existir, pule direto para o 10º passo.
- 5º passo: Não existindo a conexão acima, é necessário criá-la para que o PC enxergue o CLP. Na tela do RS Linx Classic Lite clique em Communications/Configure Drivers.
- 6º passo: Na tela Configure Drivers em Available Driver Types, selecione a opção RS-232 DFI devices, clique em Add new e, na tela seguinte, em OK.
- 7º passo: A tela Configure RS-232 DFI devices deve abrir automaticamente (vide Figura A.3). Em Comm port, selecione a opção COM1 e clique em Auto config.
- 8º passo: Clique em OK.
- 9º passo: Clique em Close.
- 10º passo: Minimize o RS Linx.

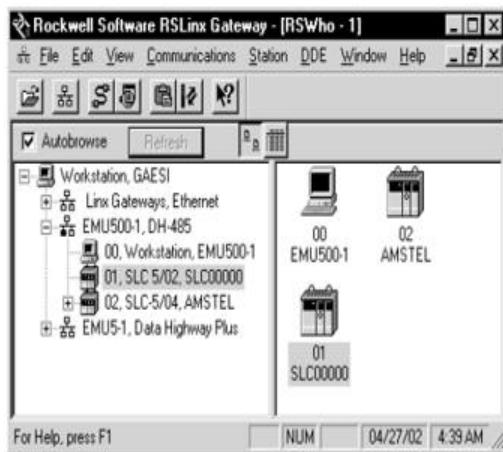


Figura A.2 - Exemplo de tela do RS Linx

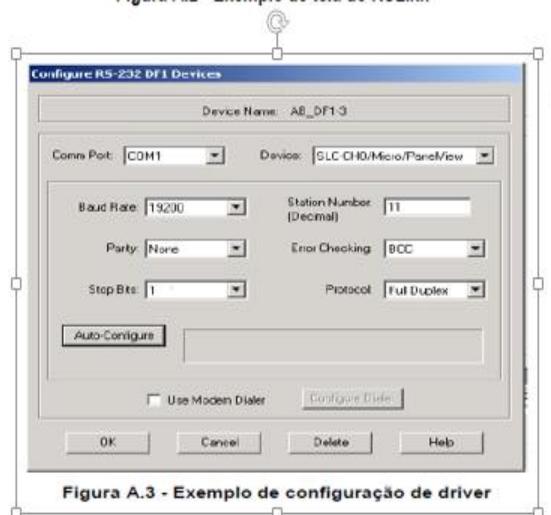


Figura A.3 - Exemplo de configuração de driver

### 1.8.2 Programação do aplicativo Ladder:

Todo novo programa Ladder aplicativo criado precisa seguir esse procedimento para que reconheça o CLP e seus cartões de expansão.

1º passo: Abrir o programa RS Logix 500 english localizado em Menu Iniciar/Todos os programas/Rockwell Software/RS Logix 500 english e Clique em file/new.

2º passo: Abrirá a tela Select processor type. Dê um nome para o seu programa em Processor name (p.ex.: exp1) e selecione o CLP a ser utilizado; no caso, o Bul 1762 Micrologix 1000.

3º passo: Confirme também se o Driver selecionado é o mesmo do item e clique em OK. A tela do seu programa irá abrir.

4º passo: Abra o project/controller/IO Configuration localizado à direita na tela.

5º passo: Na tela IO Configuration, clique em Read IO Config

6º passo: Na tela Read IO Configuration from Online processor, confirme se o Driver é o mesmo e clique em Read IO Config. Esse passo é necessário para garantir que o PC está conectado ao CLP e todas as suas expansões de entrada e saída foram reconhecidas.

7º passo: Fechar a tela IO Configuration.

8º Você está pronto para começar a programação.

9º passo: Crie seu programa clicando nos ícones referente às instruções de programação e arrastando-os para a linha do programa. Ver Figuras A.4 e A.5.

10º passo: Clique em file/save e salve seu programa.

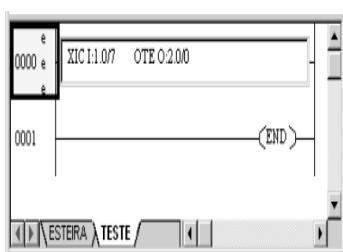


Figura A.4 - Programação ladder por linhas de instruções

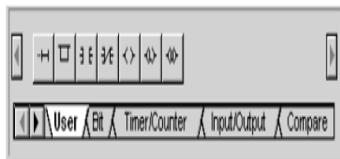


Figura A.5 - Menu de instruções

### 1.8.3 Carregando o programa do PC no CLP:

1º passo: Mude o status localizado no canto superior esquerdo de offline para download. Ver Figura A.7.

2º passo: Confirme todas as janelas que aparecerem.

3º passo: Mude o status localizado no canto superior esquerdo de offline para run. Acione a entrada do CLP correspondente ao programa que você elaborou e veja o que ocorre. Caso a saída que você esperava que fosse acionada for acionada você teve sucesso na execução.

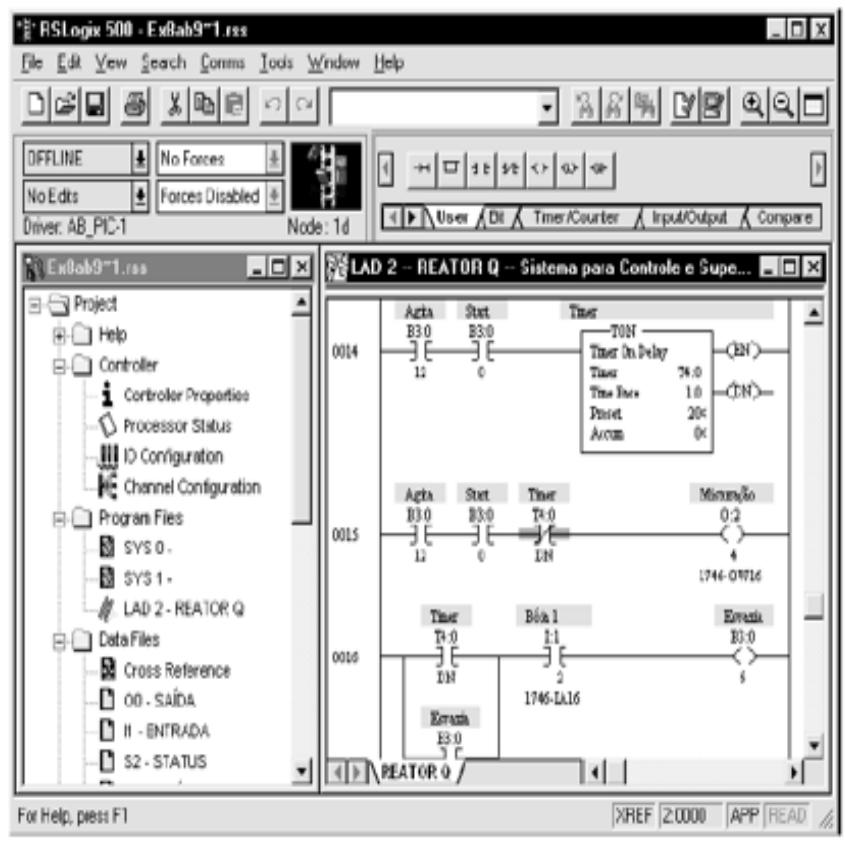
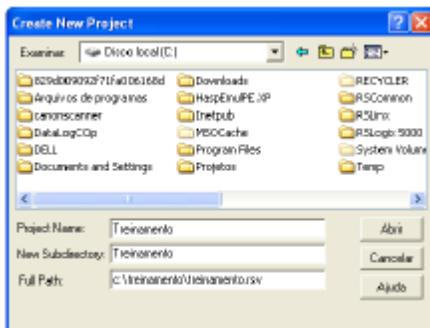


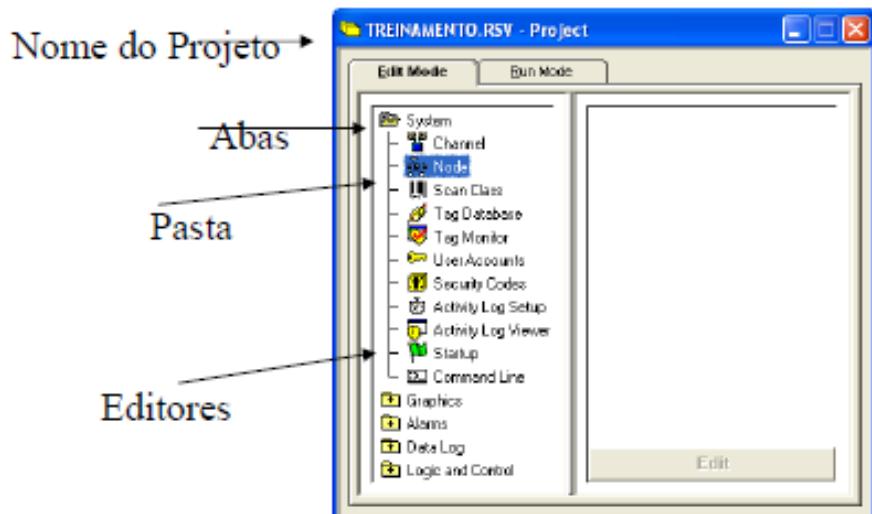
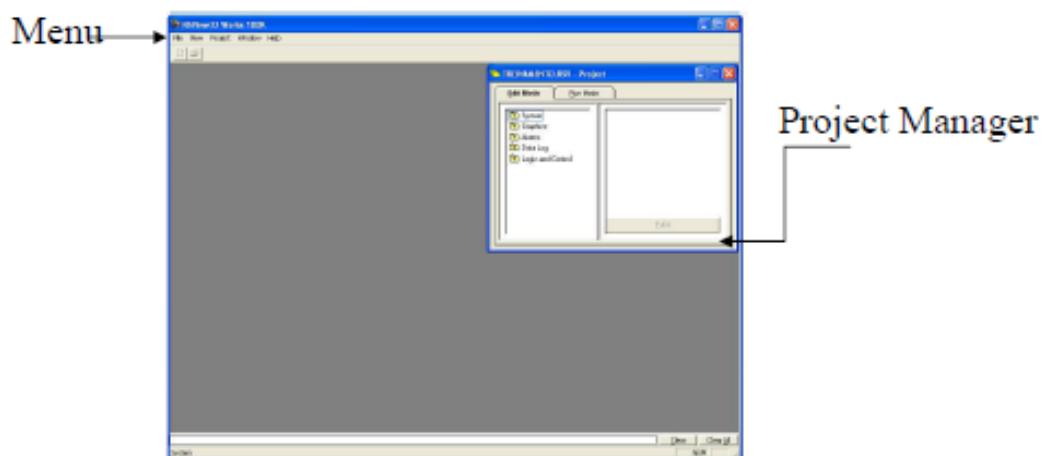
Figura A.7 - Menu Online

## 1.9 Criando Sistemas Supervisórios com RS View 32:

Para criar um novo projeto execute RS View 32 Works, clique em file>New:

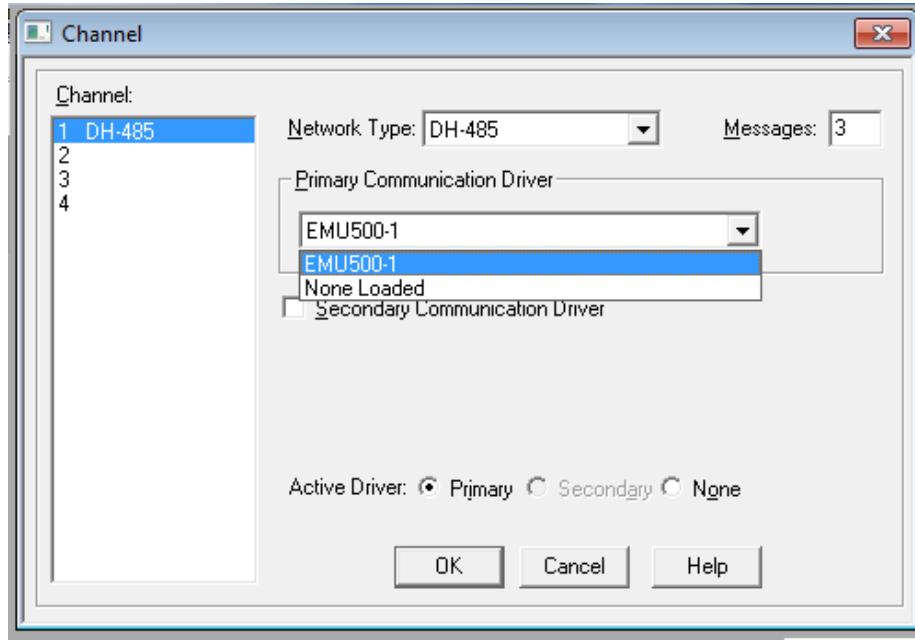


A figura a seguir mostra a área de trabalho:

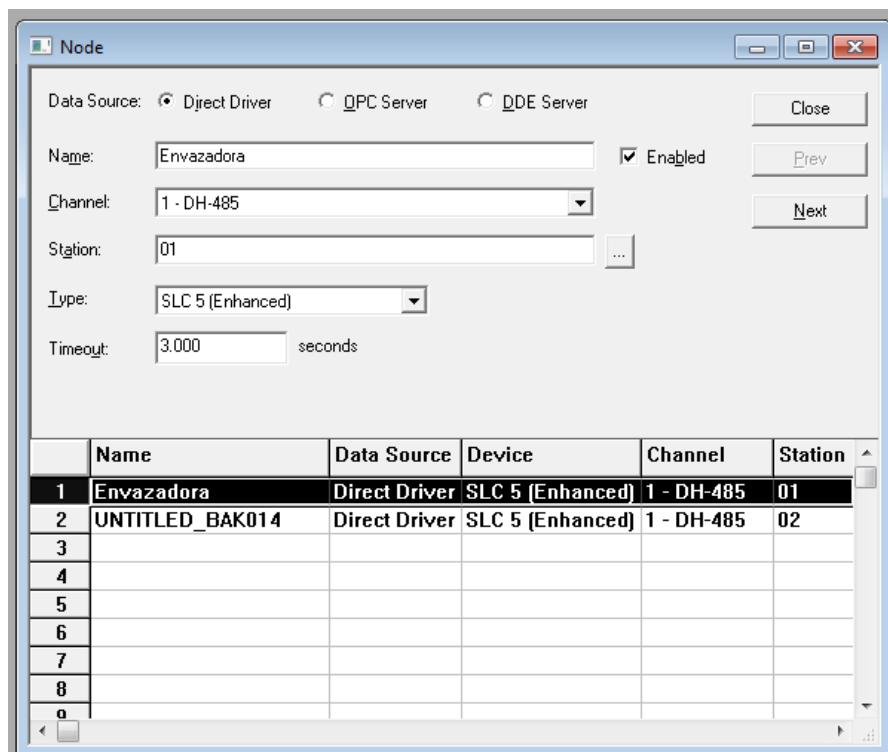


Após ter configurado a comunicação entre o CLP e o computador segundo tópico 1.8 desta apostila podemos configurar as Abas da pasta System no Project Manager.

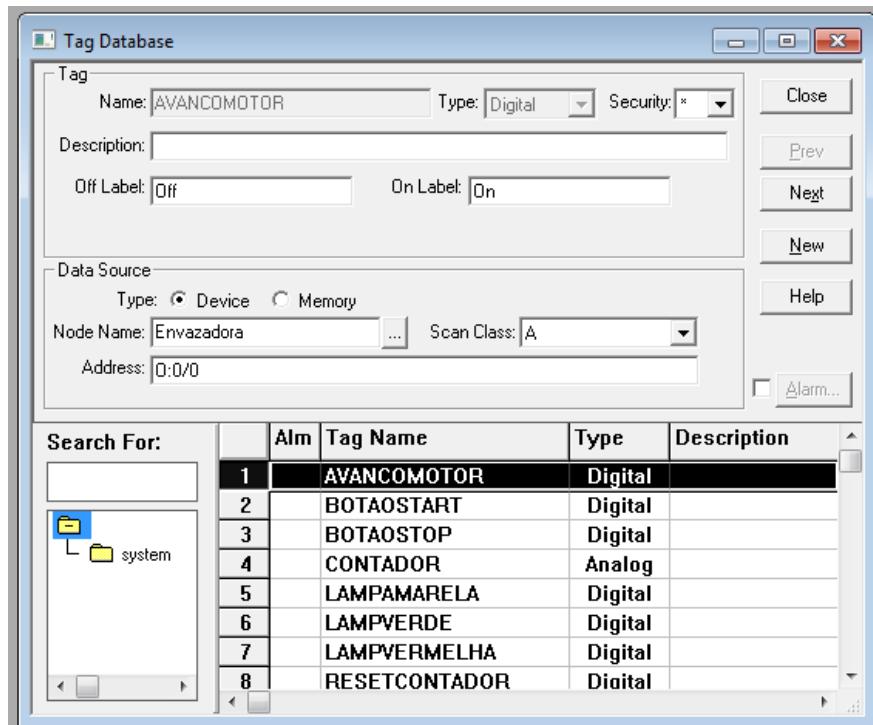
Configure a aba Chanel o canal 1 com o devido cabo de comunicação utilizado e o driver conforme já feito no RS Linx:



Na aba Node configure estabeleça um nome e configure novamente o canal e o driver(station):



Na aba Tag Database podemos definir as tags que serão monitoradas pelo RS View 32:



### 1.9.1 Tipos de Tags:

**String:** Sequencia de até 255 caracteres.

**Digital:** Assume 0 ou 1. Esse tipo de tag é muito utilizado para representar status do tipo On/Off.

**Analog:** Assume o valor real. Esse tipo de tag pode ser utilizado, por exemplo, o valor de uma temperatura em um tanque.

### 1.9.2 Tipos de Data Source:

**Device:** São tags que buscam informação de fontes externas ao RS View 32(tags que se comunicam com PLC, inversores...).

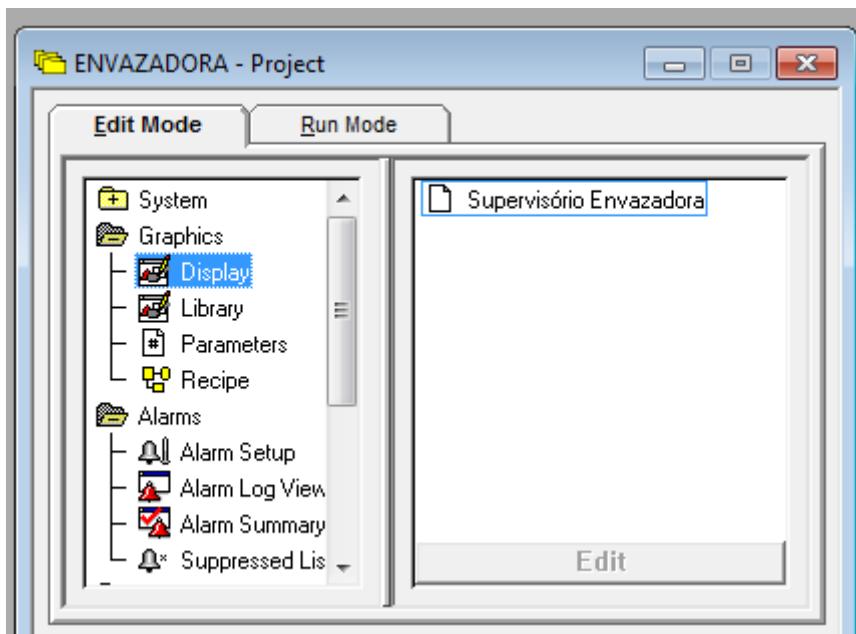
**Memory:** São tags para armazenar valores internos.

### 1.9.3 Criação de Displays Gráficos:

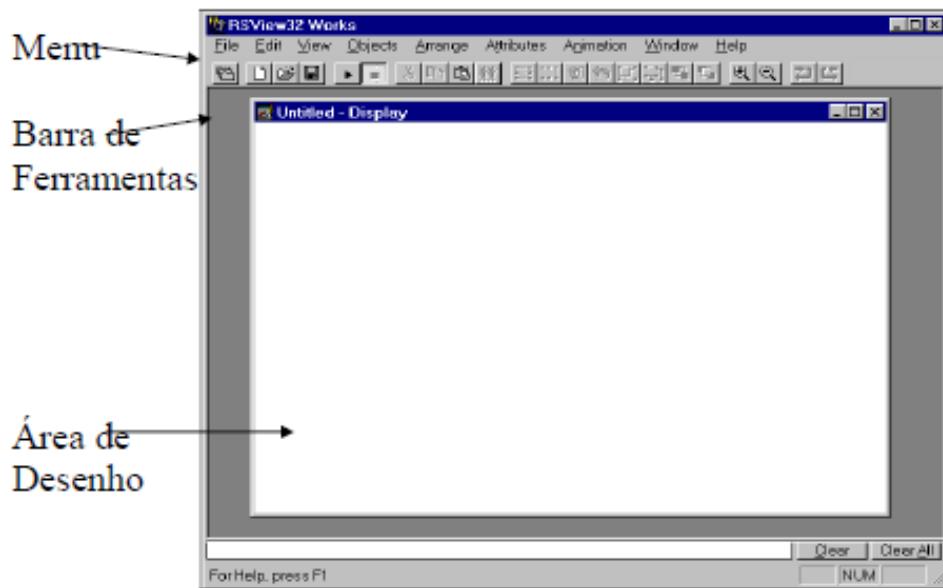
Um display gráfico representa a visão de que o operador tem de um sistema ou um processo industrial. Pode representar uma simples máquina ou até mesmo várias.

Os componentes que fazem parte do display gráfico são chamados objetos.

Para se criar um display, na pasta Graphics dê um duplo clique na aba Display ou clique em um display já existente:



Irá abrir a seguinte área de trabalho:



**Barra de Ferramentas:** Contem botões que são frequentemente utilizados existe a barra de ferramentas de desenho, de preenchimento de cores...

**Área de Desenho:** É a área destinada a criação de displays gráficos ou telas.

## 2. NORMAS DE SEGURANÇA EM ELETRICIDADE

### 2.1 NR10

A NR10 (norma regulamentadora 10) que tem como título, SEGURANÇA EM INSTALAÇÕES E SERVIÇOS EM ELETRICIDADE, é uma norma que tem como caráter regulamentar todos os serviços que envolvam eletricidade e seus riscos, além de garantir a saúde e segurança dos que estejam envolvidos direta e indiretamente nestas atividades e serviços.

Sua primeira publicação é datada no ano 1978, e sua ultima atualização começou a surgir por volta dos anos de 1990, tendo sido finalizada e publicada pelo ministério do trabalho no ano de 2004. Com modificações profundas nas questões de documentação e normatização dos serviços, seu texto de 2004, ficou popularmente conhecido entre os eletricistas e profissionais da área de elétrica com sendo A NOVA NR10. Seu maior destaque é que esta versão oficializou que todas as instalações elétricas devem atender a quesitos e as normas da ABNT (Associação Brasileira de Normas Técnicas), e caso as empresas e instalações não se enquadrem nas normas as mesmas serão penalizadas.

Para os profissionais que se envolvam diretamente nas atividades de instalação, manutenção e operação de maquinas e instalações elétricas as mudanças proporcionadas pela NR10 vieram a reduzir drástica e positivamente os índices de acidentes que envolvem eletricidade. Entre as principais mudanças se destacam a obrigatoriedade de todos os profissionais que se envolvam com as atividades elétricas passarem por um treinamento de 40h (para o curso básico), em que será apresentada a normas e suas características, e que no mínimo a cada dois anos esses mesmos profissionais terão de passar por uma reciclagem deste mesmo treinamento. Essa obrigatoriedade torna cíclico o processo de conscientização dos riscos da eletricidade para estes profissionais.

#### 2.1.1 Fases que se aplica a NR 10

A NR se aplica às fases de geração, transmissão, distribuição e consumo, incluindo as etapas de projeto, construção, montagem, operação, manutenção das instalações elétricas e quaisquer trabalhos realizados nas suas proximidades, observando-se as normas técnicas oficiais estabelecidas pelos órgãos competentes e, na ausência ou omissão destas, as normas internacionais cabíveis.

#### 2.1.2 Prontuário de Instalações Elétricas

Os estabelecimentos com carga instalada superior a 75 kW devem constituir e manter o Prontuário de Instalações Elétricas, contendo, além do disposto no subitem 10.2.3, no mínimo:

- Conjunto de procedimentos e instruções técnicas e administrativas de segurança e saúde, implantadas e relacionadas à NR 10 e descrição das medidas de controle existentes;
- Documentação das inspeções e medições do sistema de proteção contra descargas atmosféricas e aterramentos elétricos;
- Especificação dos equipamentos de proteção coletiva e individual e o ferramental, aplicáveis conforme determina esta NR;
- Documentação comprobatória da qualificação, habilitação, capacitação, autorização dos trabalhadores e dos treinamentos realizados;
- Resultados dos testes de isolação elétrica realizados em equipamentos de proteção individual e coletiva;
- Certificações dos equipamentos e materiais elétricos em áreas classificadas;
- Relatório técnico das inspeções atualizadas com recomendações, cronogramas de adequações, contemplando as alíneas de “a” a “f”.

### 2.1.3 Medidas de Proteção Coletiva

As medidas de proteção coletiva compreendem, prioritariamente, a desenergização elétrica conforme estabelece esta NR e, na sua impossibilidade, o emprego de tensão de segurança.

Na impossibilidade de implementação do estabelecido no subitem 10.2.8.2. Devem ser utilizadas outras medidas de proteção coletiva, tais como: isolação das partes vivas, obstáculos, barreiras, sinalização, sistema de seccionamento automático de alimentação, bloqueio do religamento automático.





#### **2.1.4 Segurança em Projetos**

É obrigatório que os projetos de instalações elétricas especifiquem dispositivos de desligamento de circuitos que possuam recursos para impedimento de reenergização, para sinalização de advertência com indicação da condição operativa.

#### **2.1.5 Segurança em Instalações Desenergizadas**

Somente serão consideradas desenergizadas as instalações elétricas liberadas para trabalho, mediante os procedimentos apropriados, obedecida a sequência abaixo:

- seccionamento;
- impedimento de reenergização;
- constatação da ausência de tensão;
- instalação de aterramento temporário com equipotencialização dos condutores dos circuitos;
- proteção dos elementos energizados existentes na zona controlada (Anexo II);
- instalação da sinalização de impedimento de reenergização.

O estado de instalação desenergizada deve ser mantido até a autorização para reenergização, devendo ser reenergizada respeitando a sequência de procedimentos abaixo:

- retirada das ferramentas, utensílios e equipamentos;

- retirada da zona controlada de todos os trabalhadores não envolvidos no processo de reenergização;
- remoção do aterramento temporário, da equipotencialização e das proteções adicionais;
- remoção da sinalização de impedimento de reenergização;
- destravamento, se houver, e religação dos dispositivos de seccionamento.

### **2.1.6 Segurança em instalações elétricas Energizadas**

As intervenções em instalações elétricas com tensão igual ou superior a 50 Volts em corrente alternada ou superior a 120 Volts em corrente contínua somente podem ser realizadas por trabalhadores qualificados.

**Trabalhador Qualificado:** É considerado trabalhador qualificado aquele que comprovar conclusão de curso específico na área elétrica reconhecido pelo Sistema Oficial de Ensino.

**Profissional Legalmente Habilitado:** É considerado profissional legalmente habilitado o trabalhador previamente qualificado e com registro no competente conselho de classe.

**Trabalhador Capacitado:** É considerado trabalhador capacitado aquele que atenda às seguintes condições, simultaneamente:

- receba capacitação sob orientação e responsabilidade de profissional habilitado e autorizado; e
- trabalhe sob a responsabilidade de profissional habilitado e autorizado.

A capacitação só terá validade para a empresa que o capacitou e nas condições estabelecidas pelo profissional habilitado e autorizado responsável pela capacitação.

### **2.1.7 Responsabilidade os trabalhadores nos serviços em eletricidade**

- zelar pela sua segurança e saúde e a de outras pessoas que possam ser afetadas por suas ações ou omissões no trabalho;
- responsabilizar-se junto com a empresa pelo cumprimento das disposições legais e regulamentares, inclusive quanto aos procedimentos internos de segurança e saúde; e
- comunicar, de imediato, ao responsável pela execução do serviço as situações que considerar de risco para sua segurança e saúde e a de outras pessoas.

## **2.2 NBR-5410**

A NBR-5410 é a norma que estipula as condições adequadas para o funcionamento usual e seguro das instalações elétricas de baixa tensão, ou seja, até 1000V em tensão alternada e 1500V em tensão contínua. Esta norma é aplicada principalmente em instalações prediais, públicas, comerciais, etc. Para o profissional da área funciona como um guia, sobre o que se deve ou não fazer, ela traz um texto diferenciado explicando e colocando regras em instalações de baixa tensão, e faz grande diferença conhecê-la e acima de tudo aplicá-la. Conhecer a norma e os tópicos nela propostos esclarece muitas das dúvidas dos profissionais da área.

No geral, esta norma estabelece as condições a que devem satisfazer as instalações elétricas de baixa tensão a fim de garantir a segurança de pessoas e animais, o funcionamento adequado da instalação e conservação dos bens. Ou seja, segurança das pessoas e animais que habitam a instalação, funcionamento e conservação dos bens. Confira a NBR-5410 comentada no vídeo abaixo. Este vídeo é a explicação ilustrada e comentada da norma e seus objetivos.

### **2.2.1 NBR-5410 se aplica:**

Como dito anteriormente, a NBR-5410 é uma normatização voltada às instalações prediais, porém quando se fala de instalação predial, logo pensamos na instalação residencial, por isso os tópicos abaixo esclarecem e exemplificam a aplicação desta norma.

- Áreas descobertas externas a edificações;
- Locais de acampamento, marinha e instalações análogas;
- Instalações temporárias como canteiros de obras, feiras, etc.;
- Circuitos elétricos alimentados sob tensão nominal igual ou inferior a 1000 V em corrente alternada (CA), frequência inferior a 400 Hz, ou a 1500 V e corrente contínua (CC) (modificação vinda da norma NR-10, que estabelece o que é baixa tensão);
- Circuitos elétricos que não estão dentro de equipamentos, funcionando sobre tensão superior a 1000 volts, e alimentados por uma instalação igual ou inferior a 1000 volts e corrente alternada. Circuitos de lâmpadas de descarga, por exemplo;
- Fiação e redes elétricas que não estejam cobertas pelas normas relativas aos equipamentos de utilização;
- Linhas elétricas fixas de sinal com exceção dos circuitos internos dos equipamentos
- Instalações novas e já existentes, sobre reforma;

### **2.2.2 NBR-5410 não se aplica:**

Alguns dos pontos citados pertencem a normas próprias e específicas a instalação, mesmo estando dentro das instalações de baixa tensão, por isso a NBR-5410 não se aplica aos mesmos.

- Instalações de tração elétrica;
- Instalações elétricas de veículos motores, carros elétricos, por exemplo;
- Instalações de embarcações e aeronaves;
- Equipamentos para supressão de perturbações radioelétricas, na medida em que não comprometa a segurança das instalações;
- Iluminação pública;
- Redes públicas de distribuição elétrica
- Instalações de proteção contra quedas diretas de raios, porém esta norma considera as consequências dos fenômenos atmosféricos sobre as instalações, por exemplo, seleção dos dispositivos de proteção contra sobre tensão;
- Instalações em minas;

- Instalações em cercas elétricas;

A aplicação da NBR-5410 não dispensa o seguimento de outras normas aplicadas em situações ou lugares específicos e os regulamentos que a instalação deve seguir.

### **2.2.3 Importância do cumprimento da Norma**

Ter uma instalação baseada nas normas é indiscutivelmente o correto, pois assim fica assegurado o bom funcionamento, a conservação dos bens e principalmente a segurança. Normas existem para regulamentar, trazer uma igualdade as demais instalações elétricas e melhorar o âmbito de qualidade das instalações elétricas, e a NBR-5410 existe justamente pela preocupação com as instalações elétricas de baixa tensão, pois muitos acidentes ocorrem neste tipo de instalação com usuários que nem sempre possuem qualificação. Cumprir a norma é assegurar que estas instalações estejam dentro do que é considerado um funcionamento seguro.

## **6. BIBLIOGRAFIA**

**Franchi, Claiton Moro.** Controladores Lógicos Programáveis – Sistemas Discretos / Valter Luís Arlindo de Camargo. --2.ed. – São Paulo: Érica, 2009.

**Franchi, Claiton Moro.** Inversores de Frequência: teoria e Atividades --2.ed. – São Paulo: Érica, 2009.

**G.Nascimento.** Comandos Elétricos: teoria e Atividades --1.ed. – São Paulo: Érica, 2011