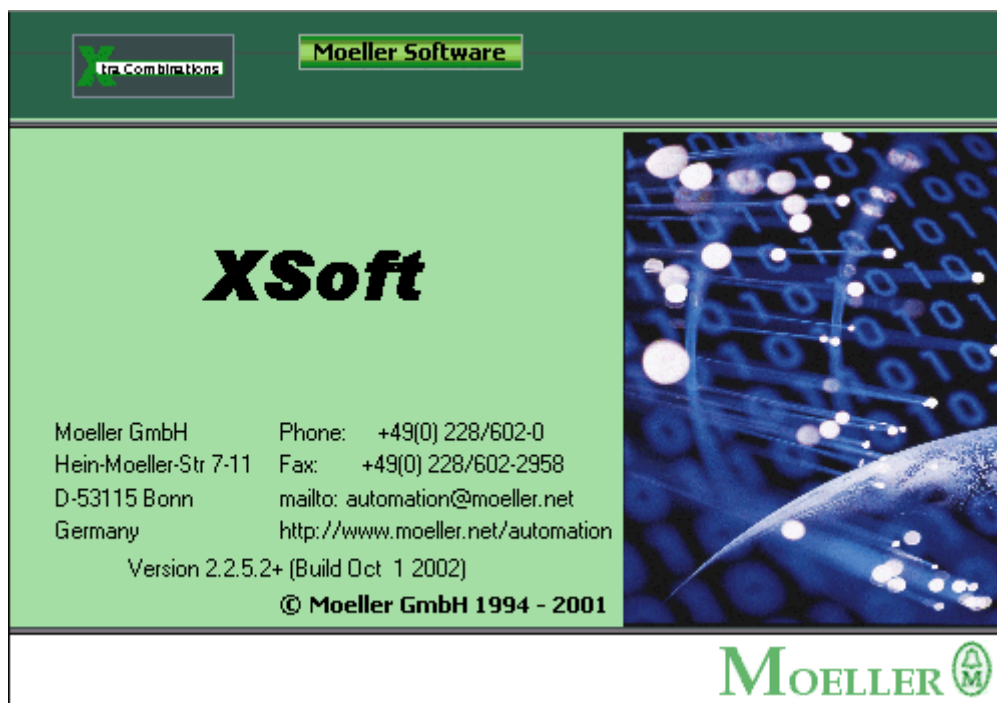


XSOFT 2.2.5



GUIA RÁPIDO DE PROGRAMAÇÃO DOS CLP'S LINHAS XC100/200/600

INTRODUÇÃO

No mercado pode-se encontrar uma variedade de diferentes linguagens e sistemas de programação. Para cada tipo de linguagem são necessários dispendiosos treinamentos para todos os tipos de controladores, tanto para a linguagem de programação como para o sistema operacional.

O usuário não consegue trocar unidades de programas entre diferentes sistemas, porque as linguagens de programação não são compatíveis entre si.

Não é possível a reutilização de funções programadas sem que seja necessária uma reprogramação, pois o modelo de dados destes CLP's necessita de endereços físicos.

Pelo exposto acima a utilização de um controlador de outro fabricante não era viável por causa do alto custo.

O principal objetivo da IEC1131.3 é reduzir, se não eliminar, todos os problemas citados acima, criando uma estruturação e forma de programação padrão, de tal forma que o usuário não se torne dependente do fabricante.

CAPÍTULO 1

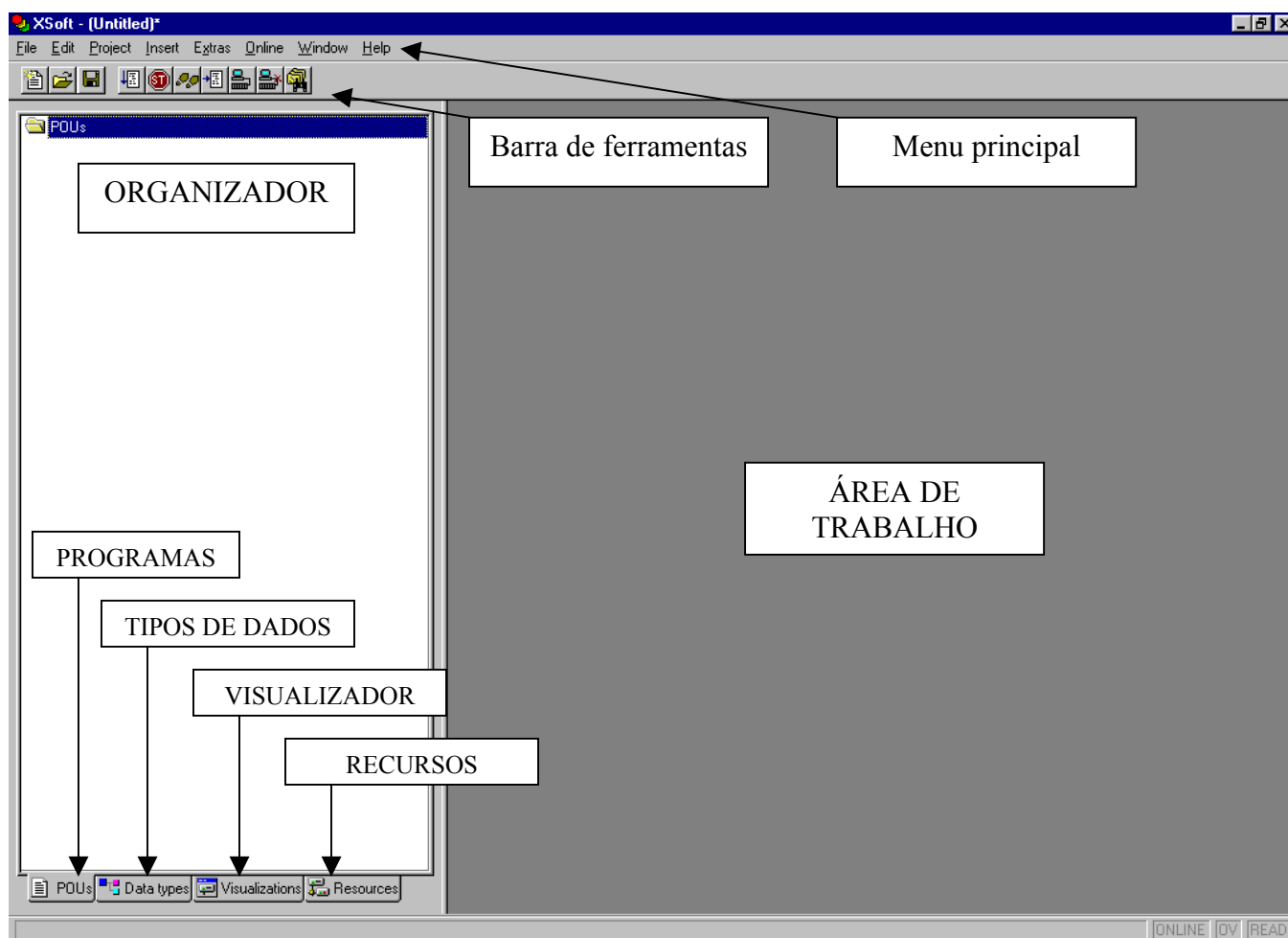
REQUISITOS E OBJETIVOS

Antes de programar é necessário um conhecimento básico da linha de controladores lógicos programáveis da Moeller. Este manual de programação rápida destina-se a auxiliar o usuário a iniciar-se na programação do Software XSOFT Versão 2.2.5 em ambiente para Windows.

CAPÍTULO 2

INICIANDO O SOFTWARE

Através do menu INICIAR (START), faça a chamada do XSOFT.



ORGANIZADOR - é usado para gerenciar os novos itens de cada pasta. Através dele inserimos novos elementos nas pastas localizadas abaixo e solicitamos a visualização/edição de alguns itens.

ORGANIZADOR / PROGRAMAS – Nesta pasta alocamos os programas do CLP, bem como suas funções e seus blocos de função.

ORGANIZADOR / TIPOS DE DADOS – Aqui nós colocamos nossos tipos de dados , Structures, Enumeration, etc...São tipos de dados criados pelo usuário conforme a IEC 61131-3

ORGANIZADOR / VISUALIZADOR – Dentre os recursos oferecidos pelo Xsoft, está a criação de telas para visualização do programa criado, tanto on-line como off-line (modo simulação) , estas telas são criadas e gerenciadas nesta pasta.

ORGANIZADOR / RECURSOS – Pasta responsável por vários itens no Xsoft. Entre eles: Configuração de hardware, configuração do sistema da CPU, criação de variáveis de rede, geração de gráficos de tendência, criação de tabelas de monitoração etc...

ÁREA DE TRABALHO – Quando selecionamos ou criamos algo em qualquer pasta do organizador, este item aparece na área de trabalho para edição ou visualização.

BARRA DE FERRAMENTAS – Aqui ficam algumas ferramentas para acesso mais rápido a algumas funções do programa, esta barra muda conforme o item selecionado na Área de Trabalho.

MENU PRINCIPAL – Aqui ficam todas as opções de acessibilidade e funcionamento do XSOFTE.

Os passos necessários para a criação de um programa são:

- **Criar um novo projeto**
- **Configurar o Hardware**
- **Criar os arquivos fontes necessários (POU)**
 - Criar as Variáveis
 - Realizar a lógica do programa
 - Criar blocos de função e /ou funções.
- **Compilar o programa**
- **Realizar os testes**
 - Configurar as telas / ferramentas de visualização.
 - Executar os testes do programa em modo OFF-line.
- **Realizar o comissionamento da máquina.**
 - Transferir o programa para o CLP
 - Executar os testes do programa em modo ON-line
 - Armazenar os programas fontes no CLP. (Quando desejado)

CAPÍTULO 3

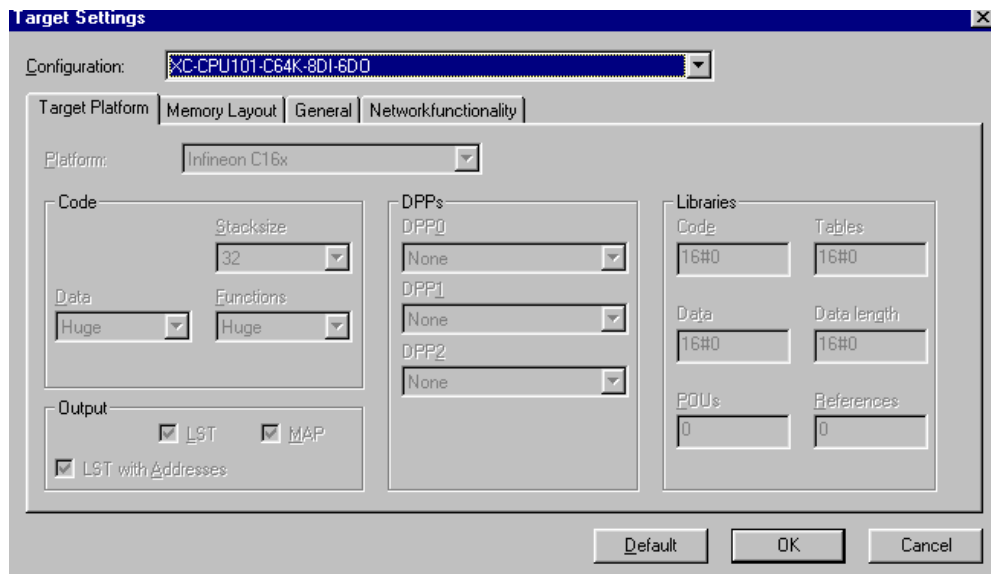
INICIANDO UM NOVO PROJETO

- Entre no menu principal opção FILE → NEW , ou seleccione o ícone . Isto irá abrir a seguinte tela:

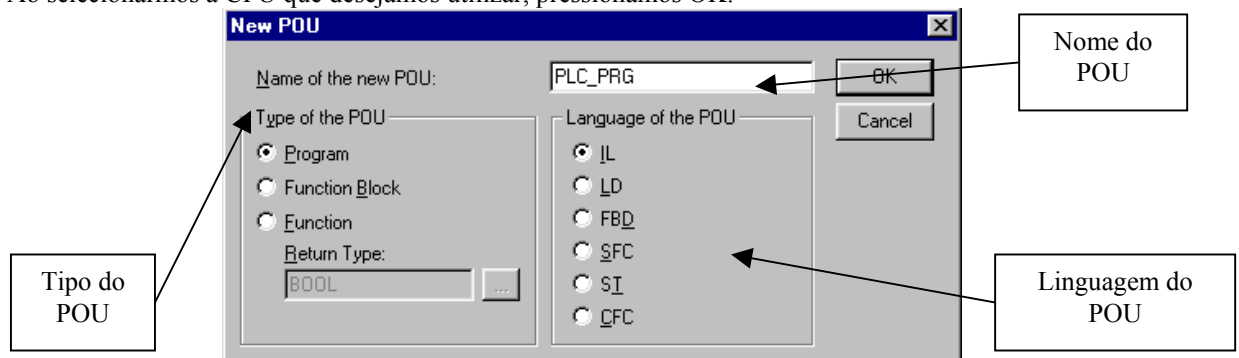


Seleção do tipo de CPU

- No item Configuration seleccione a CPU que irá utilizar para este programa, por exemplo:



- Ao seleccionarmos a XC-CPU101-C64K-8DI-6DO escolhemos uma CPU do modelo XC100. Com isto a tela do Target Settings , e o XSOFT, se modela para podermos configurar este tipo de CPU.
- O usuário não terá acesso aos itens: Target Plataform, Memory Layout e General. Estes itens só são configuráveis em outros modelos de CPU. O item Networkfunctionality , serve para configurarmos a rede deste equipamento, este tópico será abordado em outras documentações.
- Ao seleccionarmos a CPU que desejamos utilizar, pressionamos OK.



- Na opção Tipo do POU selecionamos se queremos criar um programa, um Function Block ou uma função.

Funções são elementos de software, que podem ser utilizados várias vezes dentro de um mesmo programa, de tal forma que, quando executados com um certo conjunto de valores de entrada, sempre produzem um único valor como resultado.

Blocos de funções são uma categoria da Unidade de Organização de Programa (POU). Que permitem um algoritmo específico, ou conjunto de ações, que podem ser chamados em várias instâncias dentro de um mesmo programa, de tal forma que, a serem aplicados em um conjunto de dados de entrada eles produzem um novo conjunto de dados de saída.

- No nosso caso iremos selecionar a opção Program, pois iremos criar primeiro um programa.
- Na opção nome do POU deixamos PLC_PRG , este nome designa o programa principal. Para os CLPs da linha XC100 e XC200, devemos ter sempre um programa principal com este nome, pois o compilador sabe que é por este programa que o equipamento irá começar a execução.
- Na linguagem podemos escolher entre seis linguagens de programação diferentes. Estas linguagens se dividem da seguinte forma:

CONFORME IEC 6 1131.3	REPRESENTAÇÃO	NOMENCLATURA	ABREVIÇÃO
Normalizada	Textual	Instruction List (Lista de instruções)	IL
		Structured Text (Estruturada)	ST
	Gráfica	Function Block Diagram (Blocos Lógicos)	FBD
		Ladder Diagram (Ladder / Lógica Relê)	LD
Não Normalizada	Gráfica	Sequential Function Chart (Diagrama sequencial)	SFC
		Continuos Flow Chart editor (CFC)	CFC

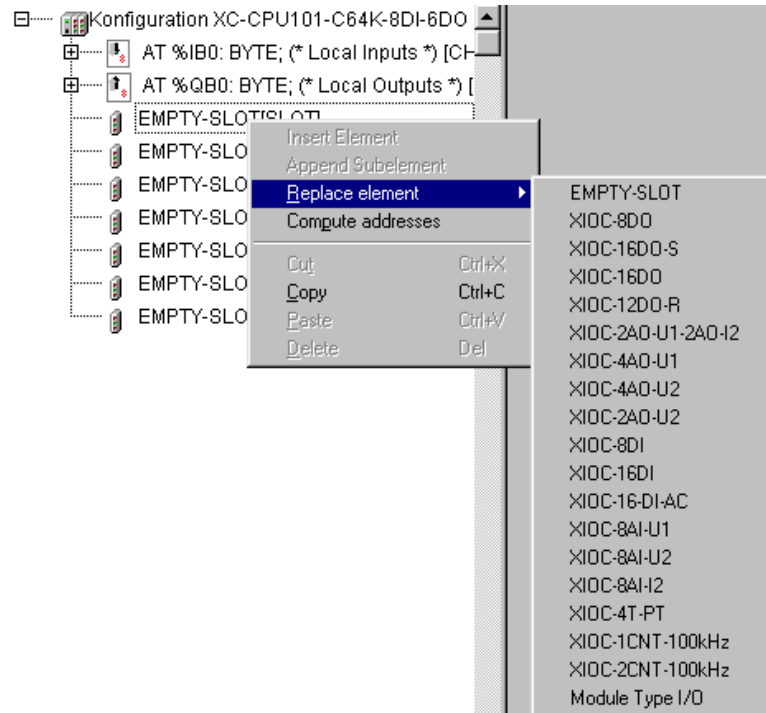
- Para este guia rápido estaremos trabalhando com a linguagem em Ladder (LD).
- Assim que pressionarmos OK na tela New program estaremos dentro do ambiente Xsoft.
- Agora nós precisamos configurar o hardware do CLP. Para isto selecione no organizador a pasta resources:

	Recursos
Global Variables	Variáveis Globais
library Counter.lib*9.1.03 13:12:26: global variables	Bibliotecas
library RTCLib.lib*4.9.02 10:06:00: global variables	Bibliotecas
library Standard.lib*7.6.02 08:26:00: global variables	Bibliotecas
PLC - Browser	Browser
PLC Configuration	Configurador
Sampling Trace	Ferramenta de Visualização (TRACE)
Target Settings	Configuração da CPU
Task configuration	Gerenciador de Tarefas
Watch- and Receipt Manager	Ferramenta de visualização

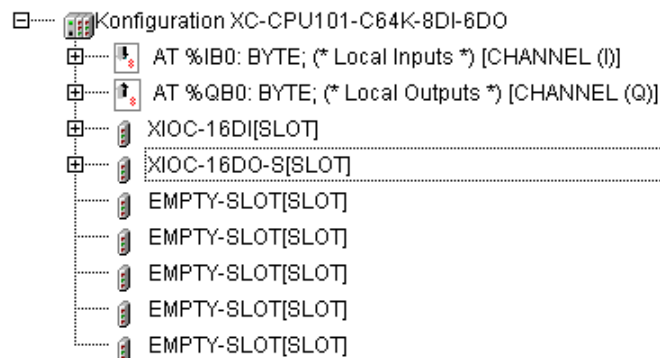
- Selecione a opção PLC-Configuration. Nesta tela estaremos configurando o Hardware do CLP.

Topologia do equipamento	Parametrização

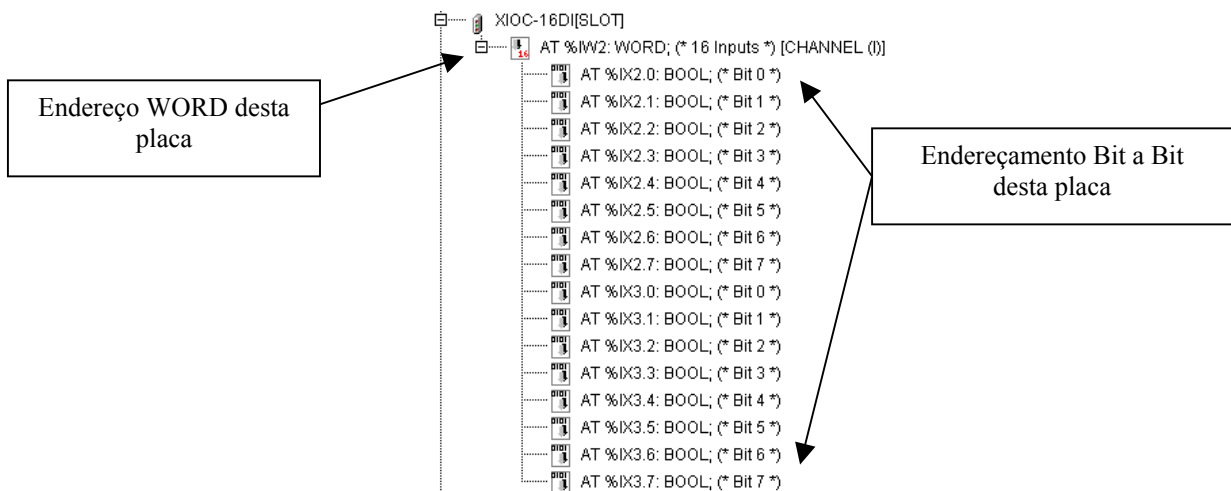
- Para adicionar mais placas selecione um dos EMPTY-SLOT com o botão direito do mouse, logo em seguida escolha a opção Replace Element:



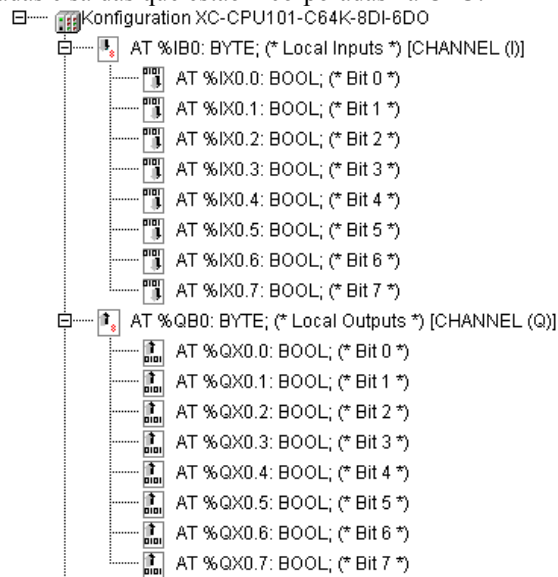
- Nesta tela escolhemos qual o módulo local que será colocado com a CPU, no caso do XC100 podemos colocar até 7 módulos locais ao lado da CPU.
- Para este exemplo selecione o módulo XIOC-16-DI.
- Coloque mais um módulo XIOC-16-DO-S ao lado.



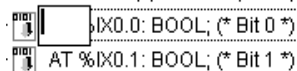
- Pressione com o botão esquerdo do mouse o símbolo + que aparece do lado esquerdo do XIOC-16DI.
- Selecione novamente o símbolo + que aparece do lado esquerdo do AT %IW2 : WORD ;



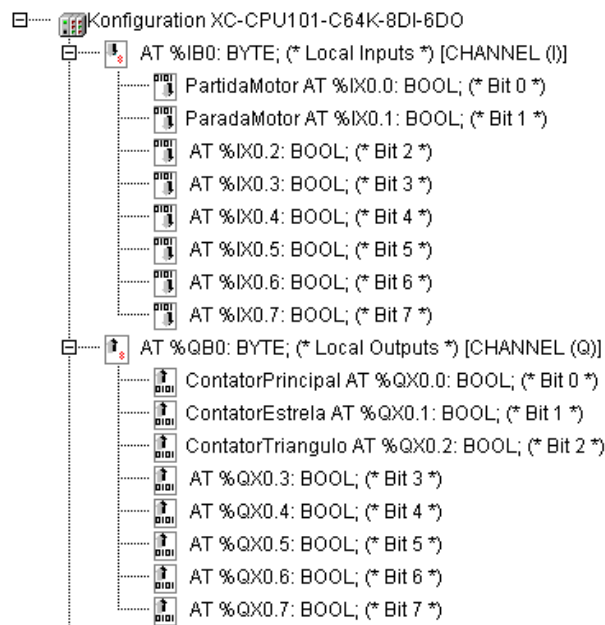
- Para maiores dados sobre variáveis consulte o anexo VARIÁVEIS neste manual!
- Abra os sub-menús das entradas e saídas que estão incorporadas na CPU.



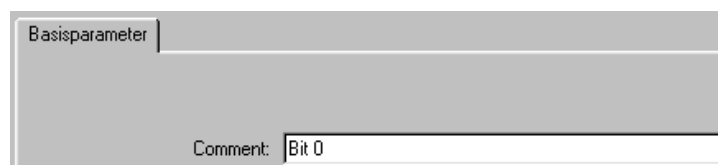
- Clique duas vezes com o botão esquerdo do mouse, entre o quadrado e o AT %IX0.0 : BOOL ;



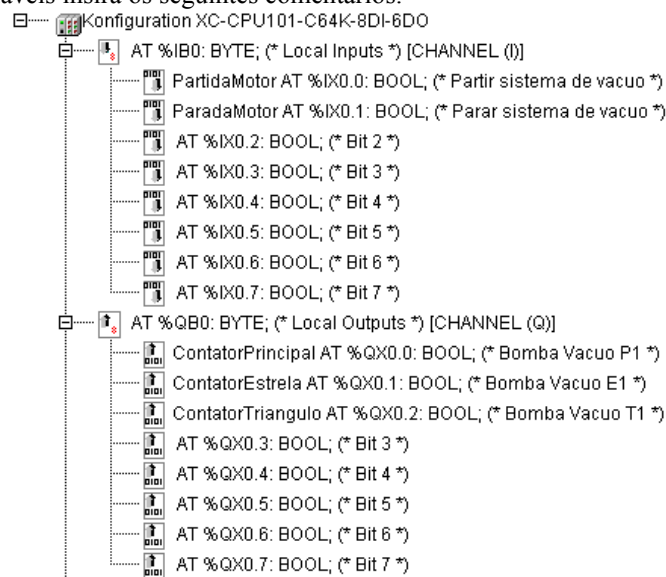
- Aqui escreveremos o nome da variável, no nosso caso PartidaMotor.
- Crie as seguintes variáveis nas entradas e saídas do CLP.




- Do lado esquerdo da tela existe a opção de inserirmos comentários para cada uma destas variáveis.



- Para cada uma das variáveis insira os seguintes comentários.

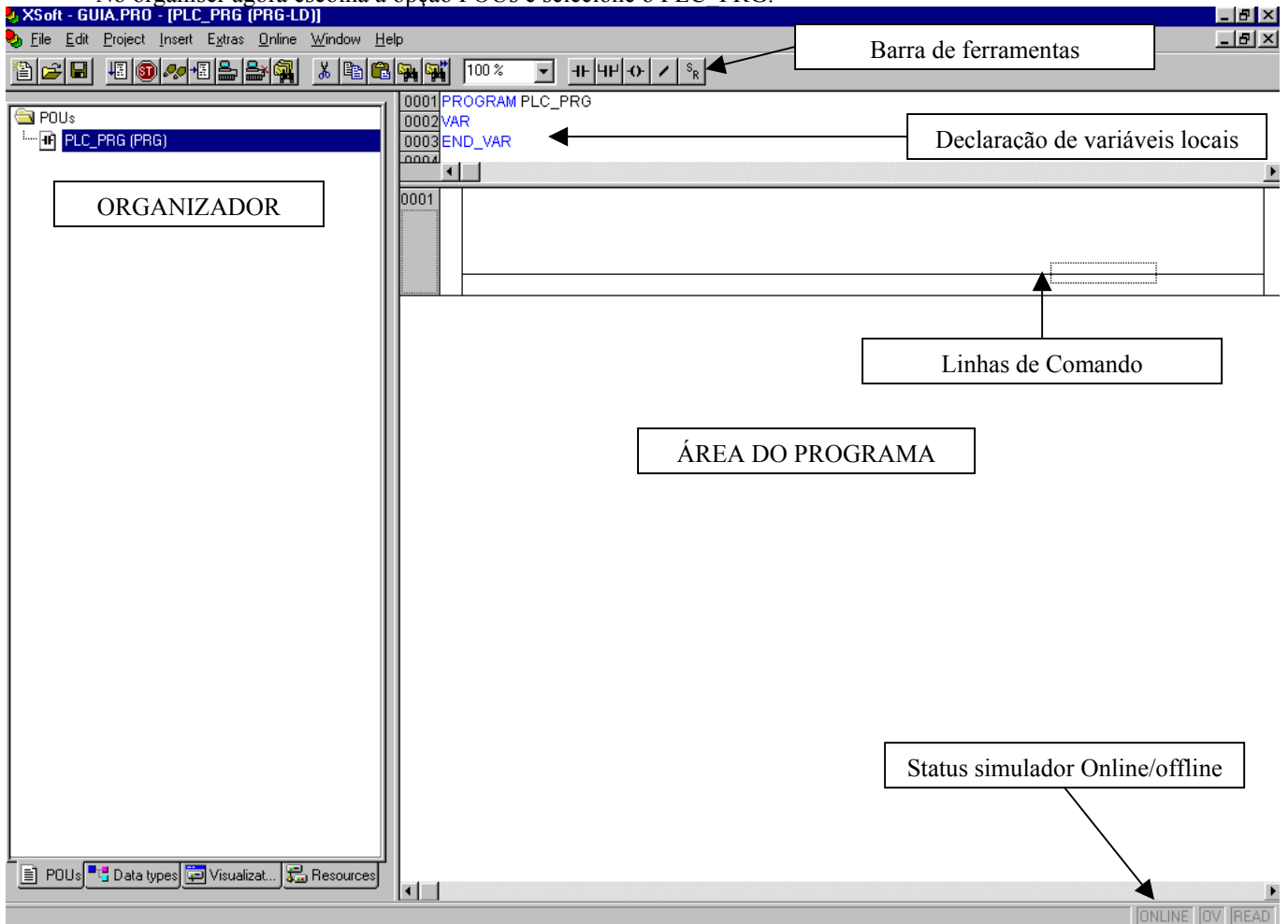



- Salve agora o que foi feito indo no menu File → Save. Ou clique no ícone .

CAPÍTULO 4

COMEÇANDO A PROGRAMAR

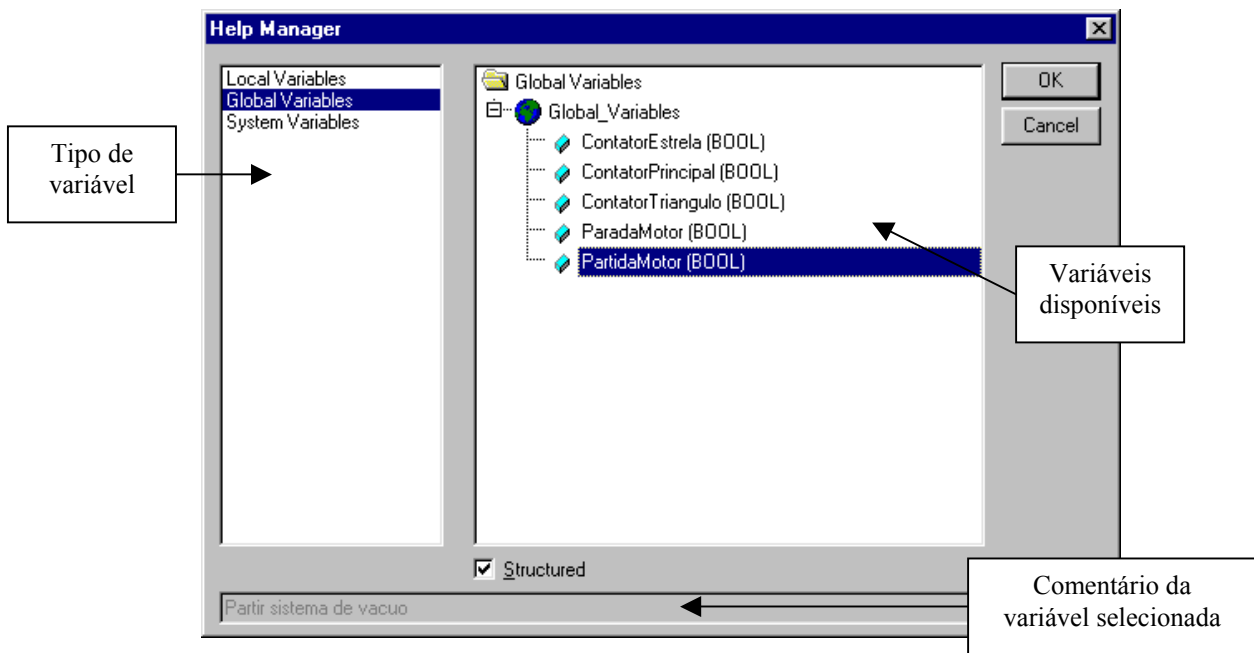
- Agora iremos criar um programa simples para conhecermos melhor o ambiente do Xsoft. O programa a ser criado é uma partida estrela-triângulo de uma bomba de vácuo.
- No organiser agora escolha a opção POU's e selecione o PLC PRG.



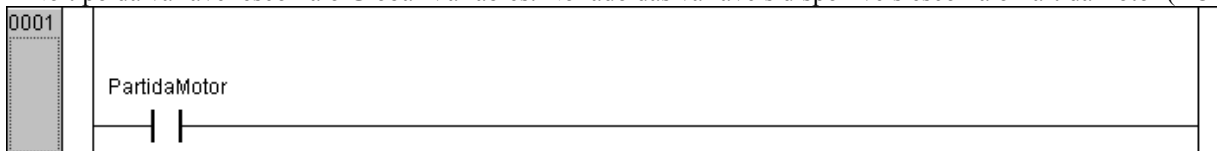
- Na área do programa clique sobre a linha de comando que aparece e clique no menu em Insert → Contact, ou pressione o ícone  na barra de ferramentas.



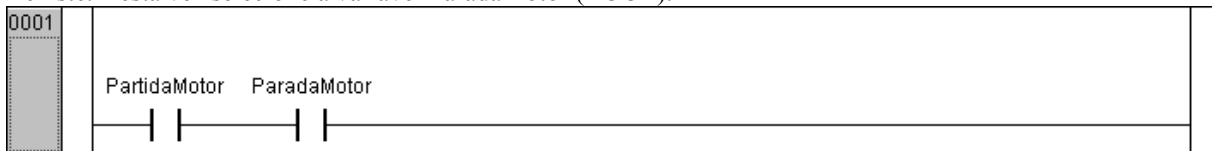
- Na linha de comandos irá aparecer um contato NA :
- Clicando nas interrogações poderemos escrever o nome deste contato. Podemos ainda pressionar a tecla F2 de nosso computador:




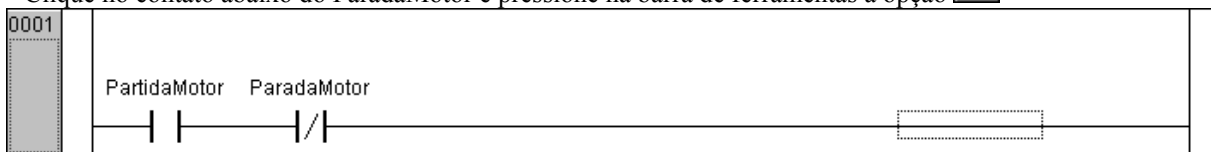
- No tipo da variável escolha o Global Variables. No lado das variáveis disponíveis escolha o PartidaMotor (BOOL).




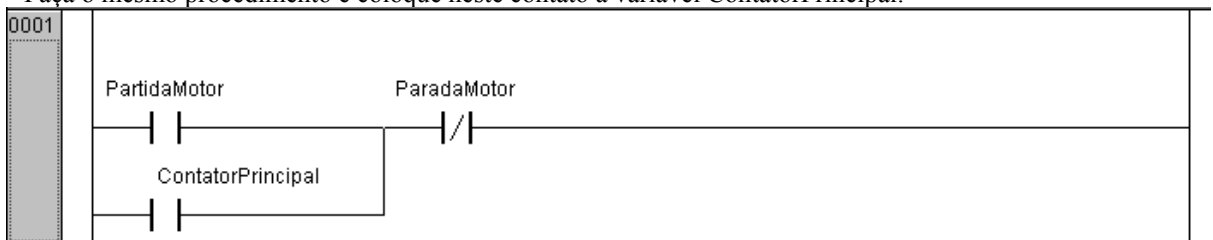
- Clique novamente na linha de comandos, e insira novamente um contato, este contato irá sair em série com o que já existe. Desta vez selecione a variável ParadaMotor (BOOL).




- Clique no contato abaixo do ParadaMotor e pressione na barra de ferramentas a opção .

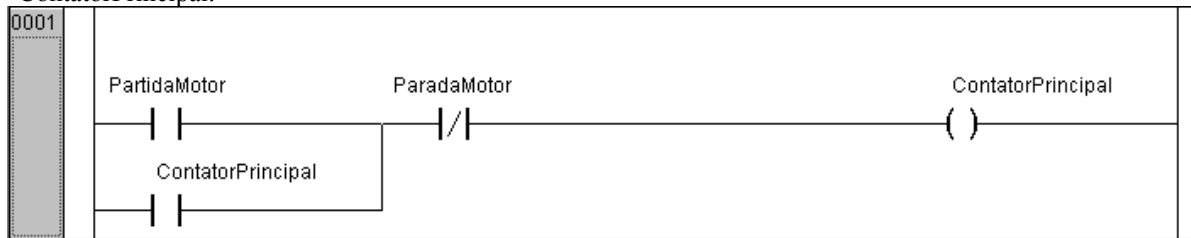


- Clique agora no contato abaixo do PartidaMotor e selecione na barra de ferramentas a opção .
- Faça o mesmo procedimento e coloque neste contato a variável ContatorPrincipal.

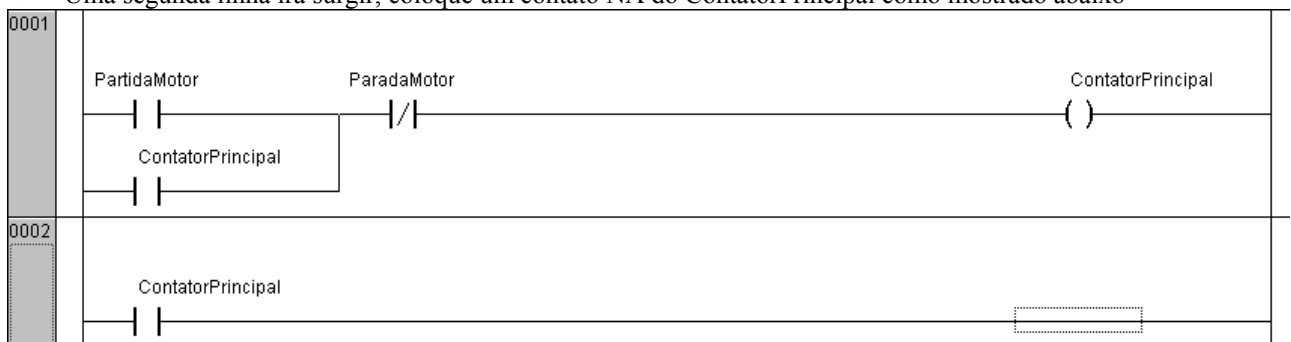


- Clicando novamente na linha de comando, selecione agora na barra de ferramentas a opção .

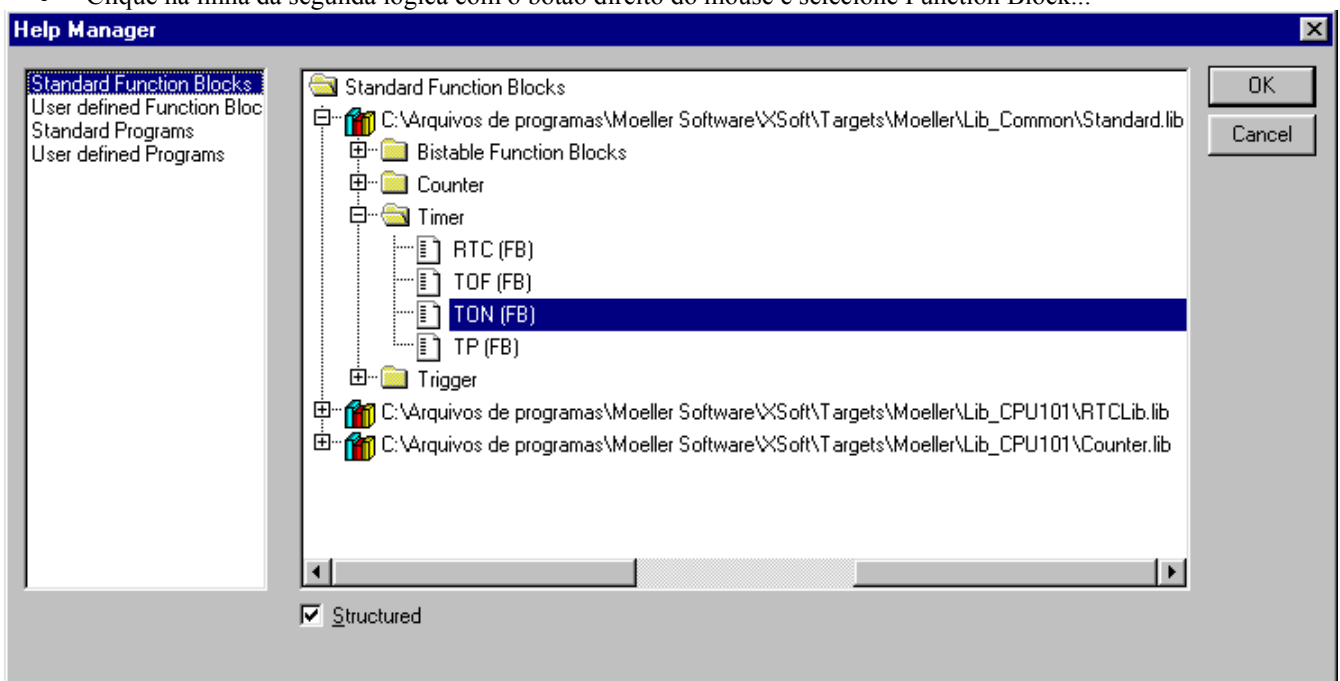
- Isto irá inserir uma bobina de saída nesta linha lógica. Utilize o mesmo procedimento para colocar o tag dela, no caso ContatorPrincipal.



- Com isto nós criamos a primeira linha lógica de nosso programa. Salve as alterações e agora vamos inserir a próxima linha.
- Vá no menu principal e selecione INSERT → Network (After).
- Uma segunda linha irá surgir, coloque um contato NA do ContatorPrincipal como mostrado abaixo

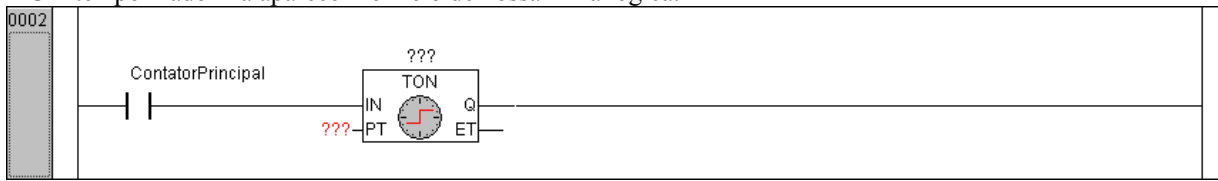


- Nós iremos criar agora o Controle dos Contadores Estrela e triângulo. Assim que o ContatorPrincipal entra, o ContarEstrela também liga, permanece alguns segundos e desliga.
- Iremos inserir então um temporizador para determinar o tempo que o contator estrela fica ligado.
- Clique na linha da segunda lógica com o botão direito do mouse e selecione Function Block...

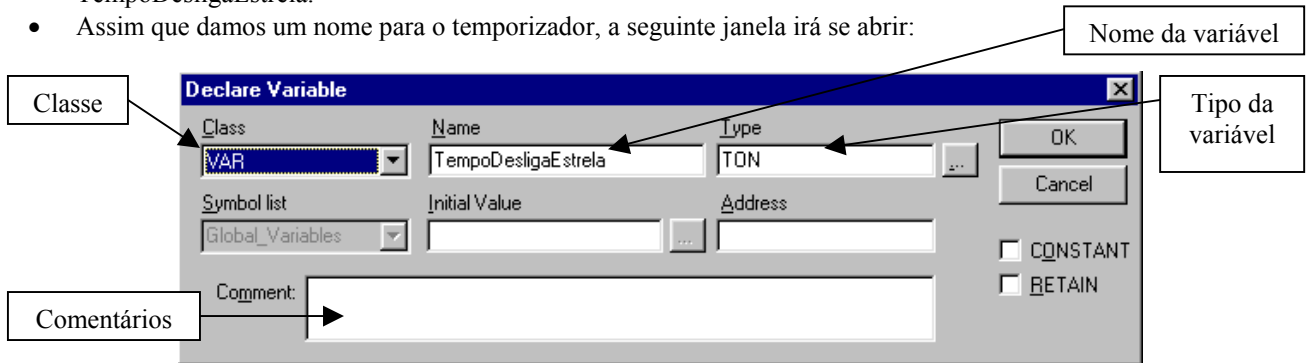


- Selecione na biblioteca Standard.lib o bloco de função TON, deixe a opção Structured ativa e selecione OK
- Veja em anexos → Bibliotecas como importar demais bibliotecas.

- Um temporizador irá aparecer no meio de nossa linha lógica.



- Vamos clicar nas interrogações acima do temporizador e colocar o nome do temporizador, vamos chama-lo de TempoDesligaEstrela.
- Assim que damos um nome para o temporizador, a seguinte janela irá se abrir:



- Isto ocorre pois o assistente de declaração verifica que não há nenhuma variável declarada como TempoDesligaEstrela, por causa disto, ele solicita que entremos com maiores dados sobre esta variável.
- Deixe esta caixa de diálogos como mostrada acima.
- O bloco de função irá aparecer na declaração local de variáveis.

```

0001 PROGRAM PLC_PRG
0002 VAR
0003     TempoDesligaEstrela: TON;
0004 END_VAR
0005
0006
    
```

Declaração de variáveis em FREE MODE

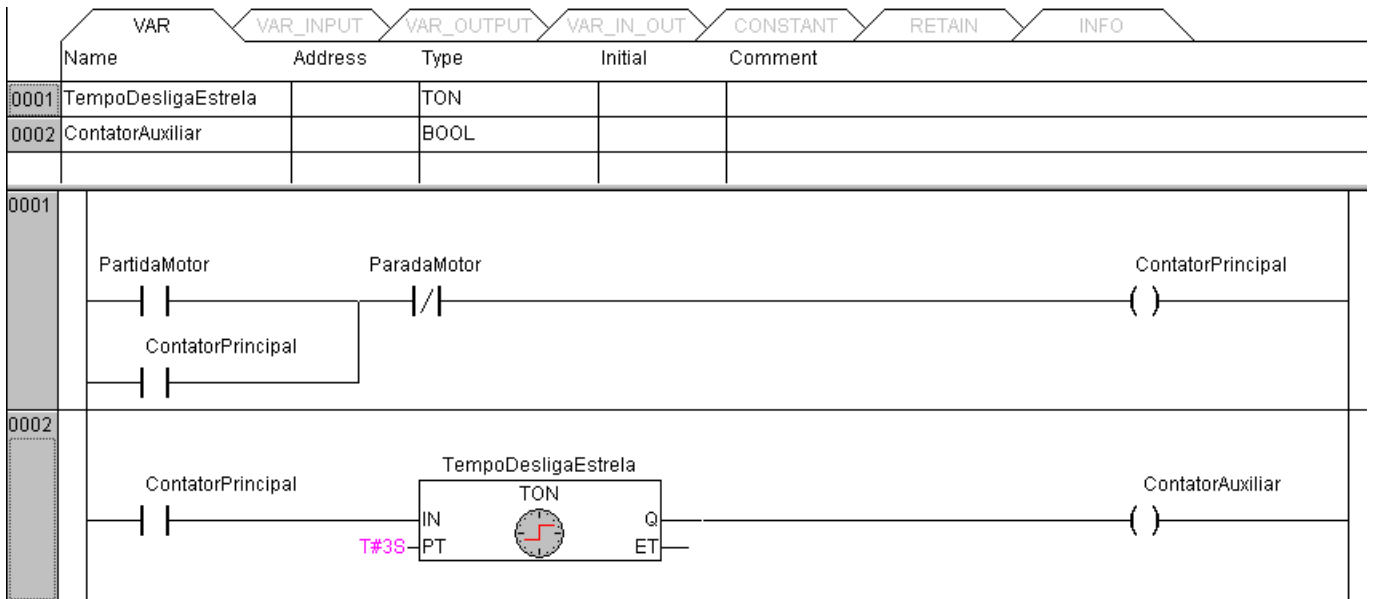
	VAR	VAR_INPUT	VAR_OUTPUT	VAR_IN_OUT	CONSTANT	RETAIN	INFO
	Name	Address	Type	Initial	Comment		
0001	TempoDesligaEstrela		TON				

Declaração como uma tabela

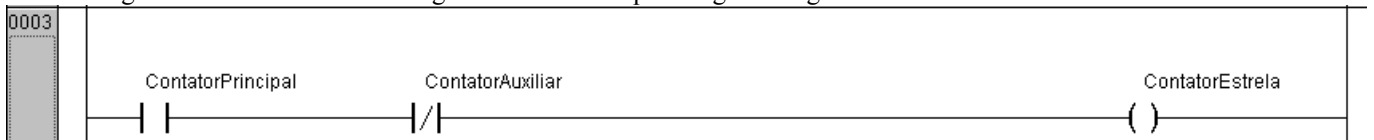
- A declaração pode aparecer destas duas formas diferentes, FREE MODE ou Tabelada. Para alternar entre estes modos de exibição, clique com o botão da direita sobre o campo da declaração de variáveis e selecione Declaration as tables
- Coloque uma bobina na linha lógica 2, logo após o temporizador e coloque o nome de ContadorAuxiliar.
- A mesma tela do assistente de declaração se abre, porém desta vez no tipo da variável selecione BOOL.
- A segunda linha do nosso programa está pronta, e a declaração de variáveis também.
- Na interrogação que surge do lado esquerdo do nosso temporizador escrevemos o tempo desejado. Por se tratar de uma entrada do tipo TIME, o tempo deve ser escrito da seguinte forma:

T#1D6H20M15S300MS Onde 1D significa o tempo de 1 dia
 6H tempo de 6Horas
 20M tempo de 20 minutos
 15S tempo de 15 segundos
 300MS tempo de 300Milésimos de segundos.

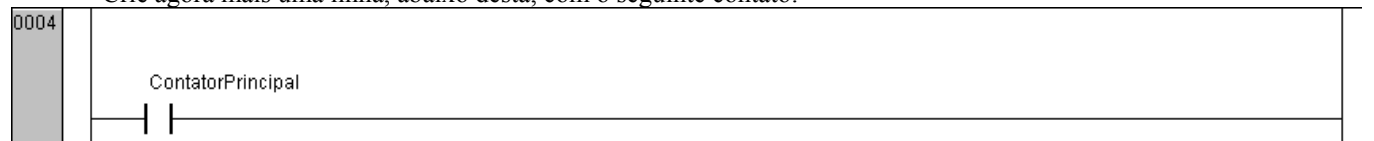
- Para criarmos um temporizador de 3 segundos então escrevemos T#3S.
- Nosso programa ficará deste jeito:



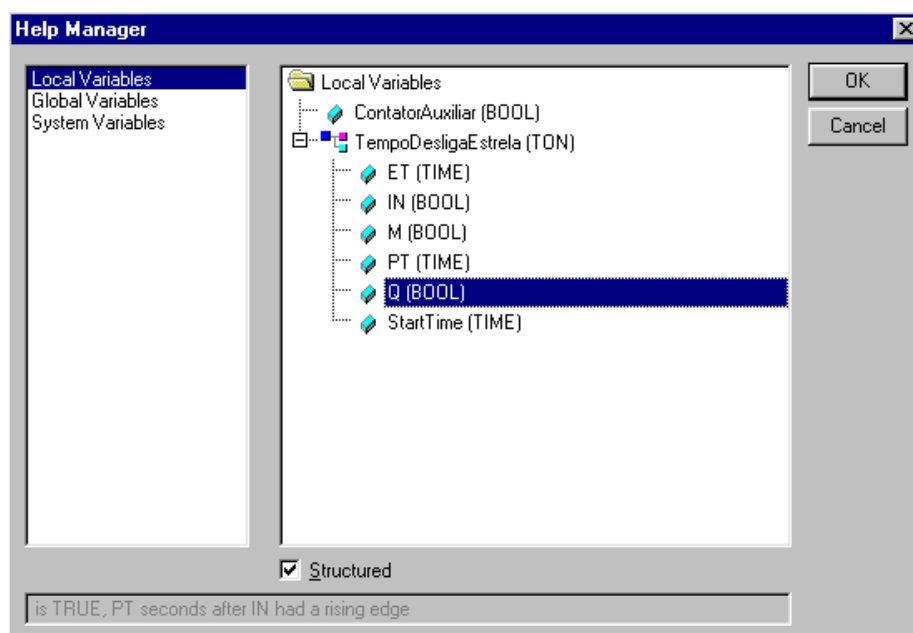
- Agora insira mais uma linha lógica abaixo e coloque a seguinte lógica:



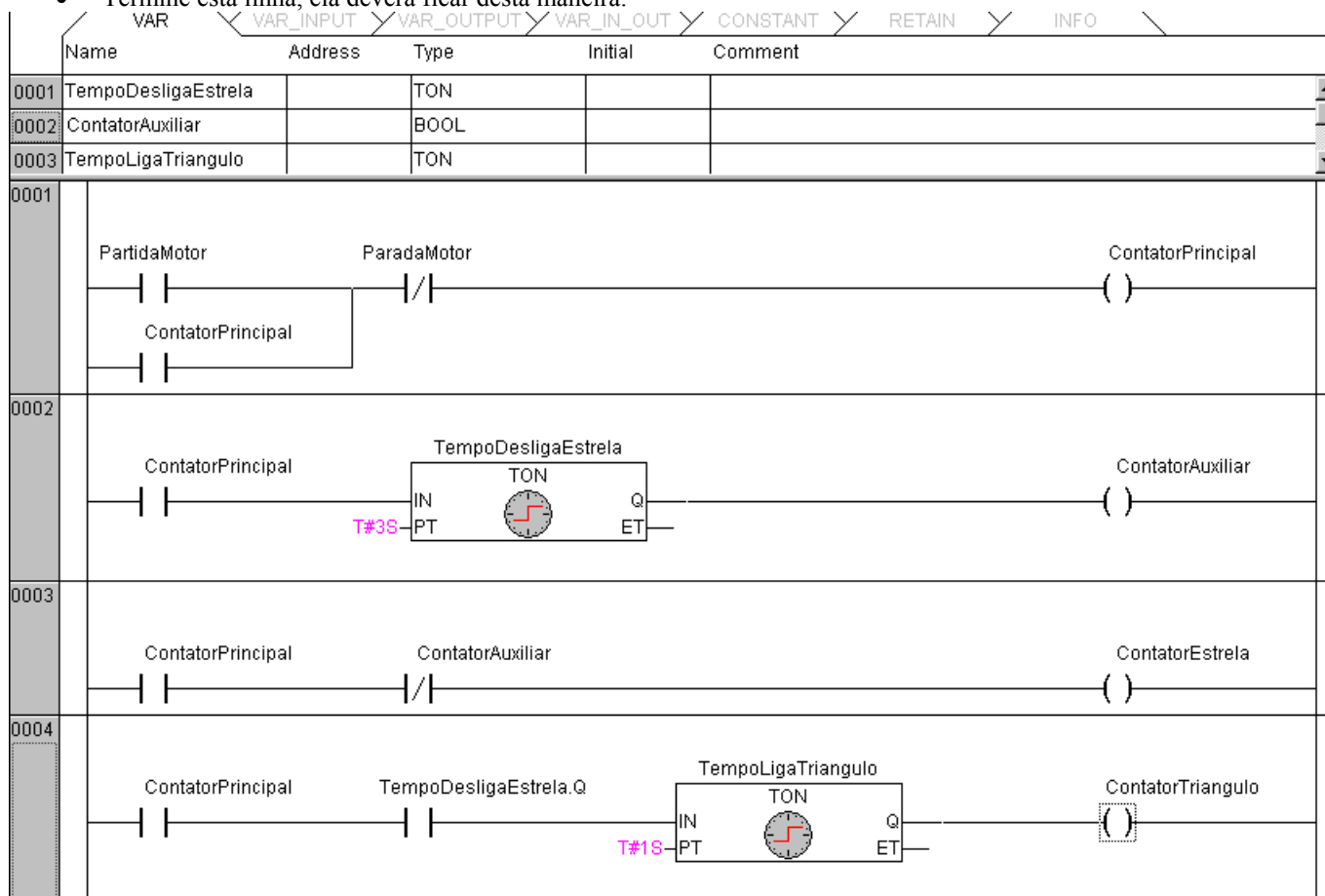
- Assim que o ContatorPrincipal ligar o ContatorEstrela ficará ligado até o tempo de 3 segundos se esgotar.
- Crie agora mais uma linha, abaixo desta, com o seguinte contato:



- Coloque agora um contato NA e pressione F2.
- Assim que a tela Help Manager aparecer selecione Local Variables → TempoDesligaEstrela → Q. Como mostrado abaixo:



- Termine esta linha, ela deverá ficar desta maneira:



- Na quarta linha lógica, no segundo contato da esquerda para direita, nós usamos o TAG TempoDesligaEstrela.Q, poderíamos ter utilizado o próprio ContatorAuxiliar sem problemas, fizemos isto exatamente para demonstrar uma funcionalidade a mais do programa.
- Salve o programa, estamos prontos para a próxima etapa.

CAPÍTULO 5

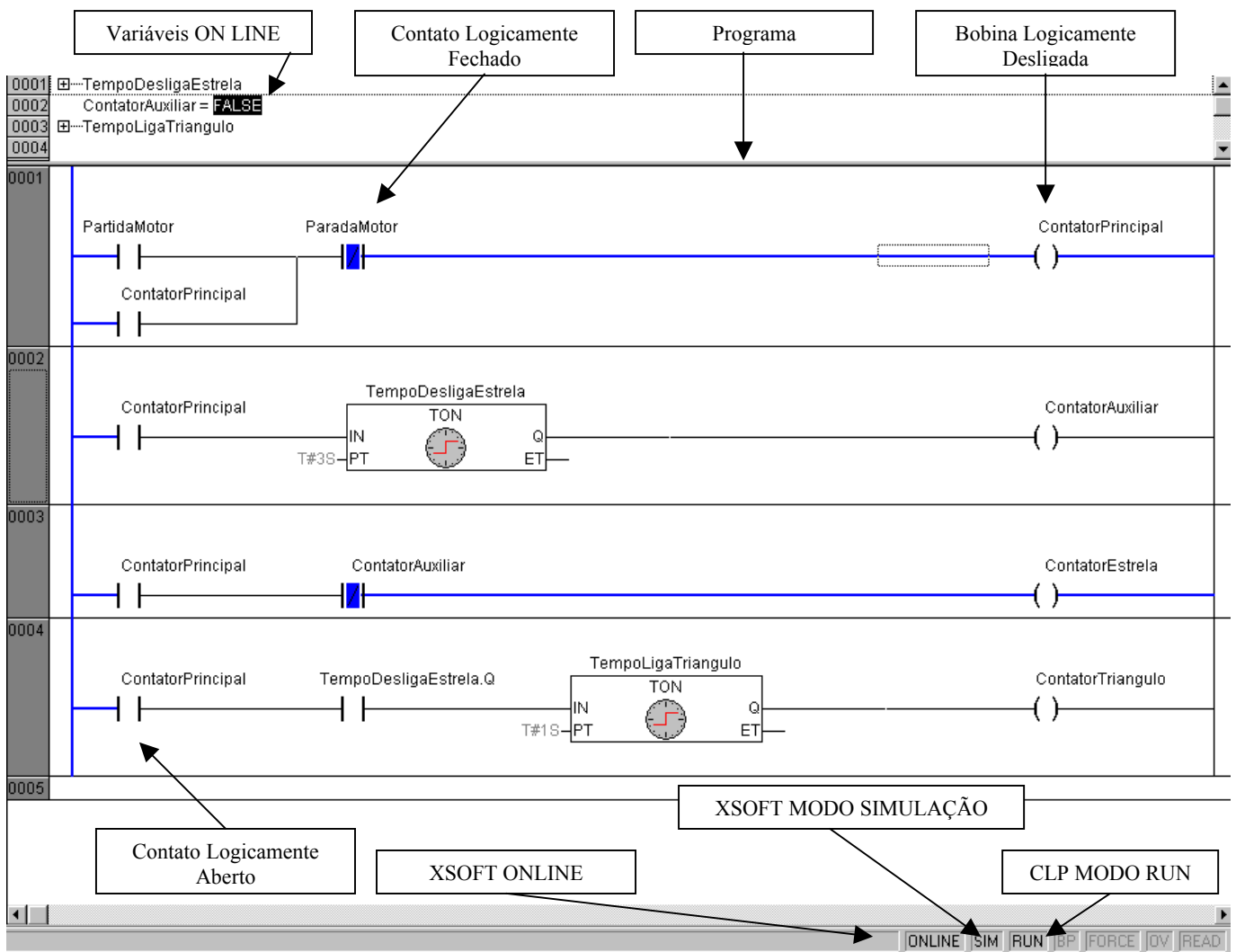
COMPILANDO O PROGRAMA

- Para compilar o programa, entre na opção Project → Rebuild All...
- Isto irá compilar todos o seu projeto. Uma tela irá aparecer abaixo de seu programa mostrando possíveis erros ocorridos no seu programa. Para localizar o erro, clique duas vezes em cima dele e você irá para a linha onde este erro está.
- Para maiores informações sobre os erros, por favor consulte o manual do XSOFT, ele contém um apêndice com os principais erros.

CAPÍTULO 6

SIMULANDO O PROGRAMA

- Para utilizar as ferramentas de simulação você deve ter o programa compilado como se você fosse transferi-lo para o CLP.
- Habilite no menu principal a opção ON LINE → Simulation Mode...isto irá colocar o programa em modo de simulação.
- Agora selecione a opção ON LINE → Login... isto fará com que o software faça um Login com um CLP virtual.
- Este CLP virtual inicializa em STOP, vá novamente para ON LINE → Run e estaremos com este CLP virtual em modo RUN. A tela que aparecerá é a seguinte:



- Para forçar entradas, saídas ou memórias internas podemos proceder de várias maneiras diferentes.
- Dê um duplo clique com o botão da esquerda do mouse sobre o tag do contato PartidaMotor, verifique que a cada duplo clique ele muda o desenho:

PartidaMotor Contato NA logicamente aberto



PartidaMotor Tag na cor preto, com um quadrado na parte inferior com as bordas azul claro preenchimento azul escuro. Contato NA logicamente aberto, forçar ON quando do comando FORCE



PartidaMotor Tag na cor preto, com um quadrado na parte inferior com as bordas azul claro sem preenchimento. Contato NA logicamente aberto, forçar OFF quando do comando FORCE.



- Se dermos mais um duplo clique o contato volta ao estado anterior.
- Dê um duplo clique no Tag e deixe-o na posição forçar ON.
- Agora pressione F7, ou vá até o menu Online → Force Values. O nosso botão de partida será forçado para a situação ON.

PartidaMotor Observe que o TAG PartidaMotor aparece em vermelho. Ao redor da quadrado azul também aparece uma linha vermelha para indicar que esta variável está sendo forçada. Contato NA logicamente fechado, forçado.



- Repare que a indicação FORCE aparece no canto esquerdo da tela, para indicar que existem variáveis forçadas em seu programa. **ONLINE SIM RUN BP FORCE OV READ**
- Se quisermos forçar várias variáveis ao mesmo tempo, podemos ir clicando com o mouse e deixando-as nos valores lógicos desejados, assim que pressionamos F7 os valores são carregados para os Tags.
- Verifique o funcionamento do seu programa, ele deve ligar o contator principal juntamente com o estrela, contar um tempo de 3 segundos, desligar o contator estrela, esperar mais um segundo e ligar o contator triângulo.
- Vamos agora retirar o comando de partida, clique novamente com o botão esquerdo do mouse sobre o TAG PartidaMotor e verifique como ele muda o status:

PartidaMotor Contato NA logicamente fechado, forçado.



PartidaMotor Tag em vermelho, aparece um quadrado na parte inferior do contato com bordas azul claro e preenchimento cinza. Contato NA logicamente fechado e forçado. Liberar comando FORCE quando ocorrer novo comando FORCE.



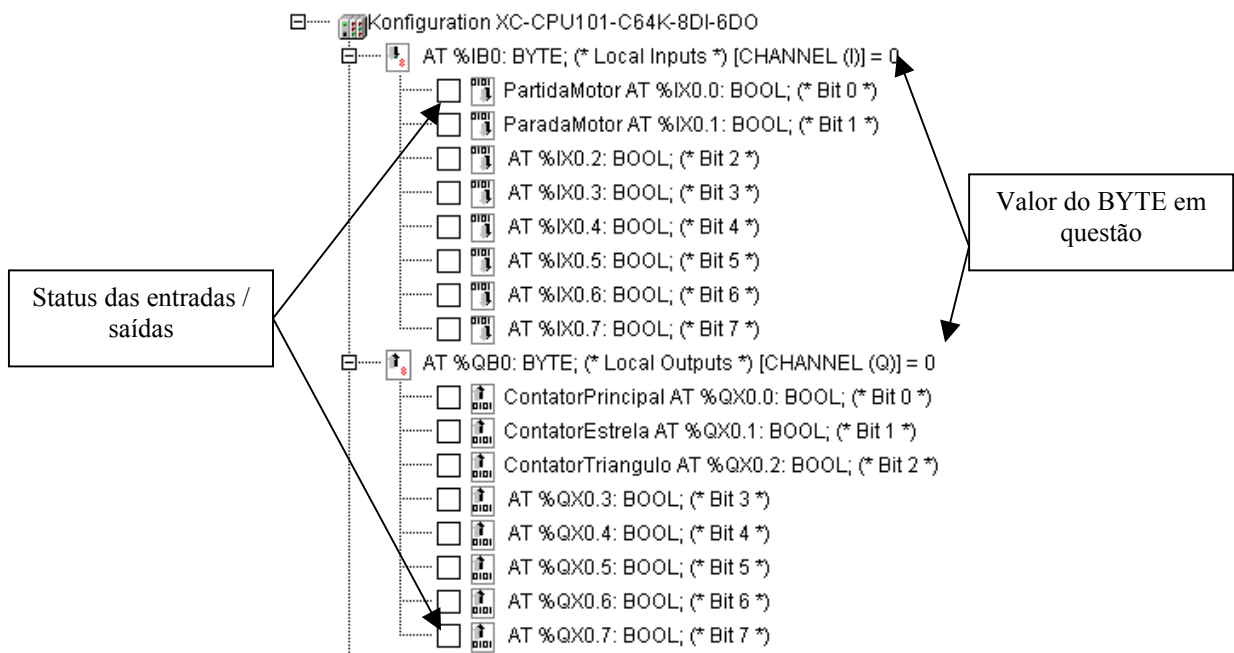
PartidaMotor Tag em vermelho, aparece um quadrado na parte inferior do contato com bordas azul claro sem preenchimento. Contato NA logicamente fechado e forçado. Forçar estado OFF quando ocorrer novo comando FORCE.



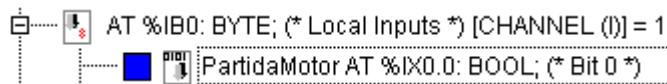
- Se estivéssemos ONLine com o CLP, bastaria colocarmos o comando Liberar comando Force, que o TAG assumiria o valor da entrada digital logo em seguida.
- Como estamos com um CLP virtual, que não possui entradas físicas, para desligarmos adequadamente este contato precisamos forçá-lo para a condição OFF e logo em seguida liberar o comando force.
- Experimente agora forçar o botão de desliga. Force-o para a condição ON (Aberto pois ele é um NF), logo em seguida para a condição OFF e então libere o FORCE.
- Estas funções estão disponíveis tanto no modo Simulação, como conectado com o CLP.

FORÇANDO VARIÁVEIS PELA CONFIGURAÇÃO

- Outra forma de realizar esta operação é forçar os elementos de entrada e saída.
- No Organizador vá em Resource → PLC Configuration



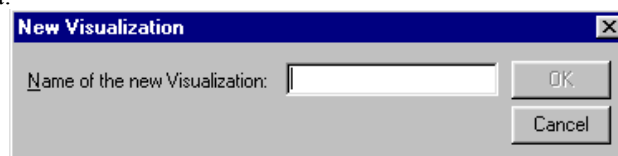
- Para forçar a entrada PartidaMotor basta clicar uma vez com o botão esquerdo do mouse no quadrado que se encontra à esquerda do TAG.



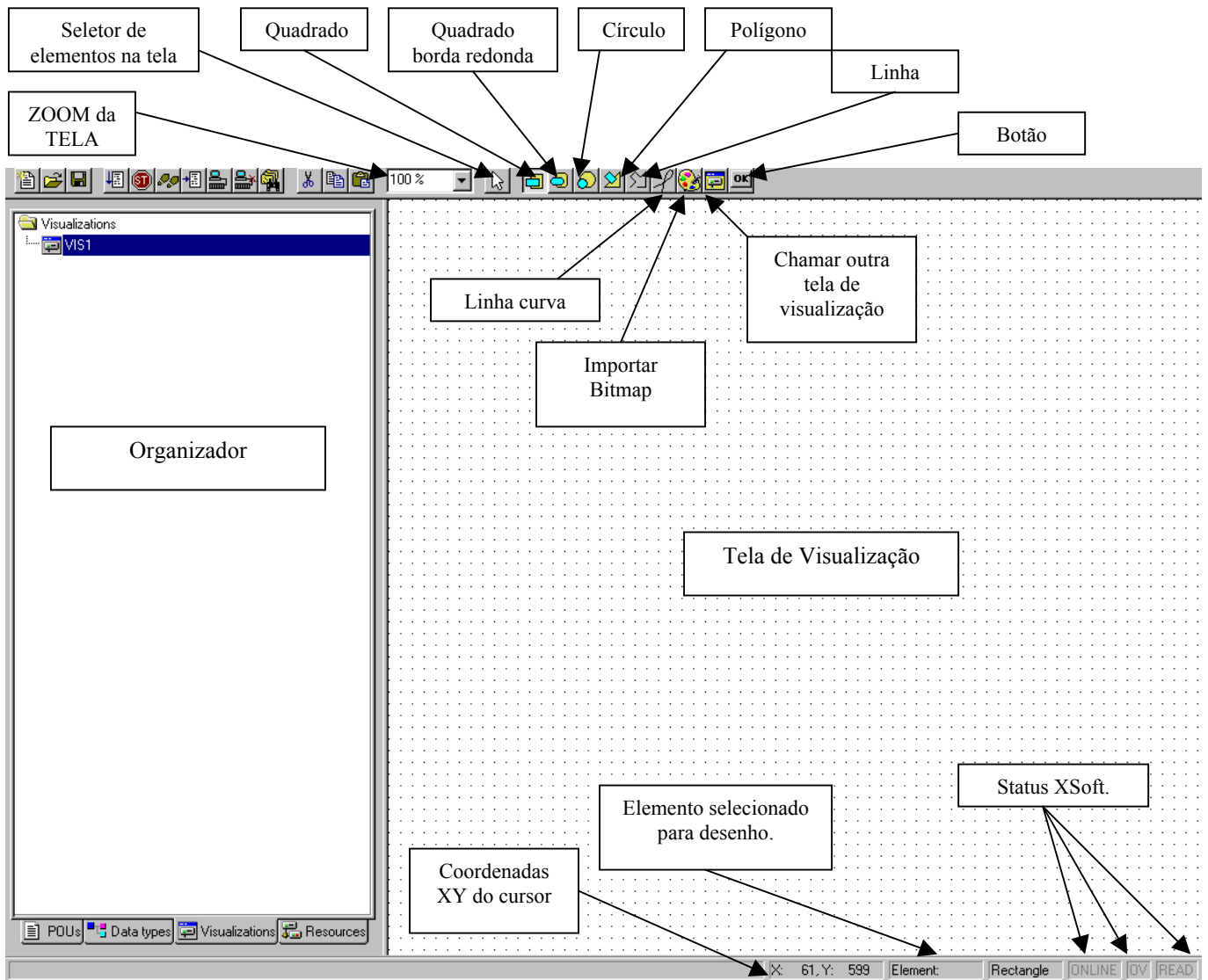
- Para desliga-lo é só clicar novamente neste quadrado.
- Acione agora o TAG ParadaMotor.


CRIANDO TELAS DE SIMULAÇÃO

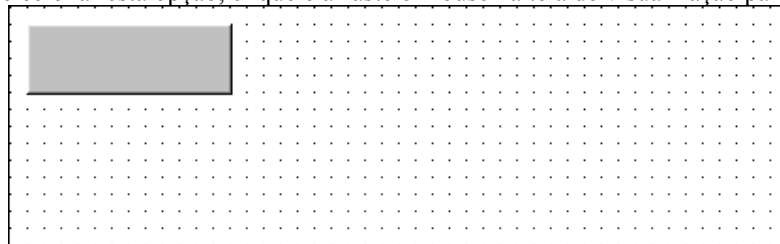
- Para criar telas de simulação é necessário estar com o simulador parado, por isso vá até a opção menu Online → Logout.
- Vá agora até o organizador e selecione a opção Visualizations.
- Clique com o botão da direita do mouse sobre a pasta Visualizations e selecione a opção ADD Object...
- Isto abrirá a seguinte tela:



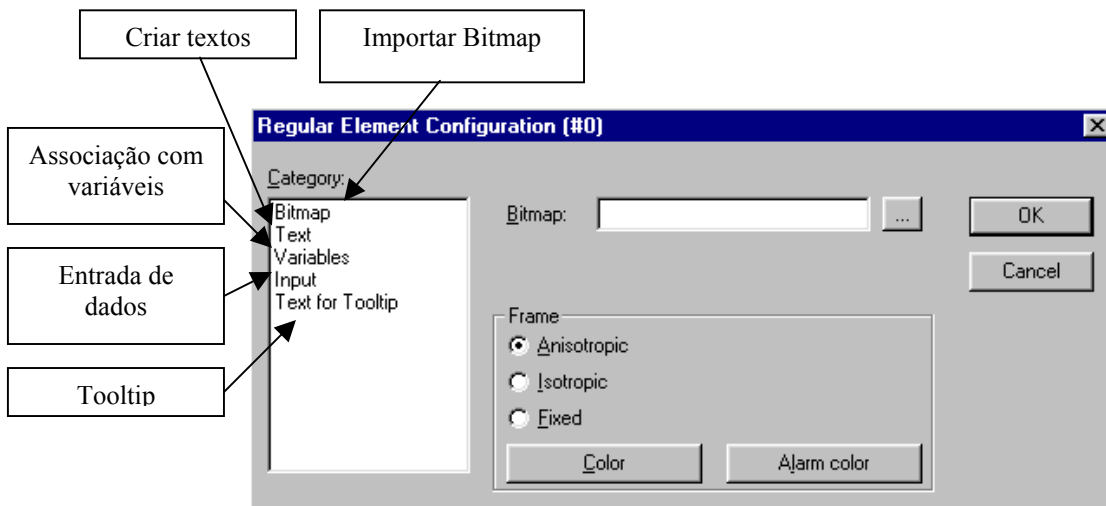
- Dê um nome para este Objeto de Visualização, por exemplo VIS1.



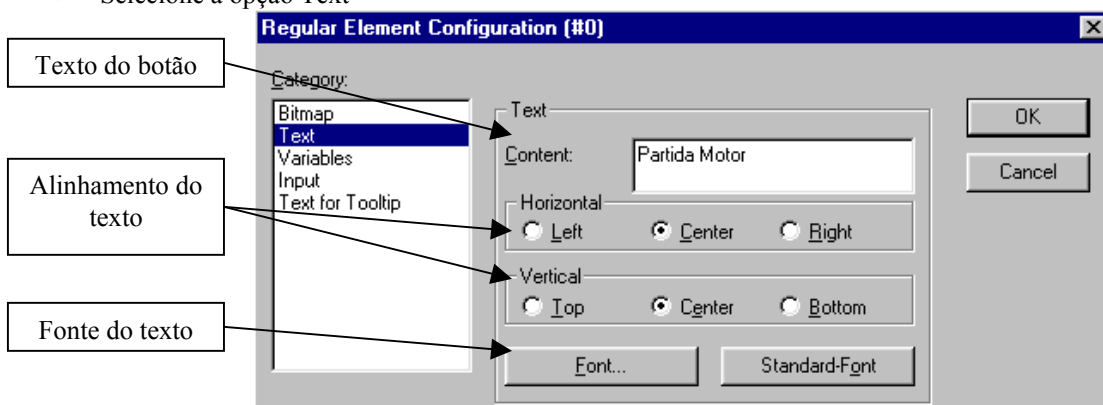
- Primeiramente vamos criar um botão de Partida. Clique no ícone  para criar um botão na tela de visualização.
- Assim que você selecionar esta opção, clique e arraste o mouse na tela de visualização para criar o botão



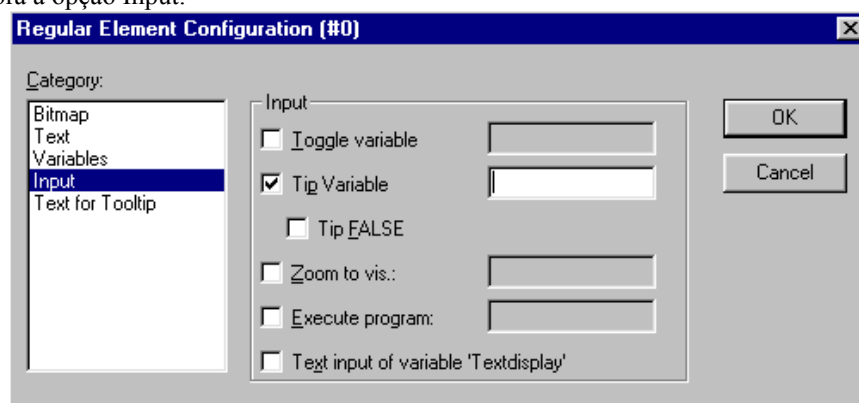
- Dê um duplo clique neste botão para editá-lo



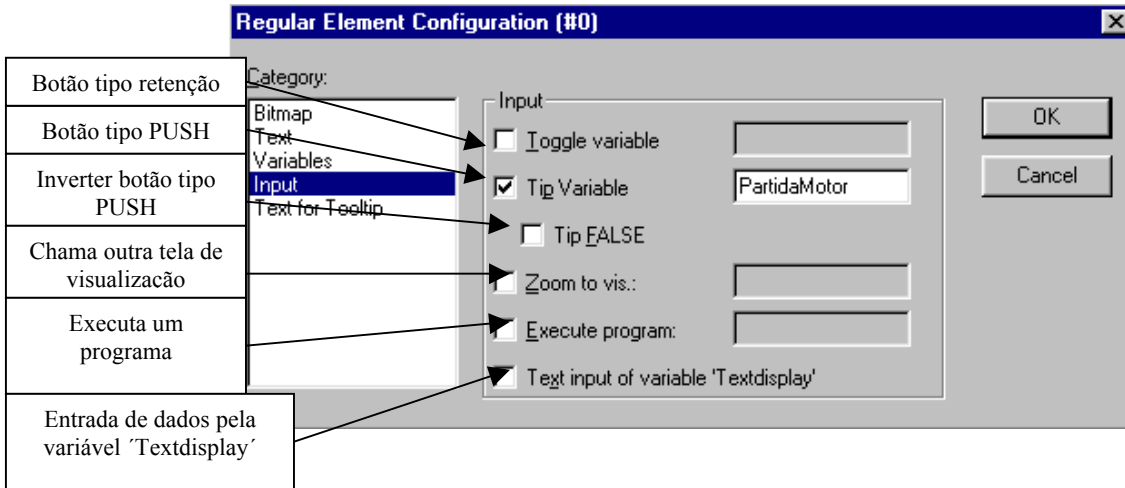
- Para cada item selecionado, teremos uma caixa de diálogo diferente.
- Selecione a opção Text



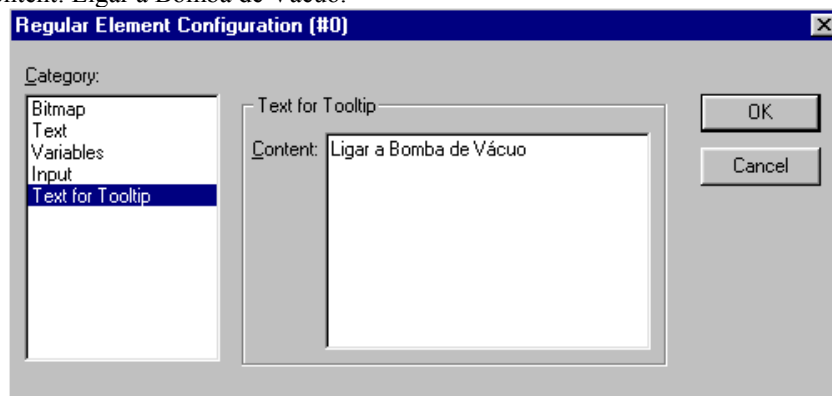
- Escreva no texto do botão Partida Motor. Mantenha o alinhamento central e escolha uma fonte de texto qualquer.
- Selecione agora a opção Input.



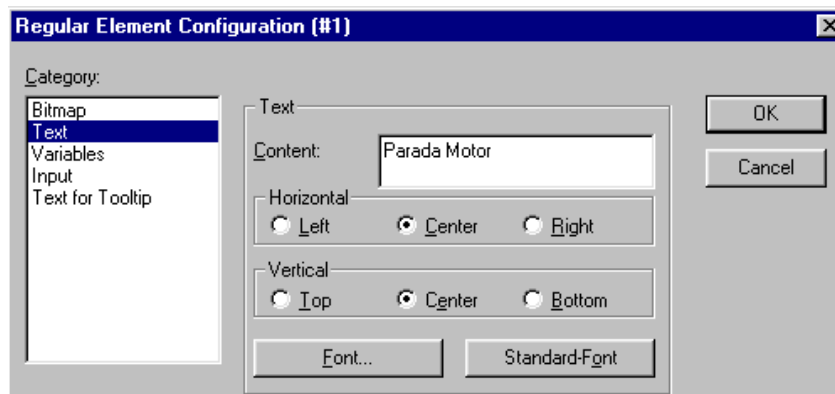
- Selecione a opção Tip Variable. Clique com o mouse dentro do quadro branco e pressione F2.
- Assim que a tela help manager abrir selecione em Global_Variables → PartidaMotor (BOOL)

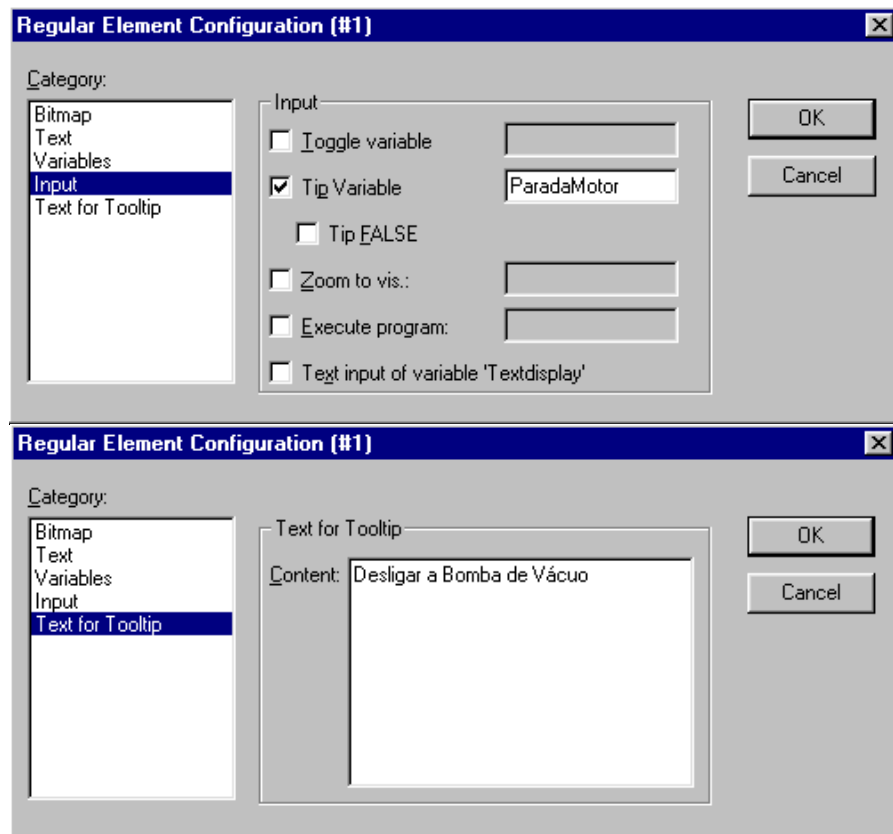



- Selecione agora Text for Tooltip
- Escreva em Content: Ligar a Bomba de Vácuo.

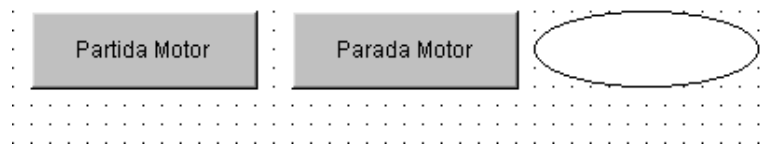


- Pressione OK e o botão de partida estará pronto. Vamos agora criar o botão de parada, basta criar um outro botão do lado deste e configura-lo da seguinte maneira.

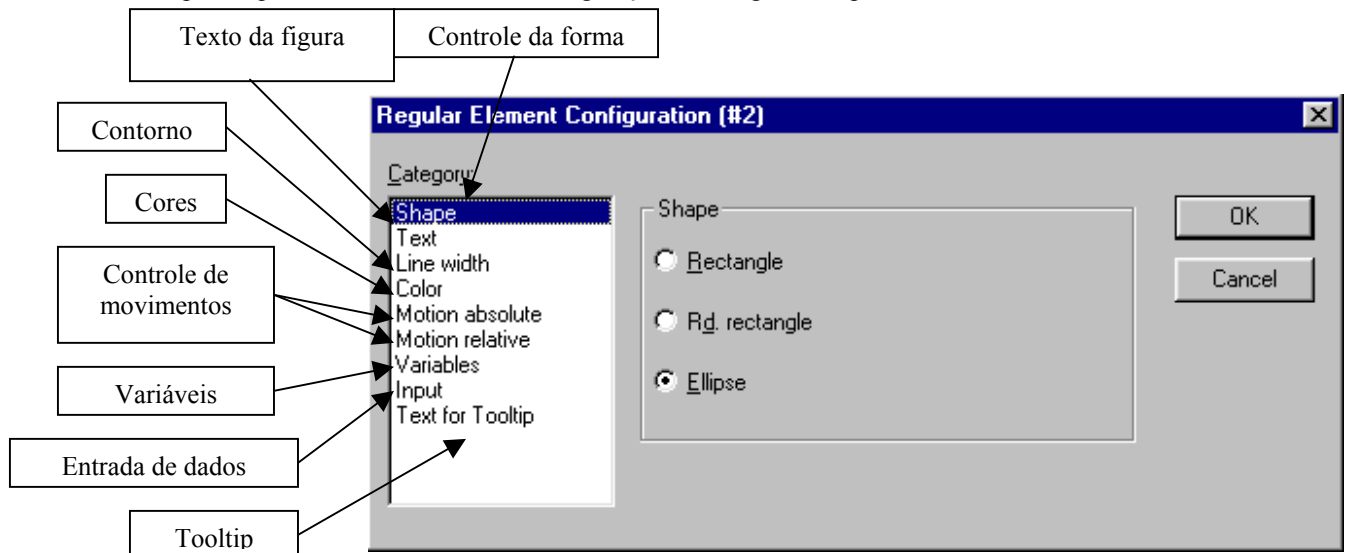




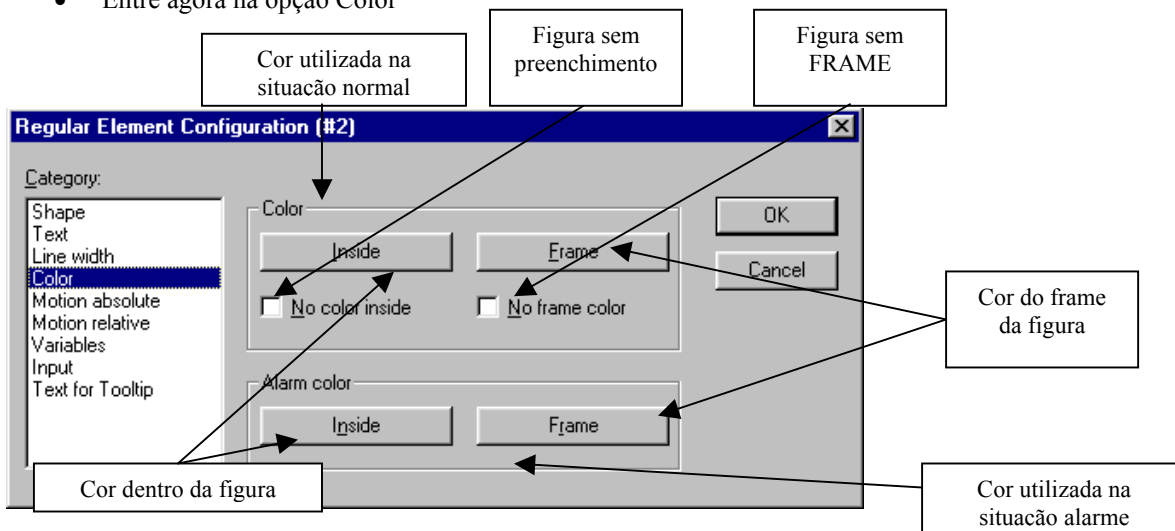
- Vamos criar agora os sinaleiros do nosso motor. Crie um círculo ao lado dos botões () conforme abaixo



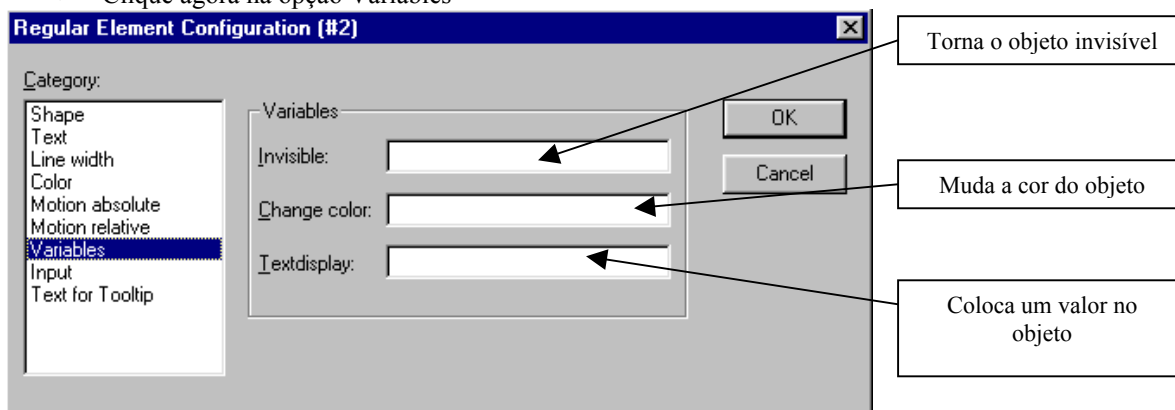
- Dê um duplo clique no círculo e a tela de configuração desta figura irá aparecer.



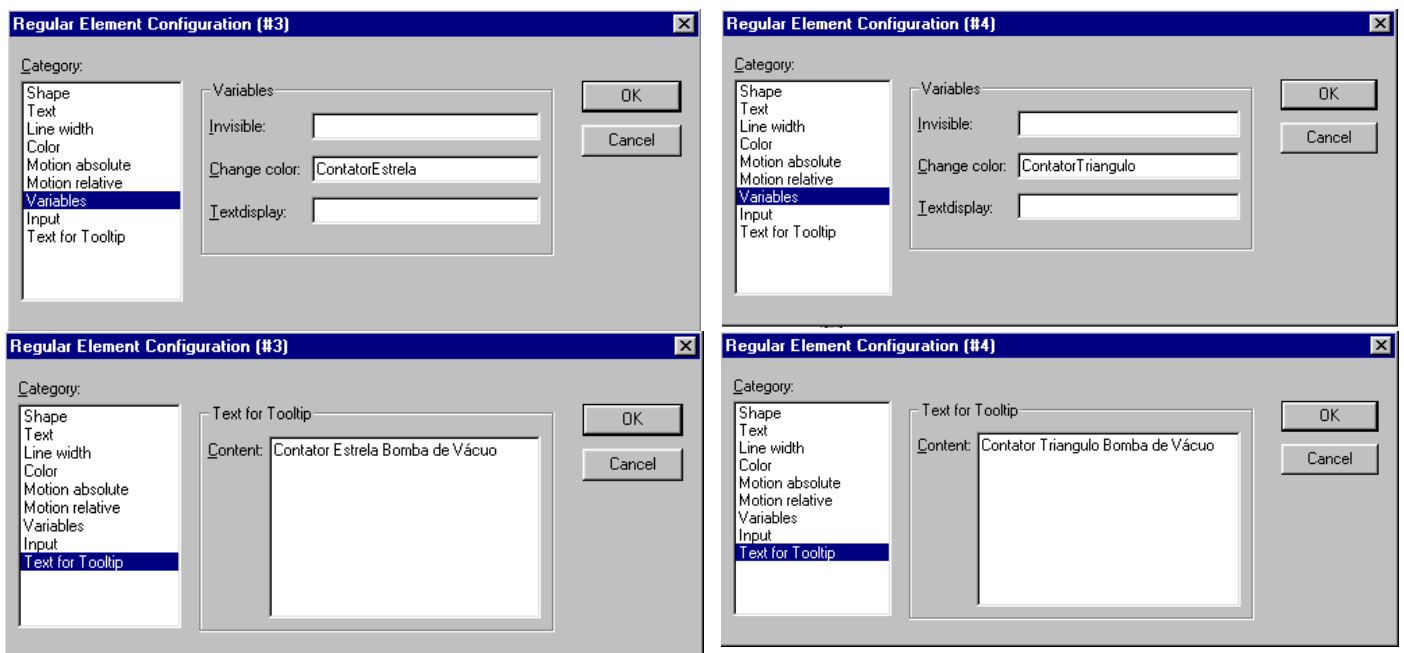
- Na opção Text → Content escreva o seguinte: Bomba de Vácuo
- Entre agora na opção Color



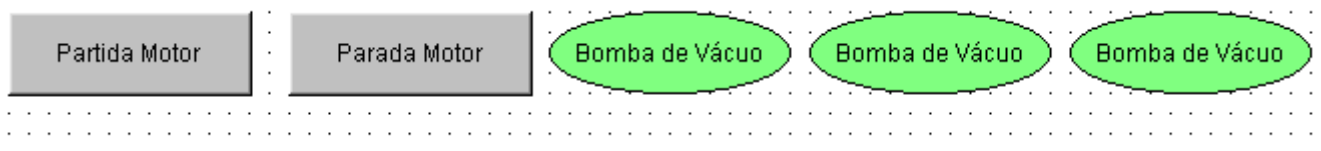
- Nesta opção podemos controlar as cores de nosso objeto.
- Em Color escolhemos a cor natural de nosso objeto e em Alarm Color escolhemos uma cor a ser utilizada quando o bit de alarme , que veremos mais adiante, esteja em nível 1.
- Clique em color → Inside e escolha a cor verde.
- Selecione agora a opção Alarm color → Inside e escolha a cor vermelha.
- Clique agora na opção Variables




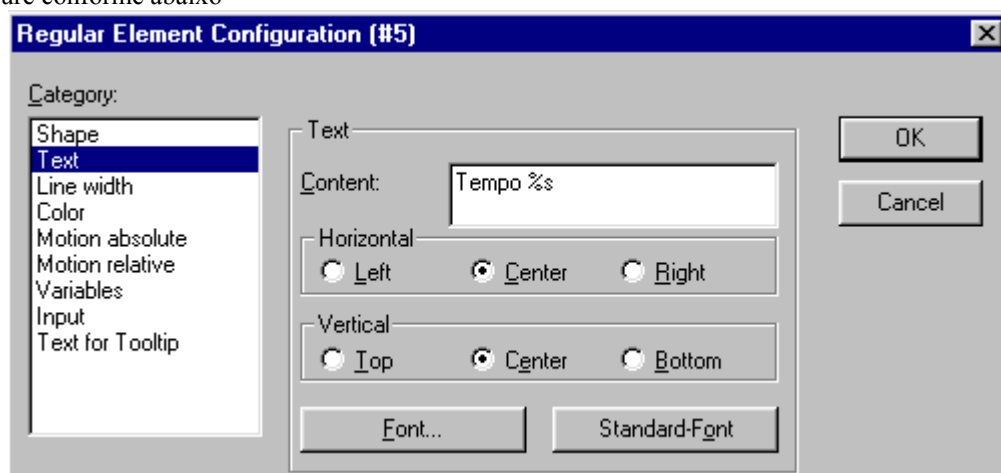
- Pressione a tecla F2 no campo Change color.
- No Help Manager selecione em Global_Variables → ContatorPrincipal.
- Selecione agora a opção Text for Tooltip e escreva o seguinte: Contator Principal Bomba de Vácuo.
- Crie o mesmo procedimento para os outros dois contatores como abaixo. Utilize a opção CTRL+C (Copiar) CTRL+V (Colar) para que os círculos saiam iguais.



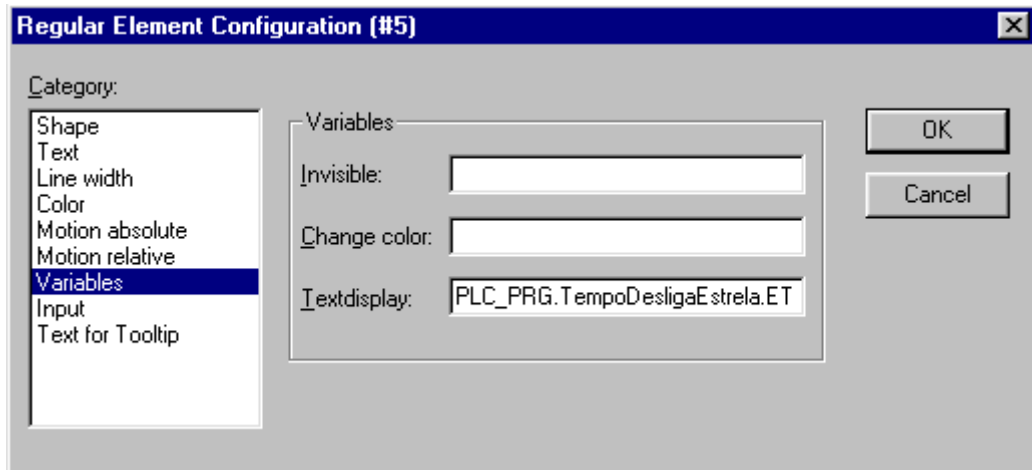
- Nosso display irá ficar desta forma:



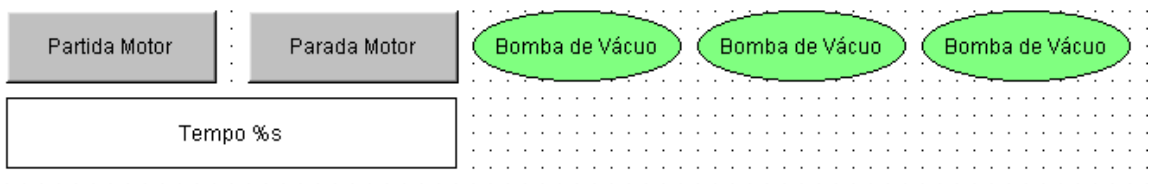
- Vamos gerar um mostrador de tempo. Faça um quadrado () logo abaixo dos botões de partida. A tela de configuração é idêntica à do círculo.
- Configure conforme abaixo



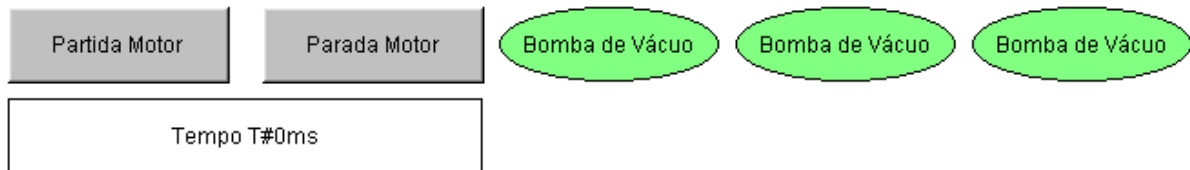
- O XSOFTE substitui a expressão %s pelo valor constante em Textdisplay. (Veja abaixo)
- Selecione agora a opção Variables.
- Coloque o cursor dentro do quadrado branco em frente a Textdisplay.
- Pressione F2 e selecione PLC_PRG → TempoDesligaEstrela → ET (TIME)



- Nossa tela de visualização está pronta:

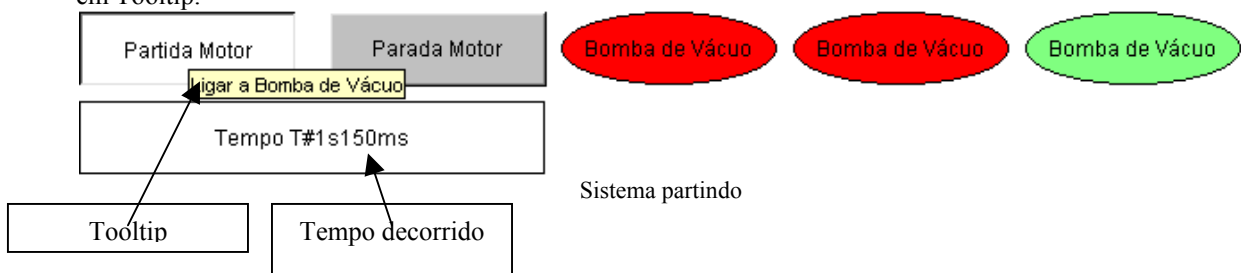


- Vamos colocar o programa mais uma vez em funcionamento
- Certifique-se que o XSOFT esteja em modo de simulação (Online → Simulation Mode...)
- Clique no menu Online → Login e logo em seguida Online → RUN.



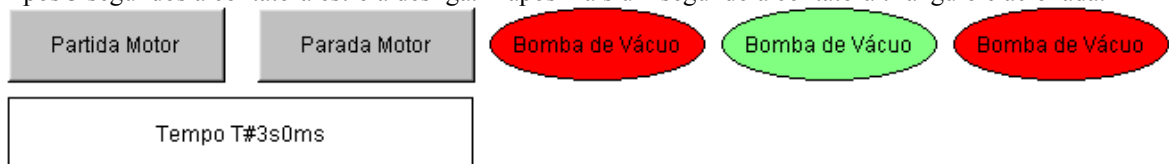
Sistema parado

- Clique no botão Partida Motor. Observe que se você segurar o mouse sobre um objeto ele irá exibir o texto digitado em Tooltip.



Sistema partindo

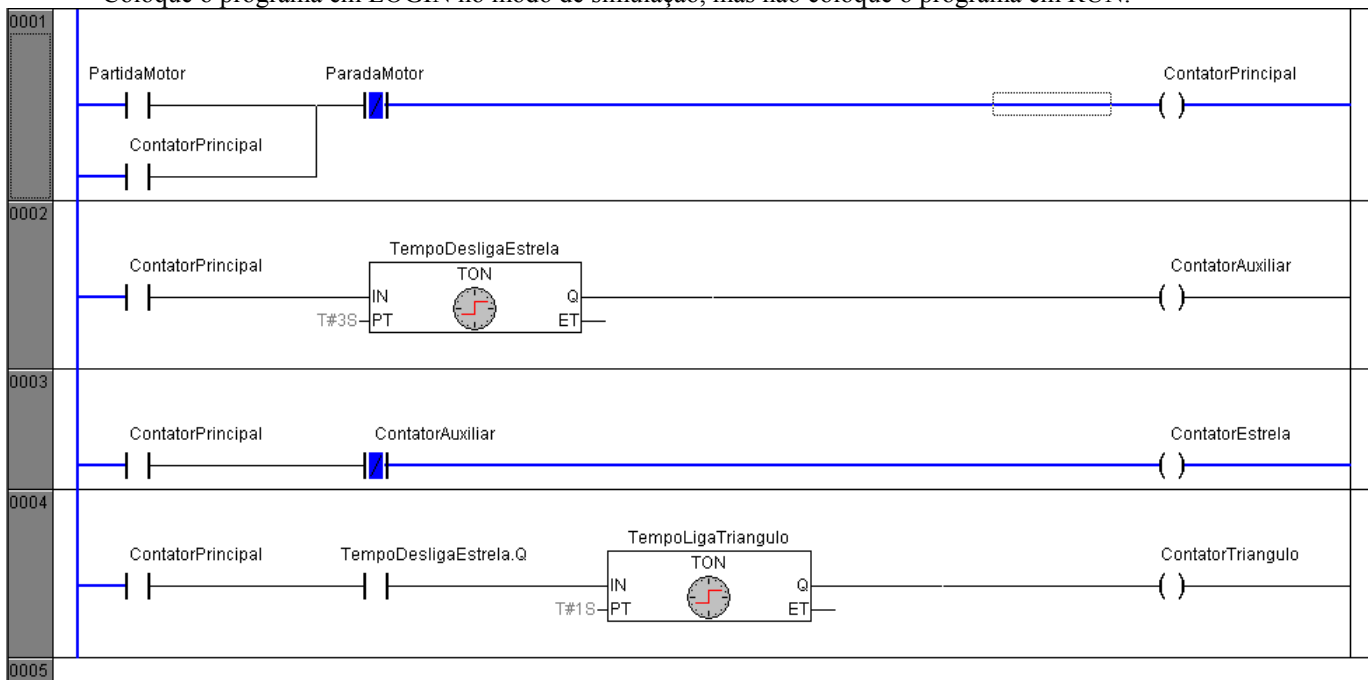
- Após 3 segundos a contadora estrela desliga. E após mais um segundo a contadora triângulo é acionada.



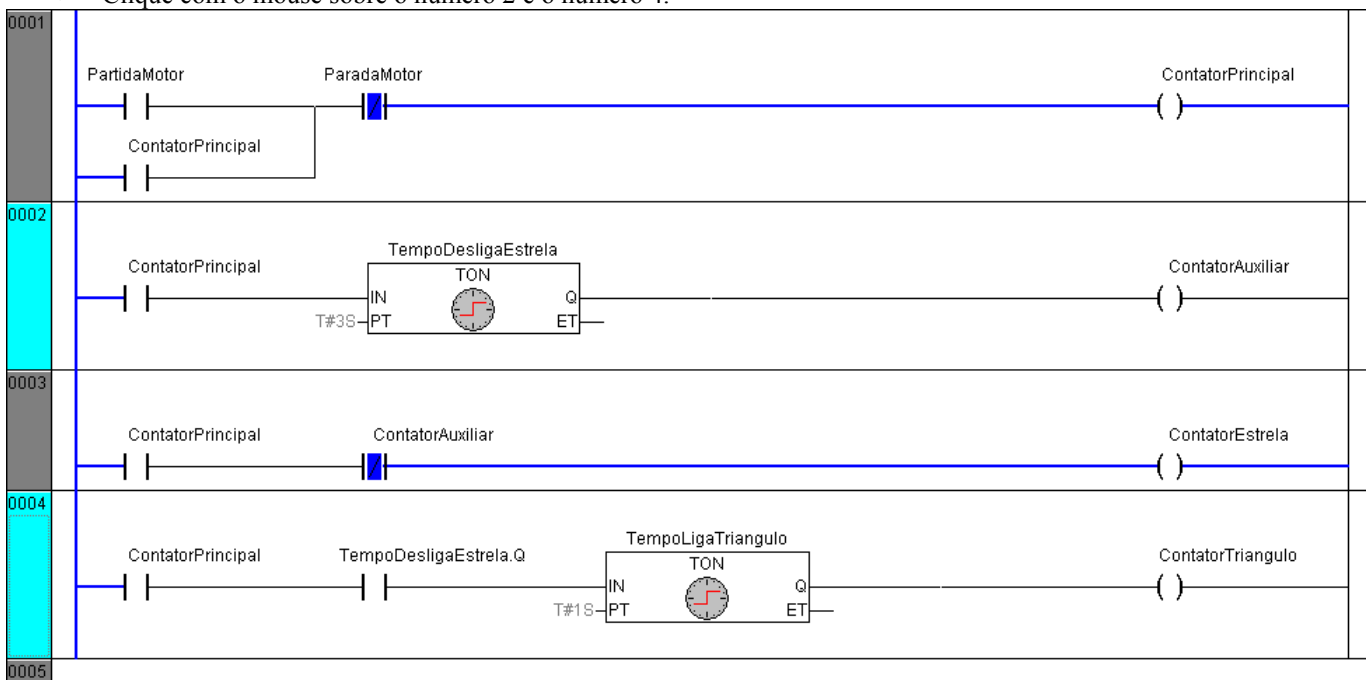
Motor em regime normal.

TRABALHANDO COM BREAK POINTS E CONTROLADORES DE CICLO

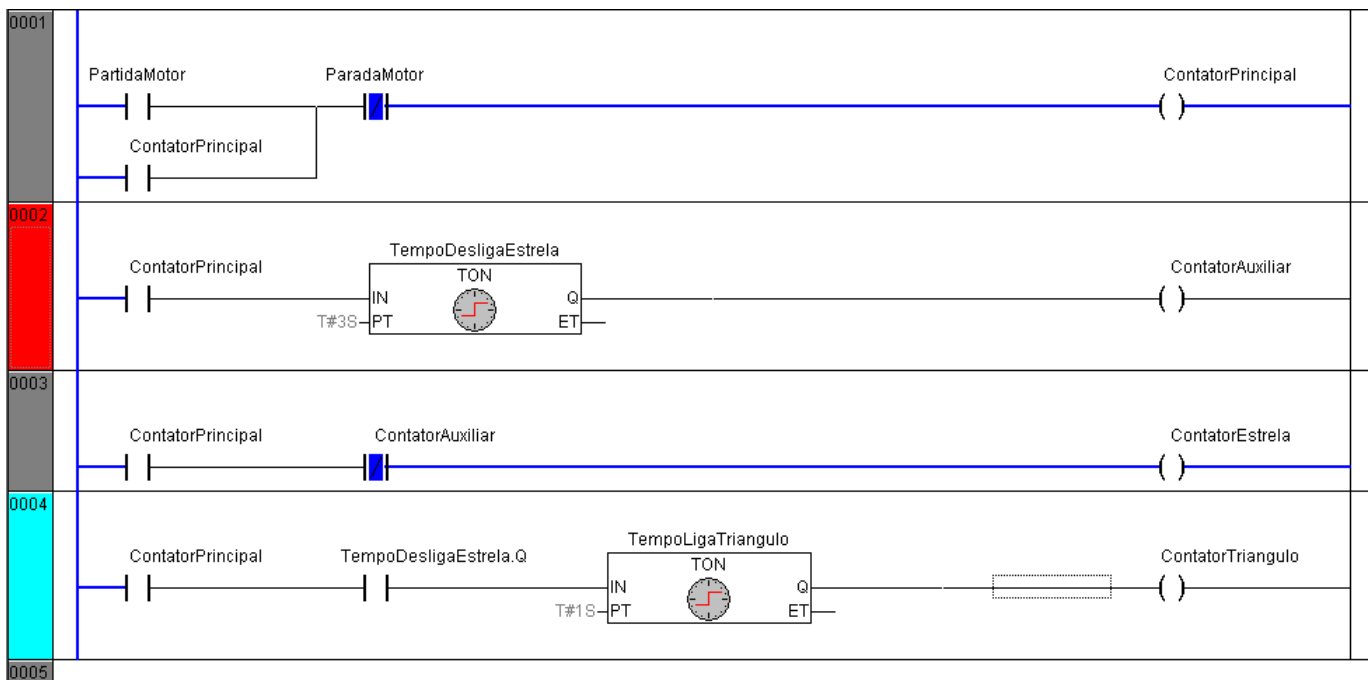
- Break Points são paradas criadas em nosso programa para nos auxiliar nas etapas de testes.
- Coloque o programa em LOGIN no modo de simulação, mas não coloque o programa em RUN.



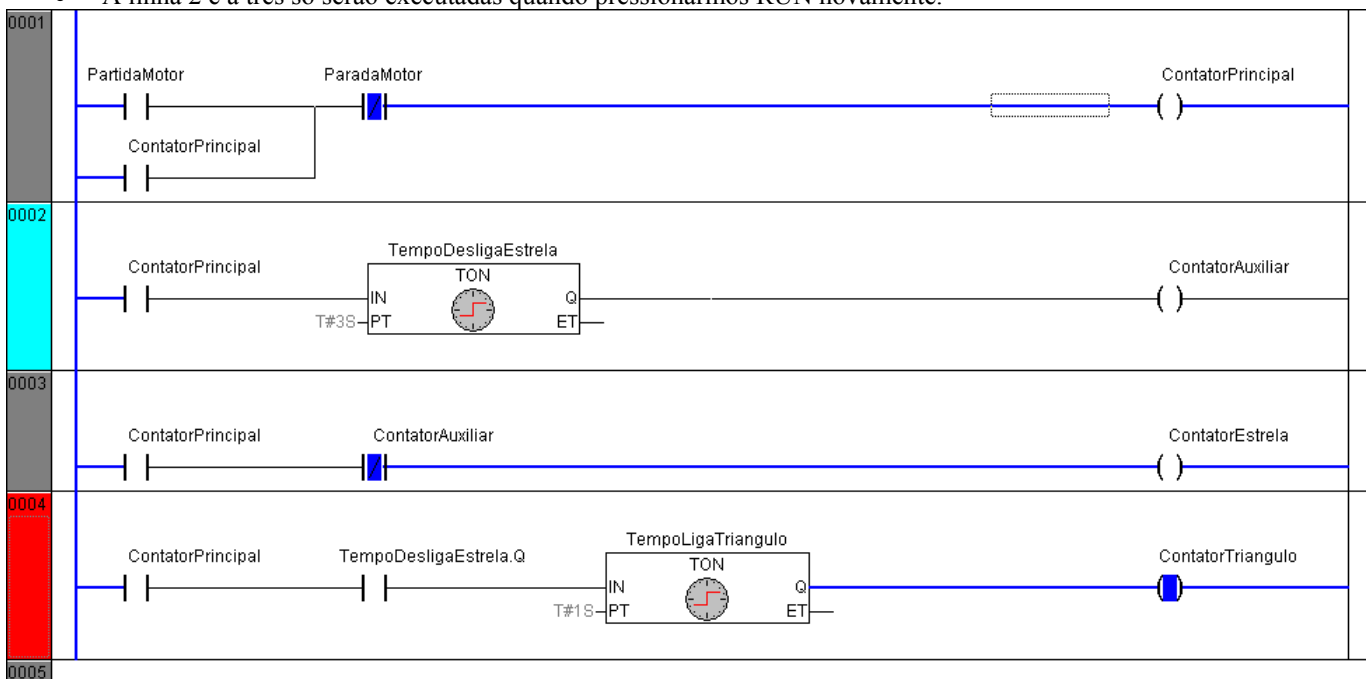
- Do lado esquerdo podemos ver a numeração das linhas lógicas.
- Clique com o mouse sobre o número 2 e o número 4.



- Acabamos de criar nas lógicas 2 e 4 break points. A cor azul claro indica onde estão os break points.
- Quando selecionarmos RUN, o programa irá rodar até chegar em um break point, então ele irá ficar parado neste ponto até darmos o comando RUN novamente.
- Vá no menu Online → RUN, ou pressione F5.

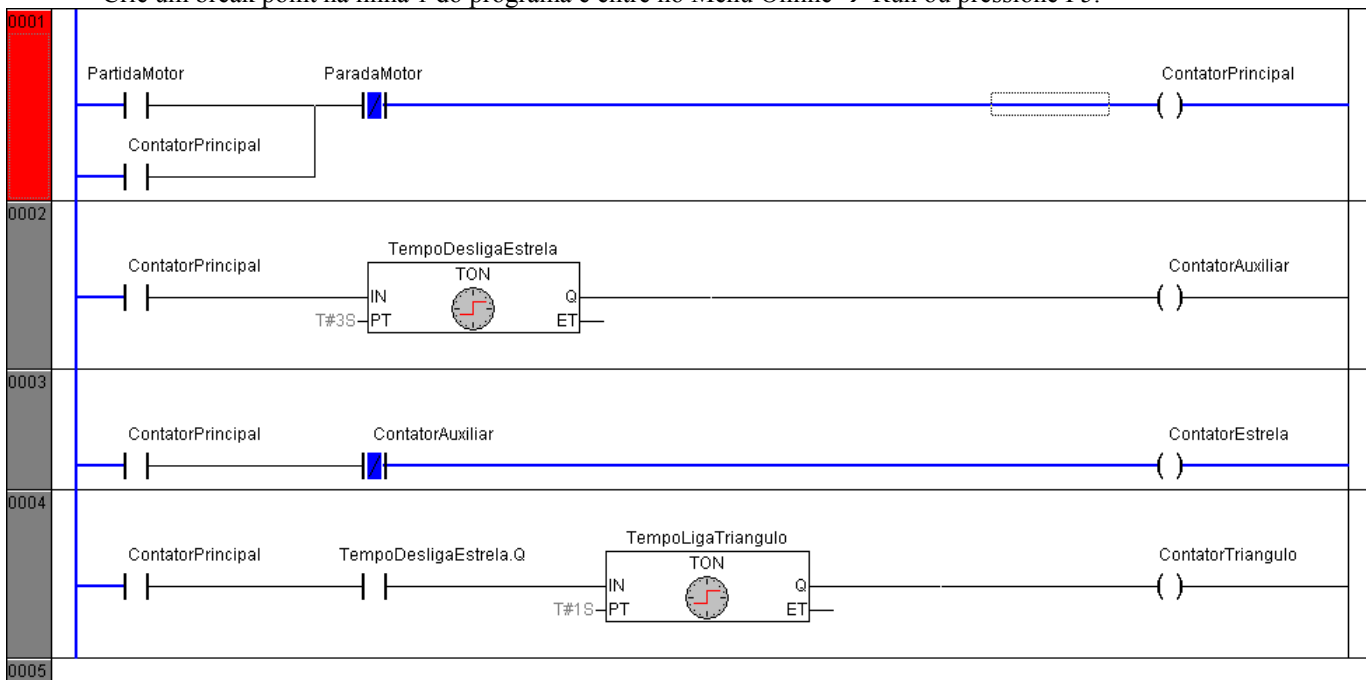


- O programa roda e irá parar na linha 2. Esta assume a cor vermelha para indicar onde o programa parou
- A linha 2 e a três só serão executadas quando pressionarmos RUN novamente.

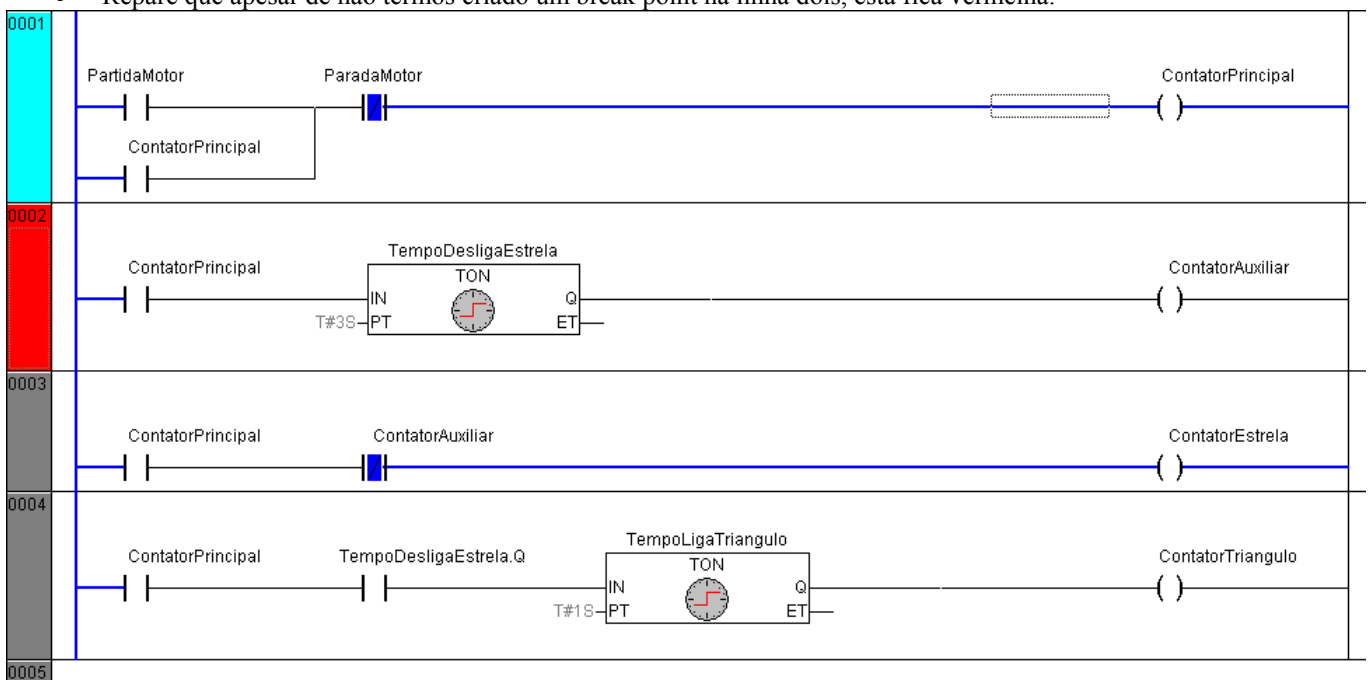


- O programa irá rodar até encontrar outro break point. No nosso caso o próximo break point está na linha 4.
- O programa só executará a linha 4, 5 e 1 se pressionarmos RUN novamente.
- O programa volta a parar na linha 2.
- Para desativar o break point clique uma vez com o botão esquerdo do mouse sobre a linha 2 e 4.
- Podemos ainda trabalhar com um controlador de ciclo, ou seja, um comando que induz o CLP a executar um ciclo de cada vez.
- Entre em Menu Online → Single Cycle ou pressione CTRL + F5. Com isto o programa irá rodar uma vez e logo em seguida entrará em stop.

- Outra ferramenta é o Step. Este comando faz com que o CLP execute uma linha lógica de cada vez.
- Crie um break point na linha 1 do programa e entre no Menu Online → Run ou pressione F5.



- Entre no menu Online → Step in , ou pressione F8.
- Repare que apesar de não termos criado um break point na linha dois, esta fica vermelha.

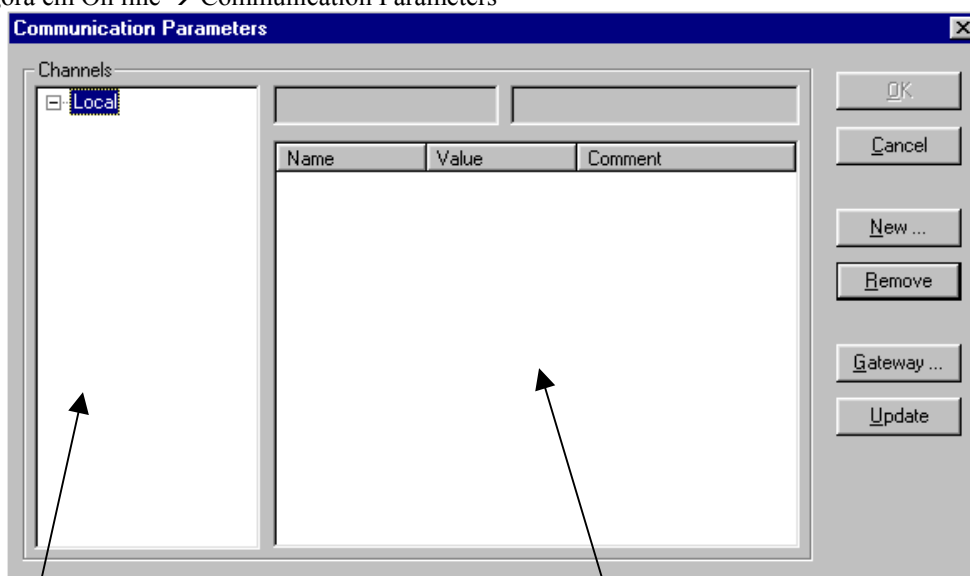


- Se pressionarmos F8 ou dermos mais uma vez o comando Step In, a linha 2 será executada e o programa irá parar na linha3.
- Caso pressionemos F5 ou executemos o comando RUN, o programa irá executar as linhas lógicas deste ponto em diante e irá parar assim que encontrar um break point.

CAPÍTULO 7

TRANSFERINDO O PROGRAMA

- Coloque o cabo no CLP, certifique-se de que ele está com a bateria.
- No menu Online, certifique-se que o Xsoft não está em modo de simulação.
- Entre agora em On line → Communication Parameters



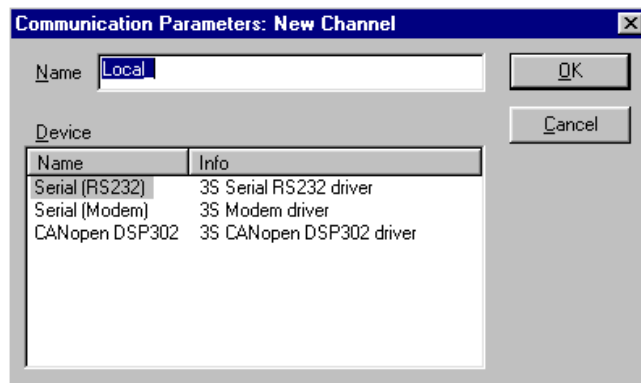
Tipo de conexão
(TCP/IP ou LOCAL)

Configuração da
conexão

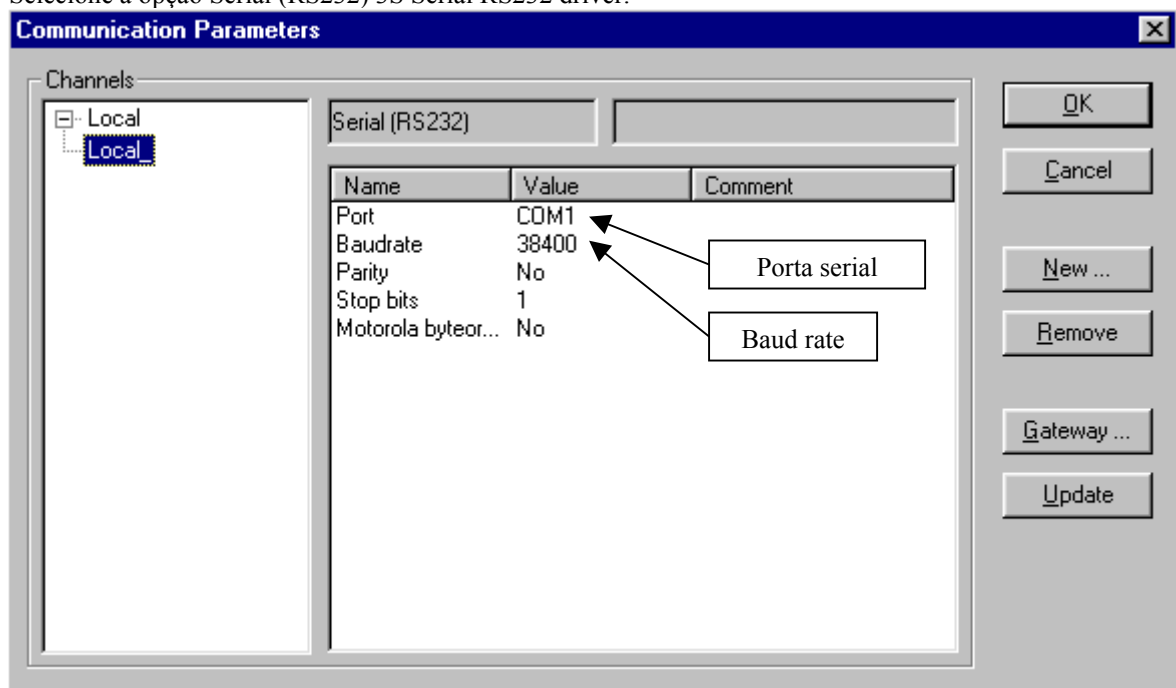
- Se estiver utilizando uma CPU XC100, então o tipo de conexão deverá estar local.
- Se for CPU modelo XC200 ou XC600, então o tipo de conexão será TCP/IP.
- Para deixar a conexão no modo local, clique em gateway e configure desta forma:



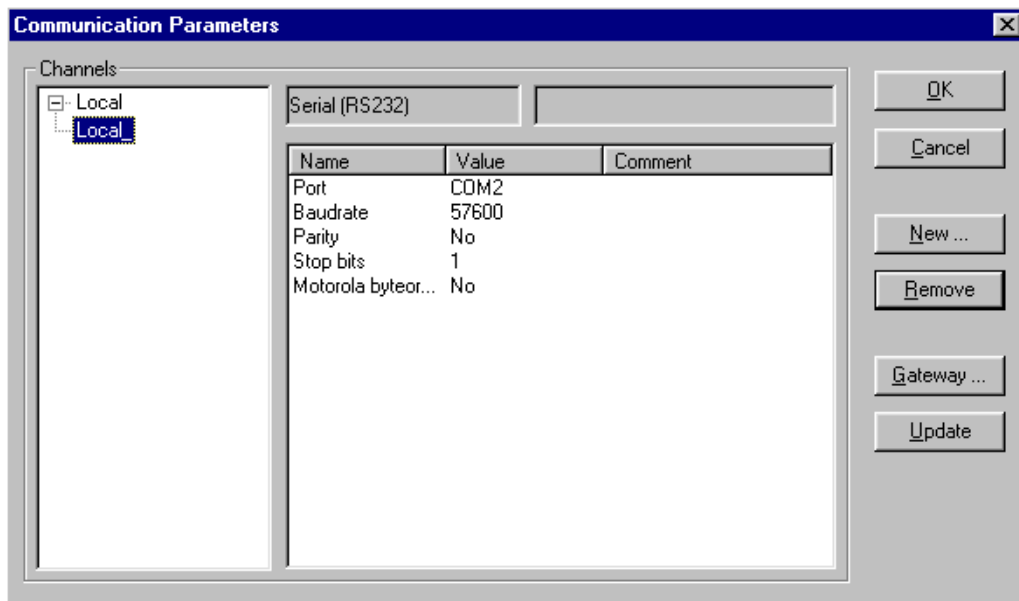
- Clique agora na opção New...



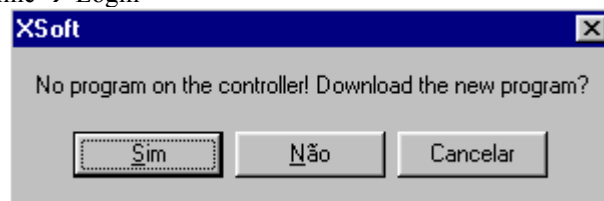
- Selecione a opção Serial (RS232) 3S Serial RS232 driver.



- Clique com o botão da esquerda sobre o COM1, ele irá ficar cinza e a cada dois cliques ele mudará a configuração, faça isso até achar sua configuração.
- Clique agora no Baud Rate, utilizando o mesmo princípio, até chegar na velocidade de 57600.



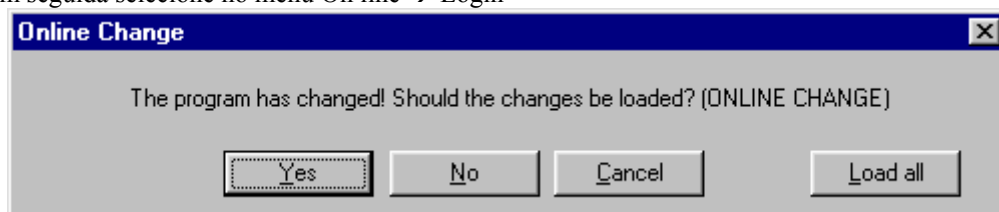
- Entre agora na opção Online → Login



- Selecione a opção Sim.
- Logo após o Download o CLP estará pronto para entrar em operação.
- Tudo o que foi explicado até aqui, tanto em monitoração como em simulação é válido quando se está conectado ao CLP.
- Para colocar o CLP em marcha selecione Online → RUN.

ALTERAÇÕES ON LINE

- Para executar qualquer alteração ON LINE no CLP proceda da seguinte forma:
- Entre em Online → Logout
- Faça as alterações necessárias no software.
- Entre no menu Project → Rebuild All...
- Logo em seguida selecione no menu On line → Login



- Se selecionarmos Yes as alterações serão feitas sem parar o processamento (ONLINE CHANGE)
- A opção Load all irá parar a CPU para executar a transferência.
- A alteração ONLINE só não pode ser feita em duas situações distintas: Alterações de hardware e inserções de novas bibliotecas.

CAPÍTULO 8

CONSIDERAÇÕES FINAIS

Este manual foi criado conforme a versão mais atual do software XSOFT. A Moeller Electric se reserva ao direito de revisar, alterar e efetuar quaisquer alterações neste manual sem aviso prévio.

Leia com atenção os manuais de programação, tente compreender mais blocos de funções através do Help do software e através do manual.

O programa aqui apresentado é apenas para efeito didático, a Moeller Electric não se responsabiliza por danos que o uso deste programa venha a causar.

Qualquer dúvida contate:

Moeller

Departamento de Engenharia de Aplicações.

Tel.: (0xx19) 3765-5520

(0xx19) 3765-5521

(0xx19) 3765-5500

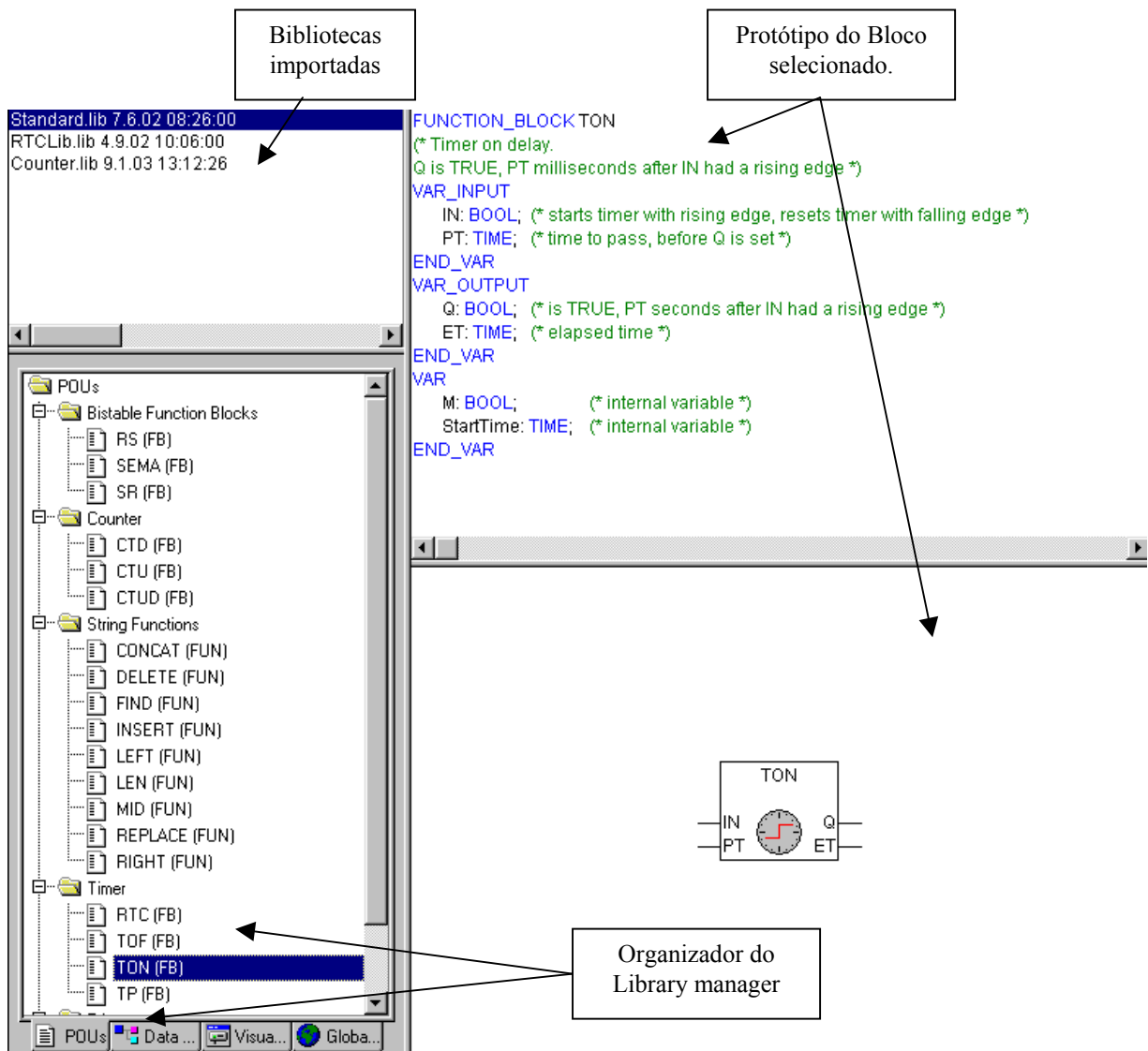
e-mail: suporte@moeller.com.br

ANEXOS

IMPORTANDO BIBLIOTECAS

As bibliotecas são conjuntos de function Blocks que podem ser exportadas e importadas. Para importar uma nova Biblioteca, proceda da seguinte forma:

- Selecione no menu a opção Window → Library Manager:



- Para importar uma nova biblioteca clicamos com o botão da direita do mouse sobre a área de bibliotecas importadas.
- Assim que a caixa de diálogo abrir selecionamos a opção Additional Library.
- E escolhemos o arquivo .LIB que desejamos importar neste projeto.

VARIÁVEIS

As variáveis geralmente são declaradas antes de serem usadas. A declaração consiste do nome da variável e o tipo de dado. Dependendo do tipo de dado atribuído a uma variável existe uma faixa de valores admissíveis. Por exemplo: o tipo de dado SINT possui 8 bits, e pode assumir valores entre -128 a 127, já o tipo USINT que também possui 8 bits, pode assumir valores entre 0 e 255.

Os tipos de dados elementares definidos pela IEC1131-3 são:

Binary	um ou vários bits em grupo de operações binária.
Integer	todos os números inteiros.
Real	números com ponto flutuante
String	conjunto de caracteres alfanuméricos.
Date/Time	datas, hora do dia e data e hora.

Junto com o tamanho da variável e sua faixa de valores, os tipos de dados também decidem que tipos de operação são permitidas para estas variáveis, por exemplo uma variável do tipo Integer só pode ser usado em funções aritméticas, já uma variável do tipo Binary só pode ser usada em funções lógicas.

Isto soluciona o problema do operador que testa diferentes tipos de dados na aplicação. Frequentes erros podem ser facilmente reconhecidos com uma rápida observação do programa. Temos os seguintes tipos de dados descritos a seguir:

Palavra Chave	Tipo de dado.	bits
BOOL	número booleano; aceita valores de 0 ou 1.	1
SINT	inteiro curto; com faixa de valores de -128 a 127.	8
INT	inteiro; com faixa de valores de -32768 a 32767.	16
USINT	inteiro curto sem sinal; com faixa de valores de 0 a 255.	8
UINT	inteiro sem sinal; com faixa de valores de 0 a 65535.	16
DINT	inteiro duplo com sinal com faixa de -2^{31} a $2^{31}-1$	32
UDINT	inteiro duplo sem sinal com faixa de 0 a $2^{32}-1$	32
LINT	inteiro longo com sinal com faixa de -2^{63} a $2^{63}-1$	64
ULINT	inteiro longo sem sinal com faixa de 0 a $2^{64}-1$	64
REAL	ponto flutuante com faixa de $\pm 10^{+/-38}$	32
LREAL	ponto flutuante com faixa de $\pm 10^{+/-308}$	64
TIME	duração	-
DATE	data.	-
STRING	Conjunto de caracteres com tamanho variável.	variável
BYTE	seqüência de 8 bits.	8
WORD	seqüência de 16 bits.	16

(*Alguns tipos de variáveis não são suportados por algumas CPUs*)

Ou seja, se desejamos fazer lógicas Booleanas devemos utilizar variáveis do tipo: BOOL, Byte ou Word; se for necessário efetuar funções aritméticas utilizamos as variáveis do tipo: INT, SINT, USINT, etc.