

INSTITUTO FEDERAL
SÃO PAULO
Campus Cubatão

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SÃO PAULO
CAMPUS CUBATÃO**

**CURSO SUPERIOR DE TECNOLOGIA EM AUTOMAÇÃO
E CONTROLE DE PROCESSOS INDUSTRIAIS
CONTÍNUOS**

APOSTILA DE

SISTEMAS SUPERVISÓRIOS

**CURSO SUPERIOR DE TECNOLOGIA EM AUTOMAÇÃO E
CONTROLE DE PROCESSOS INDUSTRIAIS CONTÍNUOS**

DISCIPLINA:
SISTEMAS SUPERVISÓRIOS MODERNO

6º MÓDULO

MONTAGEM:

PROFESSOR MARCELO S. COELHO

Revisão 13 – JULHO/2017

SUMÁRIO

CONCEITOS BÁSICOS	3
COMPONENTES FÍSICOS DE UM SISTEMA DE SUPERVISÃO.....	22
1º. LABORATÓRIO PARA CONFIGURAÇÃO DE SUPERVISÓRIO INDUSOFT	44
2º. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA DE CONTROLE COM SOFTWARE INDUSOFT E CLP STARDOM	83
.....	107
3º. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA DE CONTROLE COM SOFTWARE INDUSOFT E CLP STARDOM EM ACIONAMENTO DE ENTRADAS E SAÍDAS DIGITAIS	108
4ª. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA STARDOM + SUPERVISÓRIO COM TEMPORIZADOR	130
5ª. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA YOKOGAWA STARDOM + SUPERVISÓRIO COM ANALÓGICAS	142
6ª. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA DE CONTROLE REGULATÓRIO COM SOFTWARE INDUSOFT E CLP STARDOM	156

CONCEITOS BÁSICOS

Os sistemas supervisórios permitem que sejam monitoradas e rastreadas informações de um processo produtivo ou instalação física. Tais informações são coletadas através de equipamentos de aquisição de dados e, em seguida, manipulados, analisados, armazenados e, posteriormente, apresentados ao usuário. Estes sistemas também são chamados de SCADA (Supervisory Control and Data Acquisition).

Os primeiros sistemas SCADA, basicamente telemétricos, permitiam informar periodicamente o estado corrente do processo industrial, monitorando sinais representativos de medidas e estados de dispositivos, através de painéis de lâmpadas e indicadores.

Atualmente, os sistemas de automação industrial utilizam tecnologias de computação e comunicação para automatizar a monitoração e controle dos processos industriais, efetuando coleta de dados em ambientes complexos, eventualmente dispersos geograficamente, e a respectiva apresentação de modo amigável para o operador, com recursos gráficos elaborados (interfaces homem-máquina) e conteúdo multimídia.

Para permitir isso, os sistemas SCADA identificam todas as variáveis numéricas ou alfanuméricas envolvidas na aplicação através de tags, podendo executar funções computacionais (operações matemáticas, lógicas, com vetores ou strings, etc) ou representar pontos de entrada/saída de dados do processo que está sendo controlado. Neste caso, correspondem às variáveis do processo real (ex: temperatura, nível, vazão etc), se comportando como a ligação entre o controlador e o sistema. É com base nos valores das tags que os dados coletados são apresentados ao usuário.

Os sistemas SCADA podem também verificar condições de alarmes, identificadas quando o valor da tag ultrapassa uma faixa ou condição pré-estabelecida, sendo possível programar a gravação de registros em Bancos de Dados, ativação de som, mensagem, mudança de cores, envio de mensagens por pager, e-mail, celular, etc.

A partir do momento em que a monitoração e o controle de um processo são feitos com a ajuda de um sistema supervisório, o processamento das variáveis de campo é mais rápido e eficiente. Qualquer evento imprevisto no processo é rapidamente detectado e mudanças nos set-points são imediatamente providenciadas pelo sistema supervisório, no sentido de normalizar a situação. Ao operador fica a incumbência de acompanhar o processo de controle da planta, como o mínimo de interferência, excetuando-se casos em que sejam necessárias tomadas de decisão de atribuição restrita ao operador.

HIERARQUIA DOS SISTEMAS DE AUTOMAÇÃO

O nível mais alto dentro de uma arquitetura é representado pela rede de informação. Em grandes corporações é natural a escolha de um backbone de grande capacidade para interligação dos sistemas de ERP (Enterprise Resource Planning), Supply Chain (gerenciamento da cadeia de suprimentos), e EPS (Enterprise Production Systems). Este

backbone pode ser representado pela rede ATM ou GigaEthernet ou mesmo por uma Ethernet 100-BaseT, utilizando como meio de transmissão cabo par trançado nível 5. Esta última rede vem assegurando uma conquista de espaço crescente no segmento industrial, devido à sua simplicidade e baixo custo.

A rede de controle interliga os sistemas industriais de nível 2 ou sistemas SCADA aos sistemas de nível 1 representados por CLPs e remotas de aquisição de dados. Também alguns equipamentos de nível 3 como sistemas PIMS e MES podem estar ligados a este barramento. Até dois anos atrás o padrão mais utilizado era o Ethernet 10Base-T. Hoje o padrão mais recomendado é o Ethernet 100Base-T. Quase todos os grandes fabricantes de equipamentos de automação já possuem este padrão implementado.

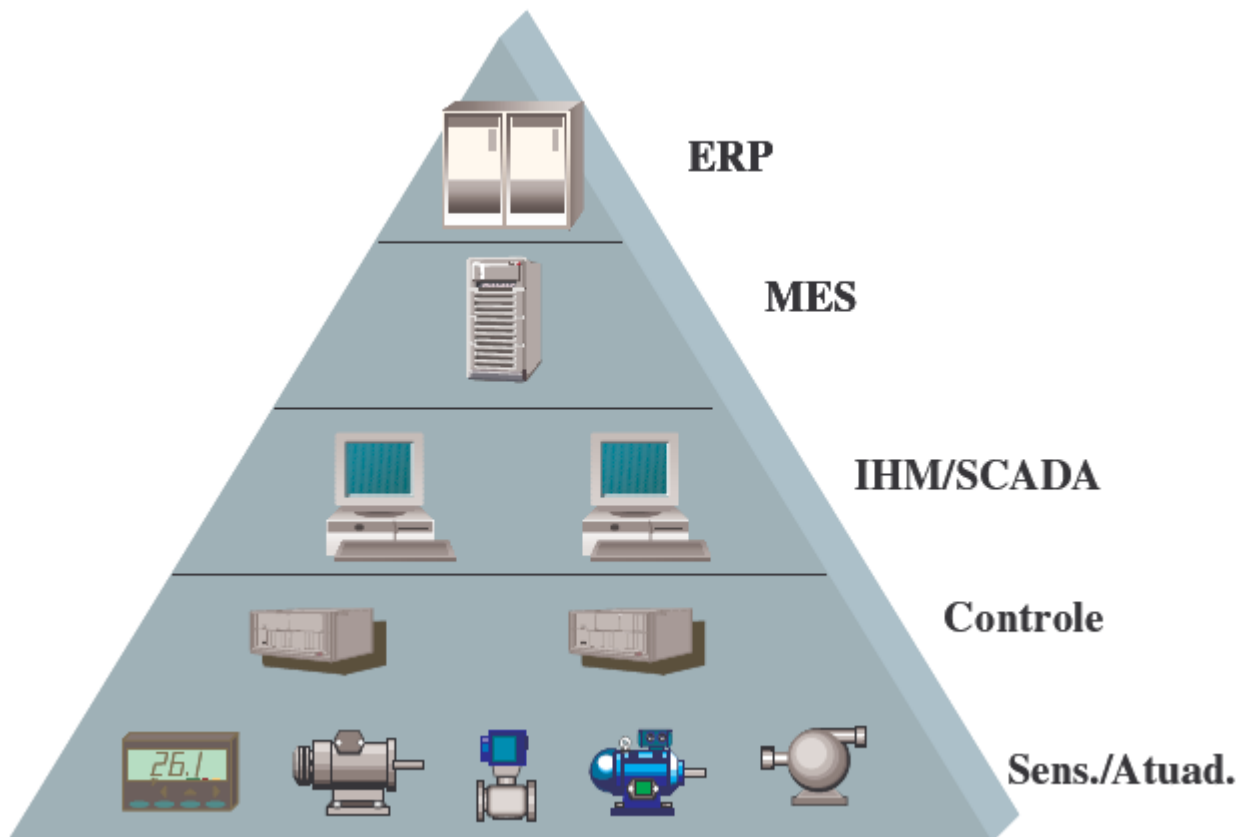


Figura 1. Hierarquia de sistemas de automação.



Figura 2. Função das camadas de uma hierarquia de automação.

Os sistemas SCADA são os sistemas de supervisão de processos industriais que coletam dados do processo através de remotas industriais, principalmente Controladores Lógico Programáveis, formatam estes dados, e os apresenta ao operador em uma multiplicidade de formas. O objetivo principal dos sistemas SCADA é propiciar uma interface de alto nível do operador com o processo informando-o "em tempo real" de todos os eventos de importância da planta.

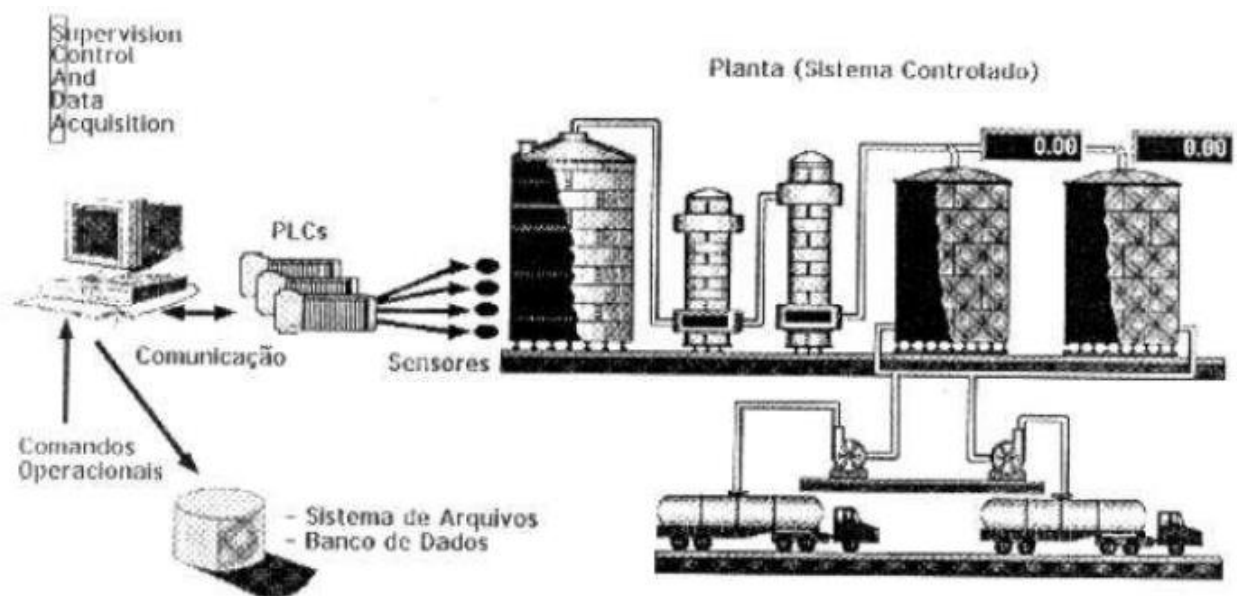


Figura 3. Exemplo de Sistema SCADA

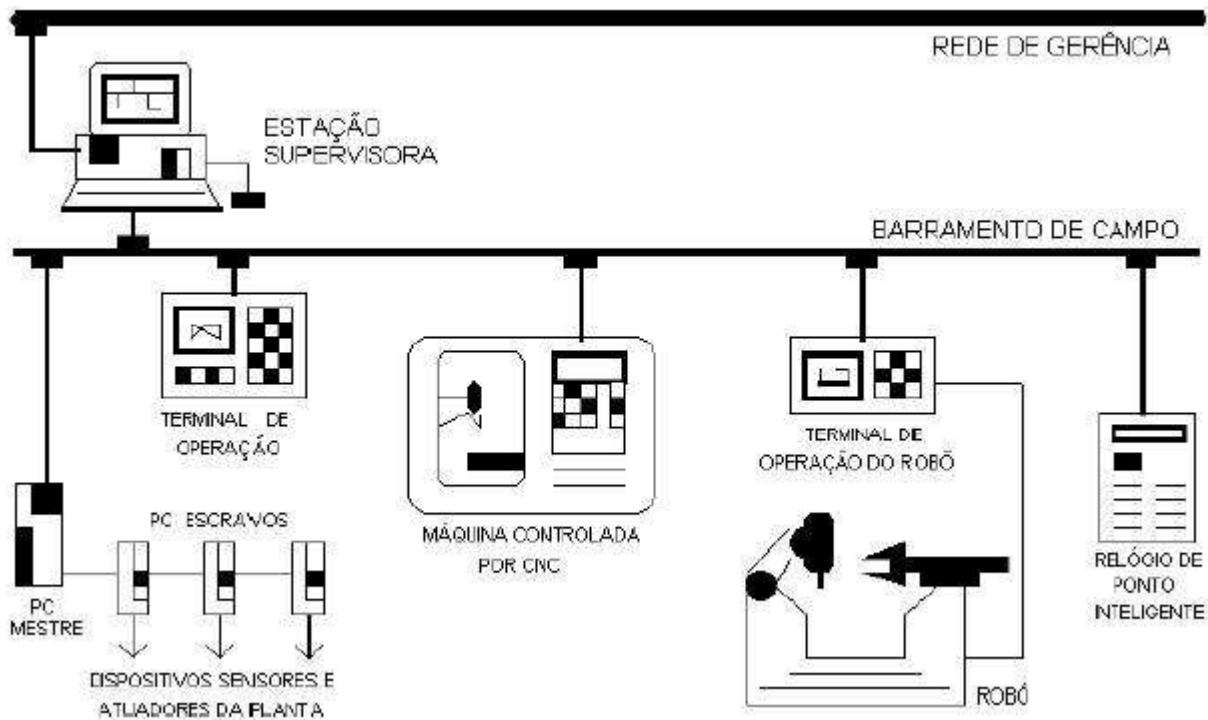


Figura 4. Exemplo de Arquitetura de uma rede SCADA.

a) Funções de supervisão:

Inclui todas as funções de monitoramento do processo tais como: sinóticos animados, gráficos de tendência de variáveis analógicas e digitais, relatórios em vídeo e impressos, etc.

b) Funções de operação:

Atualmente os sistemas SCADA substituíram com vantagens as funções da mesa de controle. As funções de operação incluem: ligar e desligar equipamentos e sequência de equipamentos, operação de malhas PID, mudança de modo de operação de equipamentos, etc.

c) Funções de controle:

Controle DDC ("Digital Direct Control") Alguns sistemas de supervisão possuem uma linguagem que permite definir diretamente ações de controle, sem depender de um nível intermediário de controle representado por remotas inteligentes. Todas as operações de entrada e saída são executadas diretamente através de cartões de I/O ligados diretamente ao barramento do micro, ou por remotas mais simples. Os dados são amostrados, um algoritmo de controle como um controlador PID, por exemplo, é executado, e a saída é aplicada ao processo (ação direta sobre uma variável manipulada). Isto, entretanto só é possível quando a velocidade do processo assim o permite. Em alguns casos, requisitos de confiabilidade tornam desaconselhável este tipo de solução.

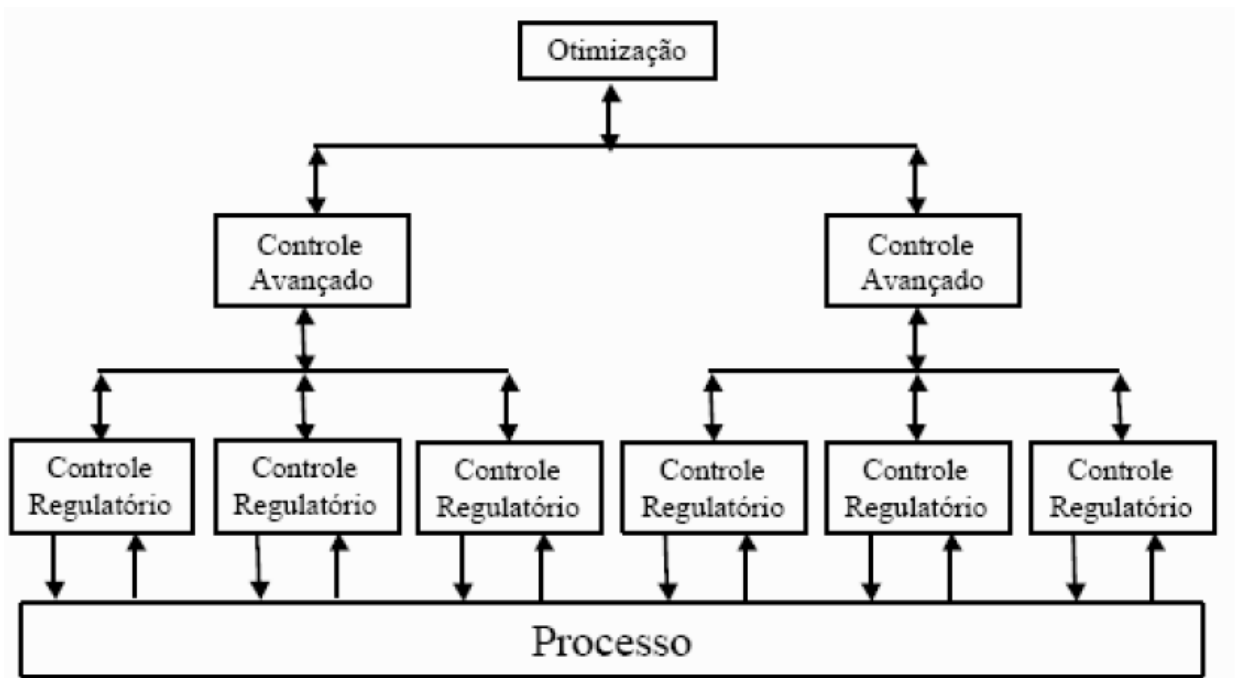


Figura 5. Hierarquia de sistemas de controle

ESTAÇÕES DE UM SISTEMA SCADA

É importante deixar claro, inicialmente, alguns conceitos importantes relacionado à aplicação dos softwares:

Estação (Nó): Estação (nó) é qualquer computador que esteja rodando um software supervisório. Estação (nó) local é aquela em que se está operando ou configurando e estação (nó) remota é aquela que é acessada através de um link de comunicação.

Estação independente ("Stand Alone"): É uma estação que desempenha todas funções de um sistema de supervisão não conectada a uma rede de comunicação.

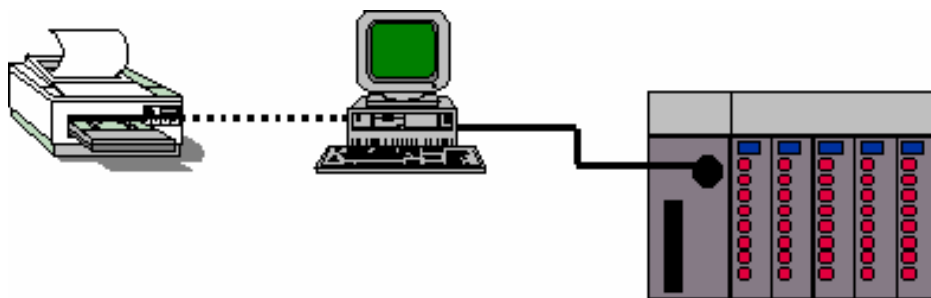


Figura 6. Estação stand alone.

Estação servidora de Base de Dados (Servidor "SCADA"): É uma estação que executa a função de aquisição de dados.

Estação de Monitoração e Operação: É uma estação que permite que o operador monitore o processo, altere parâmetros do processo, reconheça alarmes e mais algumas tarefas de operação de processo mas não permite alterar a configuração de telas nem da base de dados.

Estação de Gerência: É uma estação que permite a gerentes, supervisores ou quaisquer outras pessoas terem acesso aos dados de processo em forma de relatórios, gráficos e telas, sendo que reconhecimentos de alarme ou alteração de parâmetros do processo, entre outras tarefas de operação, não poderão ser realizadas nesta estação.

COMPONENTES LÓGICOS DE UM SISTEMA SCADA

Internamente, os sistemas SCADA geralmente dividem suas principais tarefas em blocos ou módulos, que vão permitir maior ou menor flexibilidade e robustez, de acordo com a solução desejada.

Em linhas gerais, podemos dividir essas tarefas em:

- Núcleo de processamento;
- Comunicação com PLCs/RTUs (DRIVER RUNTIME);
- Gerenciamento de Alarmes (ALARM);
- Banco de Dados (TAG'S DATABASE);
- Históricos (TREND);
- Lógicas de programação interna (Scripts) ou controle (MATH);
- Interface gráfica (VIEWER);
- Relatórios (REPORTS);
- Comunicação com outras estações SCADA (TCP/IP, DDE, ODBC);
- Comunicação com Sistemas Externos / Corporativos;
- Outros.

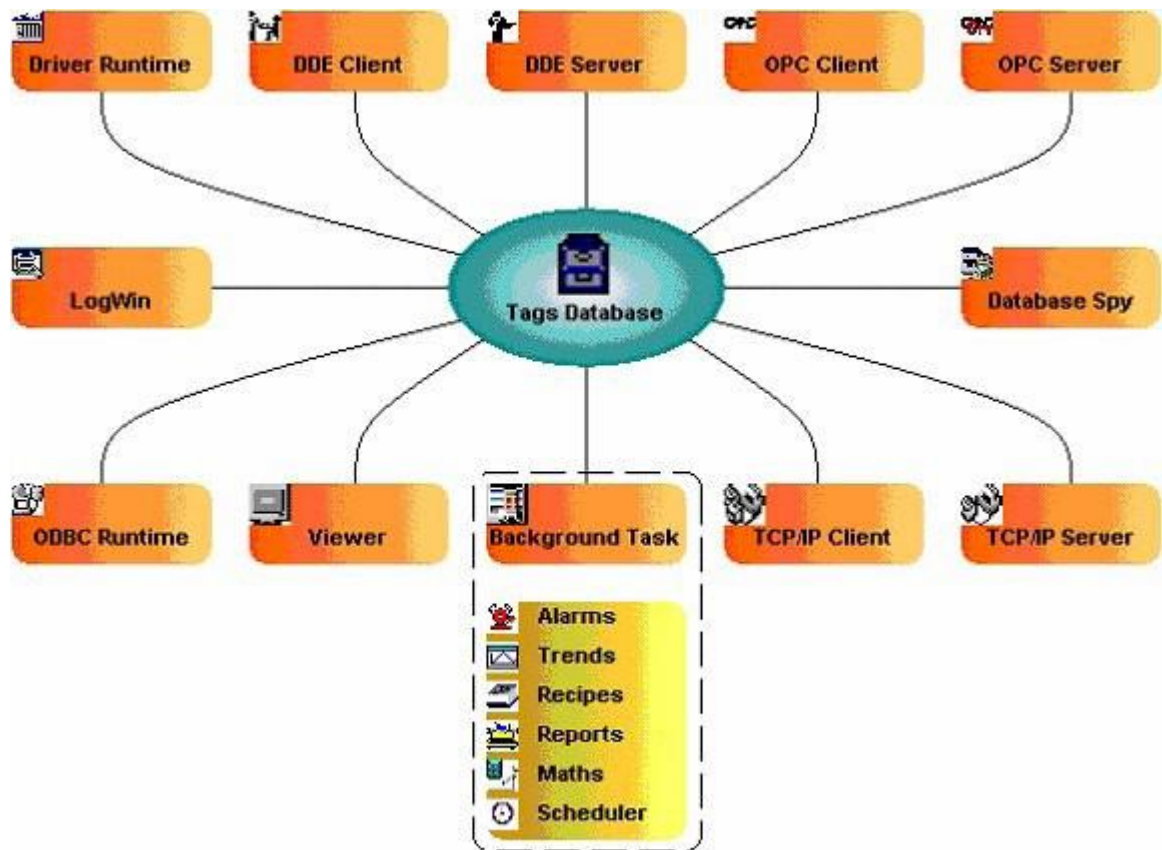


Figura 7.

A regra geral para o funcionamento de um sistema SCADA parte dos processos de comunicação com os equipamentos de campo, cujas informações são enviadas para o núcleo principal do software. O núcleo é responsável por distribuir e coordenar o fluxo dessas informações para os demais módulos, até chegarem à forma esperada para o operador do sistema, na interface gráfica ou console de operação com o processo, geralmente acompanhadas de gráficos, animações, relatórios, etc, de modo a exibir a evolução do estado dos dispositivos e do processo controlado, permitindo informar anomalias, sugerir medidas a serem tomadas ou reagir automaticamente.

As tecnologias computacionais utilizadas para o desenvolvimento dos sistemas SCADA têm evoluído bastante nos últimos anos, de forma a permitir que, cada vez mais, aumente sua confiabilidade, flexibilidade e conectividade, além de incluir novas ferramentas que permitem diminuir cada vez mais o tempo gasto na configuração e adaptação do sistema às necessidades de cada instalação.

APRESENTAÇÃO GRÁFICA

Fornecem uma representação gráfica geral da planta em substituição aos painéis sinóticos tradicionais. Cada sinótico representa uma área do processo em um certo nível de detalhe. Para se obter uma visão mais detalhada de uma determinada área pode-se recorrer a um novo sinótico, a um sinótico de hierarquia inferior (sub-sinótico), ou a uma visão de uma outra camada do mesmo sinótico (sistema "multi layer").

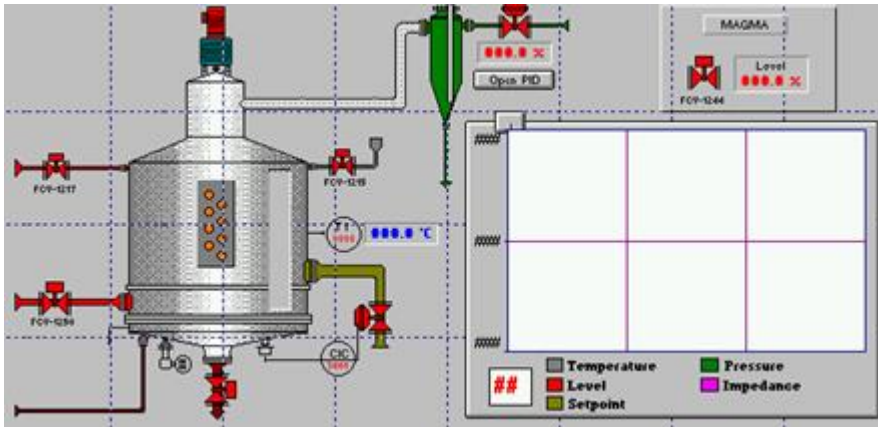


Figura 8. Exemplo de uma apresentação gráfica

No sistema gráfico ao contrário, o desenho é formado livremente pela combinação de entidades geométricas fundamentais como retas, retângulos, elipses e círculos, texto bitmapeado e vetorados ("stroke-fonts"), arcos, splines, curvas de bézier, etc. Após definidos, os símbolos são armazenados numa biblioteca. Se a representação armazenada corresponde à descrição das entidades geométricas temos um sinóptico orientado para geometria. Se o símbolo armazenado corresponde a uma configuração fixa de bits (mapa de bits), temos um editor bitmapeado. O construtor de sinópticos é a ferramenta que permite ao usuário criar novos sinópticos. De preferência deve ser possível se usar o construtor com o sistema on-line. Alguns construtores são editores gráficos que definem duas estruturas de dados básicas: uma para a máscara e outra para os campos dinâmicos. Alguns construtores, entretanto necessitam compilar a descrição de campos para obter um código executável para as animações.

GERENCIADOR DE ALARMES

Este módulo está presente em todos os sistemas SCADA. Ele recebe os eventos excepcionais do processo e os registra identificando:

- Data e hora do evento;
- Variável alarmada;
- Valor no momento do alarme;
- Descrição do evento;
- Data e hora de normalização do evento;
- Status do evento: alarmado, normalizado, reconhecido pelo operador.

Os eventos são armazenados em um buffer circular em disco ou memória que armazena os últimos eventos correspondendo geralmente a um turno de operação. Após este prazo, o arquivo pode ser salvo em disco para análise histórica, transferido para outro computador

de maior capacidade de armazenamento ou simplesmente descartado. A interface com operador se dá geralmente de duas formas: Uma janela de alarmes exibe os alarmes mais recentes. O operador pode solicitar a exibição de uma página de alarmes contendo uma certa quantidade de alarmes recentes e depois pagnar para frente e para trás até encontrar o evento de interesse.

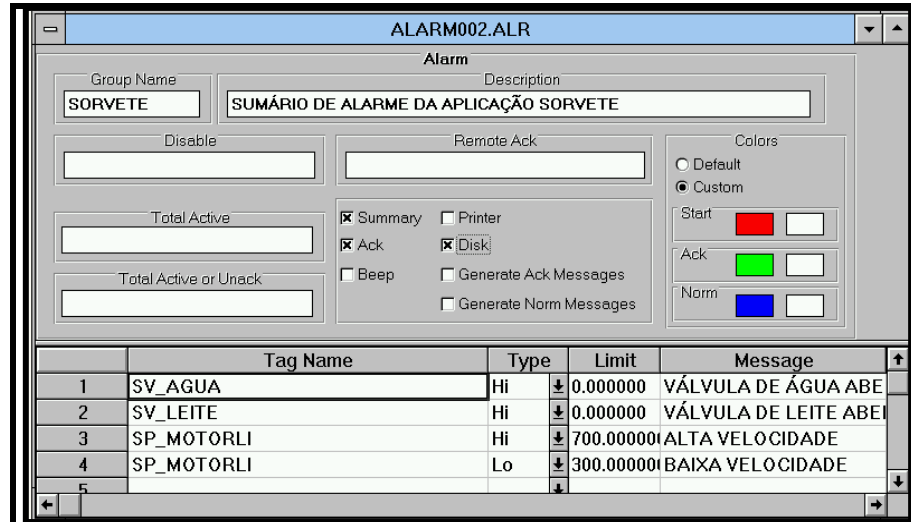


Figura 9. Exemplo de tela de *configuração de alarmes*.

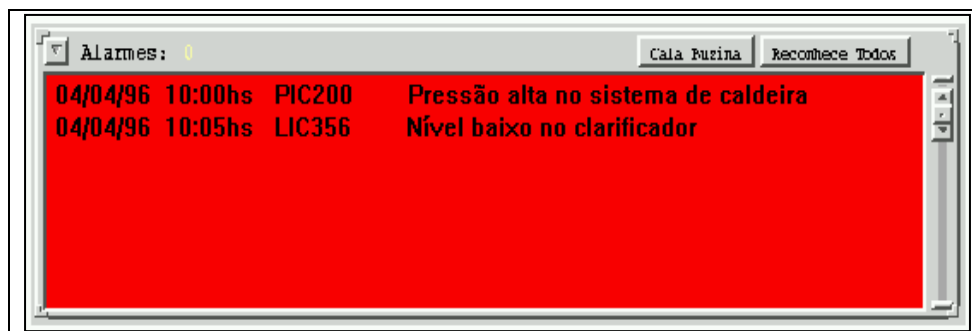


Figura 10. Exemplo de tela *sumário de alarmes*.

Quando um alarme ocorre o operador é avisado através de uma buzina, música ou por um speech maker. O operador deve declarar que está ciente do problema, reconhecendo o alarme mais recente ou todos os alarmes simultaneamente. O alarme mais recente é então substituído na tela por outro alarme que ainda não tenha tido a oportunidade de se manifestar.

REGISTRADOR DE TENDÊNCIA

Desenha um gráfico de tendência de uma dada variável, geralmente analógica, em função do tempo. Um gráfico de tendência pode ter, normalmente, de 1 a 8 penas. O usuário deverá definir duas coisas:

- Quais variáveis devem ser amostradas pelo sistema sincronamente com um relógio e com qual frequência;
- Para cada gráfico, quais das variáveis registradas devem ser exibidas simultaneamente em uma mesma tela.



Figura 11. Exemplo de um gráfico de tendência

Os períodos de amostragem que variam tipicamente de 100 ms a 1 hora devem ser escolhidos de acordo com a velocidade real do processo. É normal escolher um período para cada tipo de variável (temperatura, nível, pressão, etc). Os dados são geralmente armazenados em um buffer circular. O período total de armazenagem de dados irá depender do período de amostragem e do tamanho do buffer.

As operações possíveis neste tipo de gráfico são:

- a) Movimentação de cursor: Um cursor gráfico vertical (dial) é passeado pela figura fornecendo a leitura dos pontos onde o cursor intercepta as curvas. Serve para examinar os valores críticos da variável e o momento em que um evento ocorre.
- b) Paginação: Realiza paginação horizontal do gráfico para visualização dos pontos passados.
- c) Zoom horizontal e vertical: Realiza ampliação da parte do gráfico selecionada por uma janela de edição. A redução de escala pode ser feita para se retornar a escala original, ou para uma atingir uma escala reduzida, gerada pela compressão dos dados armazenados.

TENDÊNCIA HISTÓRICA

Quando se deseja armazenar valores de variáveis em disco por longos períodos de tempo (até 1 ano) se recorre ao registro histórico. O registro histórico possui um período de amostragem tipicamente maior que o da tendência instantânea (alguns minutos). Os valores amostrados podem ser:

- Os valores instantâneos da variável no instante da amostragem histórica;
- Valores resultantes da compactação de amostras de tendência instantânea coletados a períodos inferiores.

Nos sistemas mais recentes, podem-se visualizar os gráficos de tendência instantânea e histórica de forma concatenada em um único sistema de coordenadas. Hoje se prefere realizar todas as funções históricas de um sistema SCADA em um software externo denominado PIMS (Process/Plant Information Management System). Os PIMS possuem melhores algoritmos de compressão de dados e armazenam tipicamente 5 anos de operação em um winchester de dimensões normais (20 Gbytes). Os sistemas PIMS possuem uma riquíssima variedade de funções para manipulação dos registros históricos:

- a) Rápida importação de variáveis de processo através do tag browser. Você escolhe uma variável do dicionário de dados e arrasta o tag para o sinóptico. O número de penas simultâneas é ilimitado;
- b) Um gráfico de tendência configurado desta maneira pode ser salvo e constituir uma vista personalizada do processo de um operador em particular;
- c) Além das funções de panning e zoom, podemos inserir múltiplos cursores no gráfico para visualização dos valores das variáveis;
- d) Pode-se incluir o sinóptico em uma mensagem e enviá-la por e-mail a algum especialista que esteja em casa e queira analisar os dados;
- e) Podemos suspender a impressão de qualquer das penas por tempo indeterminado;
- f) Etc.

GERADOR DE RELATÓRIOS

Até agora os únicos dados armazenados em disco foram os relativos aos registros de tendência instantânea e histórica. Evidentemente uma das principais funções dos sistemas SCADA reside na sua capacidade de armazenar dados e produzir relatórios de produção ao final de um turno, dia ou mês. Os relatórios de produção incluindo os relatórios de balanço de massa ou energia demonstram quanto uma determinada planta produziu, quanto consumiu de insumos, de energia, etc e constituem o principal relatório de interesse gerencial. Outro alvo de interesse, desta vez, por parte do pessoal da manutenção são os relatórios de monitoramento de equipamentos que dizem quando cada equipamento

parou, por que parou e por quanto tempo ficou parado. O usuário deve definir as variáveis que farão parte do relatório e o seu período (ou instante) de amostragem. A armazenagem de um dado pode estar vinculada a um evento e não apenas a um horário. Isto é muito comum em sistemas de batelada. Depois deverá ser definido o formato do relatório e o instante de sua impressão (final do turno, dia, etc).

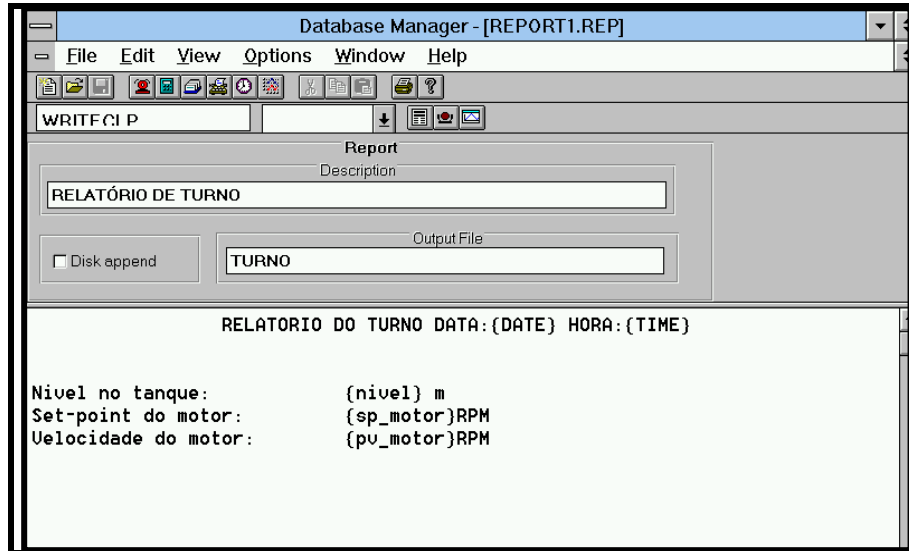


Figura 12. Exemplo de tela de configuração de relatórios.

CONSTRUTOR DE ESTRATÉGIAS DE CONTROLE

Alguns sistemas supervisórios possuem uma linguagem procedimental ou gráfica (blocos lógicos) para descrever procedimentos de controle a serem realizados em nível de supervisão.

Basicamente as funções de um construtor de estratégias podem ser agrupadas nas seguintes bibliotecas:

- a) Biblioteca de entrada e saída: Realiza leitura e escrita de variáveis na RTU, solicita dados ao operador.
- b) Biblioteca de funções matemáticas: funções convencionais + Max, Min, Média, Alarm, etc.
- c) Biblioteca de funções lógicas: And, Or, Xor, Not, Pulso, etc.
- d) Biblioteca de funções de controle: PID, PI, etc.
- e) Biblioteca de funções de tratamento de sinais: Lead_Lag, Atraso, Interpolação linear em tabela, Filtro, Função Log após ganho e offset, Função Exp após ganho e offset, Aplicação de ruído, Totalização (integração), Derivação.

Em sistemas supervisórios mais recentes, o construtor de estratégias tem sido totalmente substituído por um módulo que programa as funções típicas de um CLP utilizando as linguagens do padrão IEC 61131-3.

LOG DE EVENTOS

Registra no arquivo de diário de bordo todos os eventos relevantes de operação, com data, hora, descrição do evento e operador “logado” na hora do evento. Os eventos de interesse geralmente são:

- Eventos de configuração da base de dados;
- Eventos de operação críticos tais como: ação sobre malhas de controle, partida e parada da planta ou de equipamentos críticos, etc.

Este registro está vinculado à existência de um sistema de senhas para identificar cada operador que assuma a operação. O operador que deixa a operação deverá realizar o logoff da estação enquanto o novo realiza o login, identificando-se.

RECEITAS

Uma Receita é um conjunto de valores pré-definidos que podem ser carregados para um grupo de tags a fim de configurar um processo específico. Esta lista de tags também se chama modelo de receita.

Por exemplo, seja uma máquina que fabrica diferentes tipos de parafusos. As variáveis envolvidas no processo são sempre as mesmas, mas seus valores provavelmente irão mudar dependendo do tipo de parafuso que se quer produzir. Supondo que você tem diferentes configurações de máquina para cada tipo de parafuso, estes valores poderiam ser gravados em uma receita e serem posteriormente carregados em tags de controle, facilitando a tarefa do operador e evitando erros.

Dessa maneira, podemos criar um modelo de receita “Parafuso” com diversas receitas “Fenda Philips”, “Fenda Torx”, “Fenda Simples” e assim por diante. Para que sejam recuperados quando necessário, os modelos e os dados de uma receita são armazenados em disco, em um “arquivo de receitas”.

LINGUAGEM PARA DEFINIÇÃO DE APLICAÇÃO DO USUÁRIO

Permite ao usuário criar seus próprios programas associados à ocorrência de algum tipo de evento. Os eventos podem ter diversas naturezas. Alguns supervisórios contemplam apenas parte destes eventos. Hoje a compreensão da natureza dos eventos possíveis de ocorrer tem trazido alto grau de generalidade às ferramentas. Alguns eventos possíveis são: Variação de um dado de processo, atendimento de uma condição de processo, acionamento de uma tecla especial, clique do mouse sobre um objeto ativo (pushbutton), lançamento de uma aplicação, abertura de uma janela, dentre outros.

Seja, por exemplo, o evento gerado pelo acionamento de uma tecla especial do teclado. O programa pode ser programado para ser ativado: ao se apertar a tecla; enquanto a tecla permanecer apertada; ao se aliviar a tecla.

Outro exemplo seria um evento associado a uma condição (equação booleana envolvendo valores de variáveis de processo) que causa a execução de um Script: quando a expressão se tornar verdadeira; enquanto a expressão for verdadeira; quando a expressão se tornar falsa ou enquanto a expressão for falsa.

OS OBJETOS DE UM SUPERVISÓRIO

Cada sistema SCADA irá gerenciar um certo número de objetos também denominados entidades que descrevem as variáveis de processo controladas e os elementos habituais de uma mesa de controle. Embora este número varie de um sistema para outro, existe um consenso em relação às suas características básicas.

A atividade de configuração de um supervisório inicia geralmente na definição de cada variável de processo na base de dados. Em todo o sistema existem basicamente duas variáveis simples ou primitivas e alguma variáveis compostas formadas a partir destas.

VARIÁVEIS SIMPLES

Variável analógica: Descreve uma variável analógica de entrada. O CLP já converte esta leitura para um valor digital quantizado em 12 bits (0 a 4095).

Principais atributos:

- a) Tag: conjunto de até 12 caracteres descrevendo o nome da variável
- b) Texto: Descrição sucinta da variável
- c) Unidade de Engenharia (UEng): unidade de medida, por exemplo: ton./h, °C, metros, etc.
- d) Valor: último valor lido da variável
- e) Lim inferior: valor em UEng ser atribuído ao valor 0% da variável.
- f) Lim superior: valor em UEng a ser atribuído ao valor 100% da variável
- g) Limite HH: valor em UEng para alarme Muito Alto
- h) Limite H: valor em UEng para alarme Alto
- i) Limite L: valor em UEng para alarme Baixo
- j) Limite LL: valor em UEng para alarme Muito Baixo

k) Histerese: valor percentual de histerese para geração de alarmes. Serve para evitar múltiplos alarmes quando o valor da variável cruza uma das faixas definidas pelos valores limites.

l) Filtro: Somente quando o valor da variável variar mais que o limite especificado (filtro) é que o valor em tela será atualizado.

m) Endereço: Corresponde ao endereço da variável no CLP.

Os valores limites definem 5 faixas de operação para a variável. Quando a variável cruza uma destas fronteiras um alarme é dado para o operador. Em geral os valores da variável na tela dependem das cores atribuídas a cada faixa. Estas cores em geral são as mesmas para todas as variáveis.

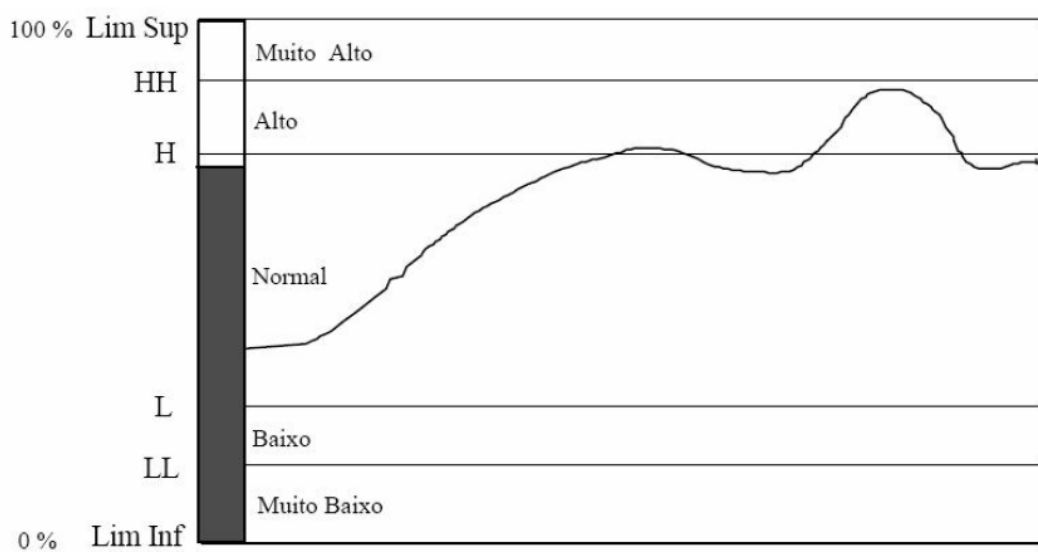


Figura 13. Faixas de uma variável analógica

Variável Discreta: Também chamada de variável digital. Corresponde a 1 bit de informação que pode assumir os valores 0 ou 1.

Principais atributos:

- Tag: nome da variável
- Texto: Descrição sucinta da variável
- Descrição do estado: texto a ser atribuído aos estados 0 e 1 da variável. Pode corresponder a 2 strings ou a um índice para uma tabela que contém os textos a serem associados a cada tipo de variável, por exemplo: Aberto/Fechado para válvulas, Ligado/Desligado para motores etc,
- Endereço: endereço da variável na memória do CLP
- Tipo: Diz se a variável desempenha alguma função importante como, por exemplo: bit de Funcionando, Pronto para Partir ou Defeito de algum equipamento.

f) Valor: valor corrente do ponto

g) Alarme: Determina a classificação do alarme associado ao ponto: Crítico, Urgente, Alarme, Evento, Sem alarme. O sistema também define que tipo de ação será realizada para cada classe de alarme.

O valor normal para uma variável discreta é sempre convencionado para todas as variáveis em conjunto para maior facilidade de interpretação por parte do operador e pessoal da manutenção.

Em alguns sistemas é possível definir agregados de variáveis discretas (bit strings), definir cores e textos para cada estado (2n), etc.

VARIÁVEIS COMPOSTAS

Enquanto alguns sistemas possuem apenas um pequeno número de primitivas, outros possuem tipos mais complexos formados pelo agregado de várias primitivas fundamentais.

De uma forma ou de outra cada sistema sempre possui uma forma de representar todas as entidades que correspondem às entidades controladas no processo ou aos instrumentos da mesa de comando convencional:

TOTALIZADOR

Totaliza uma variável analógica diretamente ou reproduz um valor totalizado pelo CLP. O input da totalização tanto pode ser um valor analógico (fluxo ou vazão) ou pulsos associados a uma variável discreta (totalizador de pulsos). Neste último caso, a cada pulso, um incremento pré determinado é aplicado ao valor totalizado.

O operador pode a qualquer tempo zerar o valor totalizado. Assim é conveniente definir três atributos básicos:

Ø Valor totalizado sem reset: não pode ser zerado pelo operador.

Ø Valor totalizado com reset: pode ser zerado pelo operador ao final de um turno, dia, mês, etc

Ø Data de reset: armazena a data e hora da última ação de reset.

CONTROLADOR PID

Descrevem um controlador PID independente de sua encarnação física. Um sistema supervisório moderno é capaz de comandar dezenas de malhas de controle que na verdade são implementadas no nível inferior de controle representado pelos CLP, Single loops e Multi loops.

Os parâmetros de configuração definidos são enviados para a memória do controlador. O operador pode então realizar a sintonia da malha durante o start-up do sistema, guiado por uma tela de tendência que plota os valores de variável medida, set-point e variável manipulada em função do tempo (tuning) ou usando uma ferramenta qualquer de sintonia.

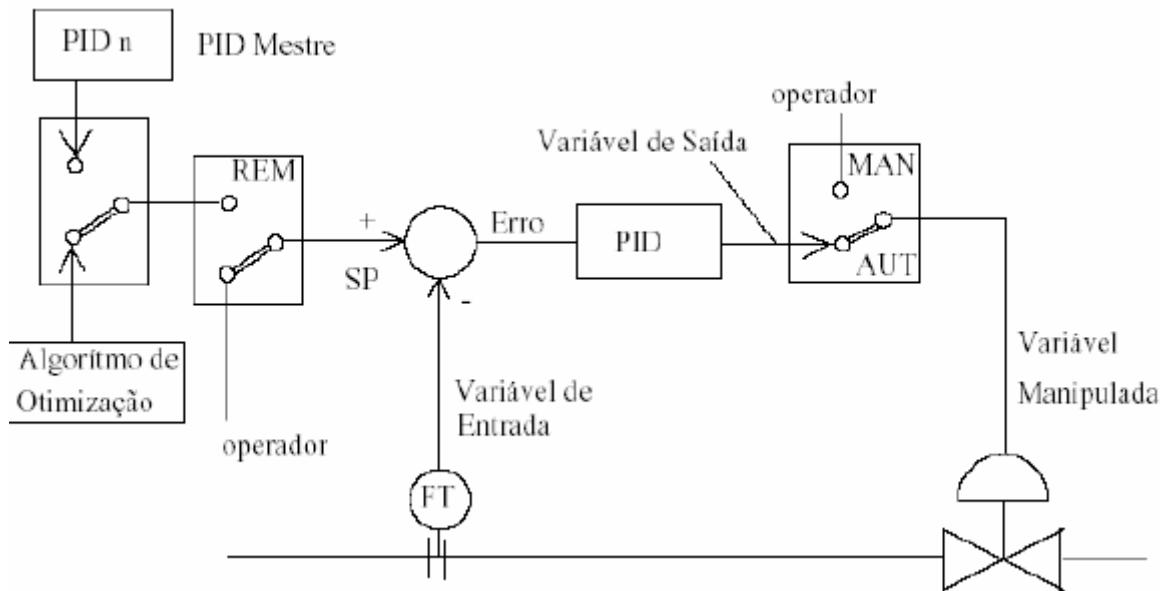


Figura 14. Malha PID típica

A partir daí as únicas operações possíveis são mudar o modo de operação da malha ou o seu set-point.

Atributos típicos:

- a) Tag: identificação da malha de controle.
- b) Texto: descrição da malha.
- c) Variável de entrada: identificação da variável de entrada (medida) da malha.
- d) Constantes: K_c , T_i , T_d para sintonia da malha. e) Modo: modo de operação da malha PID:

- Modo Manual: O operador age diretamente sobre a variável de saída do bloco PID (variável manipulada).

- Modo Automático: A ação se dá sobre o set-point da malha.

- i. Modo Automático Local: O próprio operador define o valor do setpoint.

- i. Modo Automático Remoto: O set-point é definido externamente: pela variável de saída de outra malha em cascata (malha mestre) ou por um valor decorrente de algoritmo de otimização executado pelo supervisor.

- f) Identificação da Malha Mestre (quando aplicar)

- g) Endereço: Endereço base do bloco de parâmetros PID na memória do CLP

- h) Identificação do operador que definiu ou sintonizou a malha
- i) Banda morta
- j) Limite máximo e mínimo da variável de saída k) Polarização (bias)

EQUIPAMENTO

Corresponde a um equipamento de processo qualquer: motor, classificador espiral, transportador de correia, reator, etc.

Seus principais atributos são:

- a) Tag: Identificação do equipamento
- b) Texto: Descrição do equipamento
- c) Bits de status: Cada equipamento possui uma série de bits típicos de entrada para o supervisor, que em suma são função da filosofia de programação de funcionamento destes equipamentos em nível do CLP, por exemplo:

- FUN - bit que avisa ao supervisor que o equipamento está funcionando
- PPP - bit que sinaliza que o equipamento está pronto para partir (pronto para operar)
- DEF - Bit que diz que o equipamento está com defeito (or lógico de todos os bits de defeito do equipamento)

Além destes bits hoje é normal enviar para o supervisor todos os bits de defeito que discriminam qual o problema ocorrido. Estes bits são utilizados para alarme, diagnóstico, e para guias de operação no caso dos sistemas inteligentes.

- MODO: um, dois ou três bits que discriminam o modo de operação do equipamento:
- LOCAL/REMOTO: diz se o comando está sendo dado pela central de controle ou pelo painel de comando local do equipamento.
- OPERAÇÃO/MANUTENÇÃO: diz se o equipamento está sendo comandado pela seqüência a qual pertence ou isoladamente para manutenção, ETC

- d) Bits de comando: São bits de saída do supervisor para o equipamento permitindo realizar ações de comando:

- LIGA: comando de liga para o equipamento.
- DESLIGA: comando de desliga
- MODO: definição de um novo modo para o equipamento.

e) Horímetro: Fornece o total de horas de operação do equipamento

SEQÜÊNCIA OU GRUPO

É um conjunto de equipamentos intertravados que pode ser tratado do ponto de vista lógico como uma única entidade. Cada seqüência é composta tipicamente por 8 ou mais equipamentos. Por exemplo: o conjunto de motores e outros subsistemas que compõem um moinho de bolas pode ser tratado de forma unificada pelo sistema, já que todos os equipamentos que o compõem partem e operam em conjunto de forma intertravada. Um equipamento pode pertencer a diversas seqüências diferentes. Por exemplo, um transportador de correia do pátio de um porto pode estar conectado a diversas rotas de transferência de minério de acordo com o tipo de minério sendo carregado, píer de embarque, etc.

Atributos típicos:

- a) Tag: Identificação da seqüência
- b) Texto: Descrição da seqüência
- c) Equipamentos:Relação dos equipamentos que compõem a seqüência.
- d) Bits de status: semelhantes aos bits de status de um equipamento: Funcionando, Pronto para Partir, Defeito.
- e) Modos de Operação: Basicamente a seqüência pode operar em Remoto, onde todos os equipamento estão intertravados e são operados pelo supervisor, ou em Local onde cada equipamento é operado independentemente. A partida se inicia pelo equipamento cabeça da seqüência,
- f) Bits de comando: Servem para ligar desligar ou mudar o modo de operação de todos os equipamentos de uma seqüência.

CÁLCULO

Nem todas as variáveis de interesse podem ser amostradas diretamente do processo. Às vezes estamos interessados no comportamento de uma variável que não é medida, mas que pode ser inferida a partir de diversas outras variáveis. Por exemplo, calcular a potência ativa a partir dos valores de tensão, corrente e fator de potência.

Cálculos como estes em geral são definidos a partir de uma linguagem procedimental que deve conter no mínimo:

- Funções de acesso a valores de variáveis do processo.
- Funções aritméticas fundamentais: soma, subtração, multiplicação, divisão, mod.
- Funções booleanas: and, or, xor e not.

- Funções transcendentais: trigonométricas, exponenciais, logarítmicas, radiciação, etc.
- Instruções condicionais: if ...then... else...
- Lógica de parênteses
- Funções de manipulação de texto

O tipo retornado de um cálculo geralmente é um valor real, um valor booleano ou um string. A chamada de um cálculo que envolve um outro cálculo deve referenciar sempre ao último valor calculado para se evitar chamadas recursivas infinitas.

A variável calculada deve se comportar como um ponto qualquer amostrado diretamente do processo e permitir:

- Gerar valores instantâneos para exibição em sinóticos segundo diversas formas.
- Gerar valores para base de dados histórica.

Processadores de expressões também são úteis para diversas finalidades dentro de um sistema de supervisão como, por exemplo, para criar campos virtuais em planilhas e geradores de relatórios.

Outras entidades podem ser definidas dependendo do tipo de processo como Chaves de Seleção, Chaves de Habilitação, Tanques, Constantes Paramétricas, Rotas, etc.

COMPONENTES FÍSICOS DE UM SISTEMA DE SUPERVISÃO

Os componentes físicos de um sistema de supervisão podem ser resumidos, de forma simplificada, em: sensores e atuadores, rede de comunicação, estações remotas (aquisição/controla) e de monitoração central (sistema computacional SCADA).

Os sensores são dispositivos conectados aos equipamentos controlados e monitorados pelos sistemas SCADA, que convertem parâmetros físicos tais como velocidade, nível de água e temperatura, para sinais analógicos e digitais legíveis pela estação remota. Os atuadores são utilizados para atuar sobre o sistema, ligando e desligando determinados equipamentos.

O processo de controle e aquisição de dados se inicia nas estações remotas, PLCs (Programmable Logic Controllers) e RTUs (Remote Terminal Units), com a leitura dos valores atuais dos dispositivos que a ele estão associados e seu respectivo controle. Os PLCs e RTUs são unidades computacionais específicas, utilizadas nas instalações fabris (ou qualquer outro tipo de instalação que se deseje monitorar) para a funcionalidade de ler entradas, realizar cálculos ou controles, e atualizar saídas. A diferença

entre os PLCs e as RTUs é que os primeiros possuem mais flexibilidade na linguagem de programação e controle de entradas e saídas, enquanto as RTUs possuem uma arquitetura mais distribuída entre sua unidade de processamento central e os cartões de entradas e saídas, com maior precisão e seqüenciamento de eventos.

A rede de comunicação é a plataforma por onde as informações fluem dos PLCs/RTUs para o sistema SCADA e, levando em consideração os requisitos do sistema e a distância a cobrir, pode ser implementada através de cabos Ethernet, fibras ópticas, linhas dial-up, linhas dedicadas, rádio modems, etc.

As estações de monitoração central são as unidades principais dos sistemas SCADA, sendo responsáveis por recolher a informação gerada pelas estações remotas e agir em conformidade com os eventos detectados, podendo ser centralizadas num único computador ou distribuídas por uma rede de computadores, de modo a permitir o compartilhamento das informações coletadas.

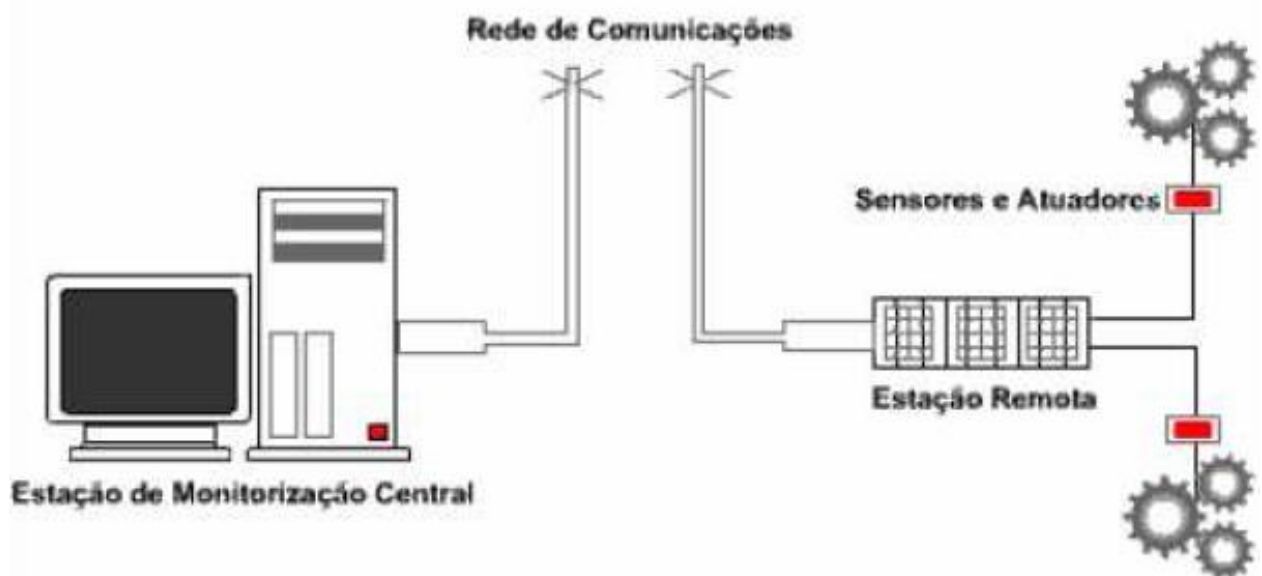


Figura 15. Sistema de supervisão e controle

MODOS DE COMUNICAÇÃO

A principal funcionalidade de qualquer sistema SCADA está ligada à troca de informações, que podem ser, basicamente:

- Comunicação com os PLCs/RTUs;
- Comunicação com outras estações SCADA;
- Comunicação com outros sistemas.

A comunicação com os equipamentos de campo, realizada através de um protocolo em comum, cuja metodologia pode ser tanto de domínio público ou de acesso restrito, geralmente pode ocorrer por polling ou por interrupção, normalmente designada por Report by Exception.

A comunicação por polling (ou Master/Slave) faz com que a estação central (Master) tenha controle absoluto das comunicações, efetuando seqüencialmente o polling aos dados de cada estação remota (Slave), que apenas responde à estação central após a recepção de um pedido, ou seja, em half-duplex. Isto traz simplicidade no processo de coleta de dados, inexistência de colisões no tráfego da rede, facilidade na detecção de falhas de ligação e uso de estações remotas não inteligentes. No entanto, traz incapacidade de comunicar situações à estação central por iniciativa das estações remotas.

Já a comunicação por interrupção ocorre quando o PLC ou o RTU monitora os seus valores de entrada e, ao detectar alterações significativas ou valores que ultrapassem os limites definidos, envia as informações para a estação central. Isto evita a transferência de informação desnecessária, diminuindo o tráfego na rede, além de permitir uma rápida detecção de informação urgente e a comunicação entre estações remotas (slave-to-slave). As desvantagens desta comunicação são que a estação central consegue detectar as falhas na ligação apenas depois de um determinado período (ou seja, quando efetua polling ao sistema) e são necessários outros métodos (ou mesmo ação por parte do operador) para obter os valores atualizados.

A comunicação com outras estações SCADA pode ocorrer através de um protocolo desenvolvido pelo próprio fabricante do sistema SCADA, ou através de um protocolo conhecido via rede Ethernet TCP/IP, linhas privativas ou discadas.

A Internet é cada vez mais utilizada como meio de comunicação para os sistemas SCADA. Através do uso de tecnologias relacionadas com a Internet, e padrões como Ethernet, TCP/IP, HTTP e HTML, é possível acessar e compartilhar dados entre áreas de produção e áreas de supervisão e controle de várias estações fabris. Através do uso de um browser de Internet, é possível controlar em tempo real, uma máquina localizada em qualquer parte do mundo.

O browser comunica com o servidor web através do protocolo http, e após o envio do pedido referente à operação pretendida, recebe a resposta na forma de uma página HTML.

Algumas das vantagens da utilização da Internet e do browser como interface de visualização SCADA são o modo simples de interação, ao qual a maioria das pessoas já está habituada, e a facilidade de manutenção do sistema, que precisa ocorrer somente no servidor. Já a comunicação com outros sistemas, como os de ordem corporativa, ou simplesmente outros coletores ou fornecedores de dados, pode se dar através da implementação de módulos específicos, via Bancos de Dados, ou outras tecnologias como o XML e o OPC.

OPC

Historicamente os integradores de sistemas tinham que implementar interfaces proprietários ou personalizados para extrair dados de dispositivos provenientes de diferentes produtores de hardware.

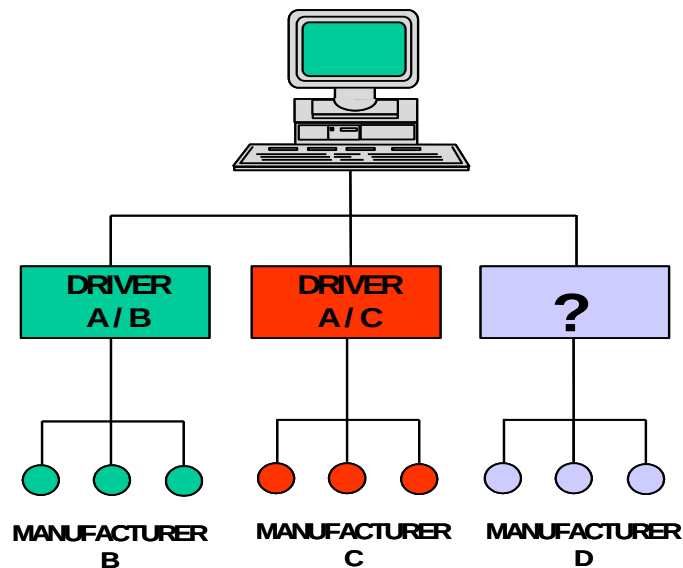


Figura 16. Utilização de driver de comunicação

Há alguns anos atrás a Microsoft introduziu as tecnologias OLE, COM e DCOM, permitindo às aplicações interoperar e comunicar com módulos distribuídos através de uma rede de computadores.

Com o objetivo de definir um standard para utilização das tecnologias OLE e COM em aplicações de controle de produção, os principais fabricantes de hardware e software constituíram uma organização, a OPC Foundation, da qual resultou o OPC (OLE for Process Control).

Atualmente está disponível uma API (Application Programming Interface) standard que permite a criação de aplicações que comuniquem com diferentes dispositivos. As vantagens do uso do OPC, entre outras, são as seguintes:

- Existência de uma única API para todos os servidores de OPC, de modo a que o código de uma aplicação cliente possa ser reutilizado em qualquer dispositivo;
- Oportunidade para desenvolver aplicações clientes em ambientes de desenvolvimento que utilizem COM e ActiveX, tais como Visual Basic, Visual C++ e Excel;
- Identificar os servidores que podem disponibilizar aos clientes determinados itens OPC. Um item OPC é um canal ou variável num dispositivo - normalmente um ponto de I/O - que um servidor monitoriza ou controla;
- Acesso distribuído e remoto através do DCOM

Um dos grandes problemas de se interfacear equipamentos e sistemas no chão de fábrica reside em se compatibilizar os protocolos da camada de aplicação. O MMS - Manufacturing Message Specification foi uma tentativa de padronização que, entretanto, fracassou por falta de adeptos. O padrão OPC foi inicialmente liderado pela Microsoft e especificado pela OPC Foundation. Este protocolo é hoje o padrão de fato da indústria. Imagine como faríamos para interfacear um sistema SCADA com um CLP há alguns anos

atrás. Imagine que este supervisor fosse o Factory Link da US Data e que o CLP fosse uma CPU da família 5 da Rockwell. O Factory Link era fornecido em várias versões, para diversos sistemas operacionais. O CLP 5 pode se comunicar com diversas redes diferentes, por exemplo com uma rede DH+. O PC pode utilizar cartões de comunicação Rockwell,

Sutherland-Schultz ou outro. O número de combinações é muito grande. Na prática, nós teríamos que utilizar ou desenvolver um driver que atendesse perfeitamente à combinação: Sistema SCADA (existem dezenas) / sistema operacional (várias opções), cartão de comunicação PC/CLP (várias fontes e possibilidade de rede). Isto implicava na existência de centenas de drivers de comunicação, que só atendiam a versões específicas da combinação de fatores apresentada acima. O protocolo OPC elimina esta situação. Um fabricante de CLP sempre fornecerá com o seu equipamento um servidor OPC. O fabricante de SCADA também fornecerá o cliente OPC. O mesmo acontece com um fornecedor de inversores, de relés inteligentes ou de qualquer outro dispositivo industrial inteligente. Um sistema SCADA também pode oferecer um servidor OPC para comunicação com outro sistema de aquisição de dados, por exemplo, um PIMS.

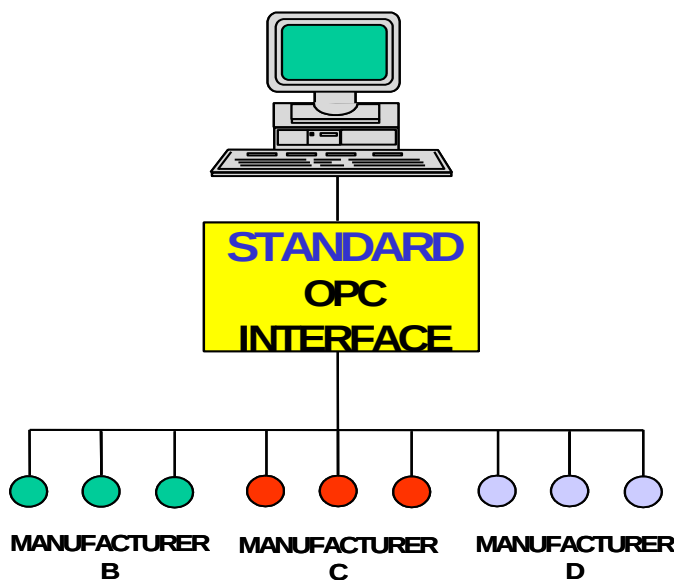


Figura 17. Utilização do OPC na comunicação

Como as aplicações precisam apenas saber como buscar dados de um servidor OPC, ignorando a implementação do dispositivo e o servidor precisa fornecer dados em um formato único: servidor OPC, a tarefa de escrever drives de comunicação fica muito facilitada.

Quais os tipos de dados providos pelo servidor OPC? O servidor OPC fornece dados de tempo real proveniente de sensores (temperatura, pressão, etc.), comandos de controle (abrir, fechar, ligar, desligar, etc.), status de comunicação, dados de performance e estatística do sistema, etc. O protocolo OPC é baseado no modelo de componentização criado pela Microsoft e denominado COM (Component Object Model), uma maneira eficiente de se estabelecer interfaces para aplicações que substitui as chamadas de procedimento e as DLL usadas inicialmente para encapsular uma aplicação. O nome OPC: OLE for Process Control foi cunhado na época em que o COM era um modelo embrionário de comunicação entre aplicativos como o nome de OLE (Object Linking and Embedding).

O padrão OPC é baseado em comunicações cíclicas ou por exceção. Cada transação pode ter de 1 a milhares de itens de dados, o que torna o protocolo muito eficiente, superando o MMS para aplicações práticas, segundo técnicos da divisão Powertrain da GM. O protocolo OPC não resolve o problema de nomes globais. Você deve saber exatamente em que servidor uma dada variável pode ser encontrada. As especificações do protocolo OPC estão disponíveis no sítio da OPC Foundation e incluem além da especificação básica para a construção de drives (OPC Data Access Specification – versão 2.05) outras especificações tais como padrão OPC para comunicação de alarmes e eventos (OPC Alarms and Events Specification - Versão 1.02), padrão OPC para dados históricos (OPC Historical Data Access Specification - Versão 1.01). padrão OPC para acesso de dados de processo em batelada (OPC Batch Specification - versão 2.00) e outros.

O servidor OPC é um objeto COM. Entre suas funções principais ele permite à aplicação cliente:

- Gerenciar grupos: Criar, clonar e deletar grupos de itens, renomear, ativar, desativar grupos.
- Incluir e remover itens em um grupo.
- Navegar pelas tags existentes (browser interface).
- Ver os atributos ou campos associado a cada tag.
- Definir a linguagem de comunicação (país) a ser usada.
- Associar mensagens significativas a códigos de erro
- Obter o status de funcionamento do servidor
- Ser avisada, caso o servidor saia do ar.

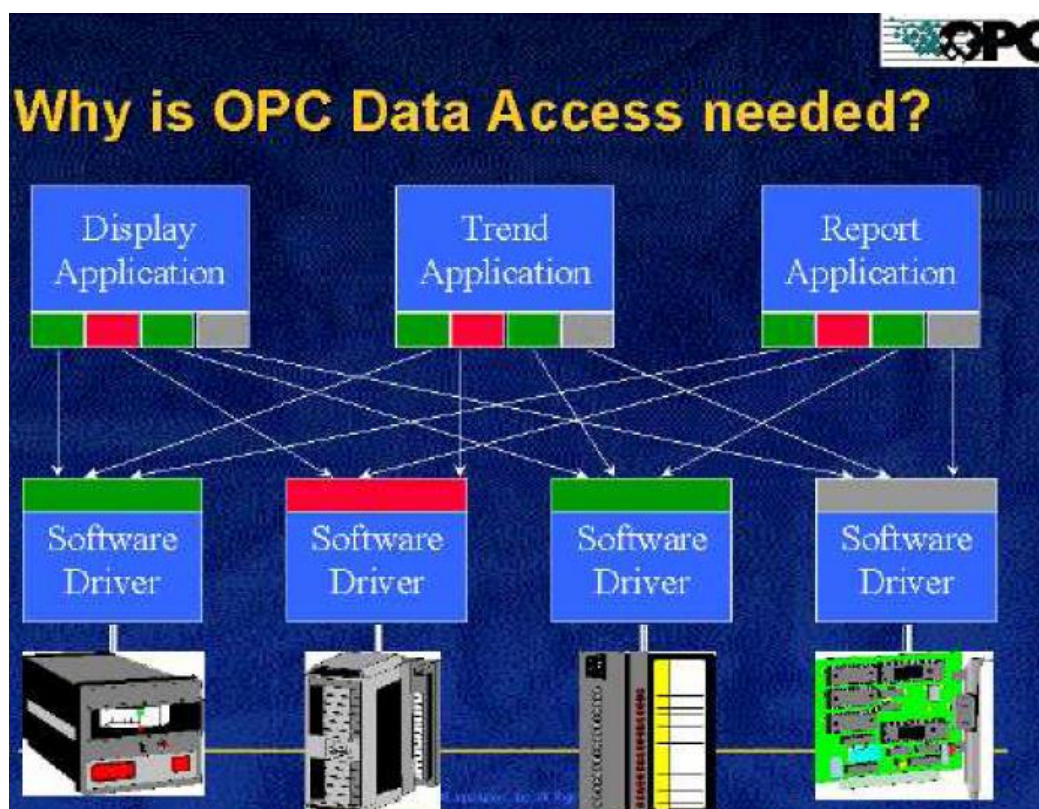


Figura 18. Acesso a dados de processo ponto a ponto antes do protocolo OPC

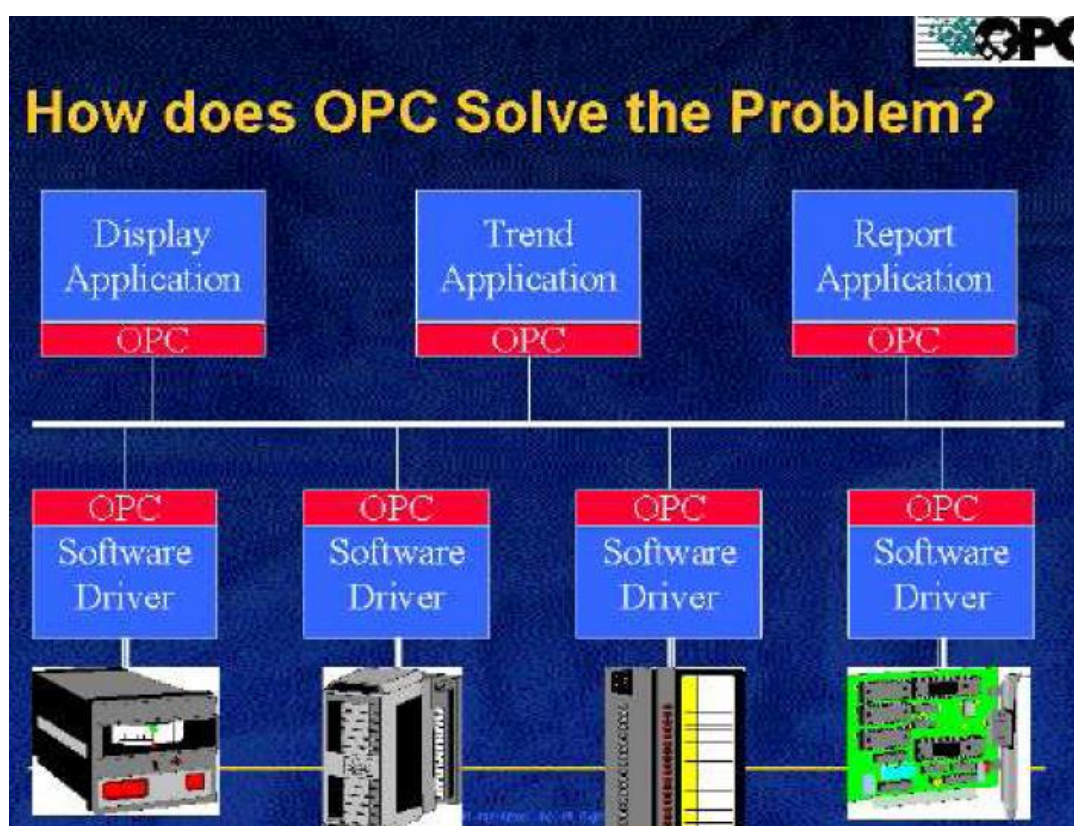


Figura 19. Acesso a dados de processo com protocolo OPC

O grupo de dados constitui uma maneira conveniente da aplicação organizar os dados de que necessita. Cada grupo de dados pode ter uma taxa de leitura específica: pode ser lida periodicamente (polling), ou por exceção. O grupo pode ser ativado ou desativado como um todo. Cada tela sinóptica, receita, relatório, etc., pode usar um ou mais grupos.

A interface de grupo permite à aplicação cliente:

- Adicionar e remover itens dos grupos.
- Definir a taxa de leitura do dado no grupo.
- Ler e escrever valores para um ou mais itens do grupo.
- Assinar dados do grupo por exceção.

Cada item é um objeto OPC que proporciona uma conexão com uma entrada física de dados. Cada item fornece ao cliente informação de: valor, time stamp, qualidade do dado e tipo de dado. É possível definir um vetor de objetos como um único item. Isto otimiza a comunicação de dados já que apenas, um time stamp e uma palavra de qualidade de dados é utilizada para cada conjunto de dados.

As leituras de dados podem ser de três tipos: leitura cíclica (polling), leitura assíncrona (o cliente é avisado quando a leitura se completa) e por exceção (assinatura). As duas primeiras trabalham sobre listas (subconjuntos) de um grupo e o serviço de assinatura envia aos clientes qualquer item no grupo que mudar de valor.

DDE/NETDDE

O DDE (Dynamic Data Exchange) é um protocolo cliente-servidor, rápido e confiável, que permite a transferência de dados entre aplicações através do uso de mensagens do Windows.

O cliente e o servidor podem ser programados para interpretar os dados como um comando. Para a troca de mensagens entre máquinas remotas, existe um mecanismo homólogo denominado NETDDE.

O DDE é totalmente bit blind, ou seja, nem o cliente nem o servidor sabem se estão a comunicar com uma aplicação de 16 ou 32 bits. Na realidade o servidor desconhece se o cliente se encontra na mesma máquina ou não.

O DDE através da rede - NETDDE (Network Dynamic Data Exchange) - usa uma hierarquia de nomes semelhante ao DDE, contudo, neste caso os nomes do serviço e tópico foram alterados, respectivamente, para servidor DDE e share, representando o primeiro o nome do computador que desempenha as funções de servidor e o segundo os nomes do serviço e tópico da aplicação servidora.

O NETDDE usa o protocolo NetBIOS, que corre sobre TCP/IP, permitindo ao NETDDE utilizar a Internet.

DCOM

O DCOM (Distributed COM), é uma extensão do COM (Component Object Model), para suporte de comunicação entre objetos existentes em diferentes computadores dispersos por LANs, WANS e Internet.

Devido ao fato de que o DCOM é uma evolução do COM, é possível utilizar os investimentos existentes em aplicações, componentes, ferramentas e conhecimentos baseados em COM, tirando partido de que o DCOM manipula os detalhes de baixo nível dos protocolos de rede.

Nos sistemas operativos atuais os processos correm isolados uns dos outros. Um cliente que precise comunicar com um componente noutro processo não pode fazê-lo diretamente, tendo que usar uma forma de comunicação entre processos disponibilizada pelo sistema operativo.

O COM disponibiliza esta comunicação interceptando e enviando a chamada do cliente para o componente no outro processo.

Quando o cliente e o componente residem em máquinas diferentes, o DCOM substitui a comunicação local entre processos com um protocolo de rede, sem que o cliente ou o objeto se perceba de que a ligação entre eles se tornou mais distante.

A tecnologia DCOM disponibiliza um mecanismo que permite de modo transparente para o utilizador a expansão para além de um único nó, de modo a tirar vantagem das capacidades de desenvolvimento e controlo distribuído.

ACTIVEX

O objetivo principal da programação orientada ao objeto é a economia de tempo através do desenvolvimento de objetos e bibliotecas de objetos reutilizáveis.

A principal diferença entre as bibliotecas de objetos e as tradicionais bibliotecas de funções é que os objetos, para além de conterem as funções que operam sobre os dados, contêm também os dados sobre os quais operam, o que os torna auto-suficientes. As linguagens orientadas a objeto, nas quais os objetos são geralmente referidos como classes, suportam o encapsulamento, a herança e o polimorfismo.

O encapsulamento significa que os dados do objeto não estão acessíveis diretamente a partir do exterior, mas sim através de métodos - funções contidas no próprio objeto que permitem aceder e manipular os seus dados - que garantem a proteção do objeto contra acessos inapropriados.

A herança permite a criação de novos objetos - filhos - com as mesmas características, ou algumas delas, de outros objetos já existentes - pais -, ou seja, ela permite herdar estruturas de dados e funções de objetos e acrescentar-lhe novas estruturas de dados e funções.

Uma vantagem da herança entre objetos é que qualquer alteração efetuada a um objeto é automaticamente propagada a todos os objetos derivados a partir dele.

O polimorfismo permite que diferentes objetos com funções semelhantes sejam tratados da mesma maneira, ou seja, dados dois objetos diferentes que suportem, por exemplo, um método chamado start, as suas respostas à invocação desse método serão diferentes em função da respectiva implementação no código do objeto.

Os objetos ActiveX são objetos baseados na tecnologia COM, que estabelece um paradigma comum para interação entre componentes de software dentro do ambiente Microsoft Windows.

A vantagem do modelo COM é que permite aos objetos interagir com outros objetos dentro e fora do mesmo programa.

ARQUITETURAS DE SISTEMAS DE AUTOMAÇÃO

O primeiro passo ao se conceber uma solução qualquer de automação é desenhar a arquitetura do sistema, organizado seus elementos vitais: remotas de aquisição de dados, PLCs, instrumentos, sistema de supervisão, etc. em torno de redes de comunicação de dados apropriadas. A escolha da arquitetura irá determinar o sucesso de um sistema em termos de alcançar os seus objetivos de desempenho, modularidade, expansibilidade etc.

As soluções irão depender das limitações de cada projeto em particular. Existem vários pontos que o projetista deve verificar ao iniciar o projeto. O melhor é estabelecer um check list de pontos importantes a serem verificados:

- Quantas são as áreas de processo ? Quais as distâncias entre as áreas ? Qual o lay out da instalação industrial?
- Haverá uma sala de controle centralizada ou apenas púlpitos de comando locais?
- Existe necessidade de um sistema de backup ? A que nível ?
- Quais são as condições ambientais ? Existe campo magnético intenso nas proximidades ? Existe interferência eletromagnética ?
- O cliente está familiarizado com novas tecnologia de redes de campo para instrumentação, sensores e acionamentos ?
- Existem sites fora da área industrial que devam ser conectados à planta ? Escritório central, centros de distribuição, unidades remotas como britagens móveis, instalações portuárias, etc. ?
- Quais as necessidades dos dispositivos em termos da velocidade de transmissão de dados ?
- Qual a capacidade de expansão dos módulos prevista para os próximos anos ?

- Existe preferência quanto ao atendimento aos padrões internacionais ou preferência por redes proprietárias ?

Uma das arquiteturas mais praticadas é a que define duas hierarquias de redes: uma rede de informação e uma rede de controle.

REDE DE INFORMAÇÃO

O nível mais alto dentro de uma arquitetura é representado pela rede de informação. Em grandes corporações é natural a escolha de um backbone de grande capacidade para interligação dos sistemas de ERP (Enterprise Resource Planning), Supply Chain (gerenciamento da cadeia de suprimentos), e EPS (Enterprise Production Systems). Este backbone pode ser representado pela rede ATM ou GigaEthernet ou mesmo por uma Ethernet 100-BaseT, utilizando como meio de transmissão cabo par trançado nível 5. Esta última rede vem assegurando uma conquista de espaço crescente no segmento industrial, devido à sua simplicidade e baixo custo.

REDE DE CONTROLE

Interliga os sistemas industriais de nível 2 ou sistemas SCADA aos sistemas de nível 1 representados por CLPs e remotas de aquisição de dados. Também alguns equipamentos de nível 3 como sistemas PIMS e MES podem estar ligados a este barramento. Até dois anos atrás o padrão mais utilizado era o Ethernet 10Base-T. Hoje o padrão mais recomendado é o Ethernet 100Base-T. Quase todos os grandes fabricantes de equipamentos de automação já possuem este padrão implementado.

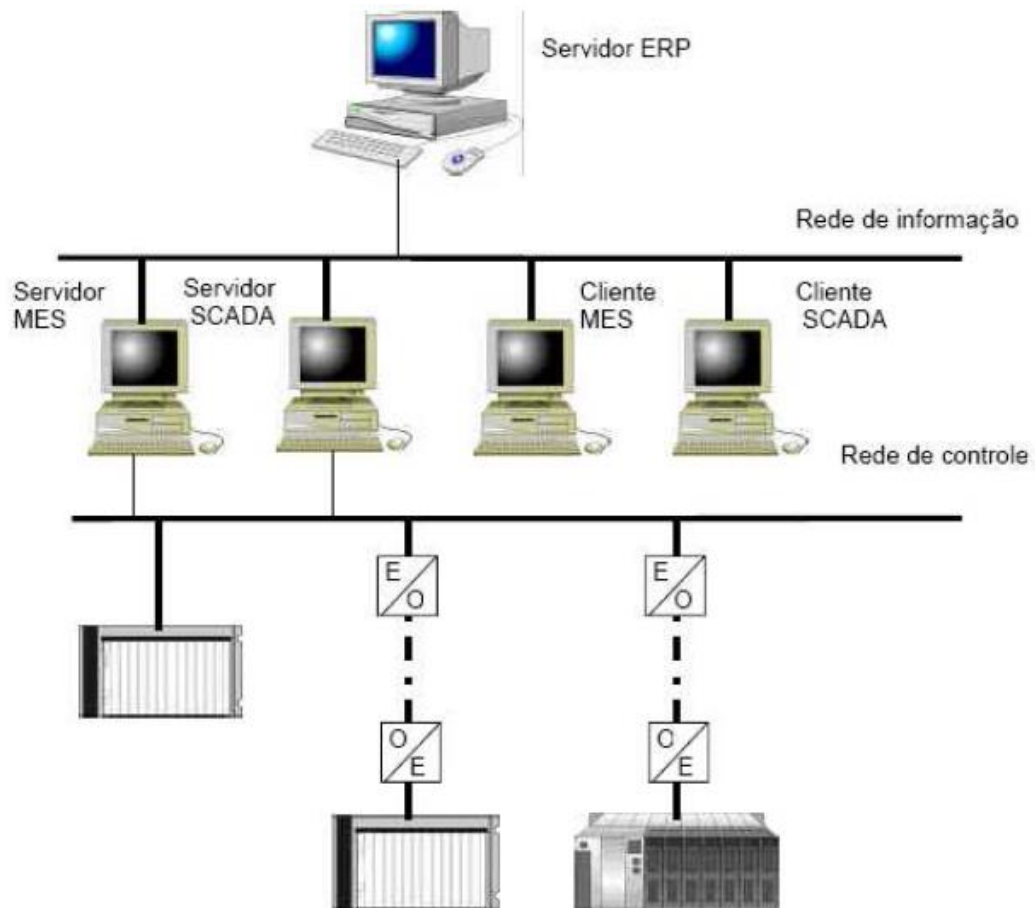


Figura 20. Arquitetura de uma rede de duas camadas - Unifilar

As estações clientes se comunicam com seus servidores através da rede de informação. As estações Servidores se comunicam com os CLPs através da rede de controle.

Do ponto de vista de segurança, é interessante isolar o tráfego de controle do tráfego de informação através de equipamentos de rede. Hoje o equipamento mais utilizado para este fim é o switch Ethernet e o padrão mais utilizado é o 100Base-T. Além de evitar os problemas de divisão de banda, típico da arquitetura barramento, o switch segmenta a rede. O switch assegura a criação de uma rede Ethernet livre de colisões. Esta nova concepção de rede é denominada de rede Ethernet Industrial

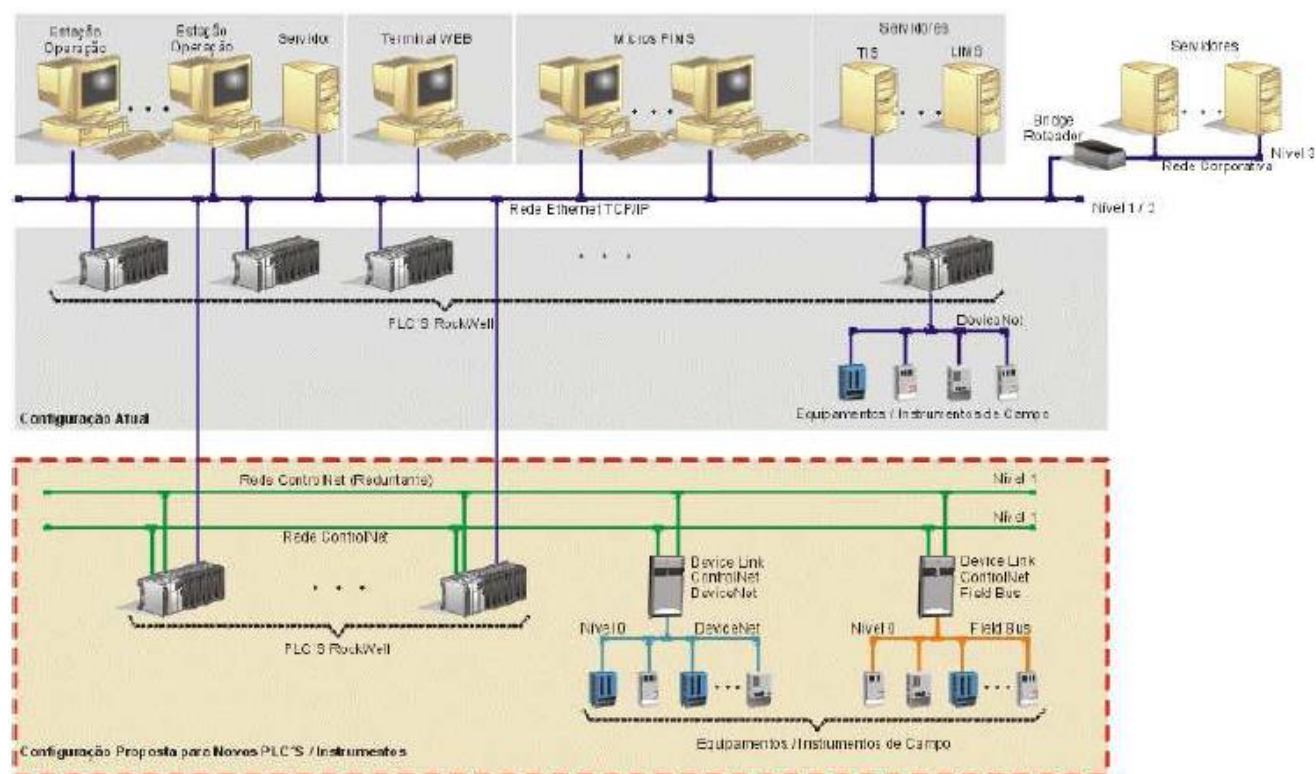


Figura 21. Arquitetura típica de uma rede de automação de múltiplas camadas

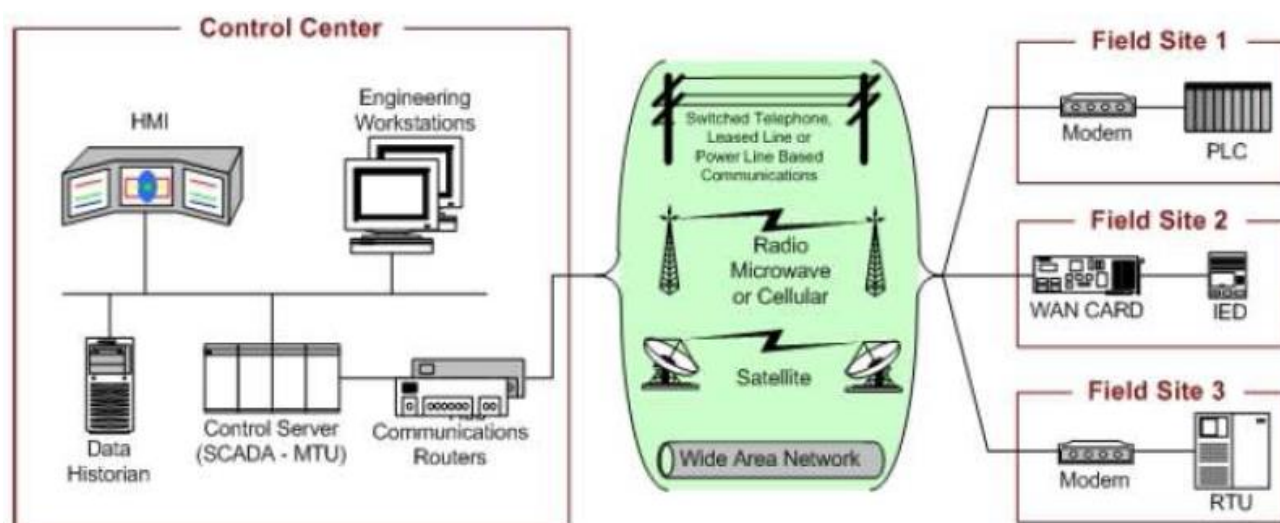


Figura 22. Arquitetura típica de uma rede de automação com supervisão remota

EQUIPAMENTOS DE INTEGRAÇÃO DE REDES.

- **Repetidor (Repeater):** Dispositivo não inteligente que simplesmente copia dados de uma rede para outra, fazendo que as duas redes se comportem logicamente como uma rede única. São usados para satisfazer restrições quanto ao comprimento do cabo, por exemplo.
- **Distribuidores de conexão (Hubs):** Servem para conectar os equipamentos que compõem uma LAN. Os equipamentos interligados a um hub pertencem a um mesmo segmento de rede, isto é os usuários do hub dividem a largura da banda. Se tivermos 10 usuários em um segmento de 100Mbps, cada usuário usufruirá em média de 10Mbps de banda de passagem. Cada hub possui de 4 a 24 portas 10Base-T com conectores RJ-45.
- **Ponte (Bridge):** Segmenta uma rede local em sub-redes com o objetivo de reduzir tráfego ou converter diferentes padrões de camadas de enlace (Ethernet para Token Ring por exemplo)
- **Roteador (Router):** Usado para interligar duas redes que possuem a mesma camada de transporte, mas camadas de rede diferentes. Os roteadores decidem sobre qual caminho o tráfego de informações (controle e dados) deve seguir.
- **Comporta (Gateway):** Usado para dar acesso à rede a um dispositivo não OSI.
- **Switch:** São os dispositivos de mais amplo espectro de utilização, para segmentar a rede a baixo custo, sem necessidade de roteamento. Sua maior limitação está em não permitir broadcasting entre segmentos.

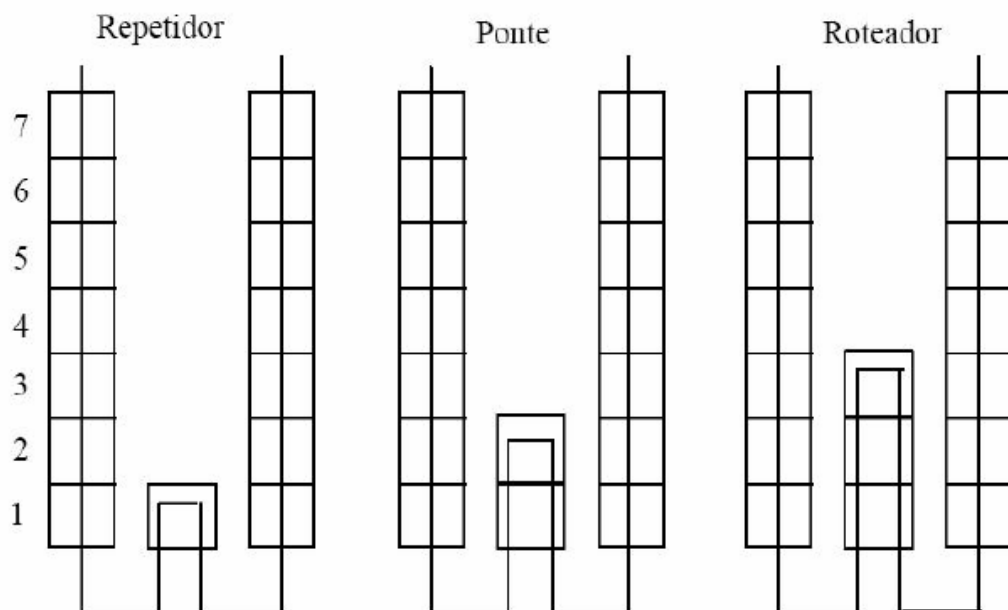


Figura 23. Conexão entre redes

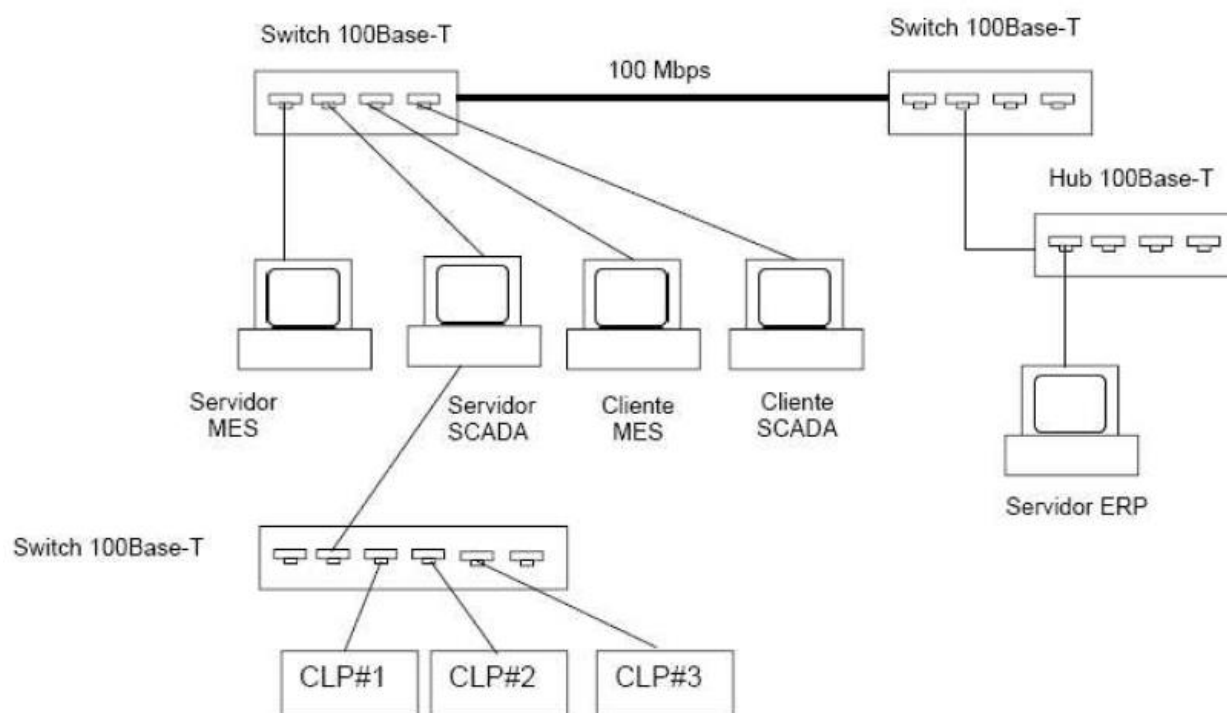


Figura 24. Arquitetura de uma rede de duas camadas (incluindo equipamentos)

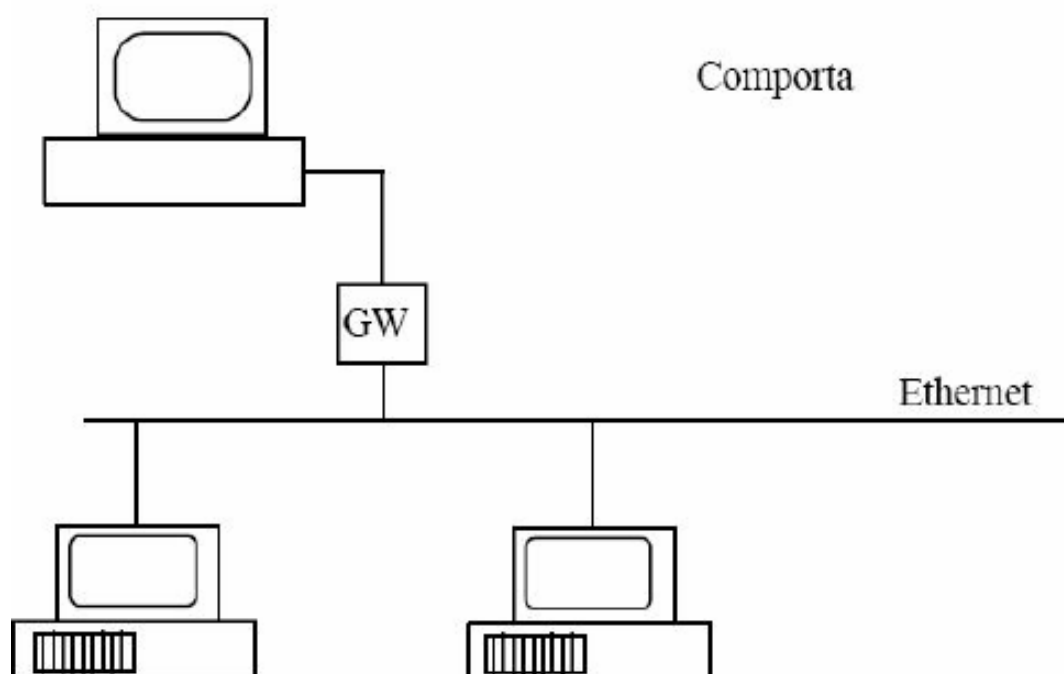


Figura 25. Uso de gateway

ARQUITETURA DE REDE ÚNICA.

As redes de controle e informação também podem estar fundidas em uma rede única. Esta topologia apresenta os seguintes inconvenientes:

O tráfego na rede de controle é de natureza diversa do tráfego na rede de informação, caracterizando-se por mensagens curtas e muito freqüentes.

O tráfego da rede de informação é em geral representado por arquivos maiores transmitidos com baixa freqüência.

Os requisitos de performance e segurança das duas redes também são diferentes.

Embora este tipo de topologia seja muito utilizado, a topologia anterior é mais recomendada por segmentar cada tipo de tráfego.

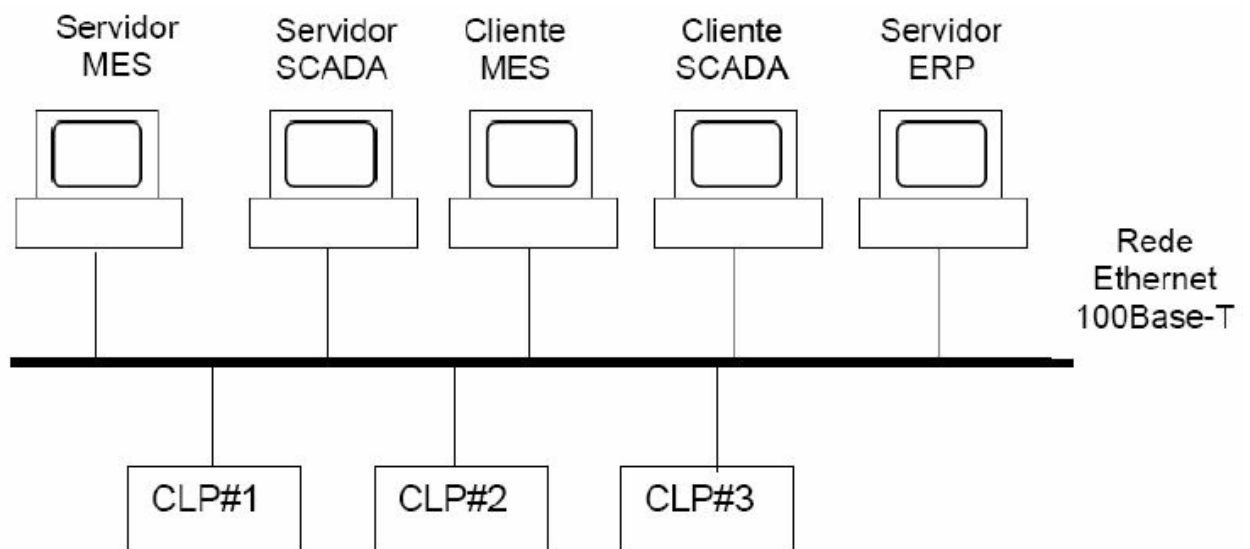


Figura 26. Barramento único

REDES PROPRIETÁRIAS

Apesar da rede Ethernet ser hoje a preferida da área industrial, muitas redes proprietárias de concepção mais antigas são ainda muito usadas. Por exemplo, a rede DH+ (Data Highway plus) da Rockwell.

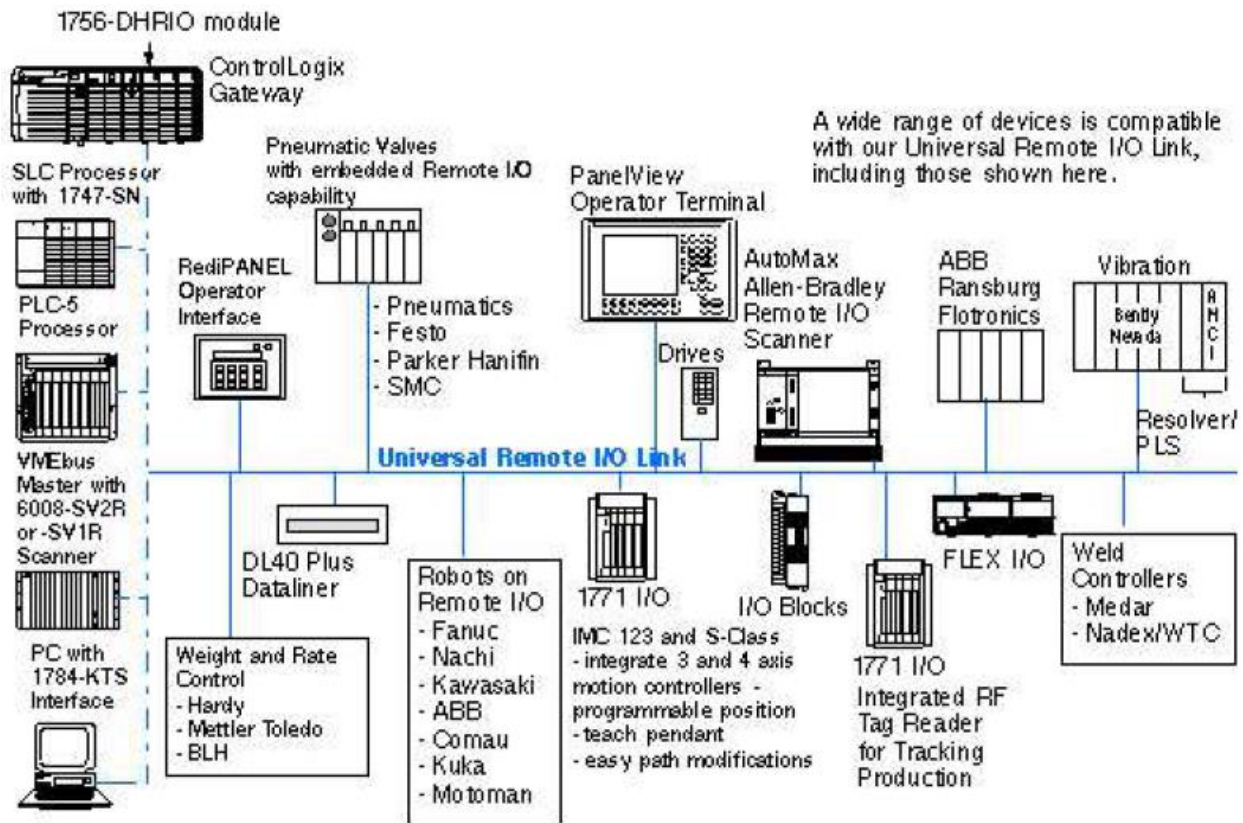


Figura 27. Exemplo de rede proprietária de nível 1: Remote IO da Rockwell

ARQUITETURA DE UM SDCCD

Os SDCCDs tradicionais se caracterizavam por um elevado nível de redundância: redundância de servidores, redundância de rede de comunicação de dados, de cartões de entrada e saída, etc. Além disso, possui sofisticados algoritmos de diagnóstico, que permitem localizar o cartão defeituoso a partir da console de operação. OS cartões de E/S com defeito podem ser trocados a quente.

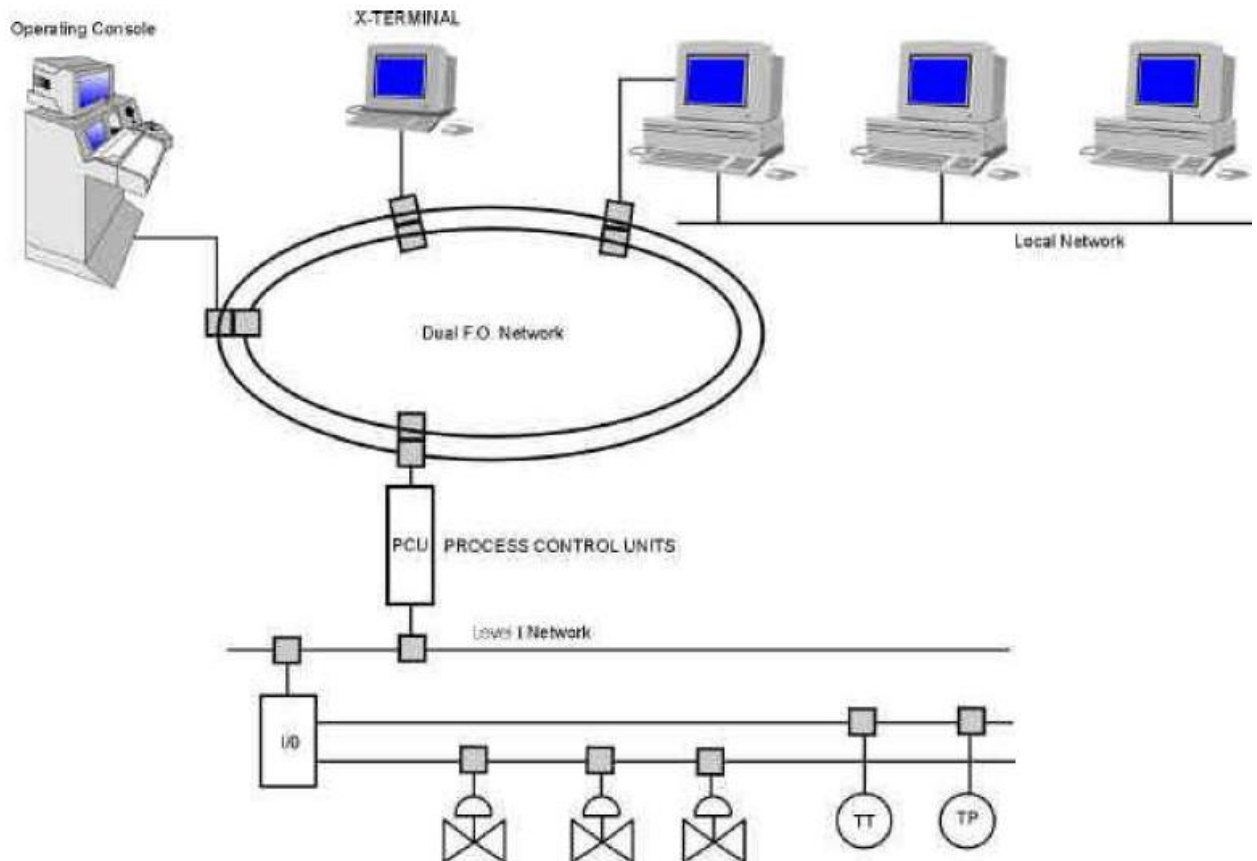


Figura 28. Configuração de um SDCD

SISTEMAS HÍBRIDOS

Atualmente os SDCDs têm um sucesso peso leve. Conhecido como sistema híbrido, este novo sistema alia a versatilidade e performance de um SDCD com o baixo custo de uma solução SCADA + CLP. São exemplos desta classe de sistema o INDUSTRIALIT (OPERATEIT/ CONTROLIT) da ABB, o Plant Web da Emerson e o PSC7 da Siemens.

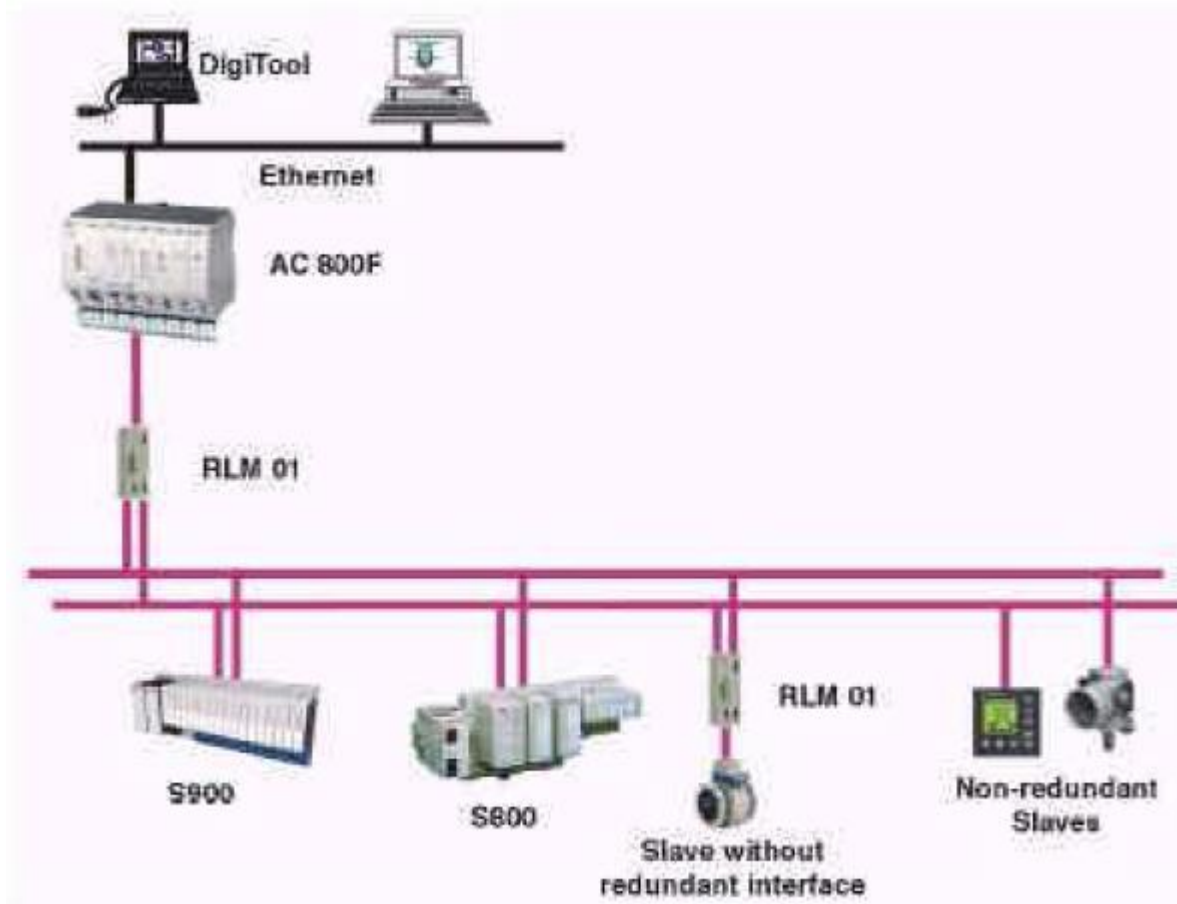


Figura 29. Sistema híbrido da ABB com rede Profibus redundante

REQUISITOS DESEJADOS PARA A ARQUITETURA

Ao se fazer um projeto de automação, estamos interessados em requisitos de desempenho tais como:

Redes de sensores e comandos:

- Baixo tempo de varredura para leitura cíclica de todos os dispositivos em rede.
- Determinismo no tempo.
- Exigência para algumas aplicações de uma rede com topologia em anel com self-healing ou outro tipo de redundância, de tal forma que um cabo partido não interrompa todo o circuito.
- Capacidade de comandar vários dispositivos sincronicamente (multicasting, broadcasting).
- Capacidade de expansão futura.
- Obediência a padrões internacionais (padrão aberto).

- Facilidade de aquisição de instrumentos de múltiplas fontes.
- Possibilidade de utilização de meio óptico para uso externo, fora das salas elétricas.
- Disponibilidade de sensores/atuadores do tipo desejado na aplicação dentro da tensão desejada (média tensão/ baixa tensão).

No caso de uma aplicação de CCM inteligente, o número máximo de nós suportados e o tempo máximo de resposta são quesitos de máxima importância.

Redes de dados (informação e controle):

- Padrão internacional e de fato (Ethernet 100Base-T é a melhor referência).
- Alto desempenho medido pela velocidade de transmissão.
- Capacidade de expansão futura
- Possibilidade de uso de fibra ótica
- Possibilidade de adição de redundância.

REDES DE CAMPO

Os CLPs são usados para ler os sensores discretos ou digitais e os valores dos instrumentos analógicos.

Caso uma rede digital não seja usada, os sinais de campo serão conectados aos cartões de entrada e saída dos CLPs. Os sinais discretos são codificados na faixa de 0 a 24VCC ou 0-110VAC ou 0-220VAC. Já os sinais analógicos são geralmente codificados na faixa de 0 a 20 mA ou de 0-10V.

Outra alternativa é o uso de uma rede digital de instrumentos e sensores. Este tipo de rede atende pelo nome genérico de fieldbus ou barramento de campo. Na verdade, devemos dividir estes tipos de rede em 3 tipos diferentes:

Redes de sensores ou Sensorbus - são redes apropriadas para interligar sensores e atuadores discretos tais como chaves limites (limit switches), contactores, desviadores, etc. São exemplos de rede Sensorbus: ASI da Siemens, Seriplex, CAN e LonWorks.

Redes de Dispositivos ou Devicebus - são redes capazes de interligar dispositivos mais genéricos como CLPs, outras remotas de aquisição de dados e controle, conversores AC/DC, relés de medição inteligentes, etc.

Exemplos: Profibus-DP, DeviceNet, Interbus-S, SDS, LonWorks, CAN, ControlNet, ModbusPlus.

Redes de instrumentação ou fieldbus - São redes concebidas para integrar instrumentos analógicos no ambiente industrial, como transmissores de vazão, pressão, temperatura, etc, válvulas de controle, etc.

Exemplos: IECSP50-H1, HART, WorldFIP, Profibus-PA.

Padronização internacional

Hoje o standard Fieldbus está padronizando 8 famílias de redes:

1. Fieldbus (mesmo que ANSI/ISA S50.01)
2. ControlNet
3. Profibus
4. P-NET
5. Fieldbus Foundation HSE
6. SwiftNet
7. WorldFIP
8. Interbus

Existe forte tendência de todas estas redes venham a utilizar a Ethernet como plataforma básica num futuro próximo e que adotem forte orientação a objetos através de blocos de função padrões.

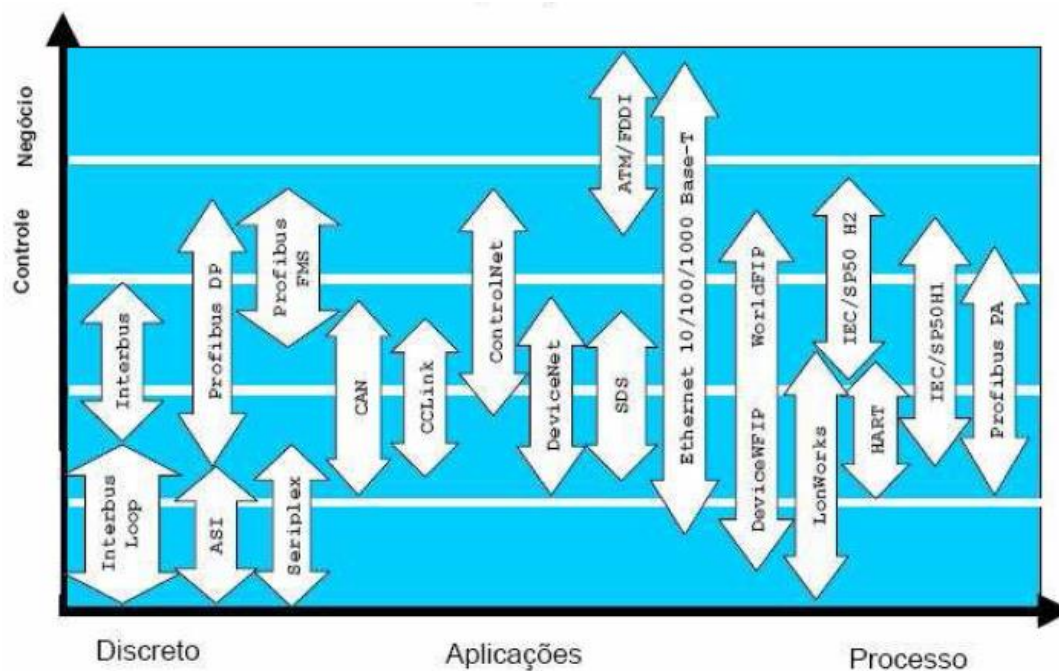


Figura 30. Faixa de aplicação das redes de campo segundo ARC

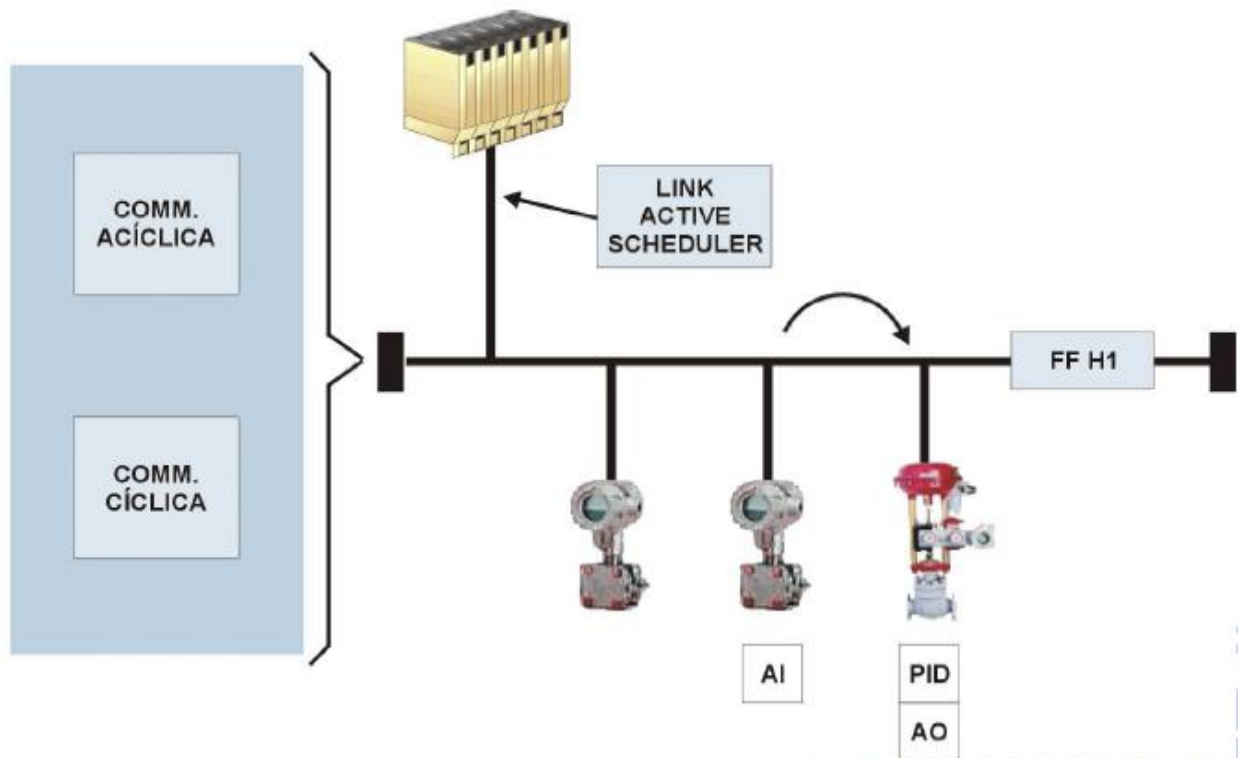


Figura 31. Uso de Redes de Instrumentos de Campo – Foundation Fieldbus

O tempo de resposta da rede depende de vários fatores, entre eles: a velocidade de transmissão dos dados, que pode ser função do comprimento da rede, do número de nodos instalados, do tamanho e natureza dos bloco de dados, etc. Os fabricantes oferecem gráficos e tabelas que nos permitem estudar o desempenho de uma rede para uma determinada aplicação.

1º. LABORATÓRIO PARA CONFIGURAÇÃO DE SUPERVISÓRIO INDUSOFT

O objetivo deste trabalho prático será criar algumas telas para que tenhamos uma aplicação onde seja possível visualizar e comandar um processo de medição e controle de nível, temperatura e pressão em 3 tanques similares.

ETAPA 1. CRIAR UM NOVO PROJETO

Para criar um novo projeto primeiro vamos abrir o software. Duplo clique no ícone  (**IWS v8.0 Wonderware InduSoft Web Studio**), ou usando o botão “Iniciar”, “Programas” -> “Wonderware InduSoft Web Studio v8.0” -> “**Wonderware InduSoft Web Studio v8.0**”.

Já no ambiente do **InduSoft Web Studio**, clique no botão localizado no canto superior esquerdo da tela, clique em **Novo** para abrir a janela “Novo” ou aperte **CTRL + N** e faça como a seguir:

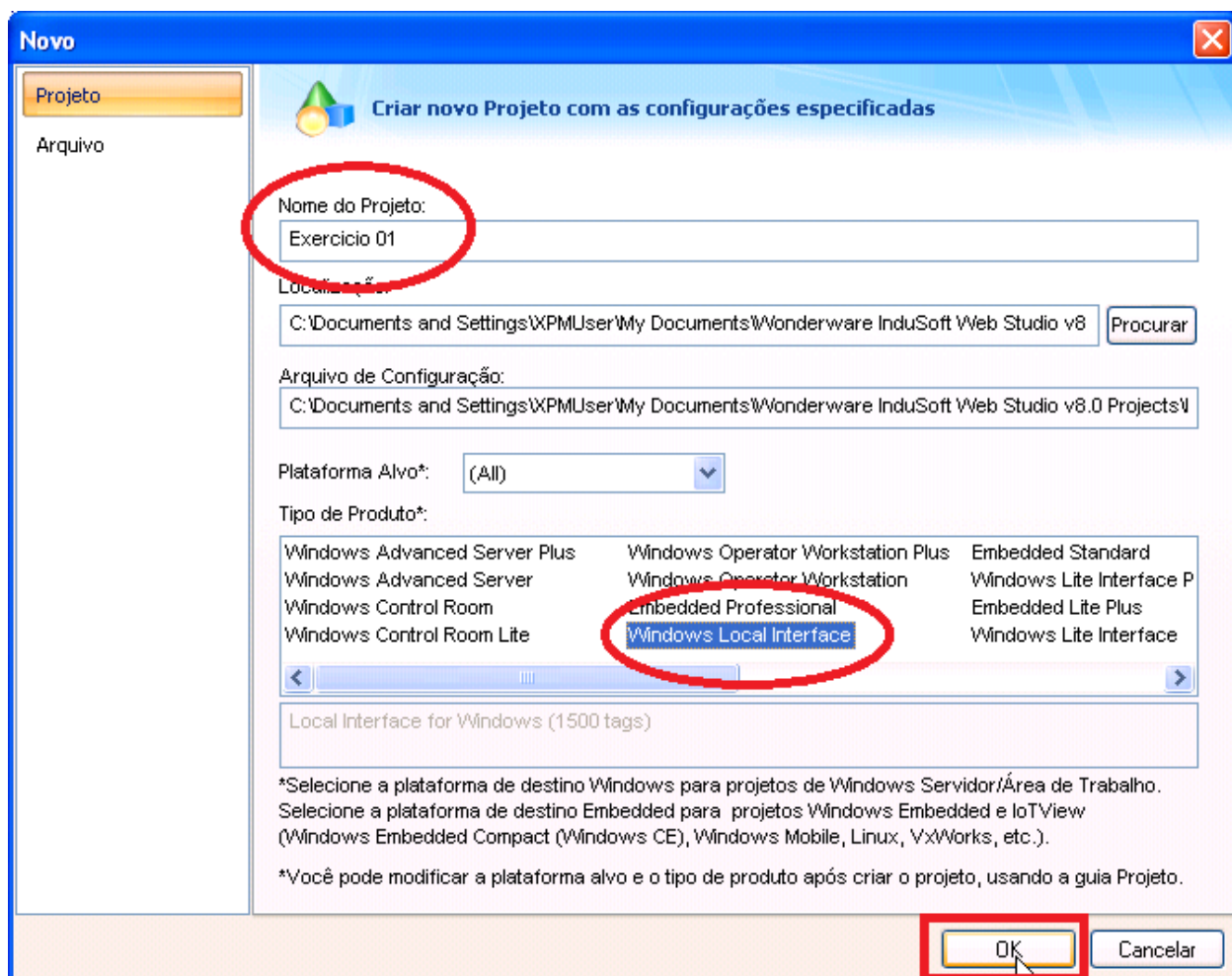


Figura 32.

Em “**Nome do Projeto**”, digite o nome desejado, exemplo: Projeto01; Exercício01.

Em “**Tipo de Produto**”, selecione “**Windows Local Interface**”, como mostrado na imagem acima.

Pressione o botão “**OK**” para abrir a janela seguinte

Na janela “**Assistente de Projeto**”, é possível selecionar algumas janelas que podem ser utilizadas como um ponto de partida para a aplicação.

Selecione “**Aplicação Vazia**” e escolha uma resolução pré-determinada pelo programa, como “**1280x800**”, ou uma resolução “**Customizado**”, onde é possível determinar a largura e altura desejada. Neste caso usaremos a resolução máxima que o monitor pode suportar para maximizar o espaço disponível, esta resolução é a “**1920x1078**”.

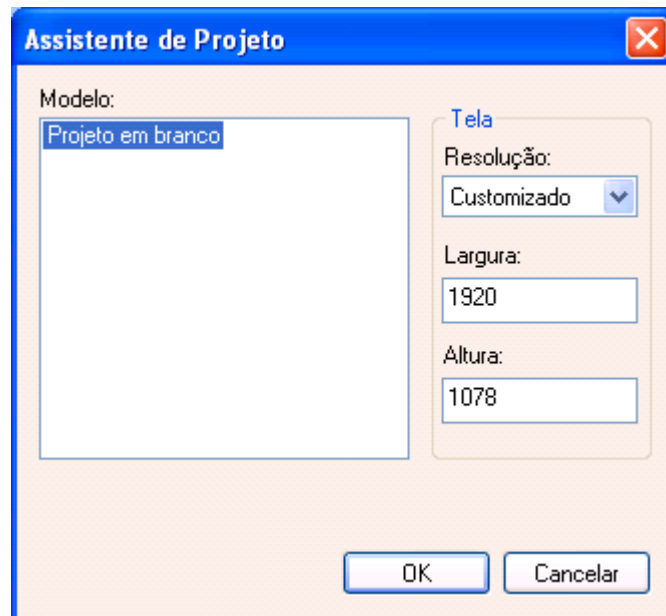


Figura 33.

ETAPA 2. CRIAR TAGS NA BASE DE DADOS “GLOBAL”

Na área de trabalho do programa, selecione a aba “**Global**”. Clique na pasta “**Tags do Projeto**” para expandi-la.

Clique duas vezes em “**Planilha de Dados**”.

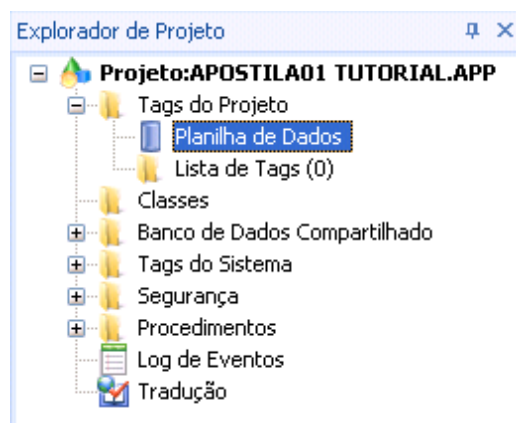


Figura 34.

A **aba Global** será preenchida e alterada durante o desenvolvimento da aplicação. Porém, para iniciarmos uma aplicação de exemplo, podemos já previamente definir alguns *tags* que sabemos que serão utilizados. Criaremos três Tanques cada qual controlado por duas válvulas. A primeira válvula enche o tanque e a segunda o esvazia.

Para sabermos o estado da primeira válvula, teremos o *tag* “**EstadoEnche**” e para a segunda “**EstadoEsvazia**”.

Como cada válvula só pode estar aberta ou fechada, estes *tags* devem ser **booleanos** (variáveis que só pode ser 0 ou 1).

Sendo três Tanques, precisaremos de três Válvulas para enchê-los e outras três para esvaziá-los. Portanto usaremos o tamanho **3** (na coluna “**Vetor**”) para rapidamente criarmos as válvulas.

Vamos configurar a “**Planilha de Dados**” como mostrado a seguir:

Crie os *tags* que enviarão os comandos para abrir ou fechar as válvulas, conforme mostrado a seguir.

	Nome	Vetor	Tipo	Descrição	Escopo
	Filtro de Texto	= ...	(Todos) ▼	Filtro de Texto	(T... ▼
1	ESTADOENCHE	3	Booleana ▼	VALVULA QUE ENCHE O TANQUE	Local ▼
2	ESTADOESVAZIA	3	Booleana ▼	VALVULA QUE Esvazia O TANQUE	Local ▼
3	COMANDOENCHE	3	Booleana ▼	COMANDO QUE ENCHE O TANQUE	Local ▼
4	COMANDOESVAZIA	3	Booleana ▼	COMANDO QUE Esvazia O TANQUE	Local ▼

Figura 35.

Esses comandos, na vida real, seriam enviados aos **CLPs** através do *driver*, endereçando esses *tags* para serem pontos de I/O (*input/output*).

Salve a configuração do banco de dados usando o ícone circulado na figura abaixo ou apertando **CTRL + S**.



Figura 36.

ETAPA 3. CRIAR TELAS NO PROJETO

A figura a seguir mostra uma das telas, no caso vemos a tela de nome Principal.

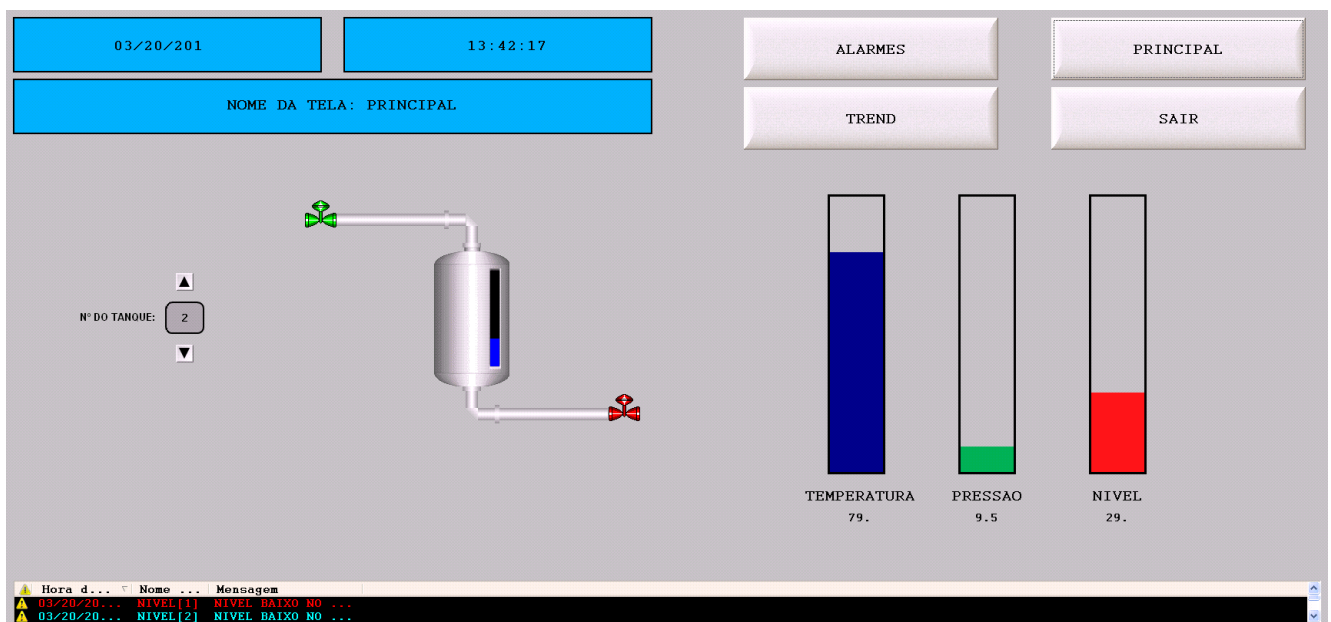


Figura 37.

Primeiramente criaremos o **CABEÇALHO** e o **RODAPÉ** em uma tela (**STANDARD**) que servirá de padrão básico para as demais telas.

Selecione a aba "**Gráficos**" na Área de Trabalho e selecione na pasta "**Telas**". Agora, clique com o botão direito na pasta "**Telas**" e clique na opção "**Inserir**" (Figura 38 e 39).

Na janela "**Atributos de Tela**" devemos configurar alguns atributos gerais sobre a tela que está sendo criada. Primeiro configure a tela nomeada como **STANDARD** como mostrado na Figura 38.



Figura 38.

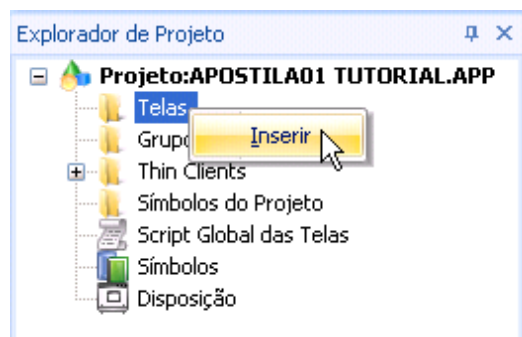


Figura 39.

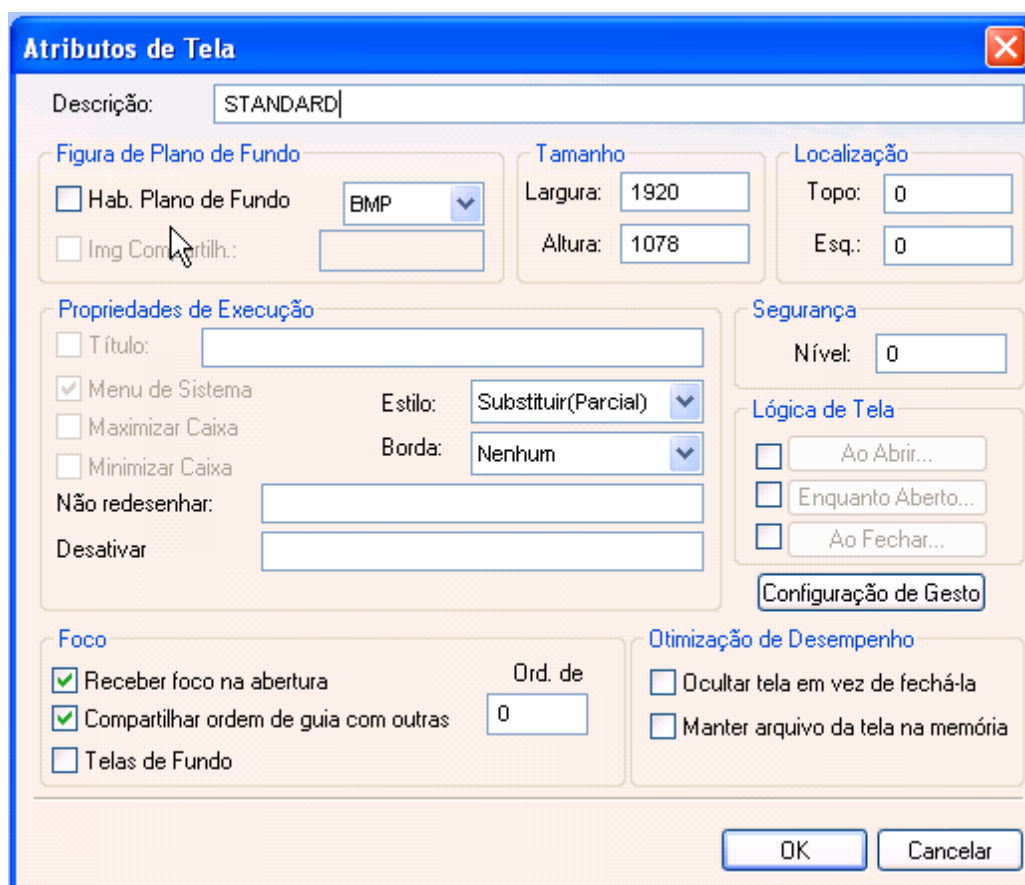


Figura 40.

Pressione o botão **OK** e uma tela vazia deverá aparecer com os atributos configurados para ela.

DESENHAR O CABEÇALHO (PARTE SUPERIOR DA TELA) E OS BOTÕES DE COMANDO

Depois de criar a tela, mudemos a cor de fundo usando o “Cor de Fundo”.

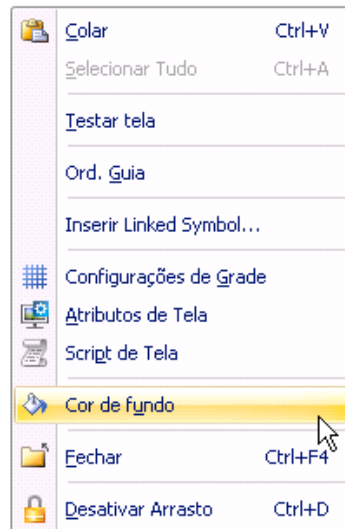


Figura 41.

Selecione o **cinza claro** nas opções de cores da janela aberta e clique em **OK**:

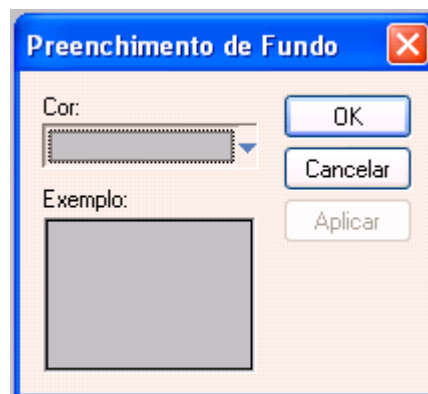


Figura 42.

Desenhe quatro botões no canto superior direito da tela.

Esses botões chamarão as outras telas e ainda permitirão fechar a aplicação. Para desenhar os botões, entre na aba “**Gráficos**” (Figura 43), e na janela de “**Objetos Ativos**” (Figura 44) clique em “**Botão**”, depois basta clicar e arrastar para projetar o botão do tamanho que desejar.



Figura 43.

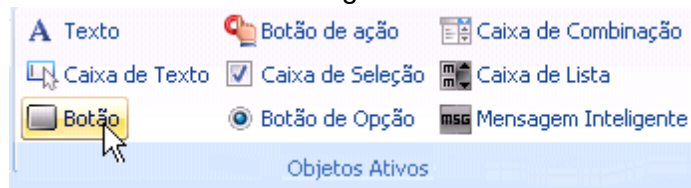


Figura 44.

Nota: É possível copiar e colar objetos das telas utilizando-se do recurso **Ctrl+C** **Ctrl+V**. Outra forma, aí exclusiva do InduSoft, é clicar no objeto a ser duplicado ao mesmo tempo que se pressiona a tecla **Ctrl**.

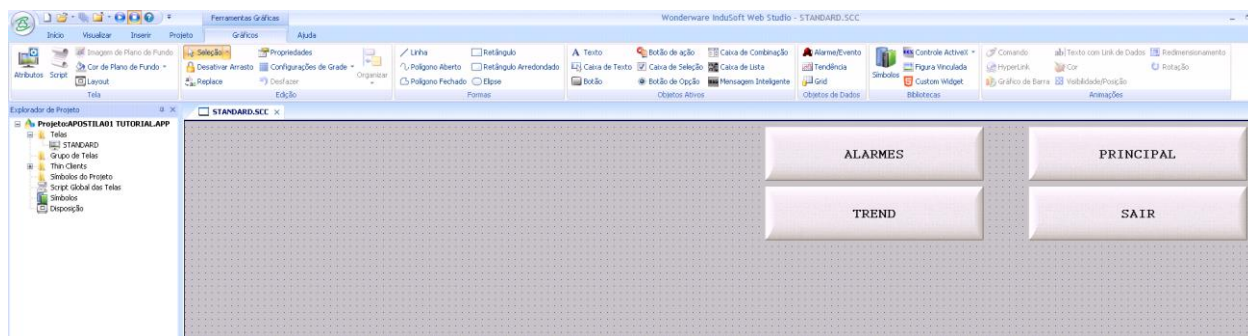


Figura 45.

Quando se clica duas vezes em um objeto desenhado, temos a janela de **“Propriedades do Objeto”**. Nesta janela são configuradas todas as propriedades, tais como comandos, animações, texto, etc. No caso dos botões (*pushbuttons*), o texto que aparecerá no botão dentro do projeto será determinado dentro de suas **“Propriedades do Objeto”** no campo de **“Legenda”**. Nomeie cada um como mostrado na Figura 46: Alarmes, Principal, Trend, Sair.

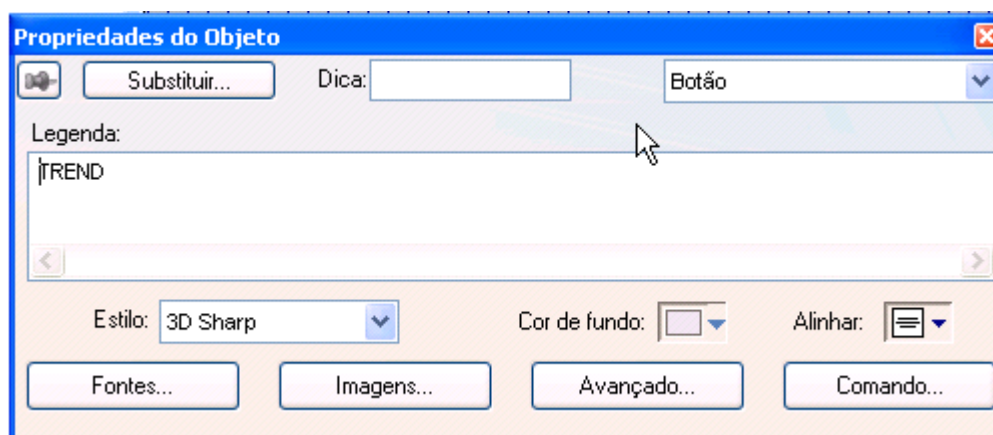


Figura 46.

Portanto, mãos a obra: crie quatro botões com as Legendas mostrados na tela abaixo:

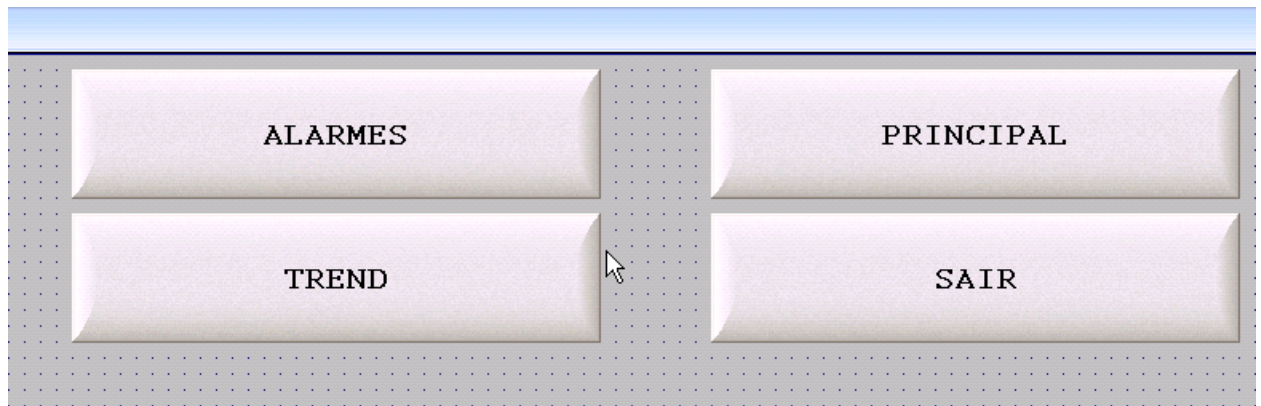


Figura 47.

Agora desenharemos alguns retângulos e textos que farão parte da tela no canto superior esquerdo. Esses campos mostrarão a data e horas atuais, bem como nome do usuário.

Desenhe agora três retângulos azuis, com contorno preto de espessura 2.

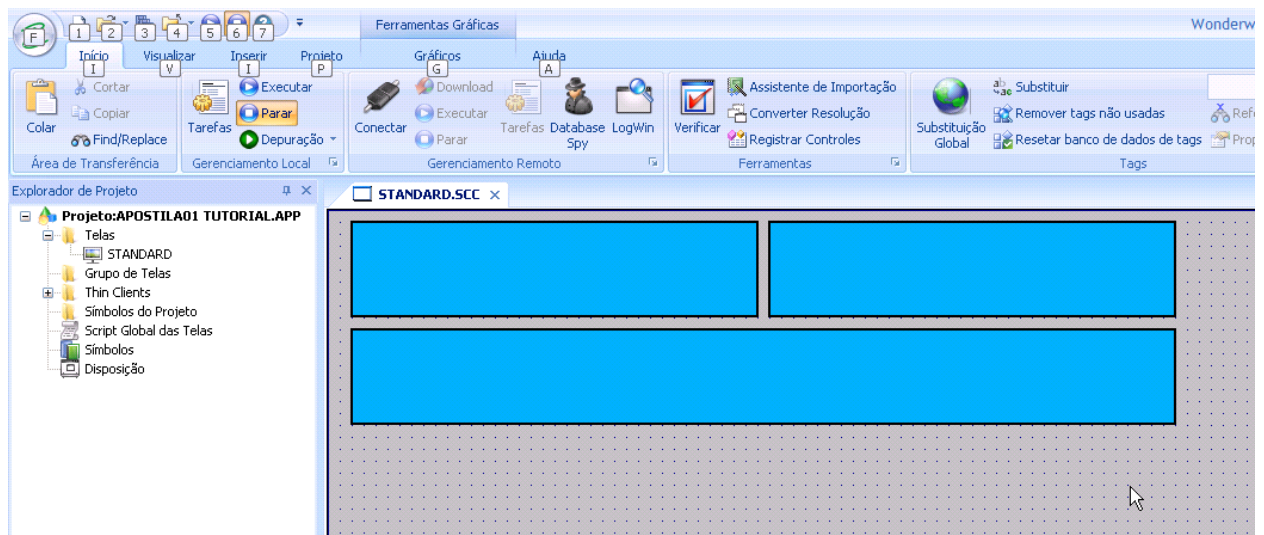


Figura 48.

Para criar o retângulo faça o mesmo processo como se estivesse criando um botão, clique e arraste, porém use a opção “**Retângulo**”, como mostrado abaixo:

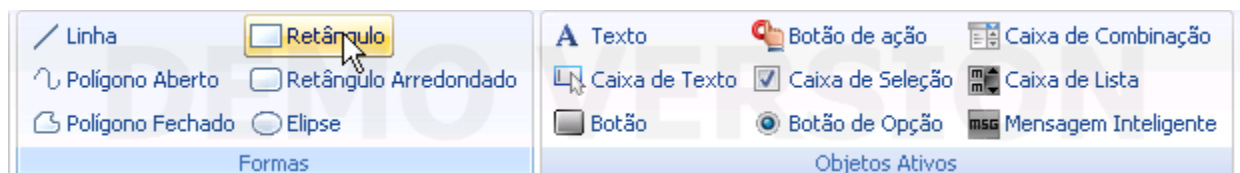


Figura 49.

Nas “**Propriedades do Objeto**” configure como a seguir, mudando a cor e a espessura de linha.

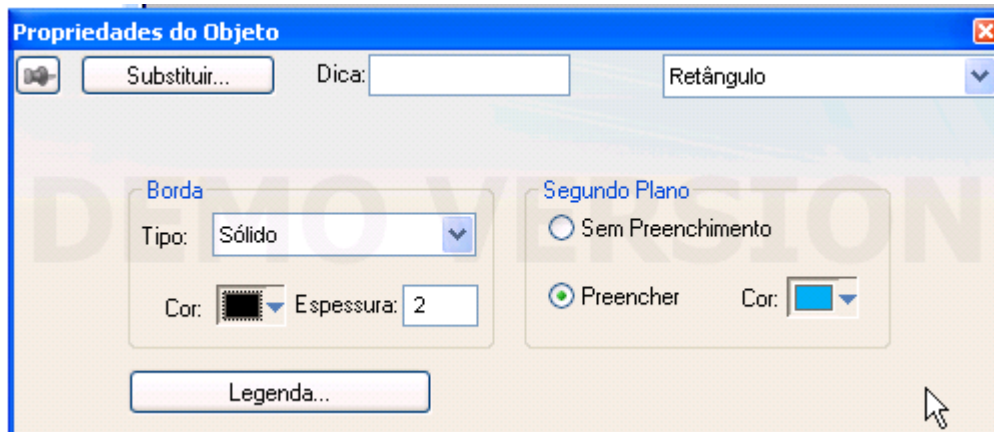


Figura 50.

Agora vamos inserir o texto nestes retângulos os quais mostrarão respectivamente: data, hora e nome da tela.

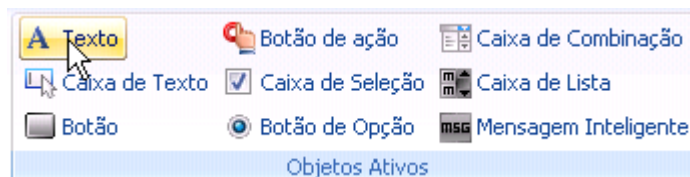


Figura 51.

Clicaremos no botão “**Texto**” para inserir caracteres de texto nos retângulos, após isso clicar dentro do retângulo, preferencialmente no centro.

Para mostrarmos o valor de um *tag* na tela, usamos o caractere “#” (que recebe vários nomes, entre eles *sustenido*, *cerquilha*, *cerquinha*, *jogo da velha*, *hash sign*, *hashtag*, etc...). O mesmo número de caracteres “cerquinha” que forem digitados será o mesmo número de caracteres do *tag* serão mostrados. Como queremos mostrar a data utilizando o formato **MM/DD/AAAA**, precisamos de **10 caracteres no retângulo da esquerda**, **8 caracteres para o retângulo da direita** e **20 (ou mais) caracteres no retângulo inferior**.

Verifique se os botões estão em tamanhos adequados e preenchem a tela de uma maneira considerável, caso não, considere aumentar o tamanho destes retângulos e(ou) botões. Pode se considerar também a diminuição da resolução de tela, porém esse não é aconselhável, visto que muito espaço utilizável será perdido.

Após a criação dos retângulos e implementação de caixas de textos dentro destes, eles devem ter uma aparência semelhante a figura a seguir.

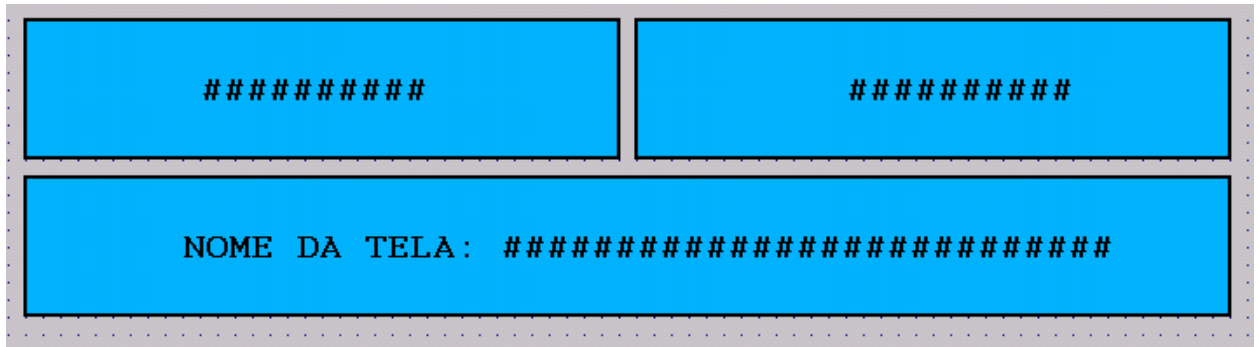


Figura 52.

Clique duas vezes no caracter “cerquinha” e configure conforme as propriedades abaixo:

Em “**Fontes**”, selecione a fonte *Courier*, tamanho 12 ou 15.

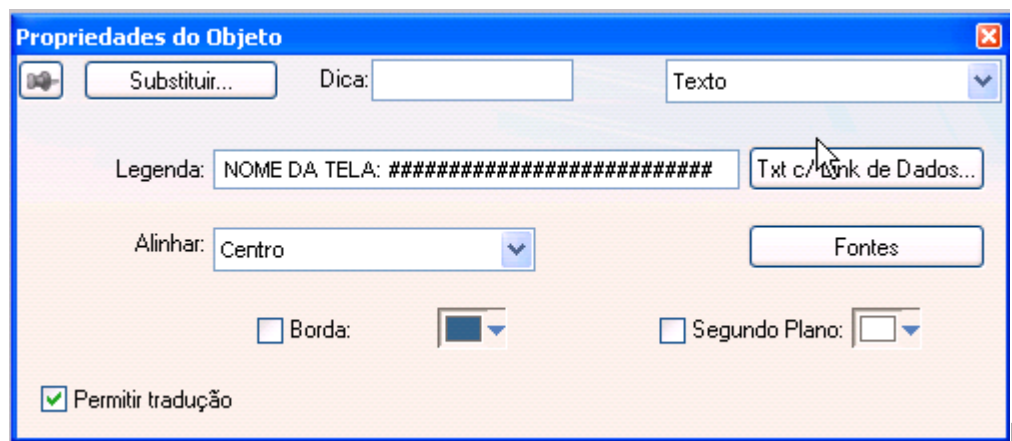


Figura 53.

Faça esse procedimento até projetar os 3 retângulos. É aconselhável usar a ferramenta de duplicar um objeto; selecione o objeto, segure **Ctrl** então clique e arraste para o lado, assim uma cópia fiel do objeto selecionado é criada. Use essa função para criar os retângulos e apenas modificar seus textos e tamanhos.

Quando o **Cabeçalho** da nossa tela estiver pronto, deve parecer com a figura abaixo:

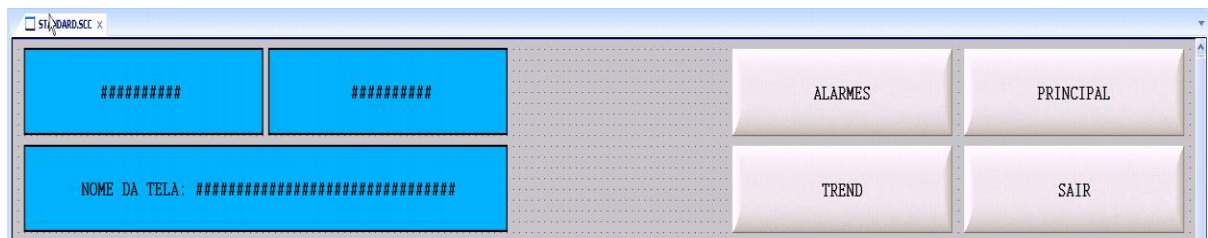


Figura 54.

Salve a tela como **STANDARD**. Para **salvar**, clique no botão localizado no canto superior esquerdo da tela, “**Salvar Como**” e digite o nome **STANDARD** no campo **Nome do arquivo**, e clique em **Salvar**.

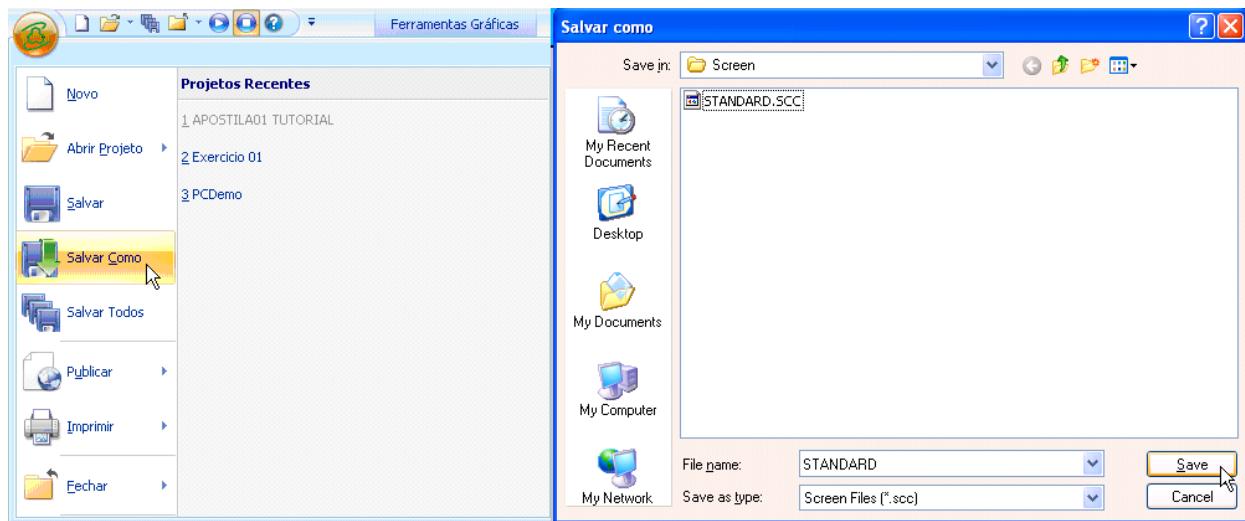


Figura 55.

CRIAR OS LINKS PARA OUTRAS TELAS NO CABEÇALHO

Agora que os objetos estão desenhados, podemos começar a aplicar em cada um deles as propriedades de comando e de display com o valor de *tags*.

Nota: Neste momento passa a ser interessante utilizar-se do recurso “**Disable Drag**” através do atalho “**Ctrl+D**”. Este recurso faz com que os objetos não possam mais ser movimentados na tela pelo mouse. Como estaremos duplo-clicando várias vezes nos objetos, às vezes estes podem mudar de posição. Com o **Ctrl+D** eles não se moverão.

Para mostrar o valor de *tags* ou mesmo expressões numa tela, aplicaremos nos textos onde houver os caracteres ##### a propriedade “**Texto com Link de Dados**”, e para dar função aos botões usaremos a propriedade “**Objeto**”.



Figura 56.

Clique uma vez sobre os caracteres ##### do primeiro retângulo para selecionar o mesmo e então clique em “**Texto com Link de Dados**”, ao fazer isso é possível alterar o que é escrito nos *hash tags* e no caso do primeiro retângulo, queremos ver o horário em tempo real.

Para ver a hora desejada colocaremos no campo “**Tag/Expressão**” a expressão **TIME** como mostrado na figura a seguir:



Figura 57.

Selecione o outro retângulo com texto ##### de 10 caracteres, faça o mesmo procedimento para inserir **"Texto com Link de Dados"**. Nesse retângulo queremos ver a data atual (DD/MM/AA), então usaremos a expressão **DATE**.

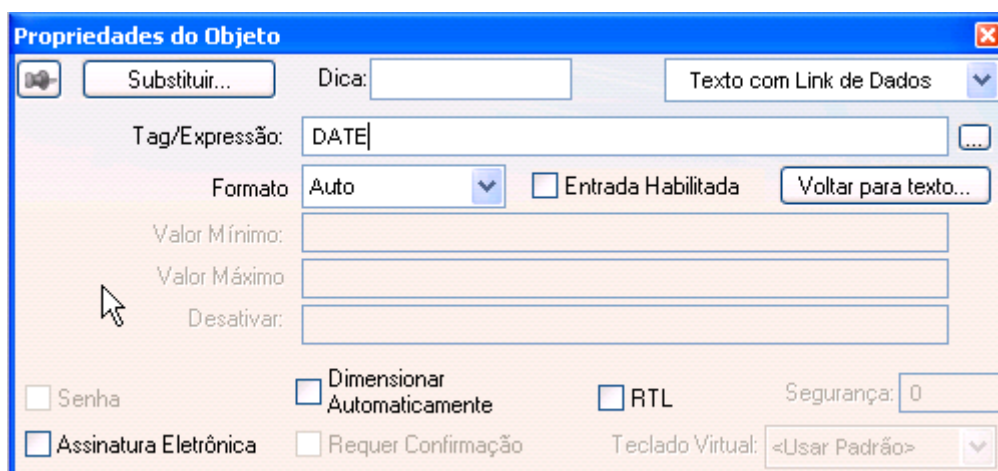


Figura 58.

Selecione o texto #####, do retângulo maior e apliquemos novamente a propriedade **"Texto com Link de Dados"**.

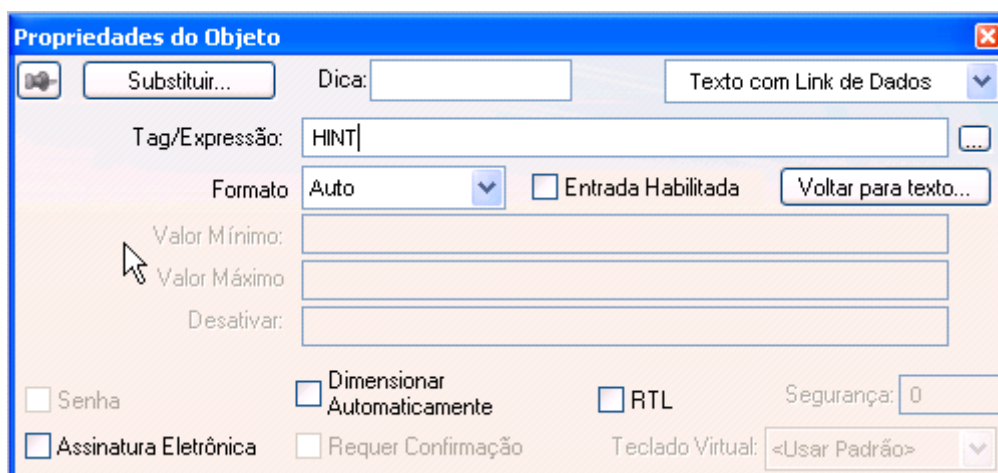


Figura 59.

A partir de agora estaremos inserindo os comandos de abrir tela nos botões do Cabeçalho da tela. Clique uma vez no botão escrito "**PRINCIPAL**" e então clique em "**Comando**" que fica ao lado esquerdo de "**Texto com Link de Dados**", com isso o botão será usado (enquanto o programa estiver sendo executado) para realizar um comando, como já dito no nome da propriedade.

Caso não tenha encontrado o botão "**Comando**", verifique a Figura 55.

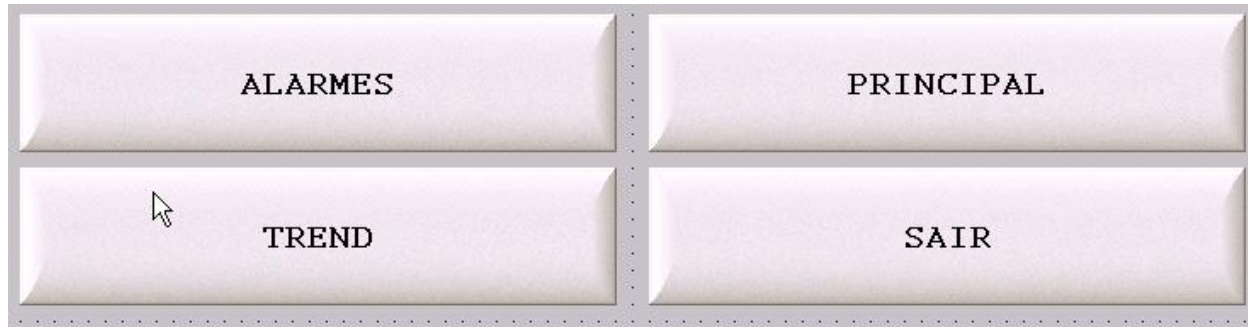


Figura 60.

Após acessar as propriedades do botão já com a propriedade "**Comando**" ativa uma janela diferente será aberta, nesta janela modificaremos o "**Tipo**" de "**VBScript**" para "**Linguagem Interna**".

Depois disso daremos uma função para cada um dos botões.

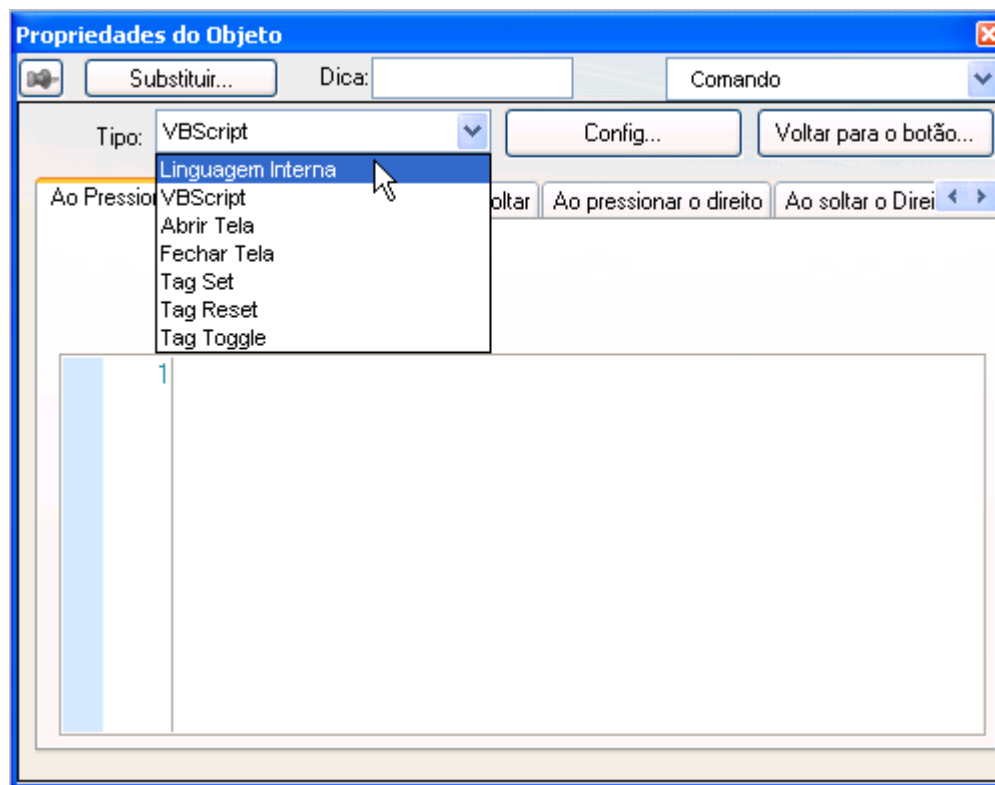


Figura 61.

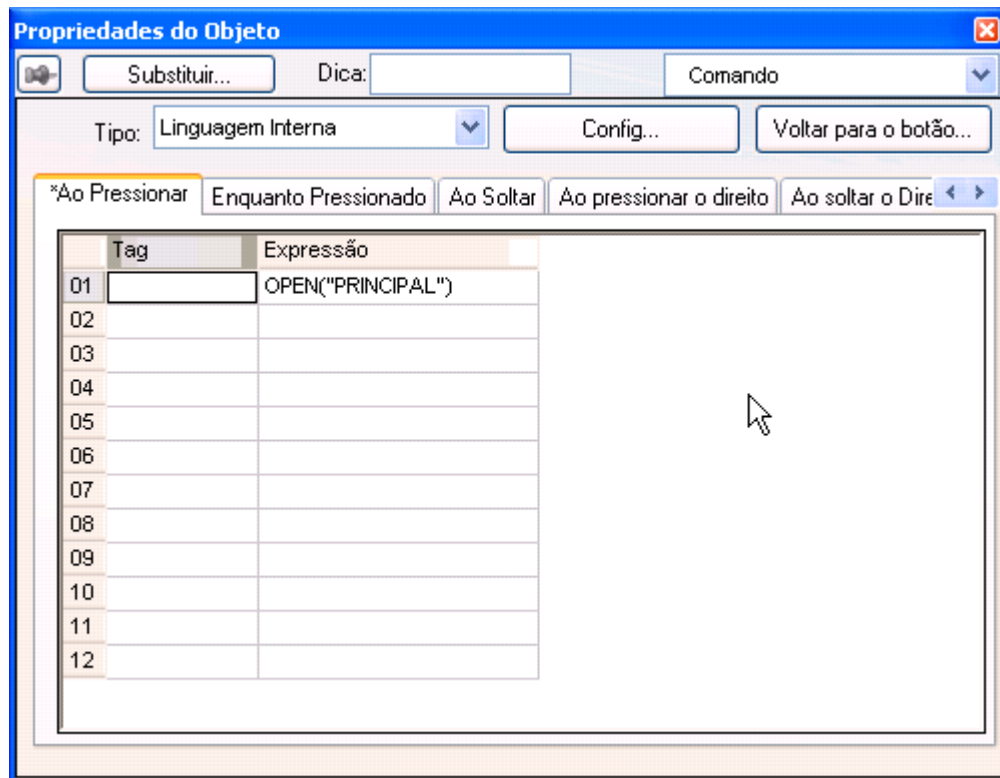


Figura 62.

A função **Open ()**, abre uma tela, que será chamada através do nome dado a ela dentro das aspas passada como parâmetro. Não é necessário que a janela tenha sido criada para que programemos um botão para chamá-la. Mas para que a função funcione aí sim a tela já deve ter sido criada. O nome da tela a ser chamada, **deve estar entre aspas** por tratar-se de uma constante do tipo **String**. Se colocarmos ao invés disso um *tag* do tipo string, sendo que este *tag* possui como valor o nome de uma tela, então podemos colocar o nome do *tag* sem aspas.

Clique duas vezes no botão "**Sair**" e repita o procedimento alterando a "**Expressão**", como visto abaixo:

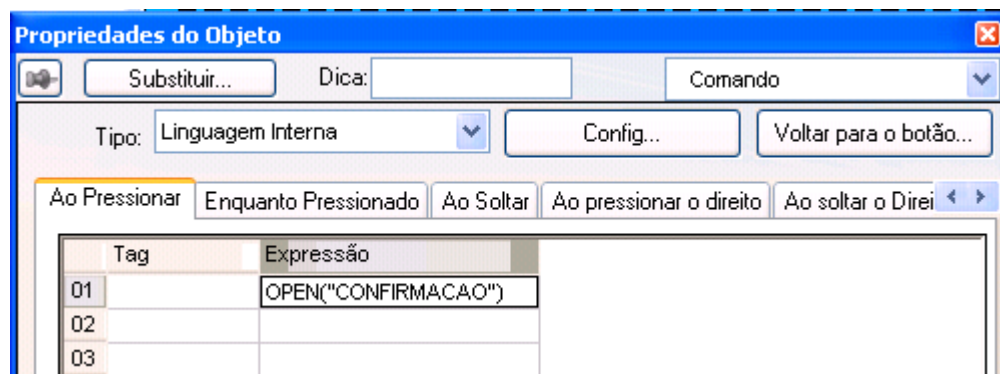


Figura 63.

Clique duas vezes no botão escrito "**TREND**" e configure a dinâmica de "**Comando**" com a expressão: **OPEN("TREND")**.

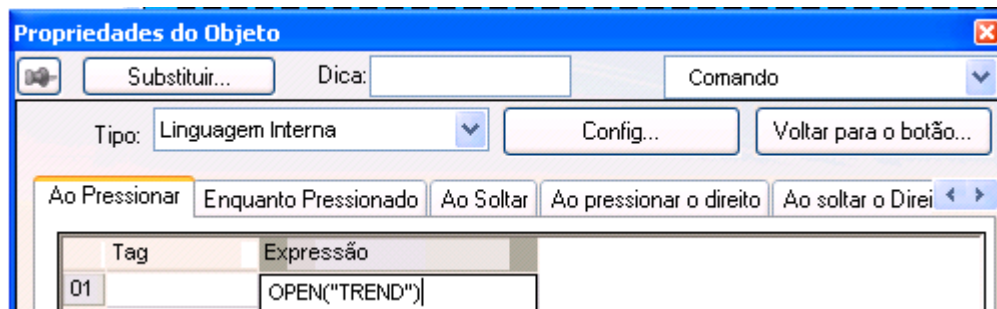


Figura 64.

Clique duas vezes no botão escrito "**ALARMES**" e configure a dinâmica de "**Comando**" com a expressão: **OPEN("ALARMES")**.

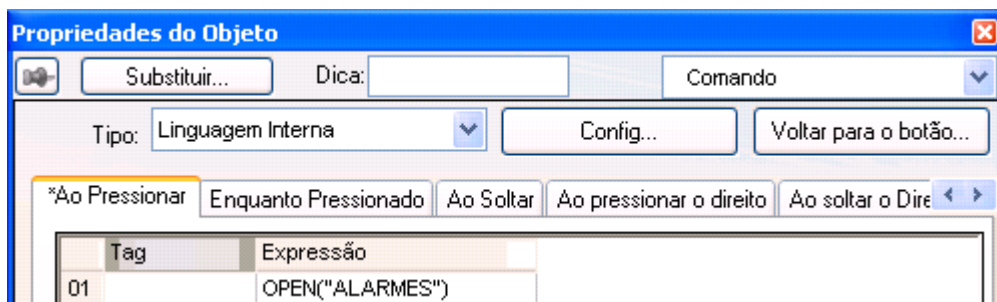


Figura 65.

Salve a tela como **STANDARD**. Para **salvar**, clique no botão localizado no canto superior esquerdo da tela, "**Salvar Como**", digite o nome **STANDARD** no campo **Nome do arquivo**, e clique em **Salvar**.

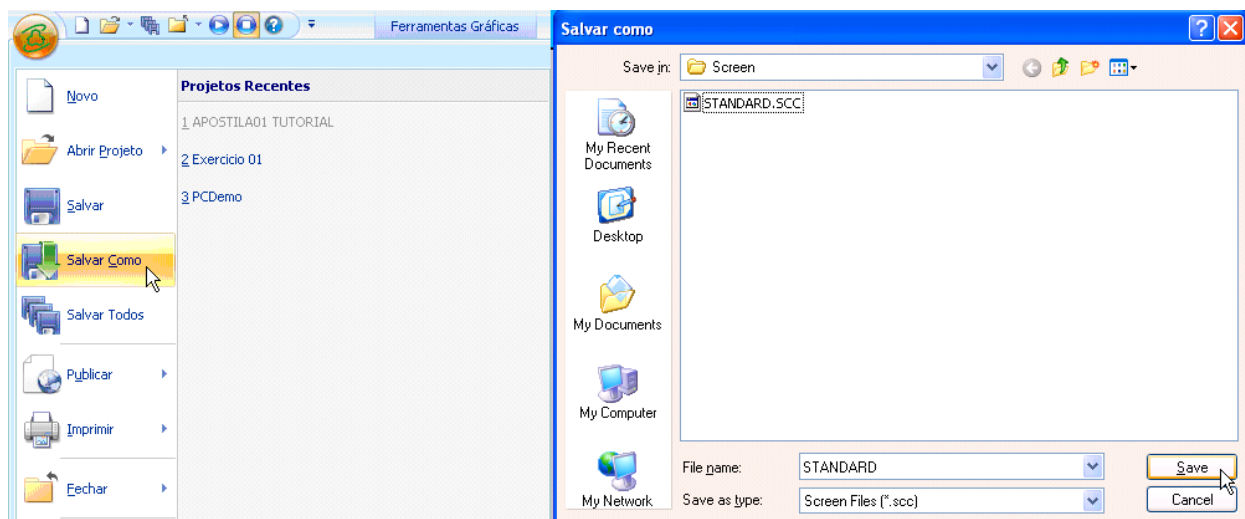


Figura 66.

CRIAR O RODAPÉ (PARTE INFERIOR DA TELA)

Vamos criar agora um **RODAPÉ**, e editar as suas propriedades.



Figura 67.

Clique no botão "**Alarme/Evento**" na mesma aba "**Gráficos**", onde encontramos os botões "**Texto com Link de Dados**" e "**Comando**", e crie no inferior da tela, clicando e arrastando como fizemos com os botões e retângulos.

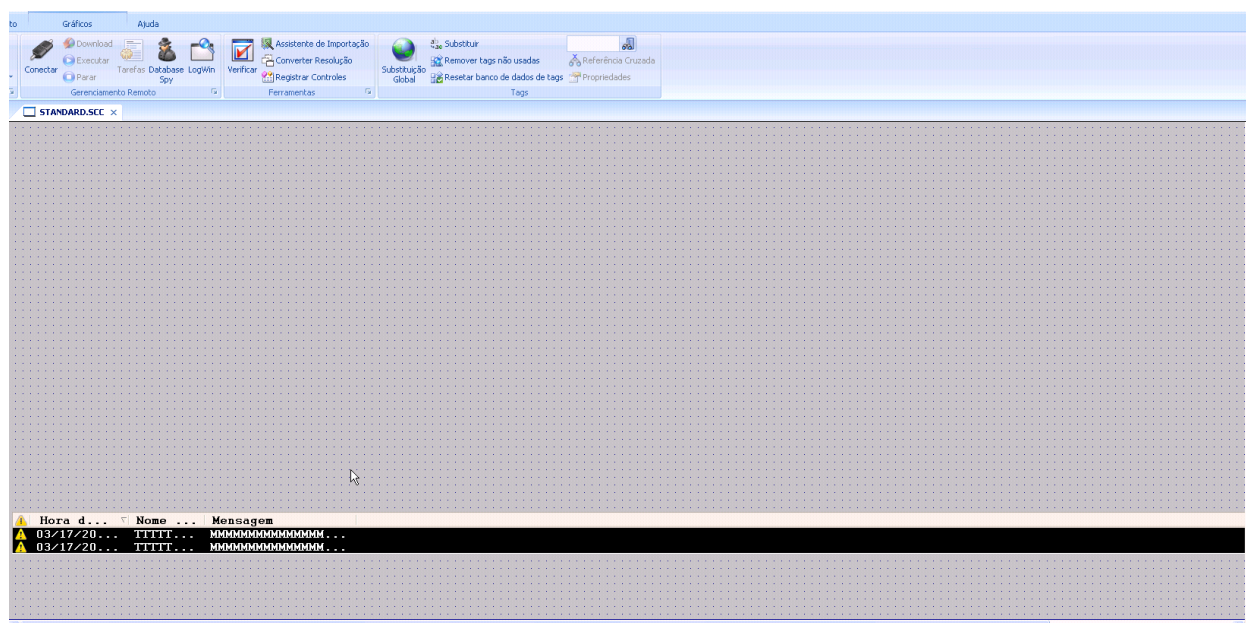


Figura 68.

Clique duas vezes na tela de Alarme recém criada, e edite suas propriedades na tela "**Propriedades do Objeto**" como mostrado abaixo:

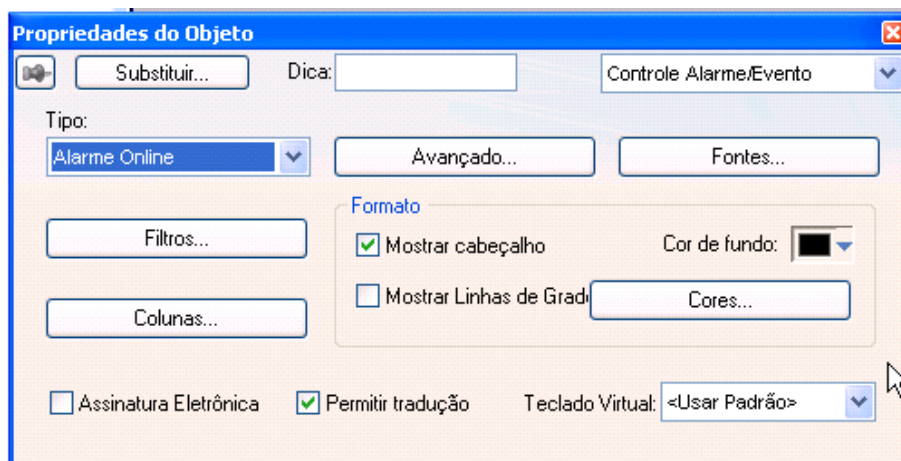


Figura 69.

Clique no botão “**Fontes**” para selecionar a fonte tipo: *Courier* - Tamanho 12 - Cor Branca (White).

O objeto então deverá parecer como este abaixo:



Figura 70.

A tela deve ser salva para que os atributos tenham efeito.

Salve a tela como **STANDARD**. Para **salvar**, clique no botão localizado no canto superior esquerdo da tela, “**Salvar Como**”, digite o nome **STANDARD** no campo **Nome do arquivo**, e clique em Salvar.

O botão “**Sair**” terá a função de fechar os módulos de RunTime(execução) da aplicação. Para tal, usamos a função "Shutdown()". Mas, aproveitaremos a ocasião para deixar o operador confirmar se realmente ele deseja sair. Então devemos criar uma tela chamada “Confirmação”, com as opções “Sim” e “Não” para confirmar o fechamento da aplicação. Esta tela será criada da seguinte forma:

CRIAR TELA CONFIRMAÇÃO

Agora crie a tela “**Confirmação**”, para inserir uma nova tela clique com o botão direito na pasta “**Telas**”, clique em “**Inserir**” e configure como na figura abaixo:

Atenção ao nomear a tela, deve-se usar **C** no lugar de **Ç**. Mude o **Tamanho** e **Localização** da tela, o **Estilo**, a **Borda** e selecione a caixa de **Título** e desmarque a caixa **Menu de Sistema**.

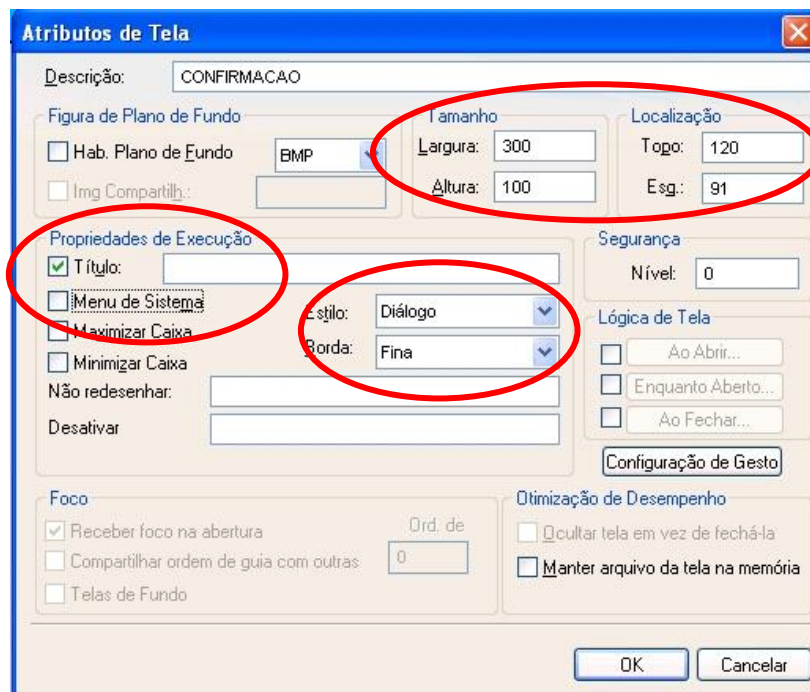


Figura 71.

Na nova tela desenhe os seguintes objetos (botões e texto) como na figura abaixo:

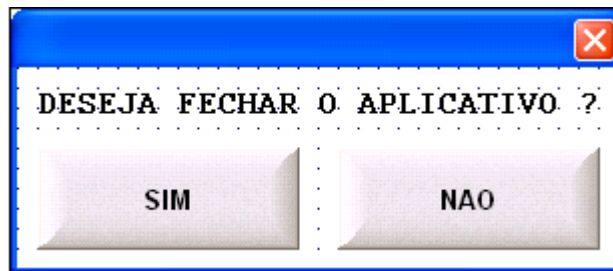


Figura 72.

Crie um “**Texto**” e os botões que devem conter a legenda de **SIM** e **NÃO**.

No botão “**Sim**”, clique duas vezes, insira a propriedade “**Comando**” e digite **Shutdown ()**.

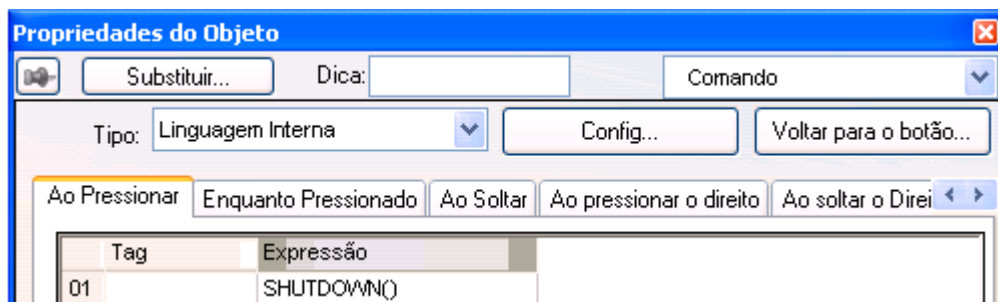


Figura 73.

No botão “**Não**”, insira a propriedade “**Comando**” e digite, na coluna “**Expressão**” o comando **Close(“Confirmacao”)**.

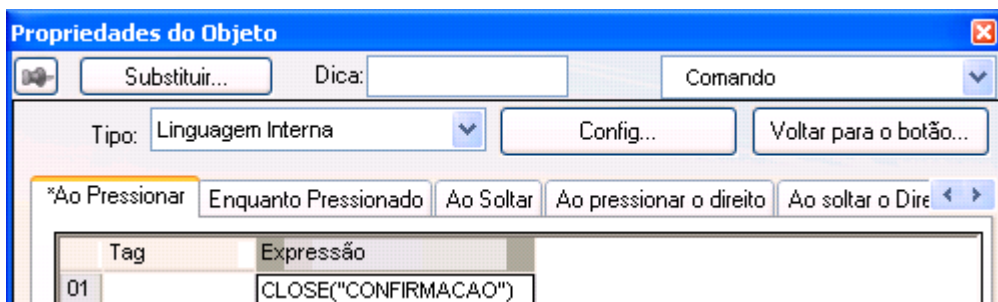


Figura 74.

Salve a tela como **CONFIRMACAO**. Para **salvar**, clique no botão localizado no canto superior esquerdo da tela, “**Salvar Como**”, digite o nome **CONFIRMACAO** no campo **Nome do arquivo**, e clique em Salvar.

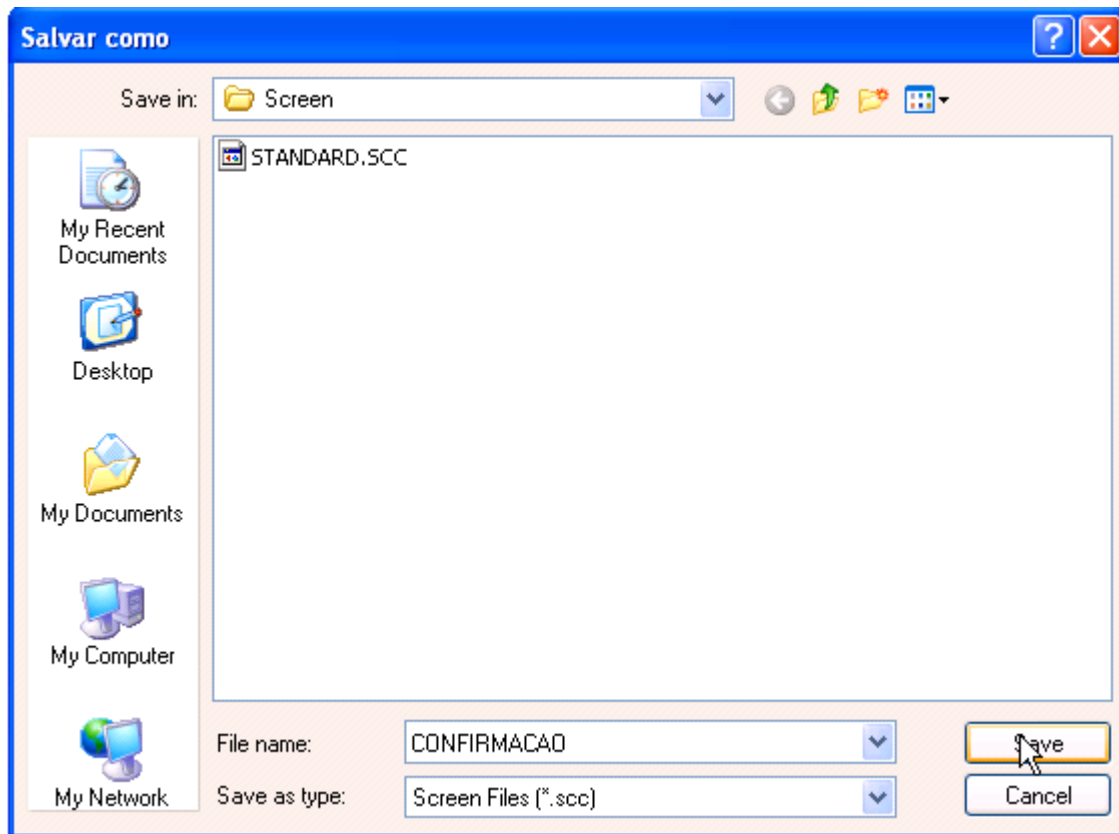


Figura 75.

CRIAR TELA PRINCIPAL COM OS ITENS DE PROCESSO E SINÓTICOS

Voltando a tela “**STANDARD**” criada anteriormente, deveremos:

Mostrar as propriedades de três Tanques (Temperatura, Pressão e Nível) em formato gráfico e numérico.

Mostrar o estado das válvulas de enchimento e esvaziamento dos Tanques através de objetos da biblioteca.

Inserir comandos de abertura e fechamento das válvulas.

Essas modificações farão com que a tela STANDARD se torne a tela PRINCIPAL, ao fazer isso deveremos futuramente mudar a descrição da tela.

Para criar os tanques, os canos e as válvulas, usaremos objetos da biblioteca. Para acessarmos a biblioteca de símbolos, clique com no ícone mostrado na figura abaixo:

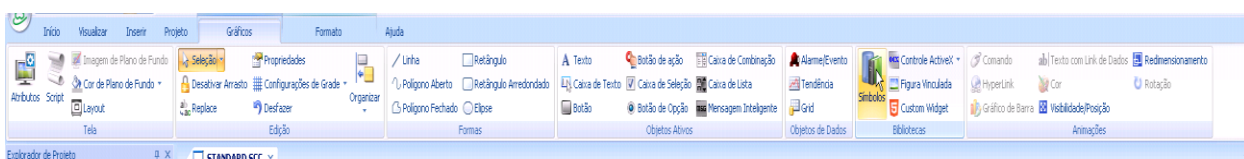


Figura 76.

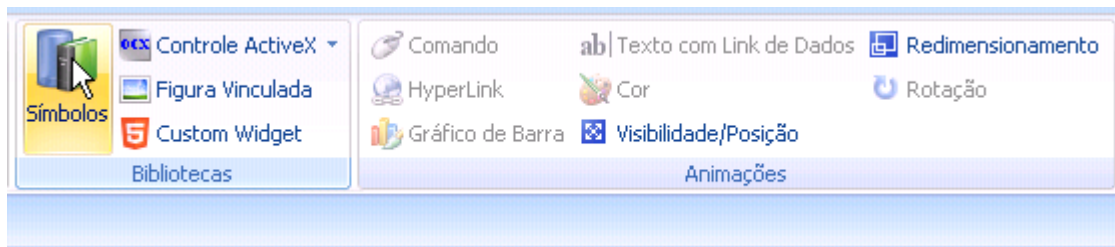


Figura 77.

Para importar um objeto da biblioteca para a tela da aplicação, simplesmente clique no objeto e depois na tela.

É claro que há outras formas de se desenhar sem ser utilizando a biblioteca, mas para os propósitos deste tutorial a biblioteca nos atenderá.

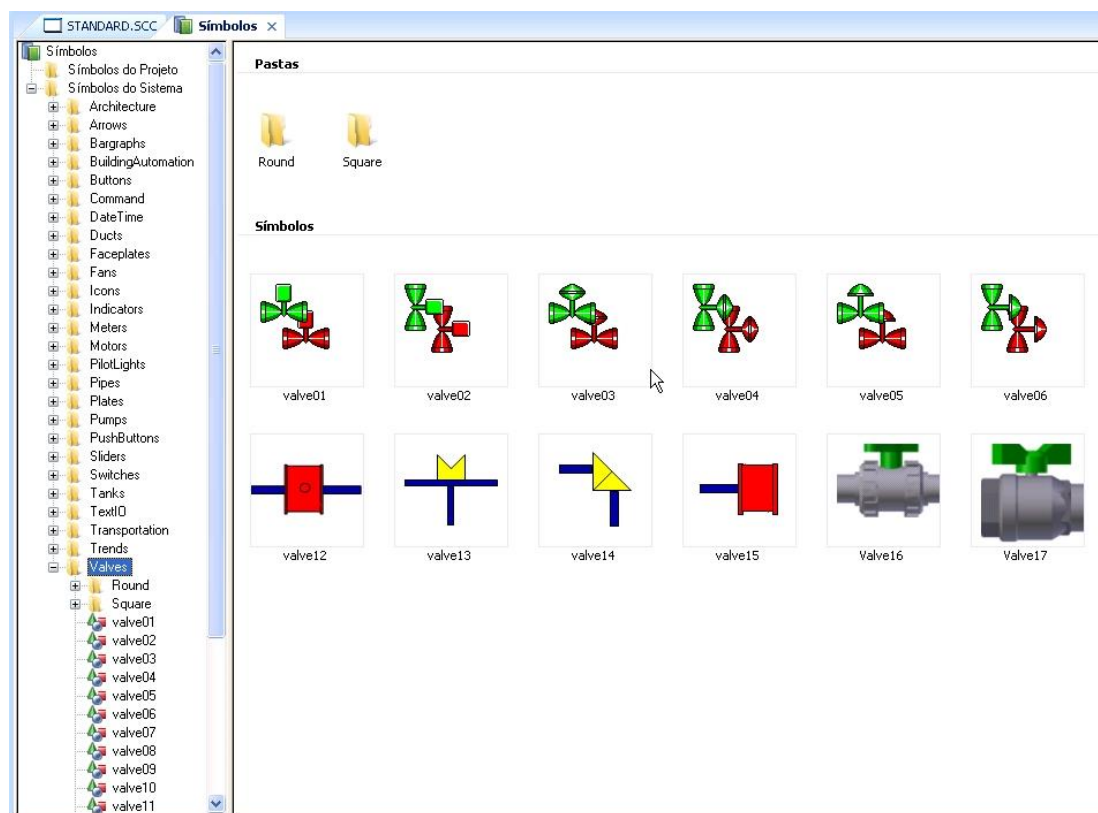


Figura 78.

Basta clicar em uma pasta qualquer que a mesma se expandirá e mostrar todos os objetos presentes nesta.

Portanto vamos importar objetos como válvulas (na janela *Valves* da biblioteca), canos (em *Pipes*) e o tanque (em *Tanks*) e as setas de *UP* e *DOWN* para aumentar ou diminuir o número de qual tanque será o ativo no momento (em *Command*), como vemos abaixo:

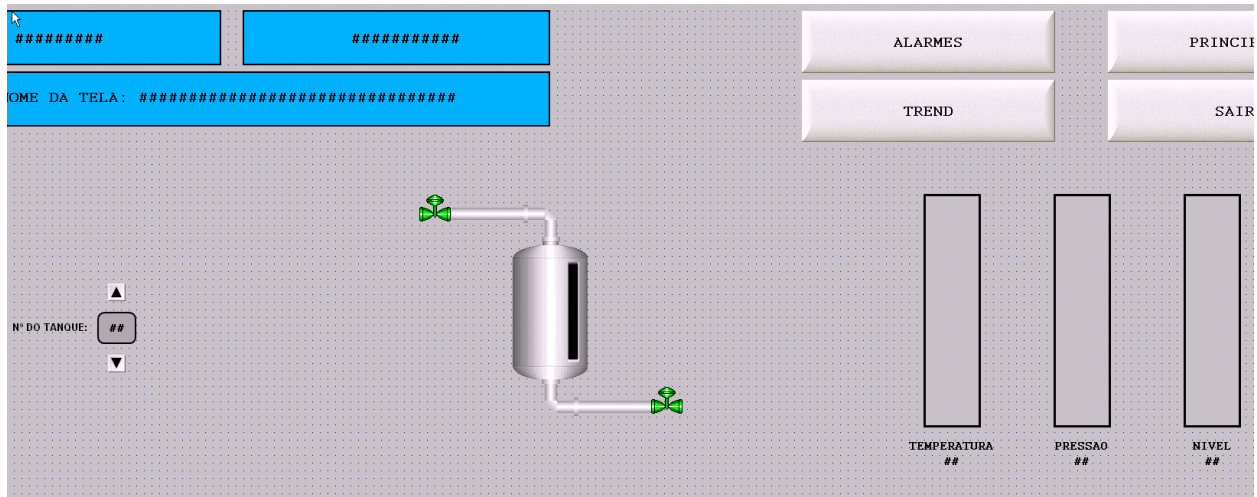


Figura 79.

Para posicionar mais adequadamente os objetos precisamos desativar o auxílio que o programa nos fornece que faz com que posicione os objetos de tantos em tantos *pixels* para qualquer direção. Para desativar isso, desmarque o quadro como mostrado nas figuras abaixo:

Clique com o botão direito na tela e acesse “**Configurações de Grade**”.

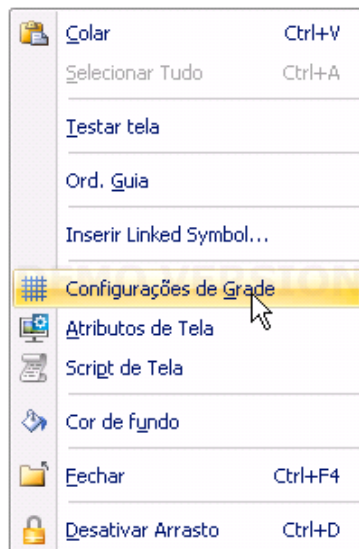


Figura 80.

Por padrão a caixa onde o cursor está em cima estará marcada, para obter essa liberdade para movimentar os objetos, basta desmarcar e clicar em **OK**. Caso queira reativar, faça o inverso.

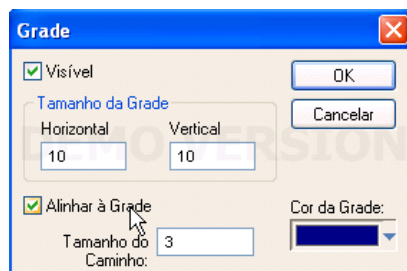


Figura 81.

Desenhemos também três retângulos na posição vertical e abaixo de cada um coloque respectivamente: TEMPERATURA, PRESSÃO, NÍVEL. Abaixo destes, coloque 3 “#” (*hash tag*), como visto na Figura 79. Estes que devem ter sua cor principal da mesma que a cor de fundo, para que fique “transparente” ao não terem nenhum valor sendo medido.

Salve a tela como **PRINCIPAL**. Para **salvar**, clique no botão localizado no canto superior esquerdo da tela, “**Salvar Como**“, digite o nome **PRINCIPAL** no campo **Nome do arquivo**, e clique em Salvar.

Clique duas vezes em **Planilha de Dados**.

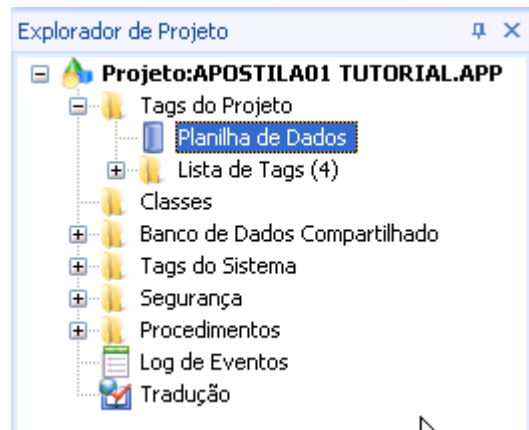


Figura 82.

Crie agora os **tags Temperatura, Nivel e Pressao** que mostrarão a Temperatura, o Nível e a Pressão dos Tanques. E mais um **tag** que será o número de um dos tanques que estará sendo visto (isso enquanto o programa estiver sendo executado) chamado **Indice** e configure conforme mostrado na tabela abaixo:

PRINCIPAL.scc Símbolos STANDARD.SCC Tags do Projeto x					
	Nome	Vetor	Tipo	Descrição	Escopo
	Filtro de Texto	= ...	(Todos)	Filtro de Texto	(T...
1	ESTADOENCHE	3	Booleana	VALVULA QUE ENCHE O TANQUE	Local
2	ESTADUESVAZIA	3	Booleana	VALVULA QUE ESVAZIA O TANQUE	Local
3	COMANDOENCHE	3	Booleana	COMANDO QUE ENCHE O TANQUE	Local
4	COMADUESVAZIA	3	Booleana	COMANDO QUE ESVAZIA O TANQUE	Local
5	TEMPERATURA	3	Real	TEMPERATURA DO TANQUE	Local
6	PRESSAO	3	Real	PRESSAO DO TANQUE	Local
7	NIVEL	3	Real	NIVEL DO TANQUE	Local
8	INDICE	0	Inteira	INDICE DO TANQUE	Local
*			Inteira		Servidor
*			Inteira		Servidor
*			Inteira		Servidor
*			Inteira		Servidor
*			Inteira		Servidor

Figura 83.

Pressione Ctrl + S para salvar ou aperte o botão de salvar no canto superior esquerdo.



Figura 84.

Os 4 primeiros tags já haviam sido criados previamente durante o exercício anterior.

Voltando a tela **PRINCIPAL** aplicaremos a propriedade “**Texto com Link de Dados**” nos *hash tags* criados abaixo de Temperatura, Pressão e Nível. E colocaremos com expressão em cada um: *A variável que desejamos mostrar*[INDICE]. Exemplo: TEMPERATURA[INDICE].

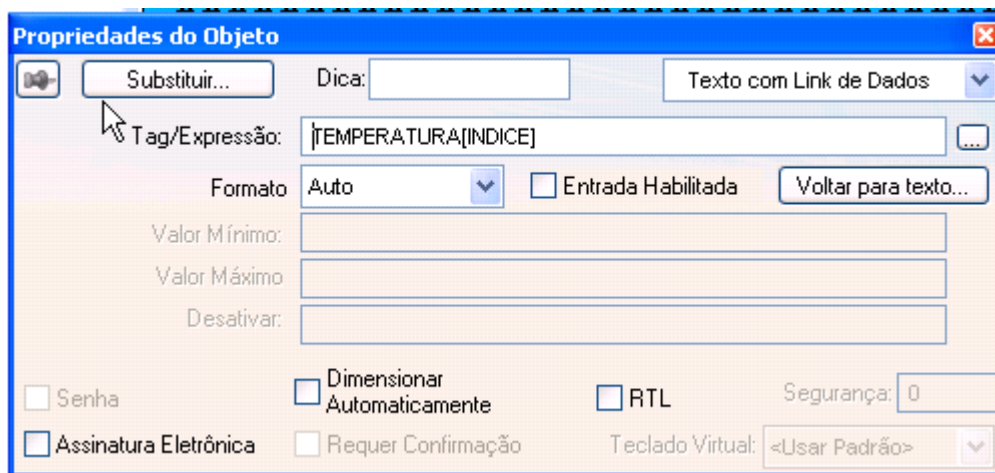


Figura 85.

Este display mostrará o valor da temperatura do tanque associado pelo *tag* **Índice**. **Índice** denominará qual (temperatura do) tanque estará sendo mostrado.

Se **Índice** = 1, ele mostrará a temperatura do tanque 1, se **Índice** = 2 ele mostrara a temperatura do tanque 2, e assim sucessivamente.

Selecione o texto “####” abaixo de Pressão e configure como mostrado abaixo.

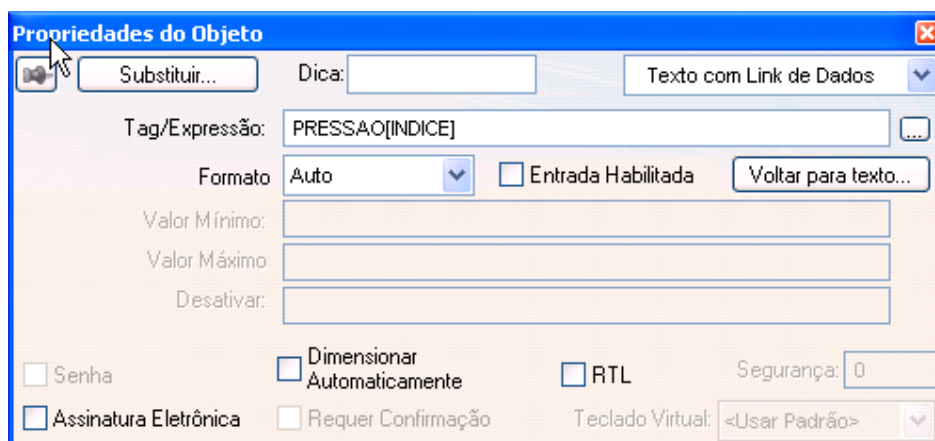


Figura 86.

Por último, selecione o texto ### abaixo de Nível. Configure sua *Tag/Expressão* como abaixo:

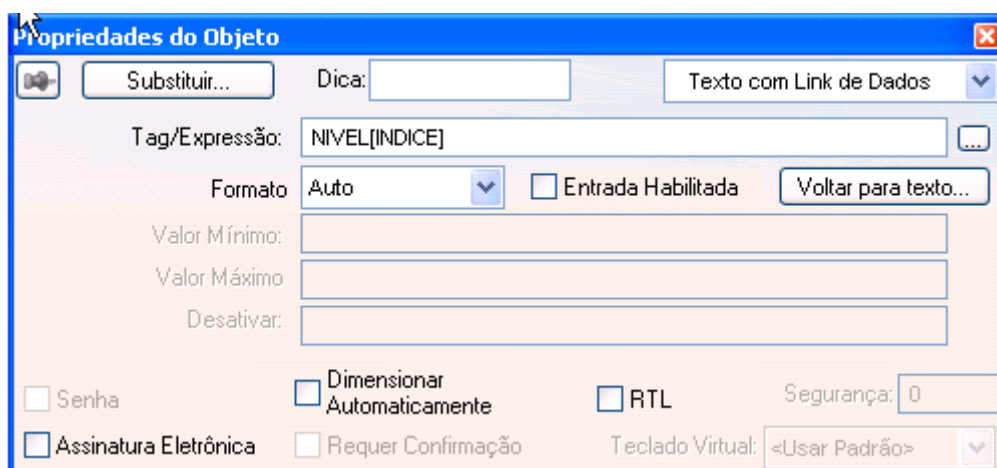


Figura 87.

A fim de mostrar os valores de Temperatura, Pressão, e Nível, todos graficamente, usaremos a propriedade **“Gráfico de Barras”** nos três retângulos desenhados acima dos textos. Clique no retângulo e então aperte o botão mostrado na figura abaixo:



Figura 88.

No nosso exemplo, o retângulo vai tendo o seu interior preenchido de acordo com o valor do *tag* associado ao seu gráfico de barra. Se o *tag* for zero retângulo vazio, 50% cheio pela metade e 100%, teremos o retângulo cheio.

Configure o primeiro retângulo como mostrado abaixo:

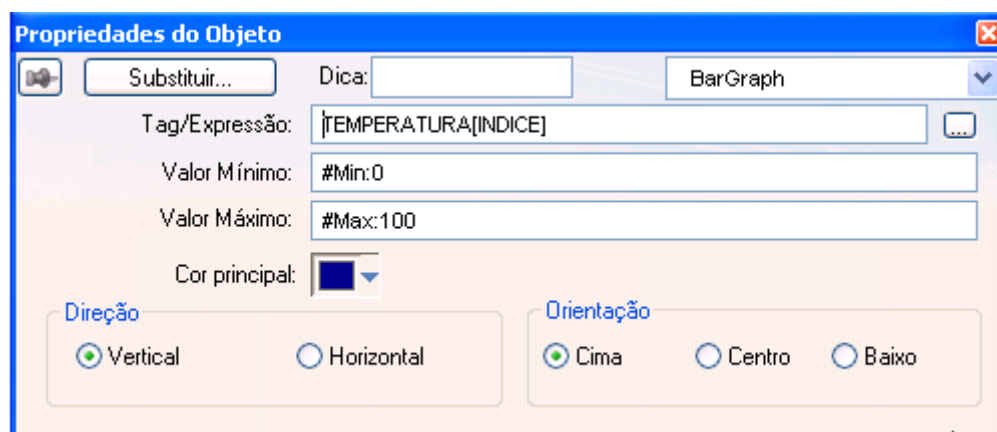


Figura 89.

Para o retângulo de Pressão:

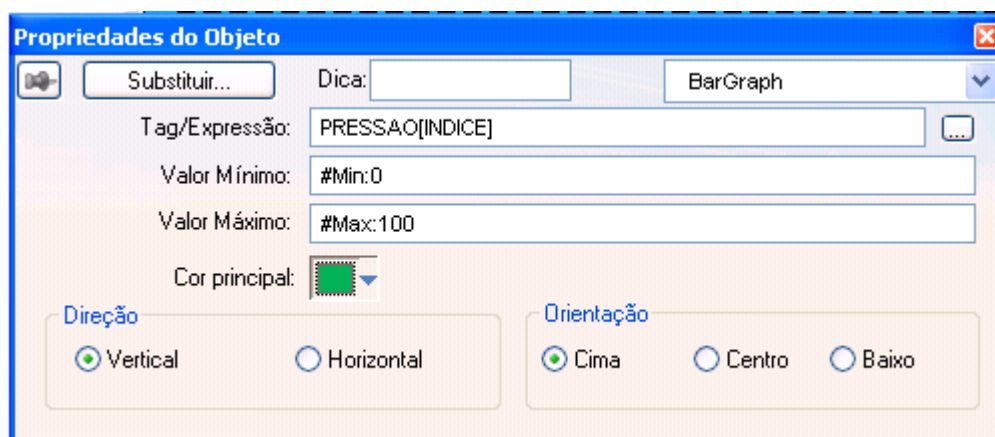


Figura 90.

E por ultimo, para o retângulo de Nível:

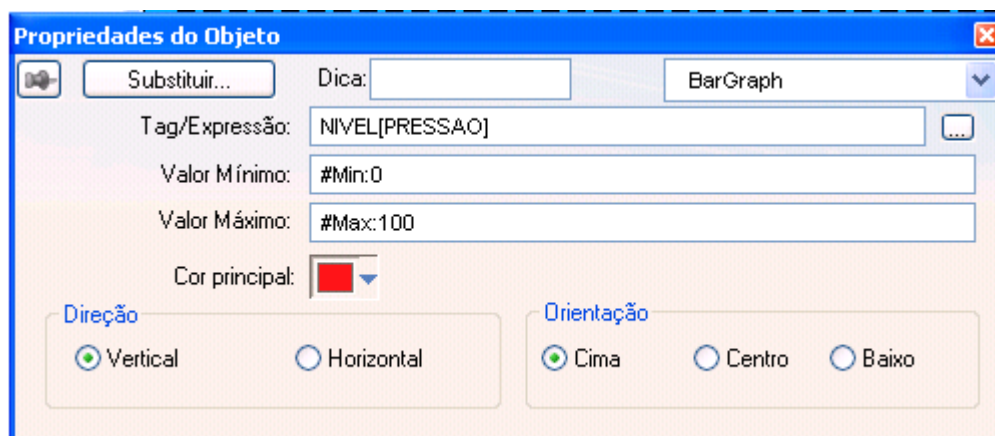


Figura 91.

Atenção ao nomear cada uma das **Expressões**.

Agora vamos às válvulas.

Chamaremos a válvula que se encontra na entrada do tanque de "**ValvulaEnche**". Clique duas vezes em cima dela e selecione as propriedades: TagCmd=**COMANDOENCHE[INDICE]** e TagState=**ESTADOENCHE[INDICE]** como mostrado abaixo:

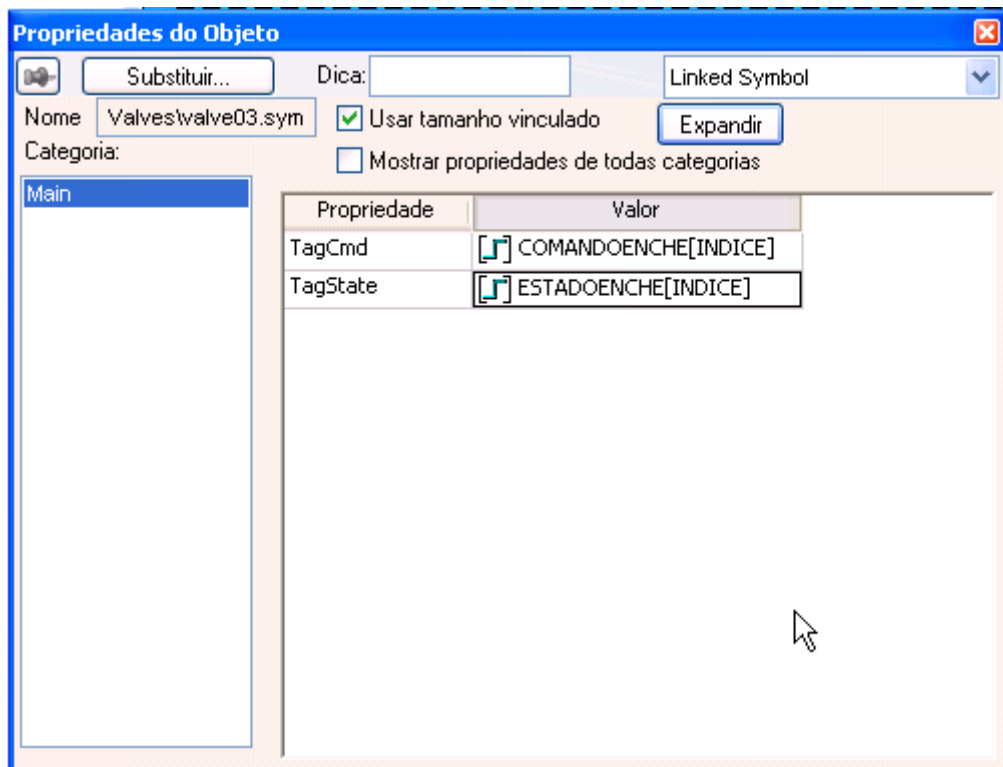


Figura 92.

Esta propriedade fará com que, sempre que o valor do tag **ESTADOENCHE[INDICE]** for “1”, a válvula se apresentará com cor verde. Pensando em uma aplicação real, o valor deste tag deverá ser **LIDO** do campo. A propriedade **TagCmd** fará com que ao teclar com o mouse sobre o objeto o valor do tag **COMANDOENCHE[INDICE]** altere entre os valores “0” e “1”. Numa aplicação real, estes tag estariam enviando comandos para abrir e fechar as válvulas para do sistema. Portanto este tag seria um tag de **ESCRITA** para o **Driver**.

Chamaremos a válvula que se encontra na saída do tanque de “**ValvulaEsvazia**”. Clique duas vezes em cima dela e selecione as propriedades: **TagCmd=COMANDOESVAZIA[INDICE]** e **TagState=ESTADOESVAZIA[INDICE]** como mostrado abaixo:

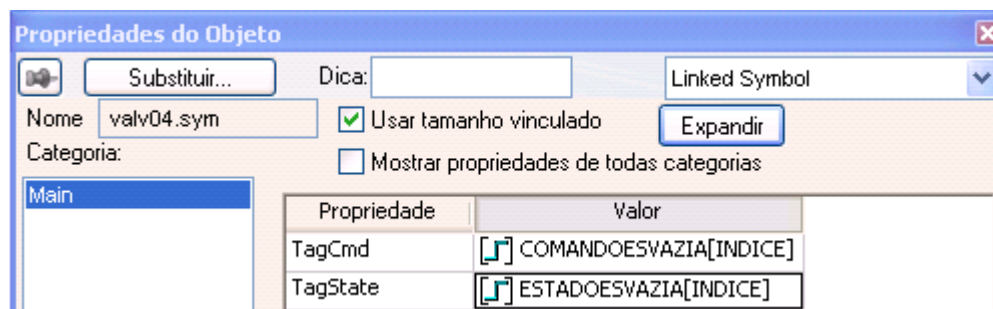


Figura 93.

Esta propriedade fará com que, sempre que o valor do tag **ESTADOESVAZIA[INDICE]** for “1”, a válvula se apresentará com cor verde. Pensando em uma aplicação real, o valor deste tag deverá ser **LIDO** do campo. A propriedade **TagCmd** fará com que ao teclar com o mouse sobre o objeto o valor do tag **COMANDOESVAZIA[INDICE]** altere entre os valores “0” e “1”. Numa aplicação real, estes tag estariam enviando comandos para abrir e fechar as válvulas para do sistema. Portanto este tag seria um tag de **ESCRITA** para o **Driver**.

Finalmente vamos criar os comandos para o tag “Índice”.

Já havíamos desenhado dois botões com setas para cima e para baixo, ao lado do número do tanque escolhido.

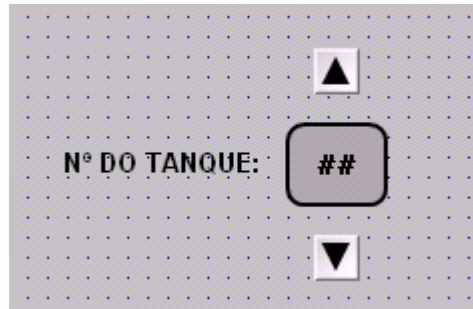


Figura 94.

Adicionaremos à propriedade “Comando” as duas setas e colocaremos “Índice” como tag a ser modificada e uma função com “IF” para somar ou subtrair um(1).

Para o botão apontado para cima: IF(INDICE<3, INDICE+1)

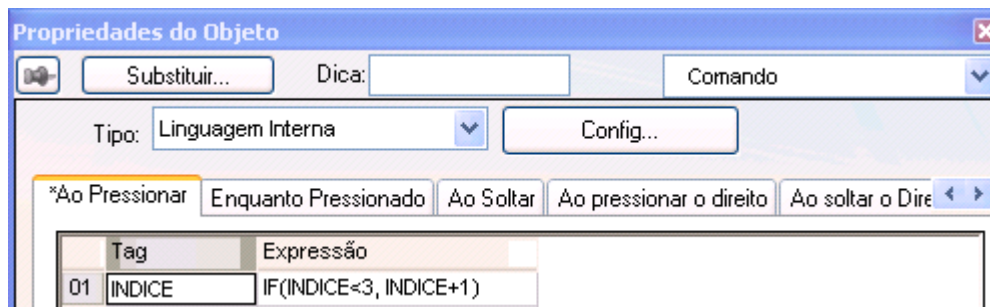


Figura 95.

Já para o botão que aponta para baixo, repita a operação e configure como mostrado abaixo: IF(INDICE>1, INDICE-1)

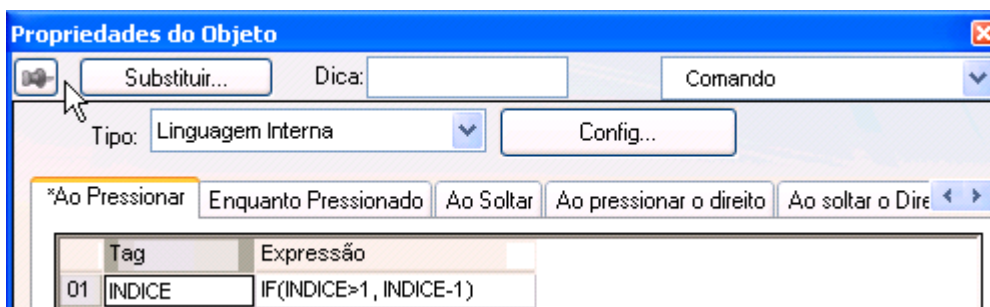


Figura 96.

Agora vamos mostrar qual o número do tanque, clique duas vezes no texto “##” e clique no ícone da propriedade “Texto com Link de Dados”, configure como mostrado abaixo.

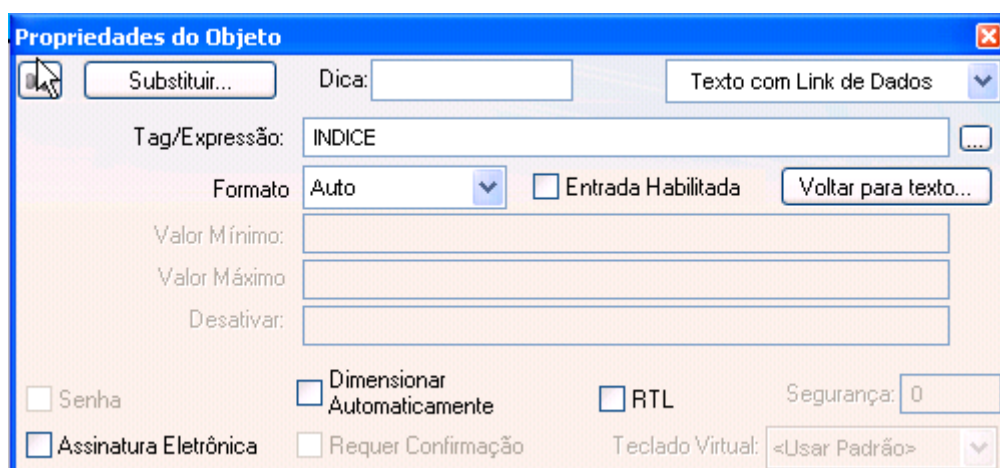


Figura 97.

Salve a tela como **PRINCIPAL**. Para **salvar**, clique no botão localizado no canto superior esquerdo da tela, “**Salvar Como**“, digite o nome **PRINCIPAL** no campo **Nome do arquivo**, e clique em Salvar.

Como editamos essa tela usando como base a **STANDARD**, a sua descrição continua como “**STANDARD**”. Faremos o seguinte:

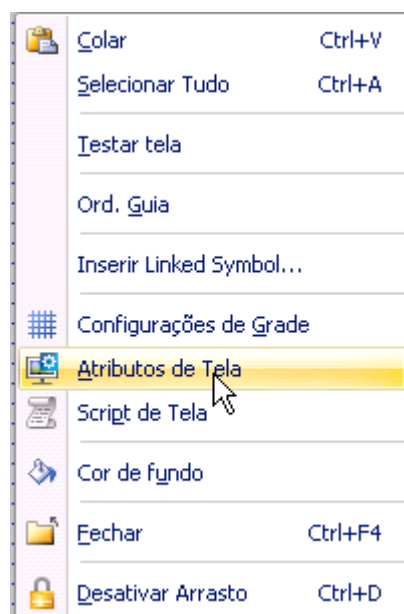


Figura 98.

Clicaremos na tela com o botão direito, iremos em “**Atributos de Tela**” e mudaremos o campo de “**Descrição**” para **PRINCIPAL**.

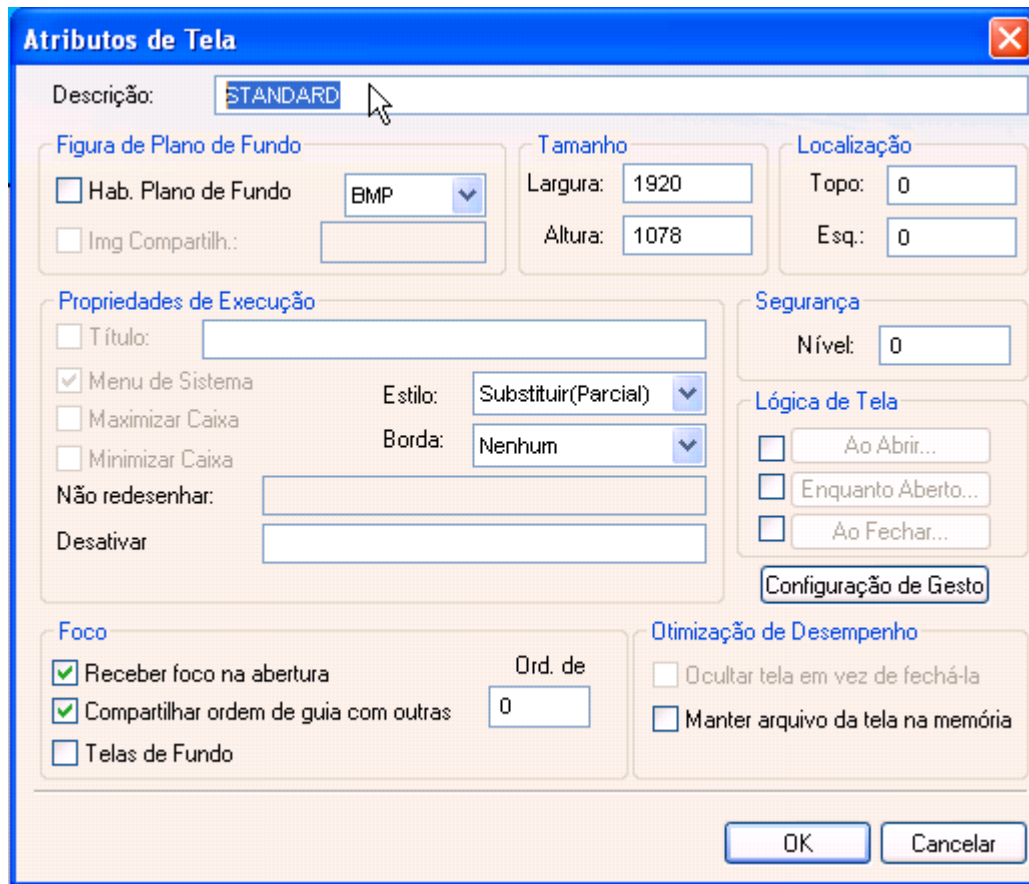


Figura 99.

Depois clique em OK.

Perceba também que após utilizar a tela **STANDARD** como base pra fazer outra tela e APÓS salvar a tela pronta, como por exemplo a tela **PRINCIPAL**, deve-se apagar tudo que foi adicionado após a criação padrão desta tela, para que futuramente ela possa ser usada como **STANDARD**

Em seguida configure a partida na aba **Projeto**, clique na opção “**Visualizador**”.



Figura 100.

Na janela “**Configurações de Projeto**”, selecione na opção “**Tela Inicial**” a opção **PRINCIPAL.SCC**, como mostrado na figura a seguir.

Esse procedimento é feito para que ao executar o programa (F5) a tela que será mostrada logo no início será a tela principal que criamos.

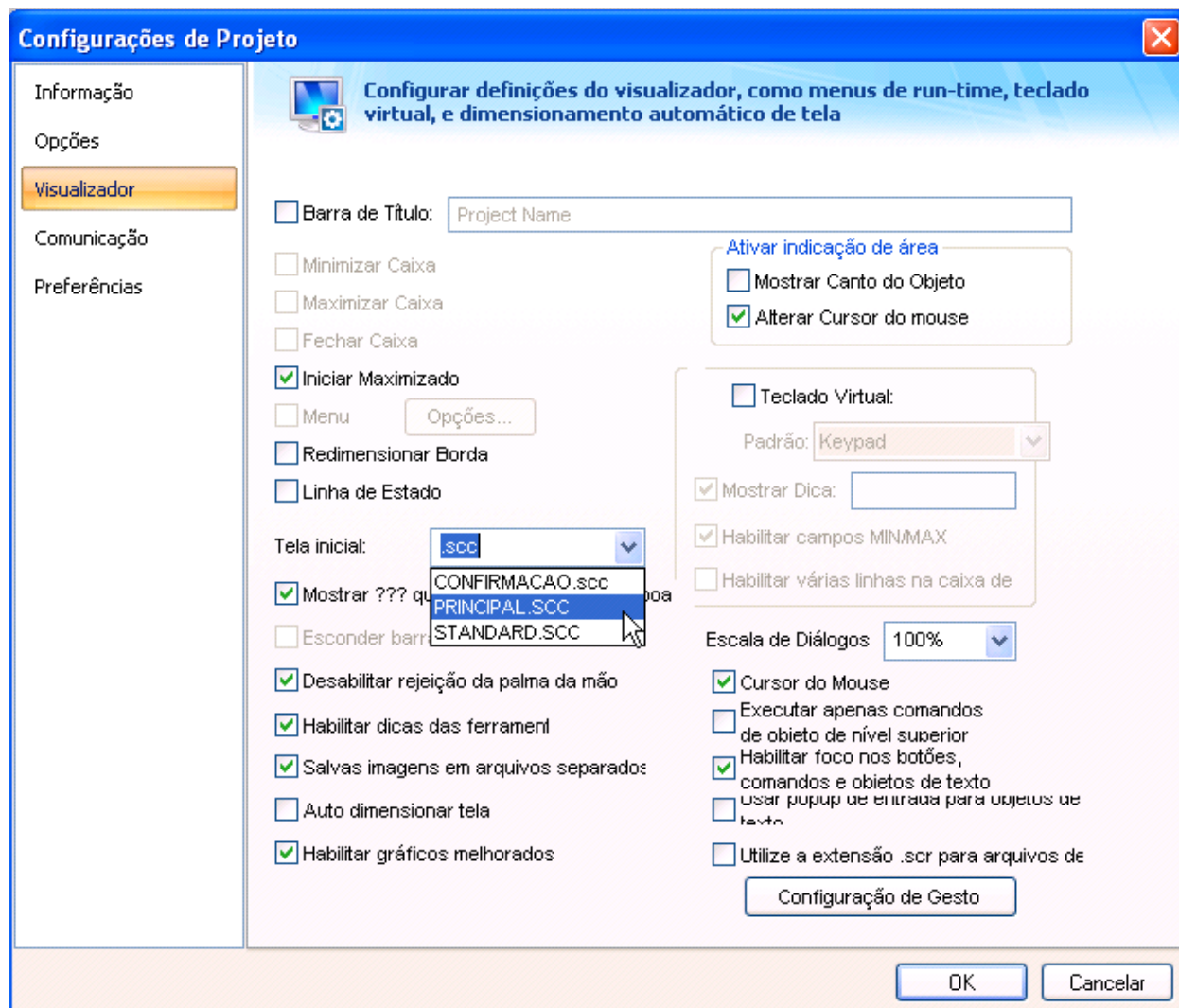


Figura 101.

Salve a aplicação e Execute a aplicação clicando no ícone “**Executar**” ou apertando **F5**.

Durante o teste a única animação que deverá funcionar até este instante será a mudança no número do tanque ao clicar nos botões de subida e descida e o botão **SAIR**, use esse botão para fechar a execução do programa.

Caso alguma dessas funções não esteja funcionando, revise o que foi feito em busca de erros.

Caso o botão **SAIR** não funcione e você, usuário, não consiga sair da tela de execução basta usar o comando **ALT + TAB** para voltar a tela de programação do **InduSoft** e apertar **SHIFT + F5** (encerra a execução) ou apertar o botão **PARE** localizado ao lado do botão **EXECUTAR** no canto superior esquerdo.

ETAPA 4. CONFIGURAR UMA PLANILHA MATEMÁTICA (MATH WORKSHEET) PARA SIMULAR VALORES DE PROCESSO

Agora é hora de desenvolvermos alguns scripts para simular valores de processo na tela **Principal**.

Clique com o botão direito do mouse na pasta "**Matemática**" localizada na aba "**Tarefas**".

Selecione o comando "**Inserir**" para criar uma nova planilha matemática.

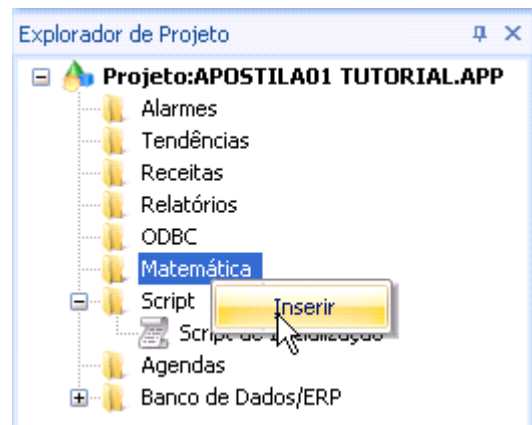


Figura 102.

O campo **Execução** controla a execução da planilha. Pode-se digitar aí um valor, um *tag*, uma expressão ou uma função que, sendo este valor verdadeiro, executa a matemática.

Portanto o nosso campo **Execução** será preenchido com o valor 1. Isto habilita a execução contínua desta planilha uma vez que 1 será sempre um valor verdadeiro (TRUE).

Figura 103.

No corpo da planilha, estaremos simulando:

Os status das válvulas, de acordo com o comando enviado.

O nível, a pressão e a temperatura dos três Tanques

Para simular o status de cada válvula, simplesmente vamos transferir o valor do Comando para o Status, uma vez que não estamos numa aplicação real.

Para os valores de temperatura e pressão brincaremos com as funções trigonométricas de seno e cosseno (sine e cosine).

Para simular a propriedade **Nível** (Nível) de cada tanque, usaremos os status das válvulas de Enchimento e Esvaziamento (fill and empty) para incrementar ou decrementar o valor da variável de **Nível**.

Portanto, com o entendimento acima, vamos configurar a planilha como mostrado abaixo:

	Nome da Tag	Expressão
	 Filtro de Texto	 Filtro de Texto
1	ESTADOENCHE[1]	COMANDOENCHE[1]
2	ESTADOESVAZIA[1]	COMANDOESVAZIA[1]
3	ESTADOENCHE[2]	COMANDOENCHE[2]
4	ESTADOESVAZIA[2]	COMANDOESVAZIA[2]
5	ESTADOENCHE[3]	COMANDOENCHE[3]
6	ESTADOESVAZIA[3]	COMANDOESVAZIA[3]
7	TEMPERATURA[1]	$(\sin((\text{Second}/30)*\pi)+1)*50$
8	TEMPERATURA[2]	$(\sin((\text{Second}/20)*\pi)+1)*50$
9	TEMPERATURA[3]	$(\sin((\text{Second}/10)*\pi)+1)*50$
10	PRESSAO[1]	$(\cos((\text{Second}/30)*\pi)+1)*50$
11	PRESSAO[2]	$(\cos((\text{Second}/20)*\pi)+1)*50$
12	PRESSAO[3]	$(\cos((\text{Second}/10)*\pi)+1)*50$
13	NIVEL[1]	if ((Not EstadoEsvazia[1] and EstadoEnche[1]) and Nivel[1] < 100, Nivel[1] + 1)
14	NIVEL[1]	if ((Not EstadoEnche[1] and EstadoEsvazia[1]) and Nivel[1] > 0, Nivel[1] - 1)
15	NIVEL[2]	if ((Not EstadoEsvazia[2] and EstadoEnche[2]) and Nivel[2] < 100, Nivel[2] + 1)
16	NIVEL[2]	if ((Not EstadoEnche[2] and EstadoEsvazia[2]) and Nivel[2] > 0, Nivel[2] - 1)
17	NIVEL[3]	if ((Not EstadoEsvazia[3] and EstadoEnche[3]) and Nivel[3] < 100, Nivel[3] + 1)
18	NIVEL[3]	if ((Not EstadoEnche[3] and EstadoEsvazia[3]) and Nivel[3] > 0, Nivel[3] - 1)

Figura 104.

Agora já podemos rodar a aplicação e verificar o comportamento do nosso processo.

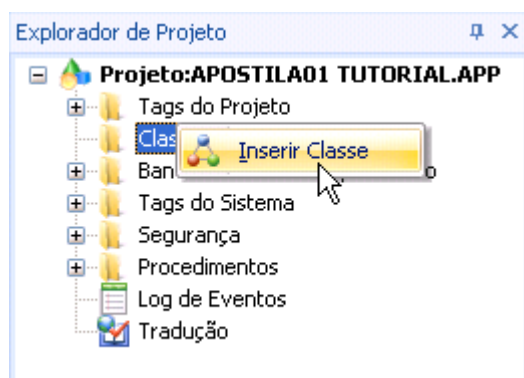
Ao ser executado, as válvulas e os botões de aumentar ou diminuir o número do tanque atual devem funcionar, assim como os níveis de temperatura e pressão de todos os tanques.

ETAPA 5. CRIAR E CONFIGURAR GRUPOS DE ALARMES (ALARMS GROUP)

Para este exercício, você deverá definir os *tags* que serão utilizados na sua aplicação para trabalhar com alarmes.

Vamos criar agora as Classes que conterão os *tags* que manipularão os alarmes.

Para criar a CLASSE, clique com o botão direito na pasta **Classes** na aba Global e escolha “Inserir Classe”.



Na caixa de diálogo, digite o nome da classe, “Alarmes” e clique em OK.



Figura 105.

Na planilha da classe que acabou de surgir, crie os nomes dos Membros da Classe.

	Nome	Tipo	Descrição
1	FILTRO	Booleana	
2	COLUNA	Booleana	
3	ACKALR	Booleana	
4	ACKALL	Booleana	
5	GRUPO	String	
6	TIPO	Inteira	
7	ATIVOS	Inteira	
8	ALARMES	Inteira	
9	NAORECONHECIDOS	Inteira	

Figura 106.

Feche a planilha, ela pode ser fechada sem ser salva.

Com isso teremos a classe criada, no entanto, temos de associá-las a algum *tag*;

Selecione o *tab* Global. Clique duas vezes na pasta “**Tags do Projeto**” para expandi-la.

Clique duas vezes em “**Planilha de Dados**”.

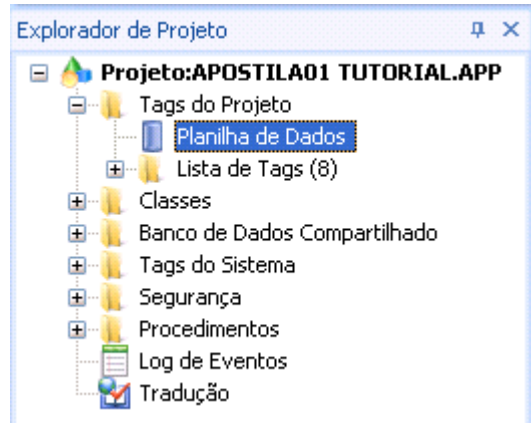


Figura 107.

Configure a janela “**Tags do Projeto**” como mostrado a seguir:

	Nome	Vetor	Tipo	Descrição	Escopo
	Filtro de Texto	= ...	(Todos)	Filtro de Texto	(T...
1	ESTADOENCHE	3	Booleana	VALVULA QUE ENCHE O TANQUE	Local
2	ESTADOESVAZIA	3	Booleana	VALVULA QUE ESVAZIA O TANQUE	Local
3	COMANDOENCHE	3	Booleana	COMANDO QUE ENCHE O TANQUE	Local
4	COMANDOESVAZIA	3	Booleana	COMANDO QUE ESVAZIA O TANQUE	Local
5	TEMPERATURA	3	Real	TEMPERATURA DO TANQUE	Local
6	PRESSAO	3	Real	PRESSAO DO TANQUE	Local
7	NIVEL	3	Real	NIVEL DO TANQUE	Local
8	INDICE	0	Inteira	INDICE DO TANQUE	Local
9	ALARME	0	Alarmes	<<< O TIPO DESSE TAG É ALARMES.	Servidor
*			Inteira		Servidor
*			Inteira		Servidor
*			Inteira		Servidor
*			Inteira		Servidor
*			Inteira		Servidor

Figura 108.

Desta forma você estará criando os *tags* Alarme e Valvula, que servirão de meios de associação entre os “**Tags do Projeto**” e as **planilhas de Classe**, no caso, a **Alarmes**.

Feche a janela “**Tags do Projeto**”.

Para criar um grupo de alarmes, vamos a aba **Tarefas**, e clique com o botão direito na pasta “**Alarmes**” seguido de um clique no botão “**Inserir**”.

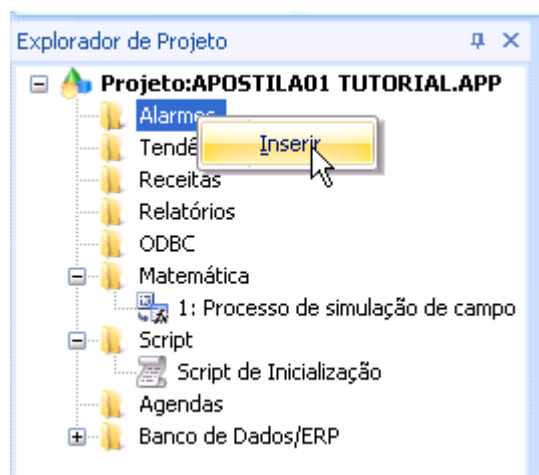


Figura 109.

Configure a planilha como mostrado abaixo:

Descrição:
ALARME DOS TANQUES

Nome do Grupo:
[Campo em branco]

Config E-mail... Avançado...

On Line

- ☒ Exibir nos Control. de Alarme
- ☒ Requer reconhecimento
- ☐ Bip
- ☐ Enviar para a impressora

Histórico

- ☒ Salvar em disco
- ☒ Gerar Msg de Reconhec.
- ☒ Gerar Msg de Normalização

Cores nos Ctrls. de Alarme

Ativação

FG [Cor Vermelha] BG [Cor Preta]

Reconhecimento

FG [Cor Verde] BG [Cor Preta]

Normalização

FG [Cor Azul] BG [Cor Preta]

	Nome da Tag	Tipo	Limite	Mensagem	Prioridade	Seleção
	Filtro de Texto	...	Filtro de Te	Filtro de Texto	Filtro	Filtro d
1	NIVEL[1]	Hi	80.000000	NIVEL ALTO NO TANQUE 1	1	1
2	NIVEL[1]	Lo	20.000000	NIVEL BAIXO NO TANQUE 1	1	1
3	NIVEL[2]	Hi	80.000000	NIVEL ALTO NO TANQUE 2	2	2
4	NIVEL[2]	Lo	20.000000	NIVEL BAIXO NO TANQUE 2	2	2
5	NIVEL[3]	Hi	80.000000	NIVEL ALTO NO TANQUE 3	3	3
6	NIVEL[3]	Lo	20.000000	NIVEL BAIXO NO TANQUE 3	3	3

Figura 110.

Criando esta planilha, você está informando ao sistema quais *tags* devem ter a função de alarmes, que tipo de alarmes, quais os seus limites, as mensagens para tais, prioridades e filtros (dados).

A caixa **“Salvar em disco”** permite escolher salvar o histórico destes alarmes em disco, em formato ASCII na pasta alarmes da aplicação. Também selecione as opções abaixo deste.

Configure o botão **Avançado** conforme a figura a seguir:



Figura 111.

Após isso, salve no botão **Salvar** no canto superior esquerdo. Agora criaremos a tela de alarmes.

ETAPA 6. CRIAR E CONFIGURAR TELA DE ALARMES (ALARM SCREEN)

1ª. PARTE: ALARME HISTÓRICO

Abra a tela “**STANDARD**” (que possui apenas os retângulos de data, hora, tela atual e botões básicos) e desenhe um objeto de “**Alarme/Evento**” como se fosse inserir um botão.



Figura 112.

Configure a tela como mostrado a seguir, inserindo os objetos:

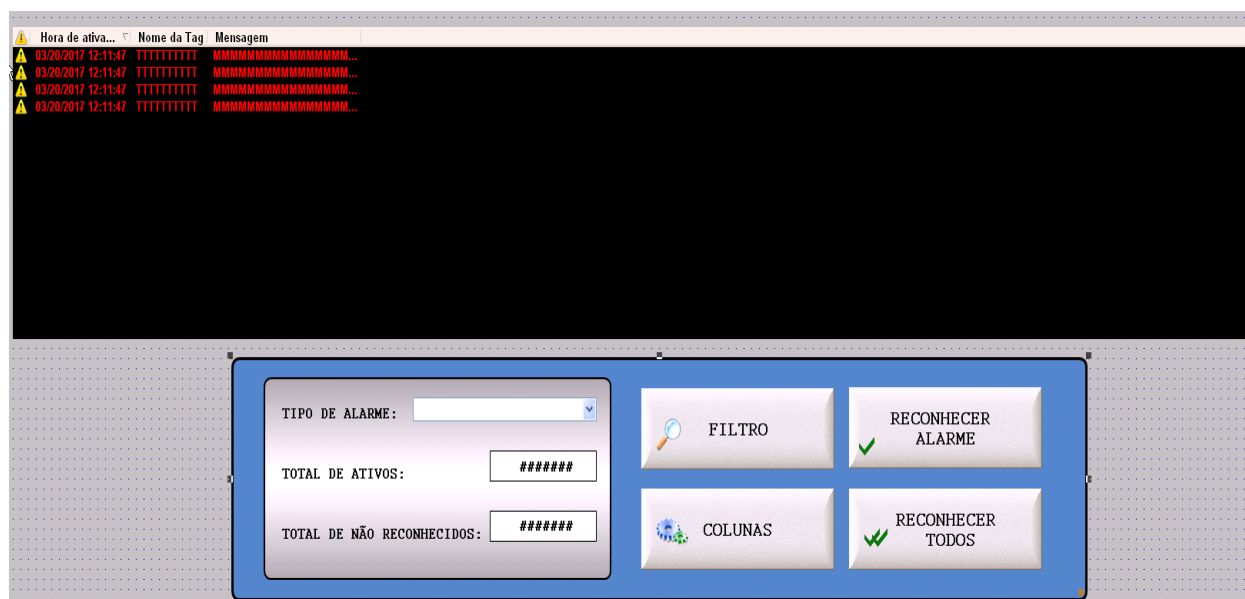


Figura 113.

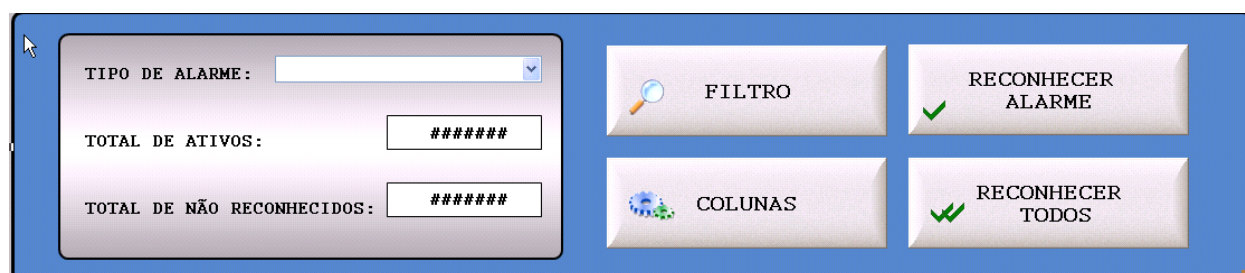


Figura 114.

Para desenhar os botões de **FILTRO**, **COLUNAS**, **RECONHECER ALARME** e **RECONHECER TODOS**, use a função “Botão” normalmente, e para seus respectivos ícones, como vistos na imagem anterior, utilize a **Biblioteca**. Os ícones desejados estão localizados na pasta “Icons”.

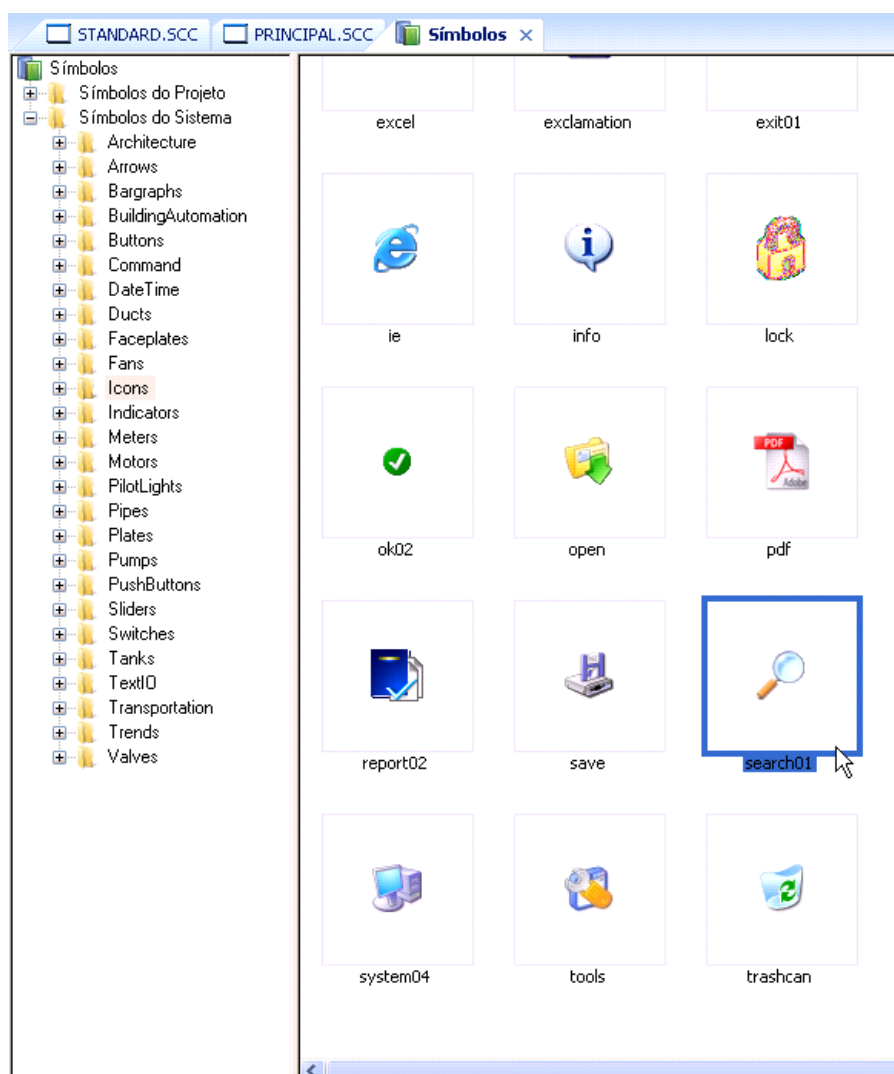


Figura 115.

Insira a propriedade “**Comando**” nos quatro botões criados e os configure como visto a seguir:

No botão “**Filtro**”:

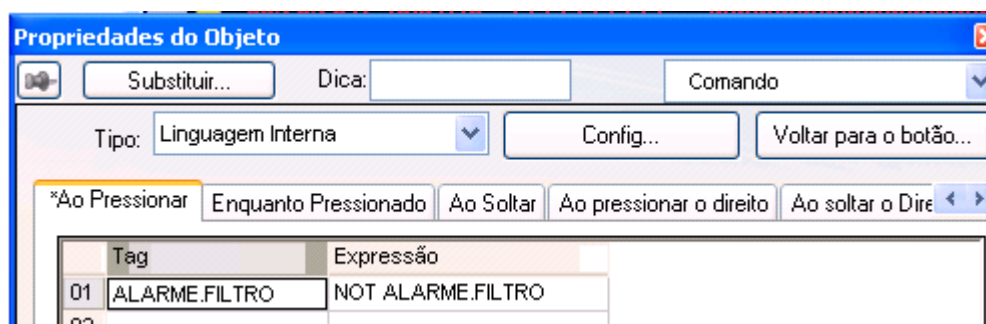


Figura 116.

No botão “**Colunas**”:

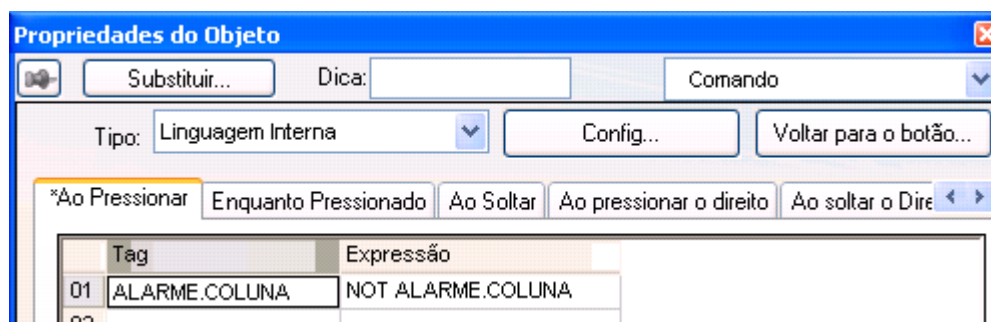


Figura 117.

No botão **“Reconhecer Alarme”**:

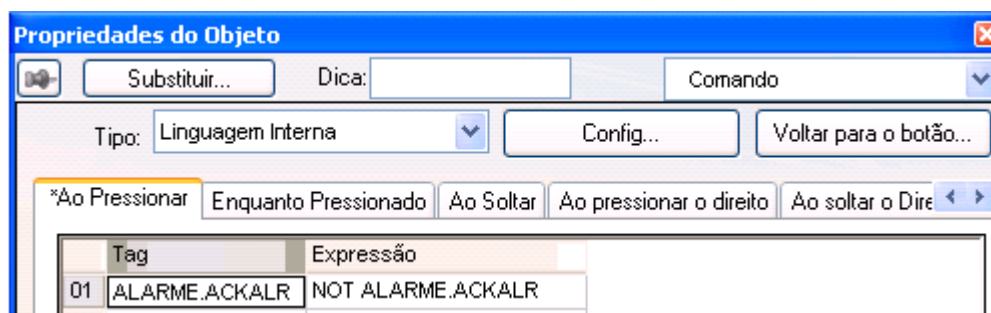


Figura 118.

No botão **“Reconhecer Todos”**:

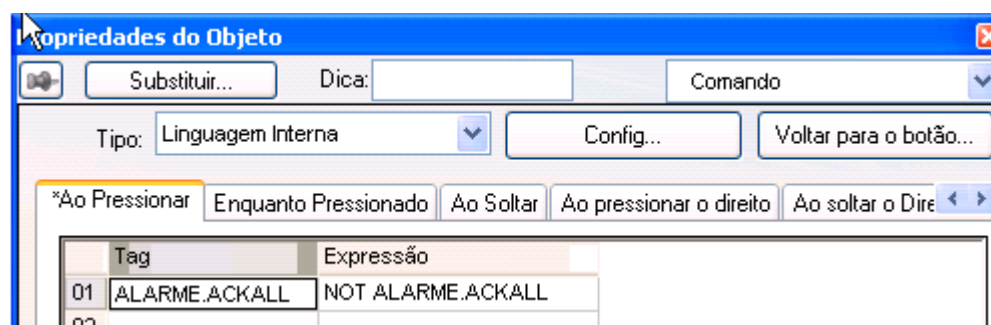


Figura 119.

Para desenhar o COMBOBOX usado no **“Tipo de Alarme”**, utilize o ícone mostrado a seguir localizado ao lado de Alarme/Evento.

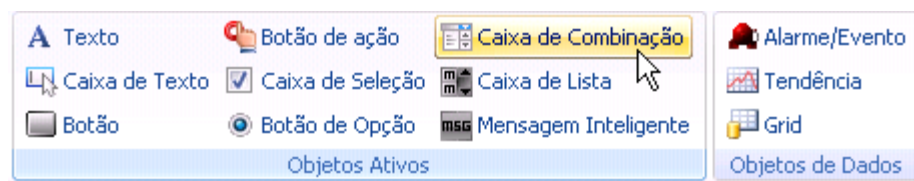


Figura 120.

E clique e arraste como se estivesse criando um botão e o posicione como mostrado a seguir:

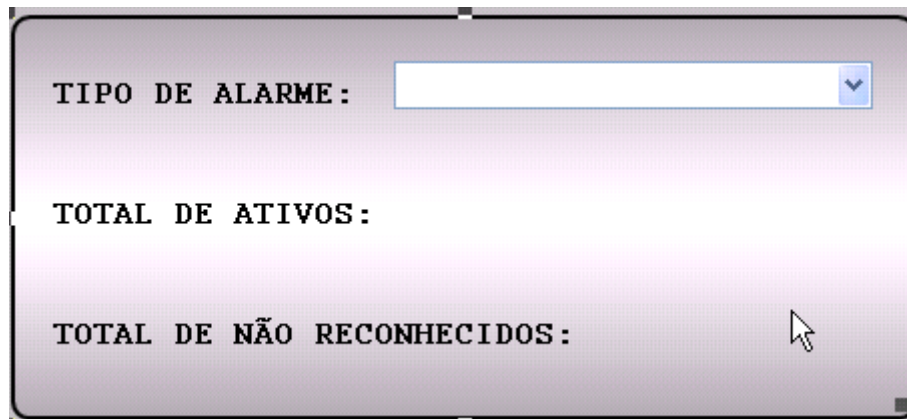


Figura 121.

Depois disso, adicione dois campos de “**Texto**” com a propriedade “**Texto com Link de Dados**”. Neste caso criaremos posteriormente dois retângulos brancos para criar um destaque maior.

No texto de “**TOTAL DE ATIVOS**” e após adicionar a propriedade “**Texto com Link de Dados**” colocaremos como expressão deste a seguinte frase: “**ALARME.ATIVOS**”.

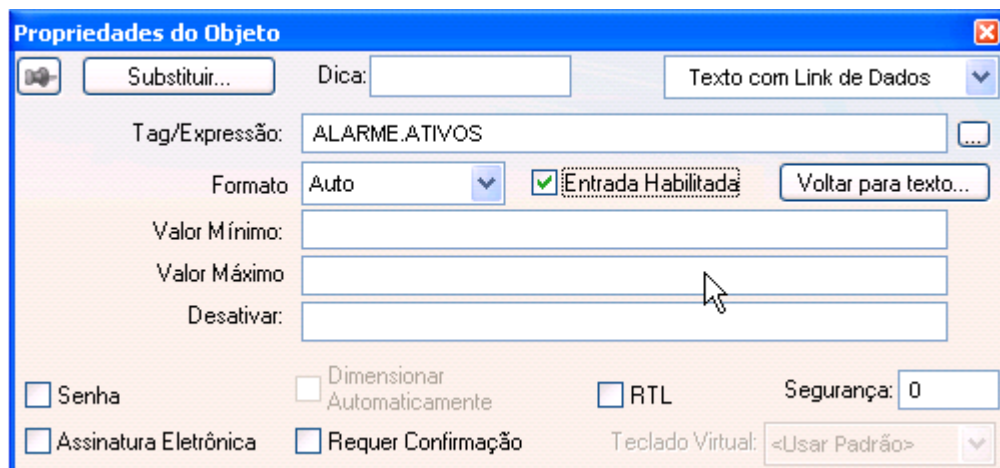


Figura 122.

Já no texto “**TOTAL NÃO RECONHECIDOS**” com a propriedade “**Texto com Link de Dados**”, colocaremos como expressão deste a seguinte frase: “**ALARME.NAORECONHECIDOS**”.

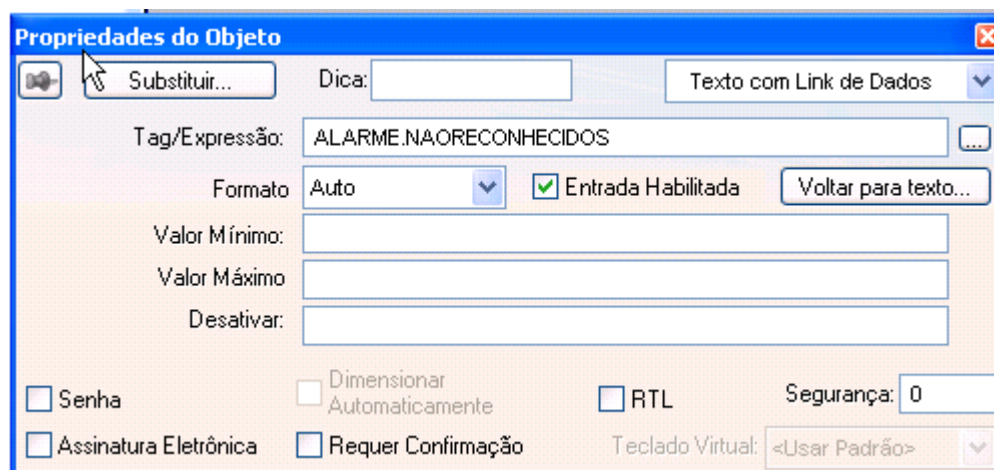


Figura 123.

Duplo-clique no objeto de **Alarme** para entrar nas suas propriedades e configure a janela “**Propriedades do Objeto**” como mostrado abaixo.

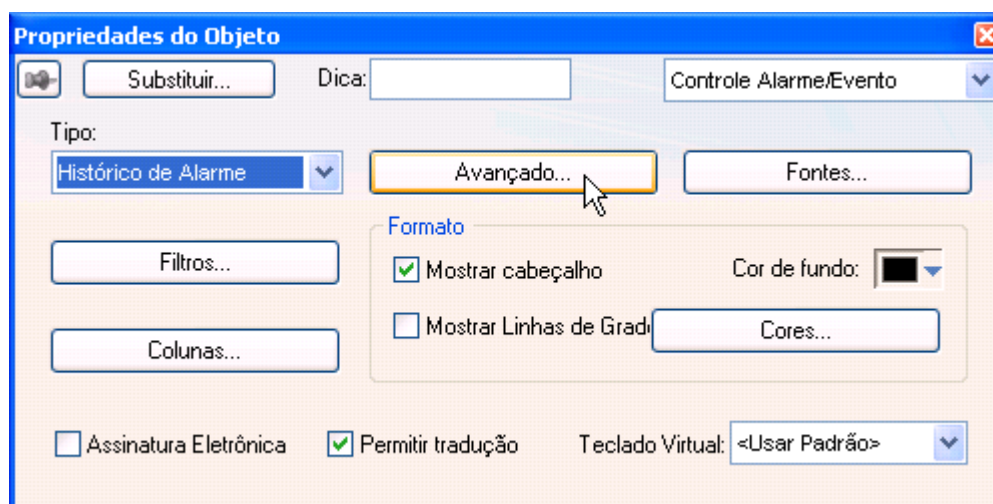


Figura 124.

Esteja certo de que o “**Tipo**” está como “**Histórico de Alarme**”. Então, clique no botão “**Avançado...**” e configure como mostrado abaixo.

Avançado

Formato Data & Hora

☒ Dia ☒ Mês ☒ Ano
☒ Hora ☒ Minuto ☒ Segundo ☐ MS

Exemplo: 03/20/2017 11:47:33

Deletar mensagem

Segurança: 0
☒ Confirmar

Reconhecimento

Reconhecer todos:
Reconhecer:
Comentar: Desativar
Desab. no duplo clique:

Valores retornados do Run-time

Total de itens: ALARME.ATIVOS
Tag Seleccionada:
Texto da Primeira:
Linha de Texto:
Resumo das:

Disparos em Runtime

Colunas: ME.COLUNA
Filtros: ME.FILTRO

Opções

☒ Auto Formatar

Salvar / Imprimir

Disparo:
Disparo PDF:
Nome de:

Naveg. de Disparos... OK Cancelar

Figura 125.

Em “Colunas” usamos “ALARME.COLUNA” e em “FILTROS” usamos “ALARME.FILTRO”

Finalizando essa configuração clique em OK.

Queremos **DUAS TELAS DE ALARMES**, uma de “**Alarmes On-line**” que estaremos criando na próxima etapa e a outra de “**Alarme Histórico**”, que estamos criando no momento. Para permitir que apenas uma tela de alarmes dessas esteja presente para visualização do operador faremos uma configuração de visibilidade.

Configuraremos agora a propriedade “**Visibilidade/Posição**” no objeto de alarme.

Selecione a caixa de Alarme (segure **shift** e clique na caixa ou apenas clique uma vez na caixa) e clique na propriedade a seguir:

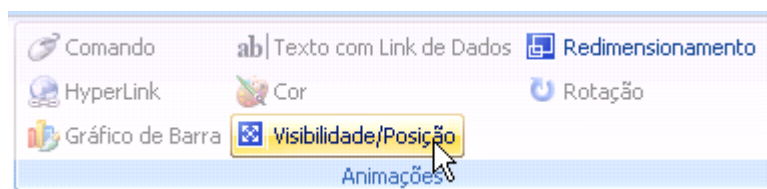


Figura 126.

Agora dê dois cliques na janela de alarme e configure a propriedade “**Visibilidade**” como mostrada a seguir:

Nota-se que logo acima do cursor a palavra “**Visibilidade**” não está escrita na íntegra na janela de “**Propriedades do Objeto**”

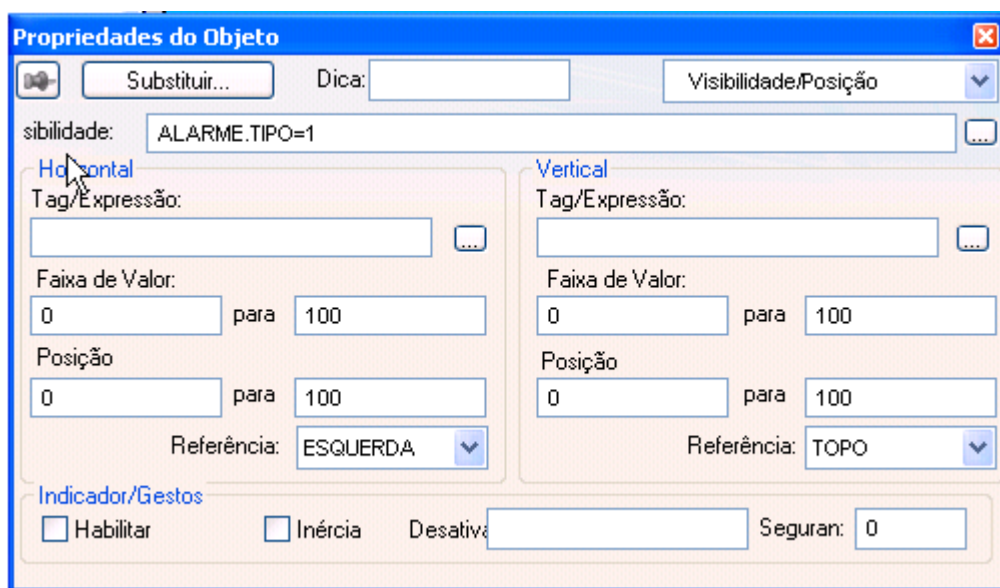


Figura 127.

Salve a tela como **ALARMES**. Para **salvar**, clique no botão localizado no canto superior esquerdo da tela, “**Salvar Como**”, digite o nome **ALARMES** no campo **Nome do arquivo**, e clique em Salvar.

2ª. PARTE: ALARME ON-LINE

Na mesma tela “**Alarmes**” recém desenhada com o objeto de **Alarme/Evento** faremos o desenho de outro objeto sobre o já desenhado. Basta clicar no **Alarme** já posicionado (para selecioná-lo), segurar **Ctrl**, clicar e arrastar pra o lado, assim criando outro **Alarme** idêntico a esse. A copia será idêntica, mas apenas por praticidade, pois iremos editar este objeto.

A figura a seguir mostra os dois objetos de alarme na mesma tela.

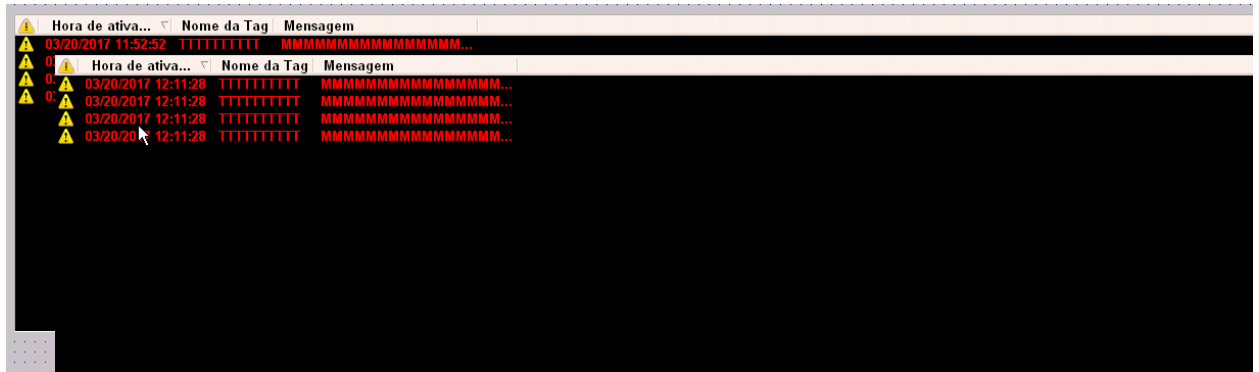


Figura 128.

Após ter criado o novo objeto de **Alarme** siga o seguinte procedimento:

Duplo-clique no objeto de Alarme e configure a janela “**Propriedades do Objeto**” como mostrado abaixo.

Primeiro mudaremos a opção “**Visibilidade**” para “**ALARME.TIPO=0**”. Feito isso iremos para as configurações do **Alarme** em si.

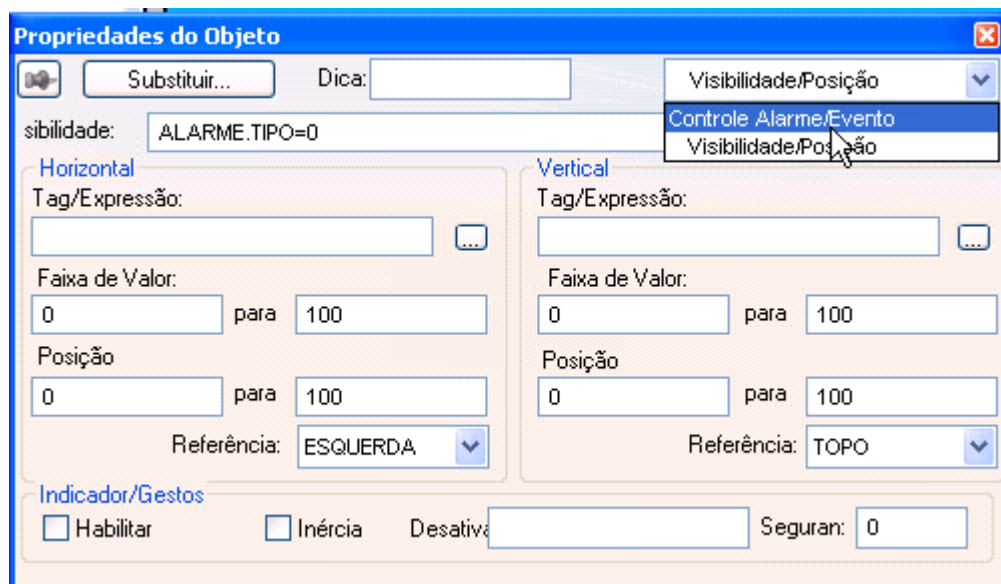


Figura 129.

Esteja certo de que a opção “**Tipo**” esteja em “**Alarme Online**”.
Clique no botão “**Avançado...**” o qual possui alguns itens a serem configurados.

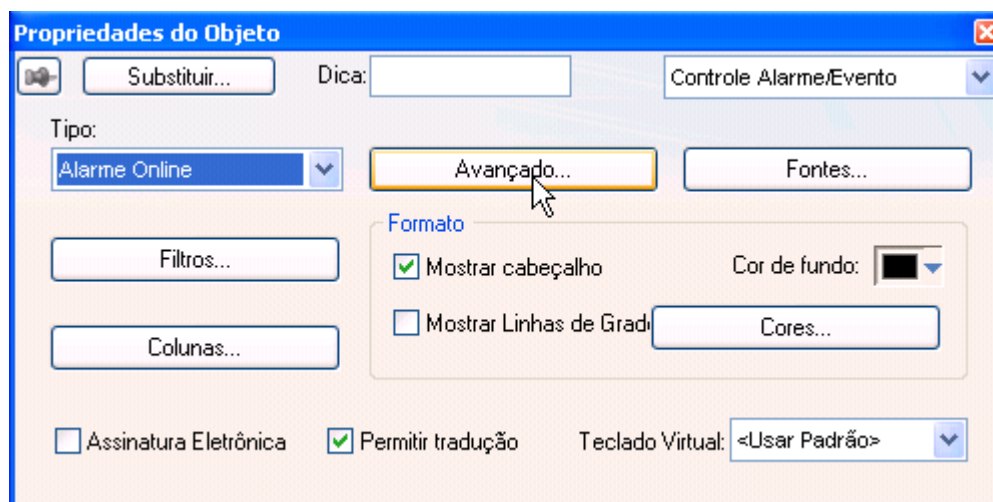


Figura 130.

Configure a janela “**Avançado...**” como mostrado a seguir:



Figura 131.

Finalizando essa configuração clique em OK.

Feche a janela e **Salve** a tela como **Alarmes**.

RECONHECIMENTO DE ALARMES

Existem duas maneiras de se reconhecer os Alarmes:

Trigando o *tag* que estaria no campo “**Reconhecimento Remoto**”, da planilha de alarmes:

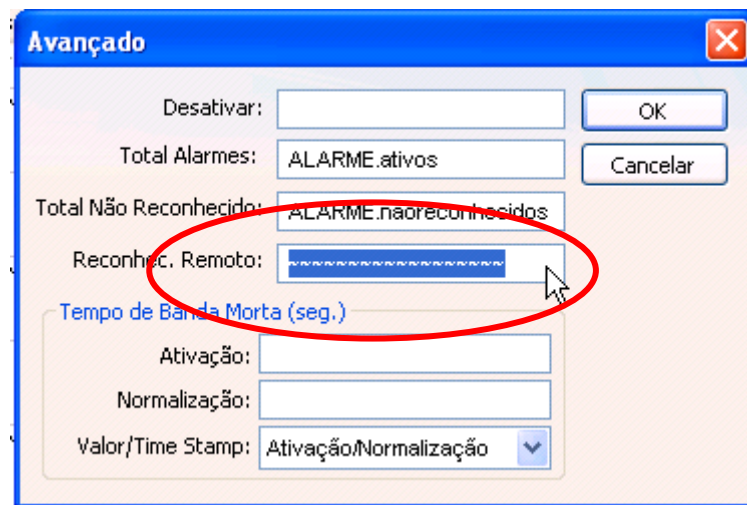


Figura 132.

Ou trigando um *tag* que estaria no campo “**Reconhecimento**” dentro de “**Propriedades do Objeto/Avançado...**” no objeto de **Alarme**.

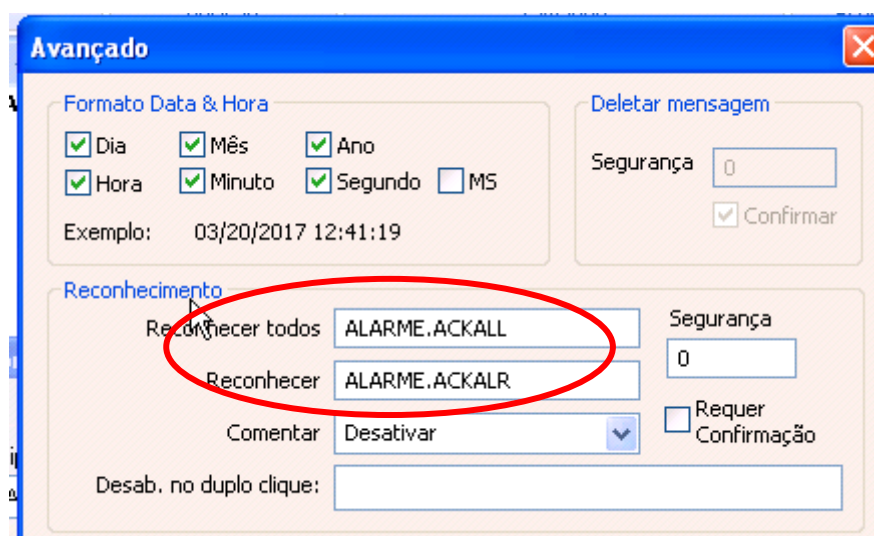


Figura 133.

Existem três *tags* internos que estudaremos agora: **Alarm**, **AckAlr** e **AckAll**.

O *tag* **AckAlr** reconhece o alarme que está ativo no Sistema. Trigando **AckAlr** o ultimo alarme ativo na lista é reconhecido.

Já o *tag* interno **AckAll** reconhece todos os alarmes ativos, independentes do grupo que eles pertençam.

Neste procedimento, usaremos estes *tags* internos.

Verifique se o botão “**Reconhecer Alarme**” e o “**Reconhecer Todos**” estão configurados como nas duas figuras a seguir:

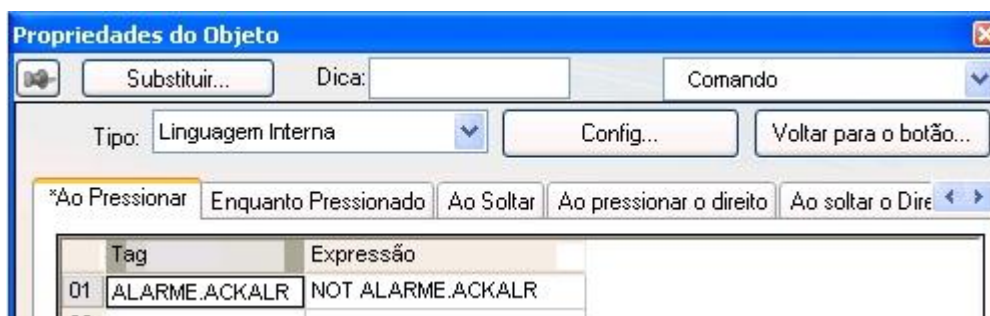


Figura 134.



Figura 135.

Vamos à configuração do objeto da **Caixa de Combinação**(*ComboBox*):

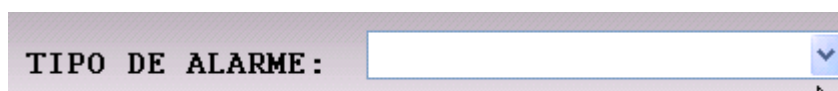


Figura 136.

Clique duas vezes no objeto e configure a janela “**Propriedades do Objeto**” como mostrado abaixo.

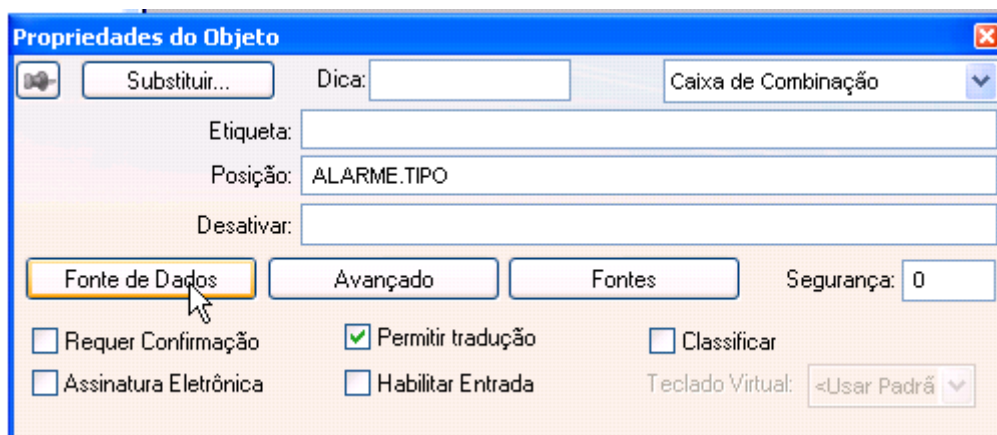


Figura 137.

Clique sobre o botão **"Fonte de Dados"**, e faça como abaixo:

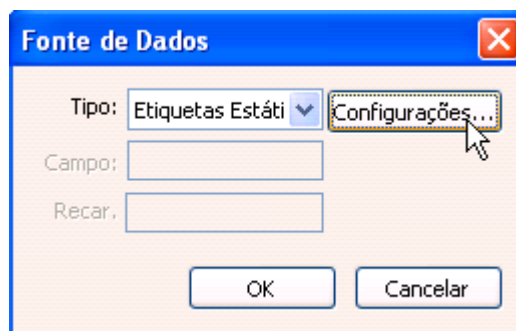


Figura 138.

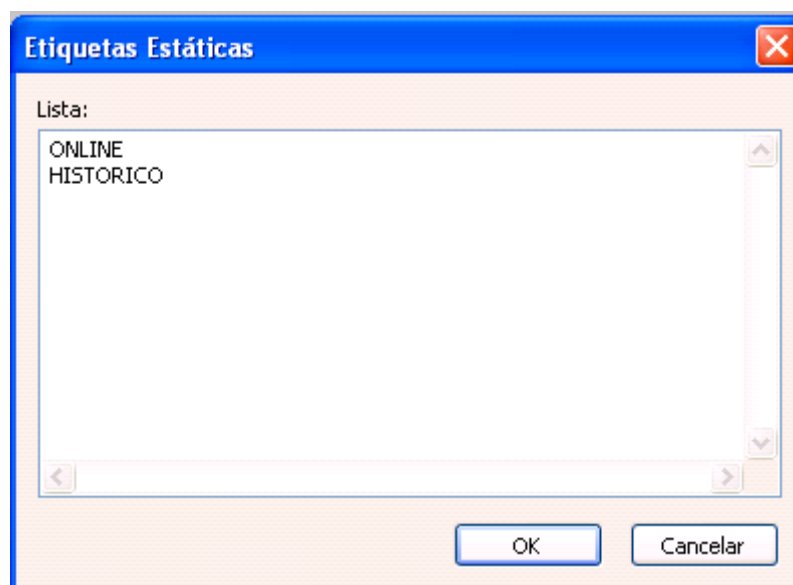


Figura 139.

A parte de histórico dos alarmes mostrará todos os alarmes que aconteceram e foram gravados em disco, de acordo com um período definido como parâmetro.

Clique em Ok e feche a janela.

FILTRO DE ALARMES

O **IWS** permite que você configure um sistema de filtros dinâmicos para a localização e apresentação de mensagens de alarmes. Certifique-se que o botão “**Filtro**” está configurado como a seguir:



Figura 140.

E que o botão “**Colunas**” desse jeito:

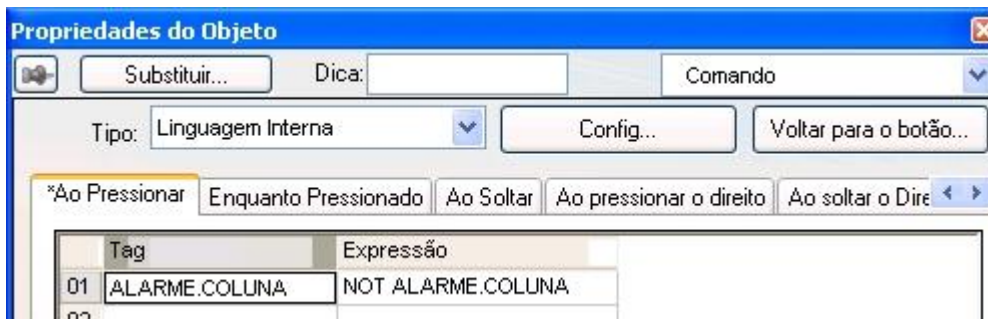


Figura 141.

Feche a janela e **Salve** a tela como **Alarmes**.

ETAPA 7. CRIAR E CONFIGURAR TELA DE TREND

A tarefa **Trend** guarda um caminho traçado pelas variáveis da aplicação. Você pode guardá-las em disco ou ainda mostrá-las em forma de gráficos.

Nosso **Trend Online** mostrará algumas variáveis e será atualizado a cada segundo.

Abra a tela **STANDARD** com apenas os botões, retângulos com data, hora e nome da tela.

Insira um objeto **Trend** através do botão “**Tendência**” e configure-o como mostrado a seguir:

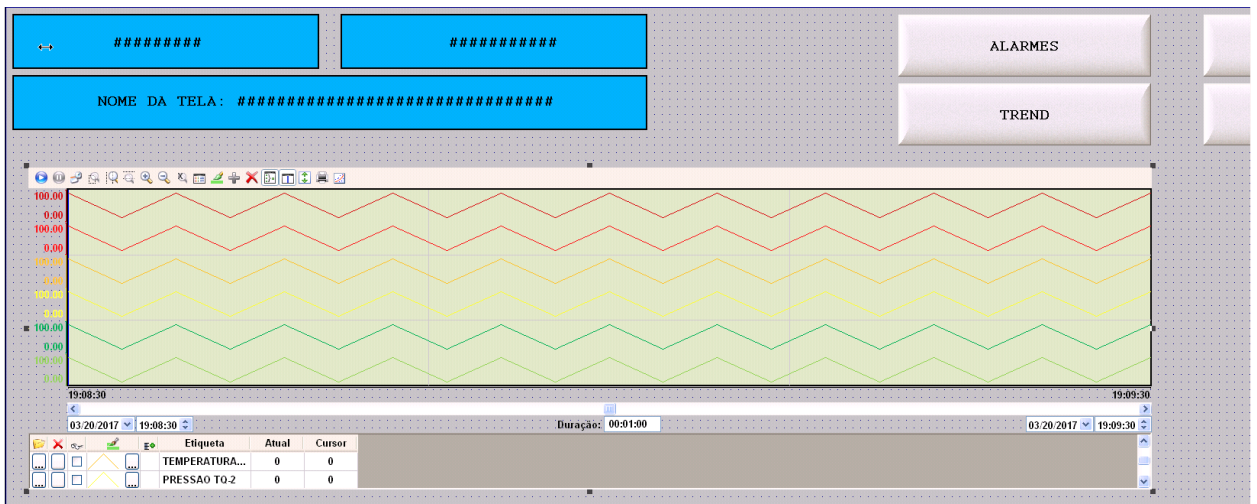


Figura 142.

Para chegar nessa configuração iremos:

Clique duas vezes no objeto e visualize a tela “**Propriedades do Objeto**” como abaixo:



Figura 143.

Clique no botão “Pontos...” e edite como mostrado abaixo:

Pontos										
Ponto	Etiqueta	Cor	Fonte Dados	Tag/Campo	Escala Min.	Escala Max.	Estilo	Opções	Histórico	Ocultar
1	TEMPERATURA TQ-1		Tag	TEMPERATURA[1]						
2	PRESSAO TQ-1		Tag	PRESSAO[1]						
3	TEMPERATURA TQ-2		Tag	TEMPERATURA[2]						
4	PRESSAO TQ-2		Tag	PRESSAO[2]						
5	TEMPERATURA TQ-3		Tag	TEMPERATURA[3]						
6	PRESSAO TQ-3		Tag	PRESSAO[3]						
7			Tag							

Figura 144.

Clique em OK.

Agora temos que dizer para o sistema quais os *tags* que queremos gravando seus valores em disco, e de quanto em quanto tempo. Clique com o botão direito na pasta “Tendências” (aba “Tarefas”) e clique em **Inserir**.

Configure a planilha como mostrada a seguir.

Descrição:

Formato Histórico:

Proprietário

Avançado...

☒ Salvar ao Disparo:

Second

☐ Salvar Na Mudança da Tag

	Nome da Tag	Banda Morta
	Filtro de Texto	Filtro de Text
1	TEMPERATURA[1]	
2	TEMPERATURA[2]	
3	TEMPERATURA[3]	
4	PRESSAO[1]	
5	PRESSAO[2]	
6	PRESSAO[3]	
*		
*		
*		
*		
*		

Figura 145.

Assim você está preparando o sistema para salvar os valores dos *tags* de temperatura e pressão dos Tanques, toda vez que o *tag* **Gravar** mudar de valor, e guardando isso em disco por até 700 dias.

A caixa “**Salvar ao Disparo**” define a taxa de gravação.

Clique no botão “Advanced” e edite como mostrado abaixo:

Configurações Avançadas de Tendência

Lote

Iniciar/Parar: Existente:

Nome: Descrição:

Deletar: ☒ Salvar dados mesmo que Lote não estiver executando

Controle de Espaço de Disco

Tempo de Histórico(dias): Comprimir Depois(dias):

Qualidade Ruim:

Tipo: Deslocamento:

Desabilitar Todo Salvamento de Dados:

OK Cancelar

Figura 146.

Clique em OK e salve a planilha.

Agora ao executar o programa todas as funções configuradas devem estar funcionando.

Barras de gráfico de Temperatura e Pressão com ajuda das funções de seno e coseno, Nível com ajuda da válvula, botões para ir até a tela de Trend e Alarmes, e tela de confirmação ao apertar o botão sair.

2º. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA DE CONTROLE COM SOFTWARE INDUSOFT E CLP STARDOM

O objetivo deste trabalho prático visa conhecer as ferramentas para programação do CLP Stardom utilizando instruções de contagem e temporização com software LOGIC DESIGNER e construção de aplicação para supervisão, ajustes e monitoramento de valores utilizando os softwares do sistema VDS.

ETAPA 1. CONFIGURAÇÃO DE HARDWARE



Duplo clique no ícone abaixo:

Se o ícone não estiver disponível na área de trabalho, siga o seguinte caminho:

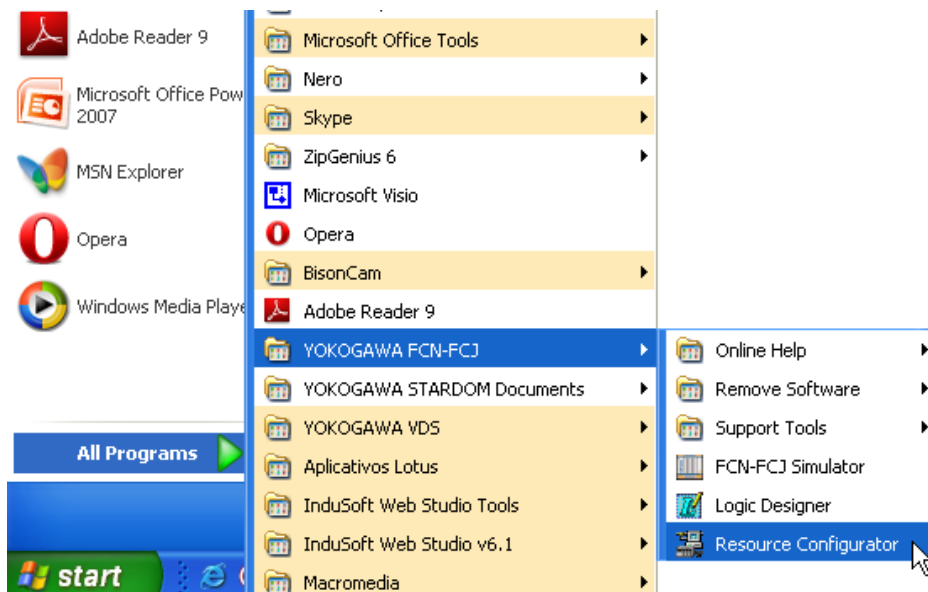


Figura 147.

Selecionar a opção “**Connection**” no menu “**File**”, conforme figura a seguir.

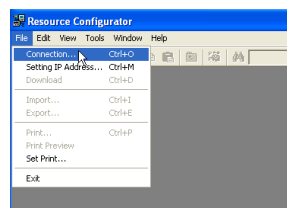


Figura 148.

Entre com o número de IP “10.111.203.12X” no campo “**Host**”, “stardom” no campo “**User Name**” e “YOKOGAWA” no campo “**Password**” e clique em “OK”.

Obs.: X=Número variável de acordo com a bancada. Vide a figura abaixo exibindo a janela “Connection”.

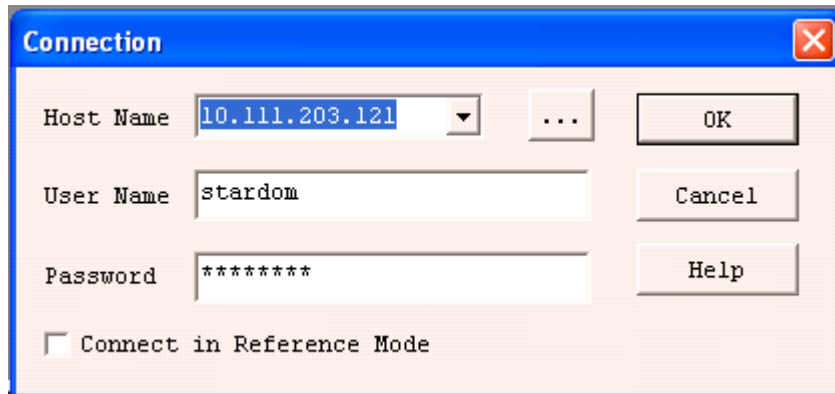


Figura 149.

O “Device Label” é um rótulo dado às entradas e saídas do sistema. Assim, torna-se mais familiar a manipulação de variáveis de campo, sabendo-se o seu nome de projeto.

Com o resource configurator em modo “tree” clique em “IOM”.

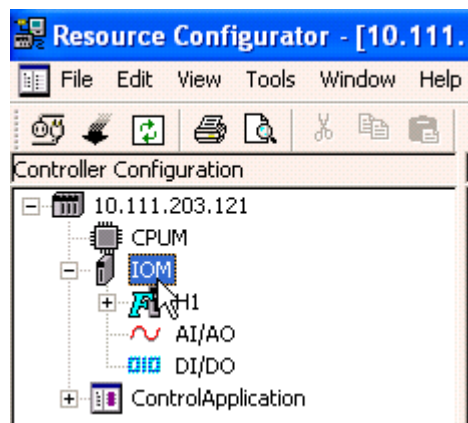


Figura 150.

Duplo-clique em DI/DO.

Ajuste o device label correspondente a 1ª. entrada digital para “BOTA01”.

Ajuste o device label correspondente a 2ª. entrada digital para “BOTA02”.

Ajuste o device label correspondente a 3ª. entrada digital para “BOTA03”.

Ajuste o device label correspondente a 4ª. entrada digital para “BOTA04”.

Ajuste o device label correspondente a 1ª. saída digital para “LAMP1”.

Ajuste o device label correspondente a 2ª. saída digital para “LAMP2”.

Ajuste o device label correspondente a 3ª. saída digital para “LAMP3”.

Ajuste o device label correspondente a 4ª. saída digital para “LAMP4”.

Selecione File-Download

Quando aparecer uma janela de prompt, clique no OK.

ETAPA 2. CRIAÇÃO DA APLICAÇÃO DE CONTROLE (PROGRAMA DO CLP)

INICIANDO UM PROJETO

Neste momento será iniciado o desenvolvimento da criação de uma aplicação de controle.

Para iniciar o Logic Designer, clique no ícone na área de trabalho como exibido na figura ao



Selecione “File-New Project”.

Selecione o ícone “STARDOM NPAS” e clique “OK”.

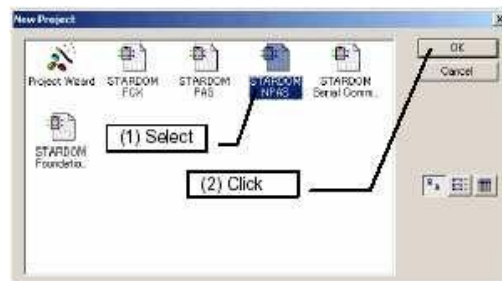


Figura 151.

ESPECIFICANDO O “TARGET” DA FCN/FCJ

Duplo-clique em “TargetSetting” na árvore do projeto.

Entre com o número de IP “10.111.203.12X” e clique em “OK”. Obs.: X=Número variável de acordo com a bancada.

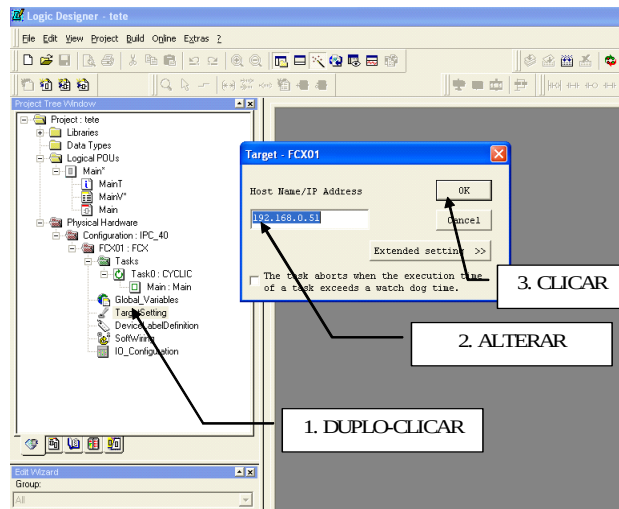


Figura 152.

DEFININDO DEVICE LABEL PARA AS VARIÁVEIS

Nesta aplicação iremos utilizar duas “variáveis globais” que farão as vezes dos sinais de entrada e saída (digitais) deste exercício.

Duplo-clique sobre “DeviceLabelDefinition” na árvore do projeto.

Dentro da janela defina as variáveis como descritas abaixo. Após finalizar os ajustes das variáveis, clique em “Ok” e clique em “YES” na janela de diálogo de confirmação. Os valores da coluna “Address” não precisam ser preenchidos, pois os mesmos são automáticos.

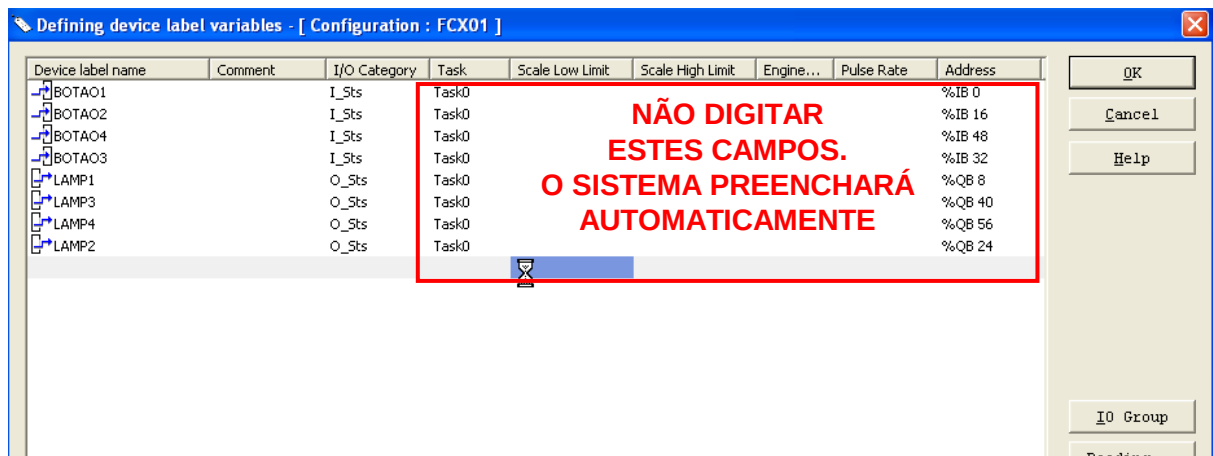


Figura 153.

CRIANDO UM PROGRAMA

Iremos agora construir uma aplicação de controle usando a unidade POU (Program Organization Unit).

Clique com o botão direito sobre “Logical POU’s” na árvore do projeto e então selecione “Insert – Program” para aparecer a janela exibida abaixo.

Neste exercício, insira no campo “Name” a informação “COMANDOS” e selecione “LD” (LADDER) no campo “Language” e clique em “Ok”.

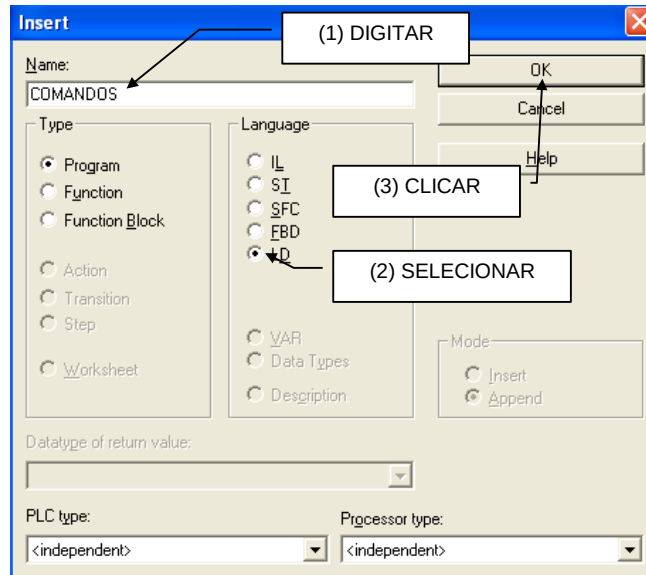


Figura 154.

Verifique no seu projeto se como na figura abaixo a estrutura com três ícone foram criados:

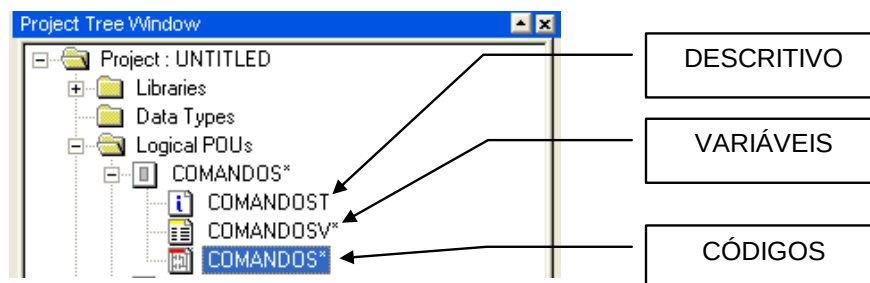


Figura 155.

Duplo-clique sobre “COMANDOS” (ícone de código) criado no projeto.

Clique numa posição específica dentro da área de trabalho a fim de determinar onde ficará a linha de programa. Em seguida, clicar sobre o ícone “Contact Network” para iniciar uma linha básica de programa.

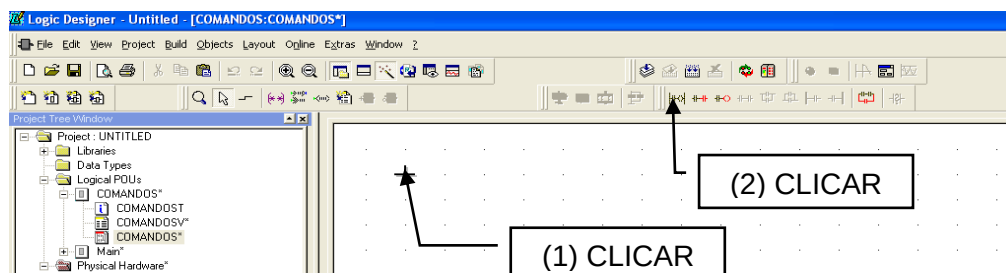


Figura 156.

Duplo-clique no contato aberto da linha de programa, conforme mostrado na figura a seguir. Em seguida, no campo “Global Variable Groups:”, selecione “Physical Hardware–Configuration –FCX01–Default”. Mude a opção dentro de “Scope” para “Global” e clique na opção “Show all variable of worksheet”. Selecione o tag “BOTAO1_BOOL” na opção “NAME”. Não esqueça de ativar a opção “OPC”, que permitirá a comunicação com o supervisor. Clique em “OK”.

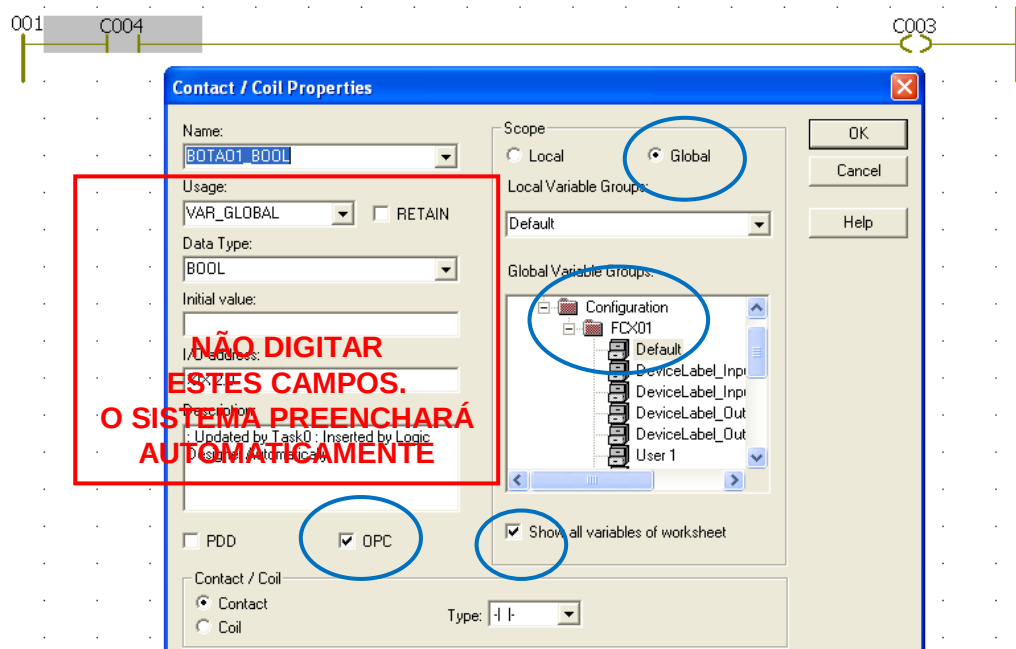


Figura 157.

É importante informar que ao definir o tag “BOTAO1_BOOL” como variável vinculada a instrução “Contato-Aberto” automaticamente o sistema cria uma variável de escopo Local com o mesmo nome. Esta variável de escopo local é um espelho da mesma variável externa de escopo Global.

Clique sobre o contato recém criado com a variável “BOTAO1_BOOL” e em seguida, clicar sobre o ícone “Contact Right” para inserir mais um contato a direita do anterior, na mesma linha de programa.

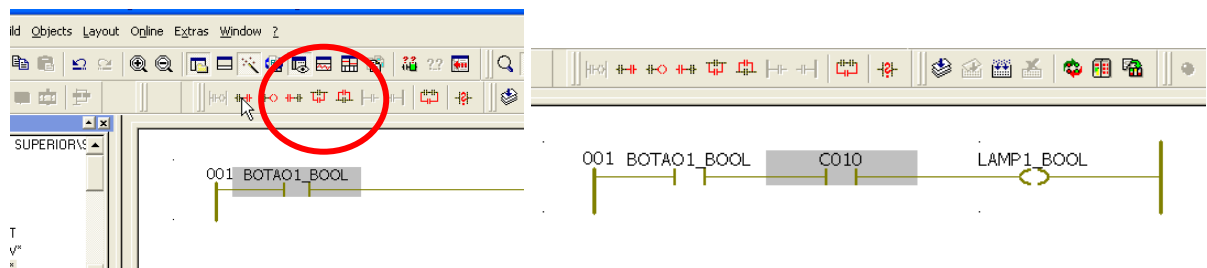


Figura 158.

Duplo-clique no novo contato aberto da linha de programa, conforme mostrado na figura a seguir. Em seguida, no campo “Global Variable Groups:”, selecione “Physical Hardware–Configuration –FCX01–Default”. Mude a opção dentro de “Scope” para “Global” e clique na

opção “Show all variable of worksheet”. Selecione o tag “BOTAO2_BOOL” na opção “NAME”. Não esqueça de ativar a opção “OPC”, que permitirá a comunicação com o supervisão. Clique em “OK”.

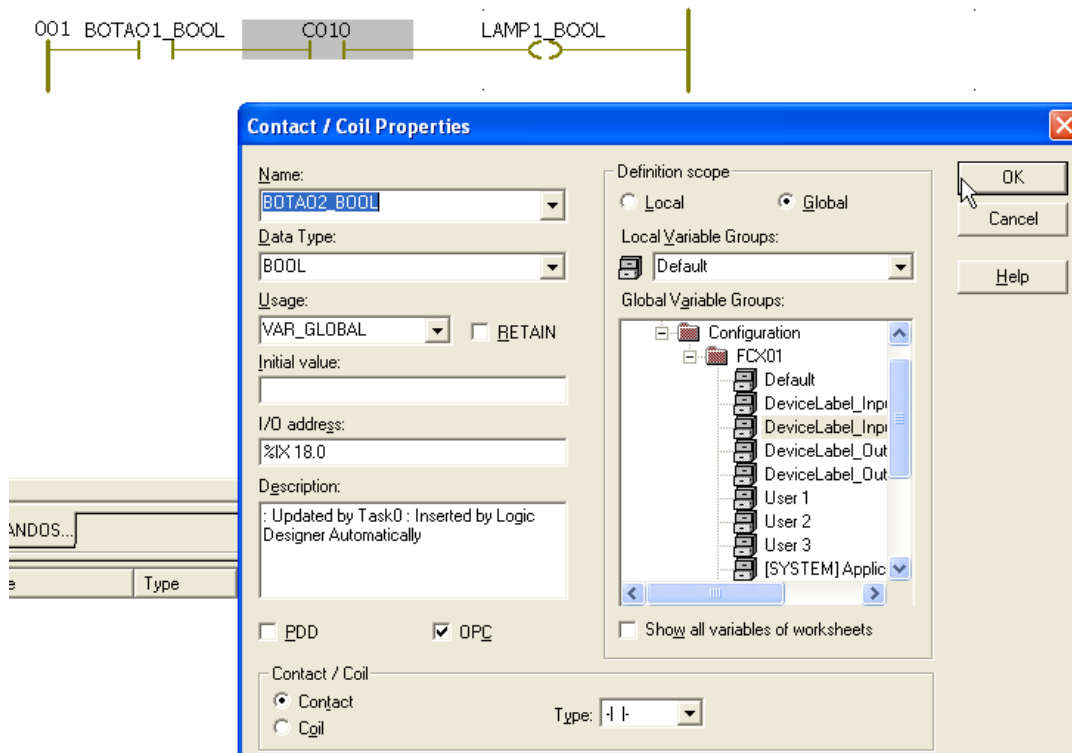


Figura 159.

Duplo-clique na bobina da linha de programa, conforme mostrado na figura a seguir. Em seguida, no campo “Global Variable Groups:”, selecione “Physical Hardware– Configuration –FCX01–Default”. Mude a opção dentro de “Scope” para “Global” e clique na opção “Show all variable of worksheet”. Selecione o tag “LAMP1_BOOL” na opção “NAME”. Não esqueça de ativar a opção “OPC”, que permitirá a comunicação com o supervisão. Clique em “OK”.

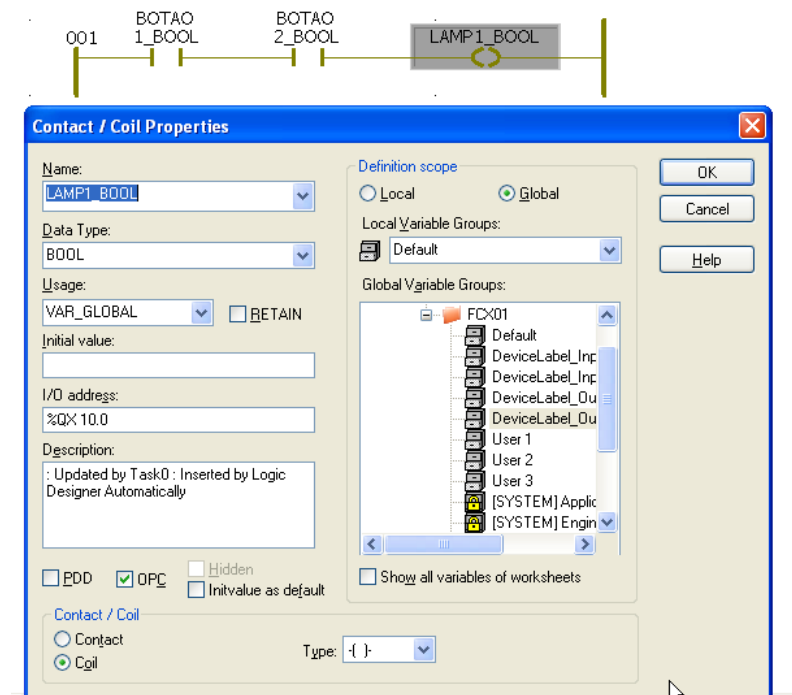


Figura 160.

Crie uma nova linha de programa idêntica a mostrada no passo anterior substituindo por BOTAO3_BOOL, BOTAO2_BOOL e LAMP2_BOOL.

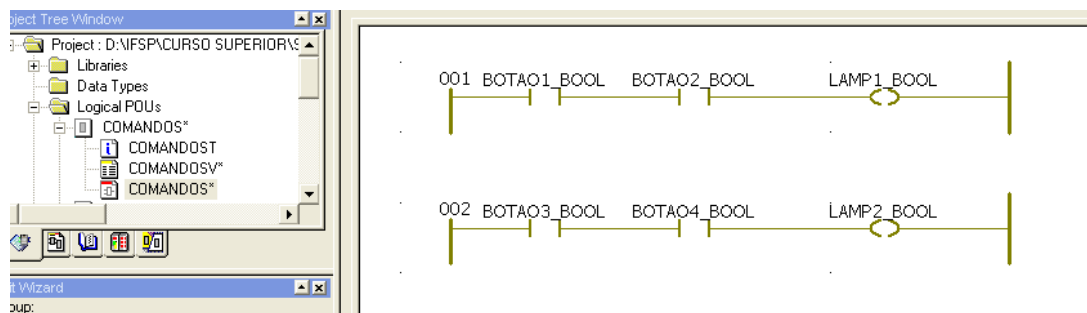


Figura 161.

AJUSTANDO UMA “TASK”

Selecione “Task0” exibida na árvore do projeto. Clique com o botão direito do mouse e selecione “Properties”. Task0 é do tipo “cyclic”. Após confirmação do tipo da “task”, clique em “Cancel” para fechar esta janela.

Selecione “Task0” com o botão direito do mouse e selecione “Settings”. Irá surgir uma janela “Task setting for IPC_40”. Nesta tarefa, valores como “interval” e “watchdogtime” serão mantidos.. Após confirmação, clique em “Cancel” para fechar esta janela.

Nota:

◁ “Default task”: É a que possui menor prioridade pois é executada apenas quando nenhuma outra está em execução.

◁ “Cyclical Task”: Executada de acordo com a sua especificação de intervalo (interval). O que a torna prioritária.

◁ “System Task”: Uma task desse tipo é chamada a ser executada apenas pelo sistema quando na ocorrência de um erro durante a operação com a “FCX”.

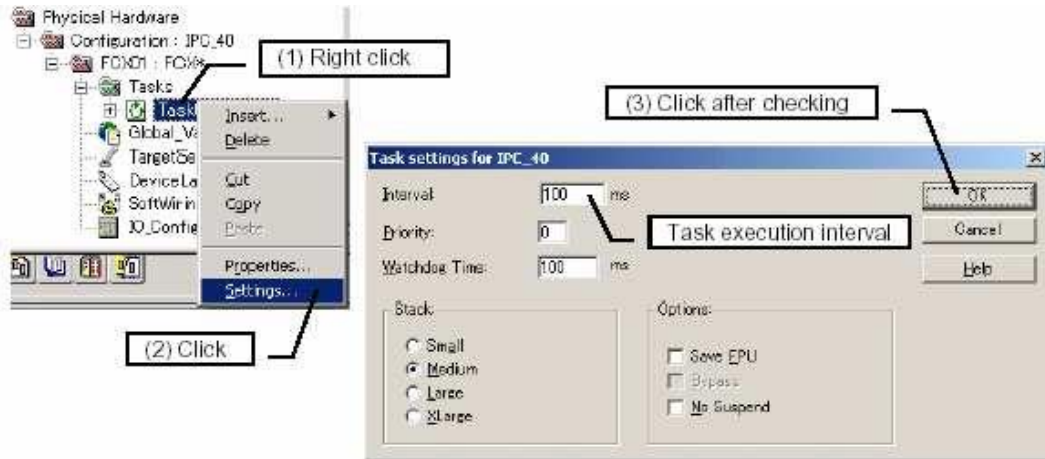


Figura 162.

ASSOCIANDO UMA “TASK” A UM PROGRAMA

Selecione “Task0” e clique com o botão direito do mouse. Selecione “Insert - Program Instance”, através do menu.

Entre no campo “Program Instance” e insira “_COMANDOS” e selecione “COMANDOS” no campo “Program Type”. Clique em “Ok”.

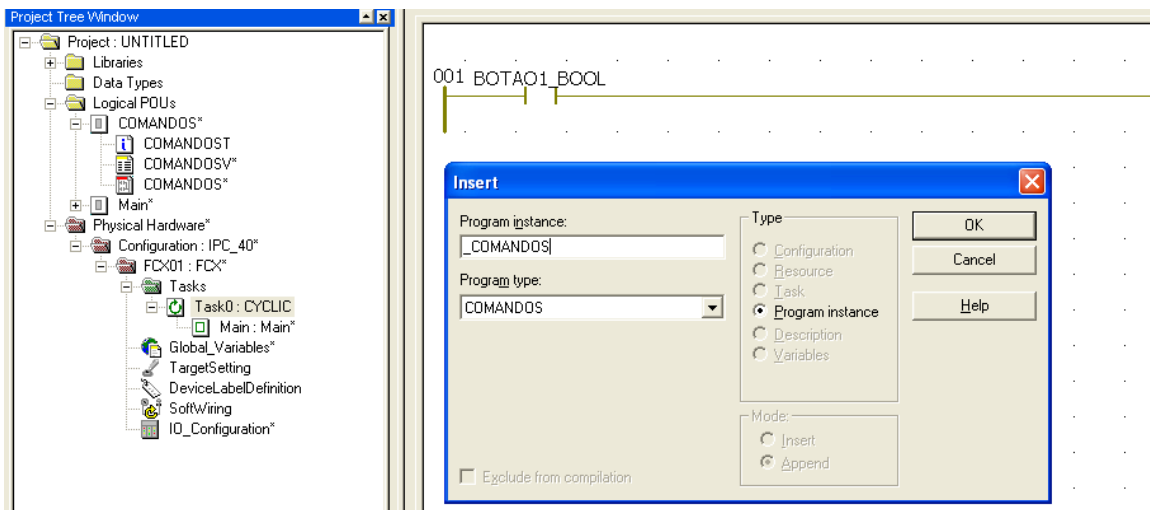


Figura 163.

FAZENDO O “BUILDING” DO PROJETO

Neste ponto, estaremos convertendo para código de máquina toda a aplicação desenvolvida até o momento.

Selecione “Build – Rebuild Project” através da barra de menu. Uma mensagem aparecerá informando que a compilação está em progresso.

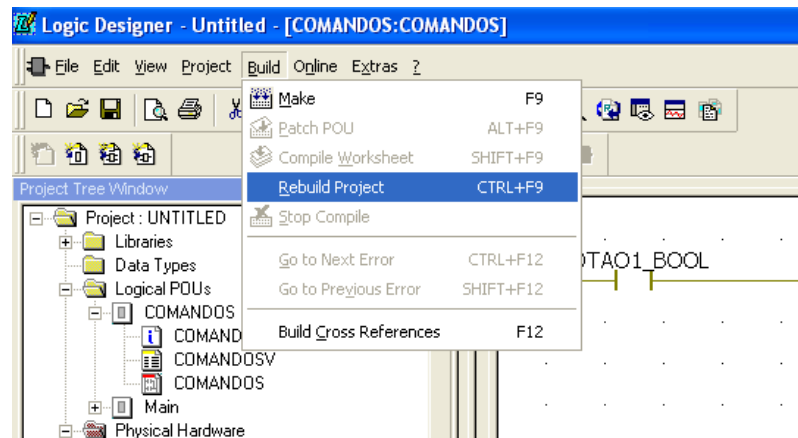


Figura 164.

Quando concluída a compilação, o resultado da mesma será exibida na janela de mensagem como na figura abaixo. Toda mensagem que exibe “0 Error(s)” indica que a compilação foi concluída com sucesso.

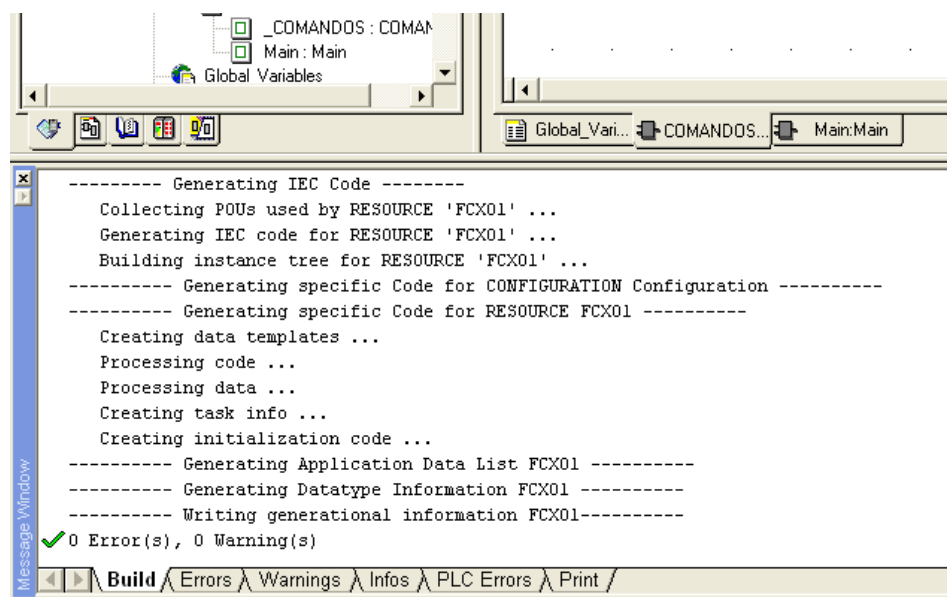


Figura 165.

FAZENDO O “DOWNLOAD” DE UM PROJETO

O que será feito agora é a carga do projeto compilado para a “FCX”.

Selecione “Online – Project Control” através da barra de menu. O diálogo do “Project Control” aparecerá como na figura abaixo:

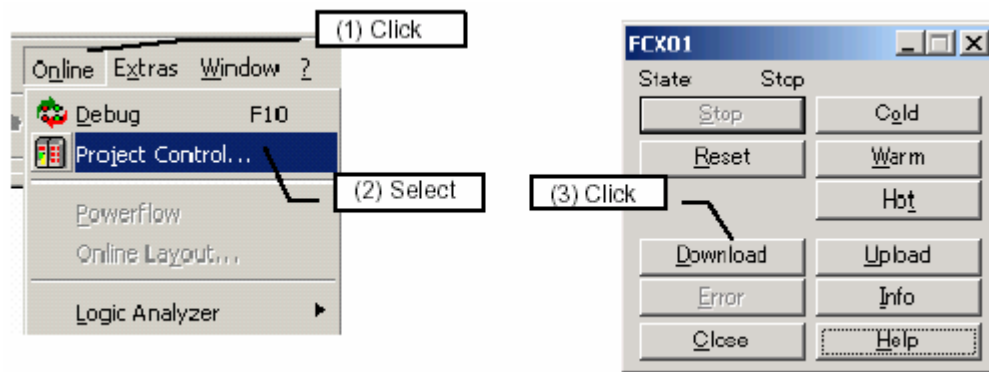


Figura 166.

Clique em “Download”. O diálogo deste botão será exibido.

Clique no botão “Download” abaixo do campo “Project”.

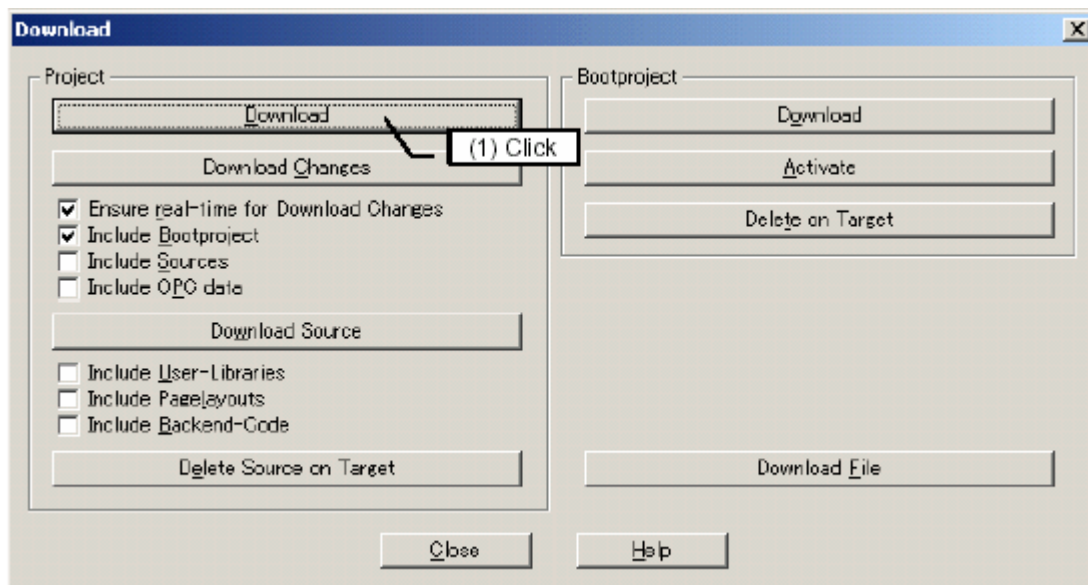


Figura 167.

Se algum outro projeto foi carregado anteriormente, aparecerá uma janela informando que este projeto será sobrescrito. Clique em “YES”.

Na parte inferior do Logic Designer será exibida uma barra de progressão referente ao “download”. Aguarde até que haja a indicação operação realizada.

INICIALIZANDO O PROJETO

Neste passo, a aplicação ainda não foi executada. Para alterar o estado do CLP, selecione um dos modos proposto na tabela abaixo:

Clicar no ícone “Project Control Dialog”, conforme figura abaixo.

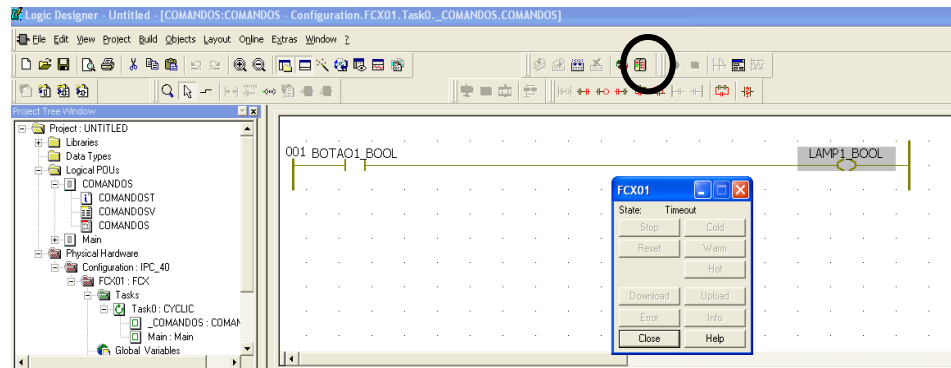


Figura 168.

Neste exercício, iremos optar pela inicialização a frio (“Cold”). Vide figura abaixo:

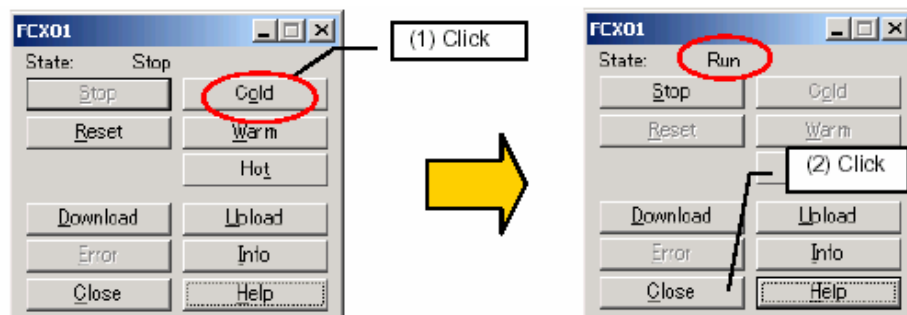


Figura 169.

ATENÇÃO:

Antes de salvarmos o projeto verificaremos se todas as *tags* estão com a propriedade **OPC** ativadas, para isso iremos em “**Global Variables**” como visto na figura abaixo:

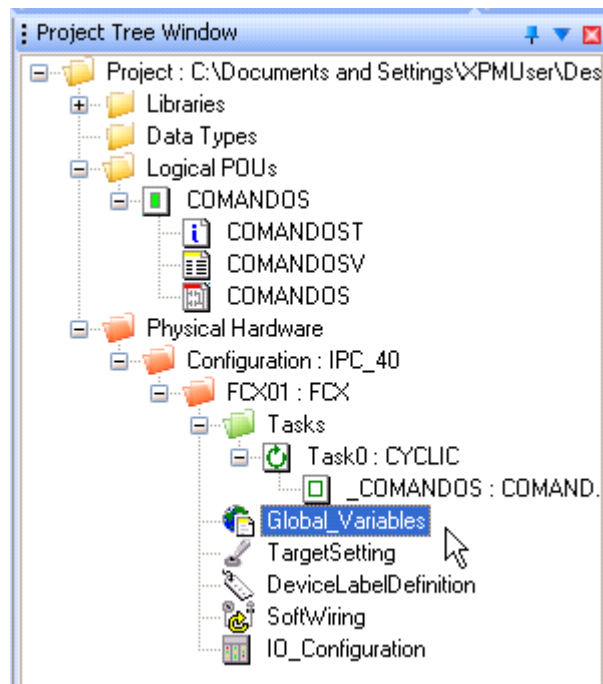


Figura 170.

Ao clicar 2 vezes, uma tabela com as **TAGS** e **PROPRIEDADES** de cada uma se abrirá na parte esquerda. Nessa tela teremos de localizar as variáveis que criamos: **BOTAO** e **LAMP**.

Após localizar estas, verifique se a propriedade **OPC** está ativa. Caso não esteja, basta clicar no quadrado referente a este.

Name	Type	Usage	Description	Address	Init	Retain	PDD	OPC	TB	Hid...	Init...	Default Hid...
User 2												
User 3												
[SYSTEM] Interface Flags												
[SYSTEM] System Variables												
[SYSTEM] Task Information												
[SYSTEM] Extended System Variables												
[SYSTEM] RAS Information												
[SYSTEM] Status Definition												
[SYSTEM] Engineering Constant												
[SYSTEM] Application Flags												
DeviceLabel_Input_A												
DeviceLabel_Input_D												
BOTA01	DTag_I_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IB 0								
BOTA01_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IX 2.0								
BOTA02	DTag_I_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IB 8								
BOTA02_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IX 10.0								
BOTA03	DTag_I_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IB 16								
BOTA03_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IX 18.0								
BOTA04	DTag_I_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IB 24								
BOTA04_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IX 26.0								
DeviceLabel_Output_A												
DeviceLabel_Output_D												
LAMP1_RB	DTag_O_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IB 32								
LAMP1_RB_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IX 34.0								
LAMP2_RB	DTag_O_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IB 40								
LAMP2_RB_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IX 42.0								
LAMP3_RB	DTag_O_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IB 48								
LAMP3_RB_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IX 50.0								
LAMP4_RB	DTag_O_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IB 56								
LAMP4_RB_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%IX 58.0								
LAMP1	DTag_O_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QB 32								
LAMP1_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QX 34.0								
LAMP2	DTag_O_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QB 40								
LAMP2_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QX 42.0								
LAMP3	DTag_O_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QB 48								
LAMP3_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QX 50.0								
LAMP4	DTag_O_Sts	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QB 56								
LAMP4_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QX 58.0								

Figura 171.

A seguir vemos o “LAMP1_BOOL” com o **OPC** ativo.

LAMP1_BOOL	BOOL	VAR_GLOBAL	/: Updated by Task0 : Inserted by Logic Designer Automati...	%QX 34.0								
------------	------	------------	--	----------	--	--	--	--	--	--	--	--

Figura 172.

SALVANDO UM PROJETO

Selecione “File – Save Project As/Zip Project As” no menu do “Logic Designer”.

Entre com um nome para o seu projeto (LABO2, por exemplo) e clique em “Save”.

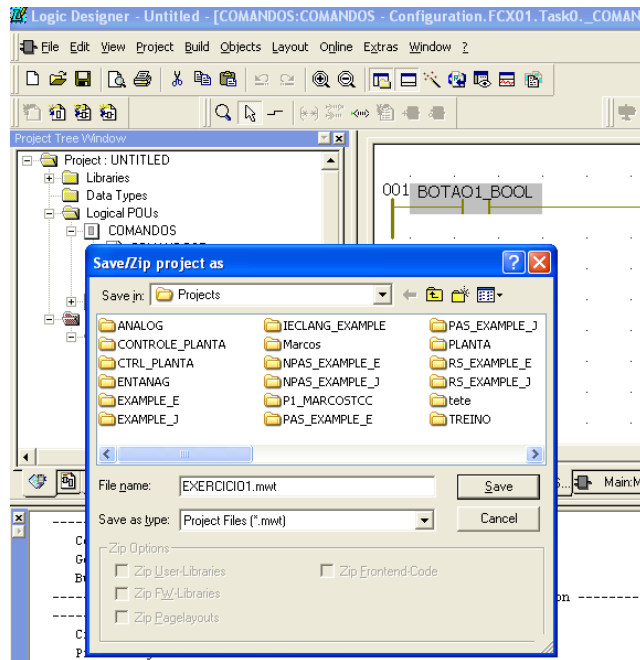


Figura 173.

ETAPA 3. CRIAR UM NOVO PROJETO NO INDUSOFT

Para criar um novo projeto primeiro vamos abrir o software. Duplo clique no ícone  (**IWS v8.0 Wonderware InduSoft Web Studio**), ou usando o botão “Iniciar”, “Programas” -> “Wonderware InduSoft Web Studio v8.0” -> **“Wonderware InduSoft Web Studio v8.0”**.

Já no ambiente do **InduSoft Web Studio**, clique no botão localizado no canto superior esquerdo da tela, clique em **Novo** para abrir a janela “Novo” ou aperte **CTRL + N** e faça como a seguir:

Em “**Nome do Projeto**”, digite o nome desejado, exemplo: Projeto02; Exercício02.

Em “**Tipo de Produto**”, selecione “**Windows Local Interface**”, como mostrado na imagem acima.

Pressione o botão “**OK**” para abrir a janela seguinte

Na janela “**Assistente de Projeto**”, é possível selecionar algumas janelas que podem ser utilizadas como um ponto de partida para a aplicação.

Selecione “**Aplicação Vazia**” e escolha uma resolução pré-determinada pelo programa, como “**1280x800**”, ou uma resolução “**Customizado**”, onde é possível determinar a largura e altura desejada. Neste caso usaremos a resolução máxima que o monitor pode suportar para maximizar o espaço disponível, esta resolução é a “**1920x1078**”.

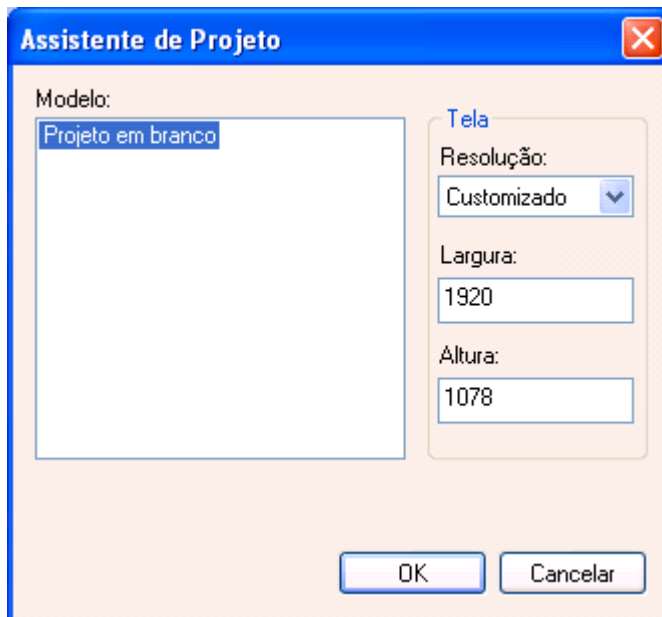


Figura 174.

ETAPA 2. CRIAR TAGS NA BASE DE DADOS “DATABASE”

Na área de trabalho do programa, selecione a aba "Global". Clique na pasta "Tags do Projeto" para expandi-la.

Clique duas vezes em "Planilha de Dados".

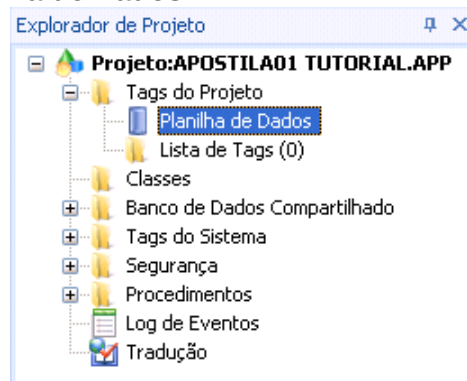


Figura 175.

Vamos configurar a "Planilha de Dados", como mostrado a seguir:

Crie os *tags* que enviarão os comandos "para abrir ou fechar as válvulas", conforme mostrado a seguir.

	Nome	Vetor	Tipo	Descrição	Escopo
	Filtro de Texto	= ...	(Todos)	Filtro de Texto	(T...
1	BOT_ENCHE	2	Booleana		Local
2	BOT_ESVAZIA	2	Booleana		Local
3	VALV_ENCHE	0	Booleana		Local
4	VALV_ESVAZIA	0	Booleana		Local

Figura 176.

Salve a configuração do banco de dados usando o ícone:



ETAPA 3. CRIAR TELA NO PROJETO

A figura a seguir mostra a tela de nome LAB02.

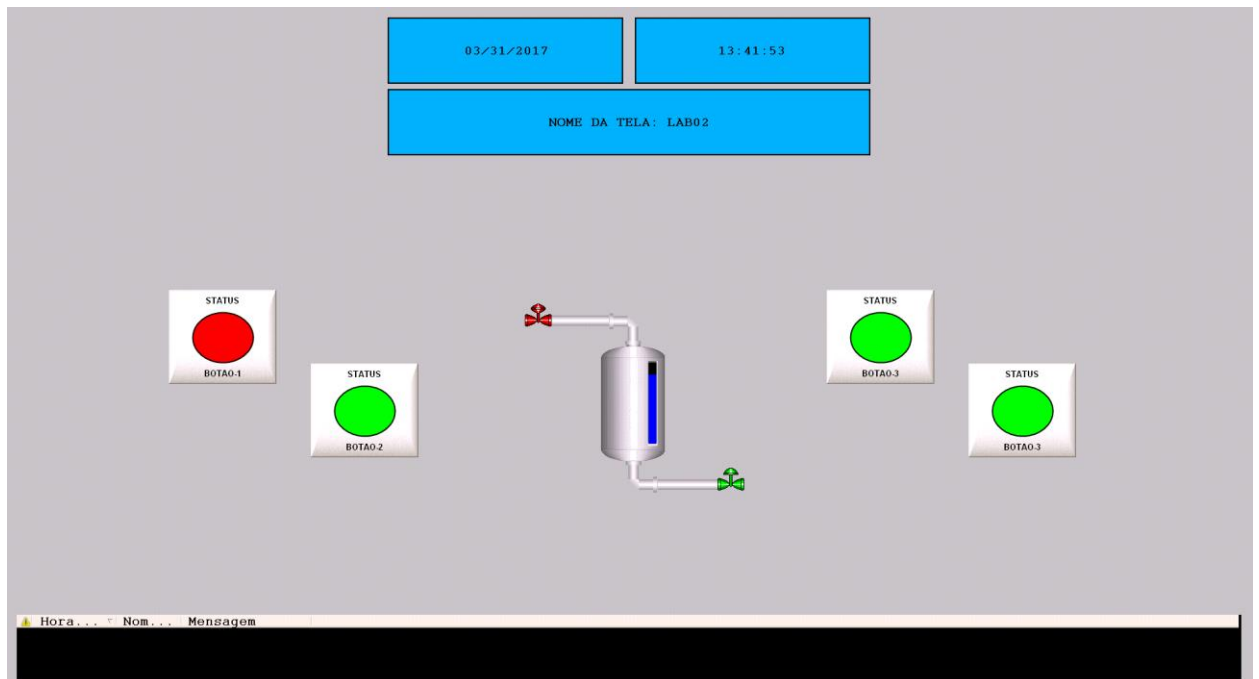


Figura 177.

Após projetar a sua tela, ela deve ter uma aparência semelhante a Figura 178. Nota-se que a barra de alarmes está presente na mesma apenas por questão estética, visto que ela não tem função em nosso programa. É possível considerar a adição de elementos gráficos por questões de aparência e(ou) um botão de sair (como fizemos no exercício 1, criando a tela de confirmação).

Selecione a aba "**Gráficos**" na Área de Trabalho e selecione na pasta "**Telas**". Agora, clique com o botão direito na pasta "**Telas**" e clique na opção "**Inserir**".

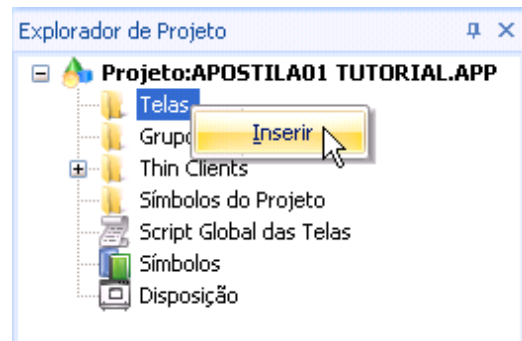


Figura 178.

Na janela "**Atributos de Tela**" devemos configurar alguns atributos gerais sobre a tela que está sendo criada.

Estaremos nomeando essa tela de LAB02 e usando o cabeçalho/rodapé semelhantes aos de nosso ultimo projeto.

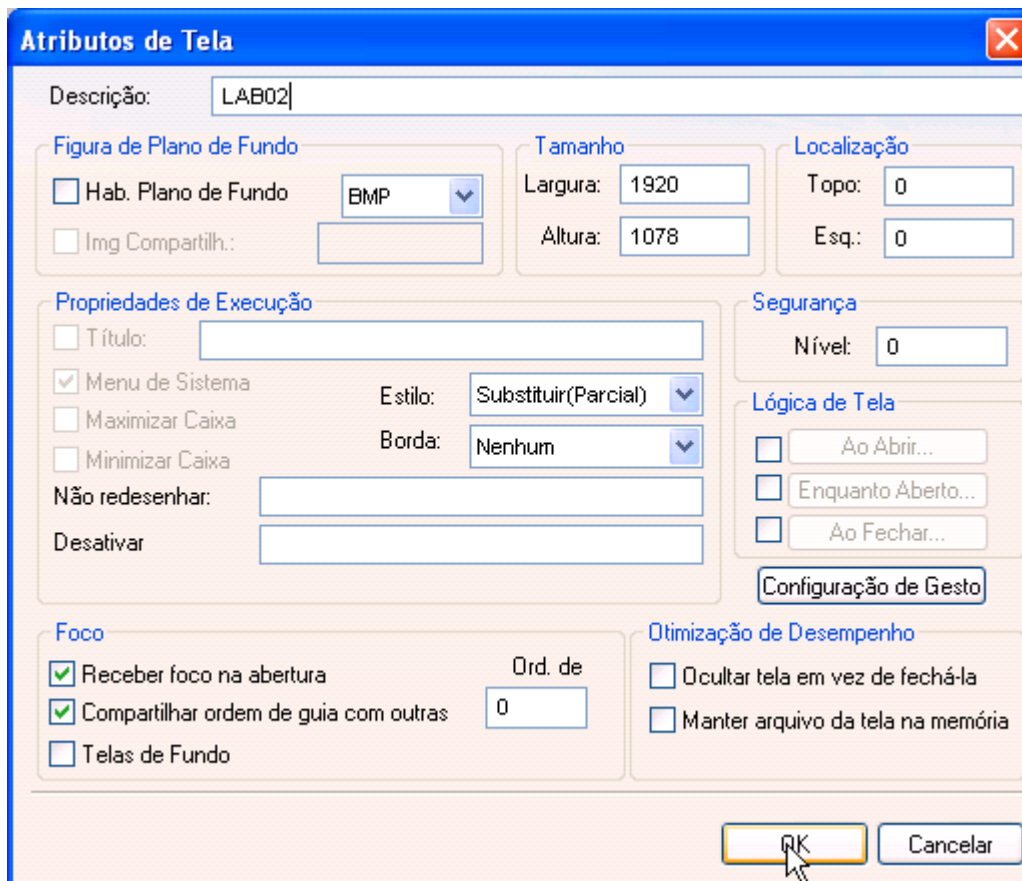


Figura 179.

Pressione o botão OK e uma tela vazia deverá aparecer com os atributos configurados para ela.

Depois de criar a tela, mudemos a cor de fundo usando o "**Cor de Fundo**".

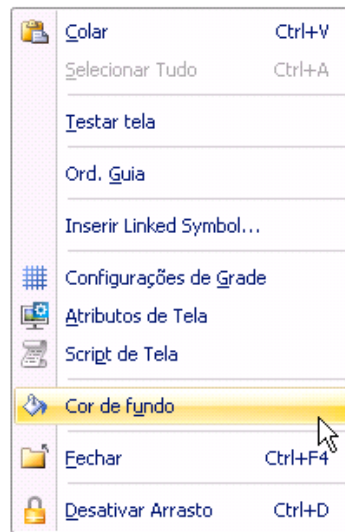


Figura 180.

Selecione o **cinza claro** nas opções de cores da janela aberta e clique em **OK**:

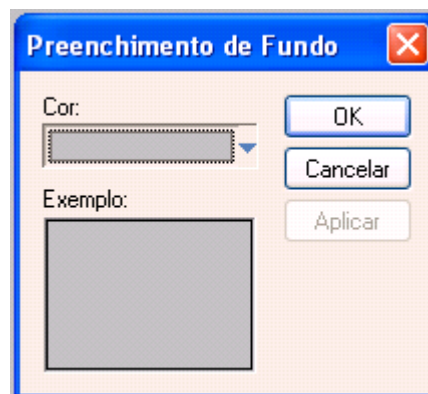


Figura 181.

Salve a tela como LABO2. Para salvar, clique no menu File na barra superior, e no item Save As e digite o nome LABO2 no campo Nome do arquivo, e clique em Salvar.

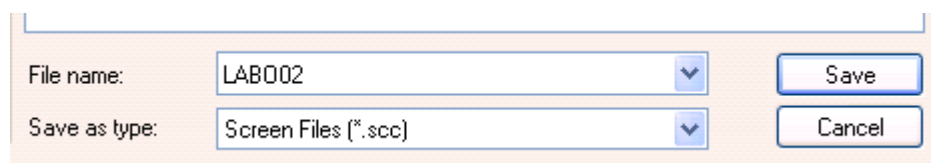


Figura 182.

Para criar os tanques, tubos e as válvulas estaremos usando objetos da biblioteca. Para acessarmos a biblioteca de símbolos, clique com no ícone mostrado na figura abaixo:



Figura 183.

Para importar um objeto da biblioteca para a tela da aplicação, simplesmente clique no objeto e depois na tela.

É claro que há outras formas de se desenhar sem ser utilizando a biblioteca, mas para os propósitos desta atividade, a biblioteca nos atende bem.

Portanto vamos importar objetos como válvulas (na janela Valves da biblioteca), tubos (em Pipes) e o tanque (em Tanks) como mostrado abaixo. Obs.: Não criar o retângulo com o texto “STATUS BOTAO1” a partir da biblioteca, criaremos manualmente.

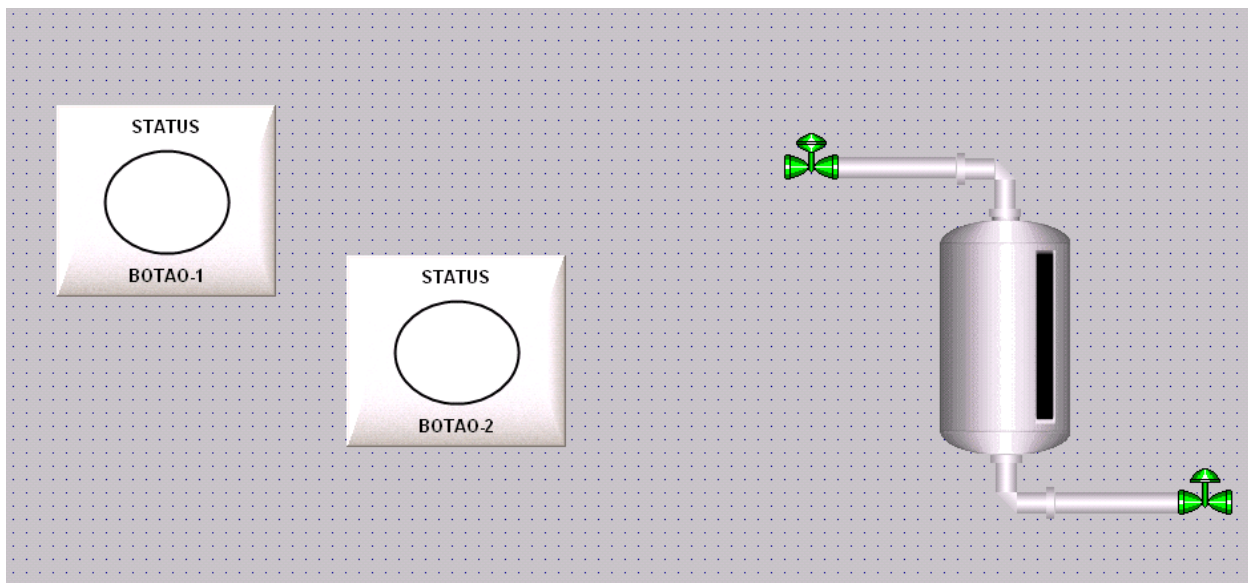


Figura 184.

O conjunto para monitorar o status do botão deverá ser composto por 3 objetos:



Depois de criados e posicionados corretamente, estes objetos serão agrupados por você.

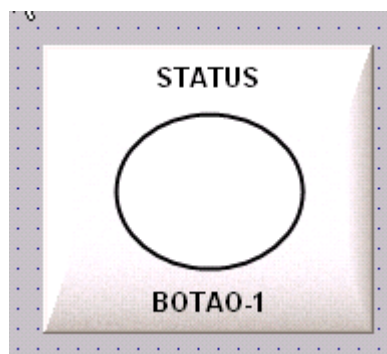


Figura 185.

Depois de agrupado e posicionado, o objeto elipse deverá ser associada uma propriedade do tipo “**Cor**”, cuja localização é mostrada na figura a seguir.

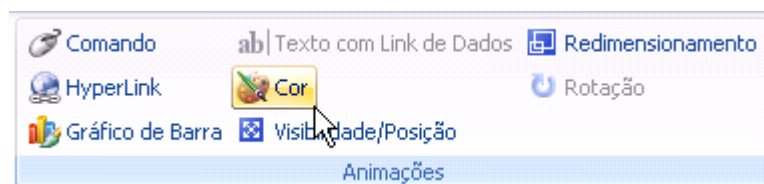


Figura 186.

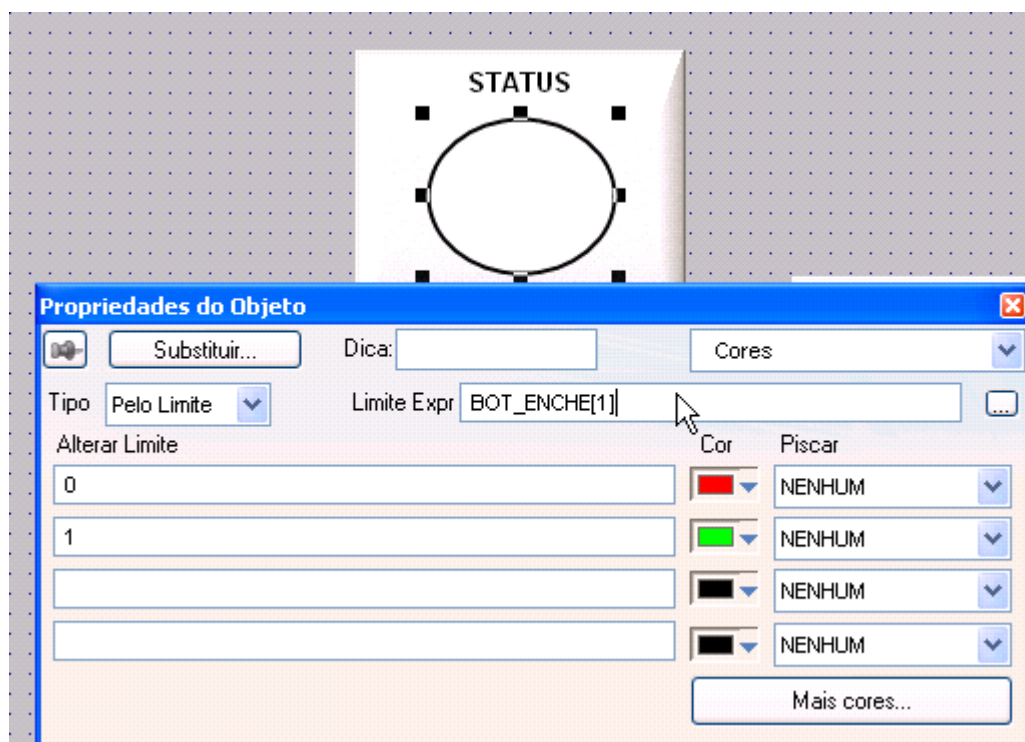


Figura 187.

Repetir o mesmo procedimento para os quatro botões, sendo que em cada um teremos uma tag e uma numeração, sendo essas: BOT_ENCHE[1], BOT_ENCHE[2], BOT_ESVAZIA[1], BOT_ESVAZIA[2].

Agora vamos às válvulas.

As válvulas que escolhemos na biblioteca são objetos previamente preparados para mudança de cor comandados pelo valor ou tag definido no campo "TagState".

Chamaremos a válvula que se encontra na horizontal de "**ValvulaEnche**". Clique duas vezes em cima dela e selecione a propriedade "**Linked Symbol**", e configure como mostrado abaixo:

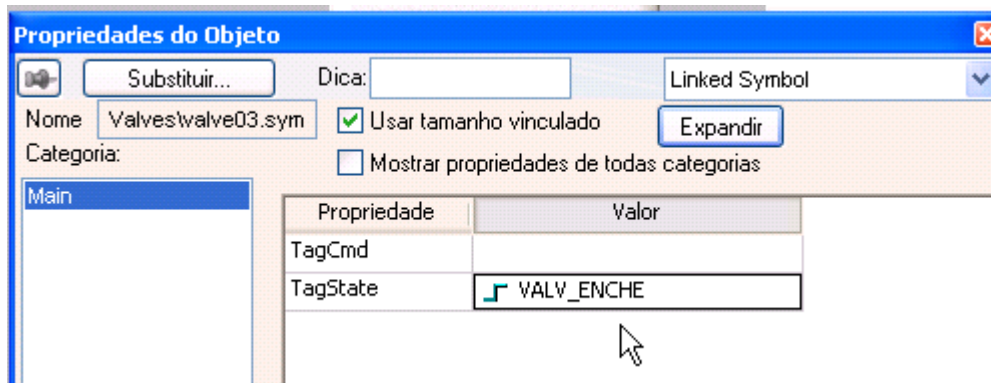


Figura 188.

Esta propriedade fará com que, sempre que o valor do tag **VALV_ENCHE** for "1", a válvula estará verde. Pensando em nossa aplicação, o valor deste tag deverá ser **LIDO** do campo pelo botão que está conectado ao CLP.

Chamaremos de "**ValvulaEsvazia**" a válvula que está na posição vertical.

Esta válvula, quando aberta, esvaziará o tanque.

Clique duas vezes em cima dela e selecione a propriedade "**Linked Symbol**", e configure como mostrado abaixo:

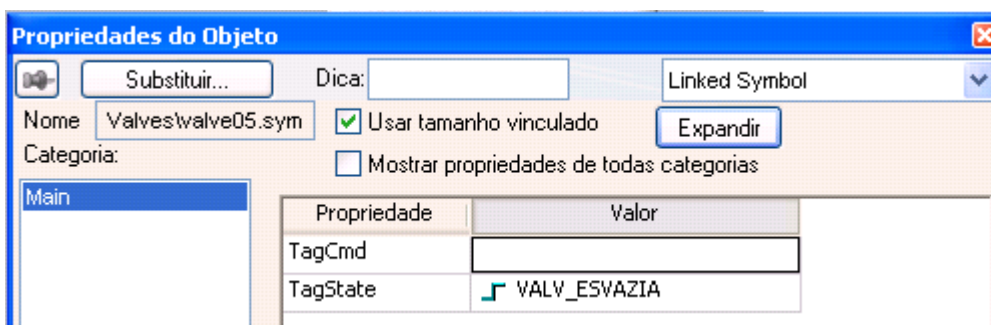


Figura 189.

Salve a tela como LABO2. Para salvar, clique no menu File na barra superior, e no item Save As e digite o nome LABO2 no campo Nome do arquivo, e clique em Salvar.

Em seguida configure a partida no menu **“Projeto”**, clique na opção **“Visualizador”**.



Figura 190.

No aba **“Configurações de Projeto”** digite dentro da janela “Tela Inicial”: LABO2.

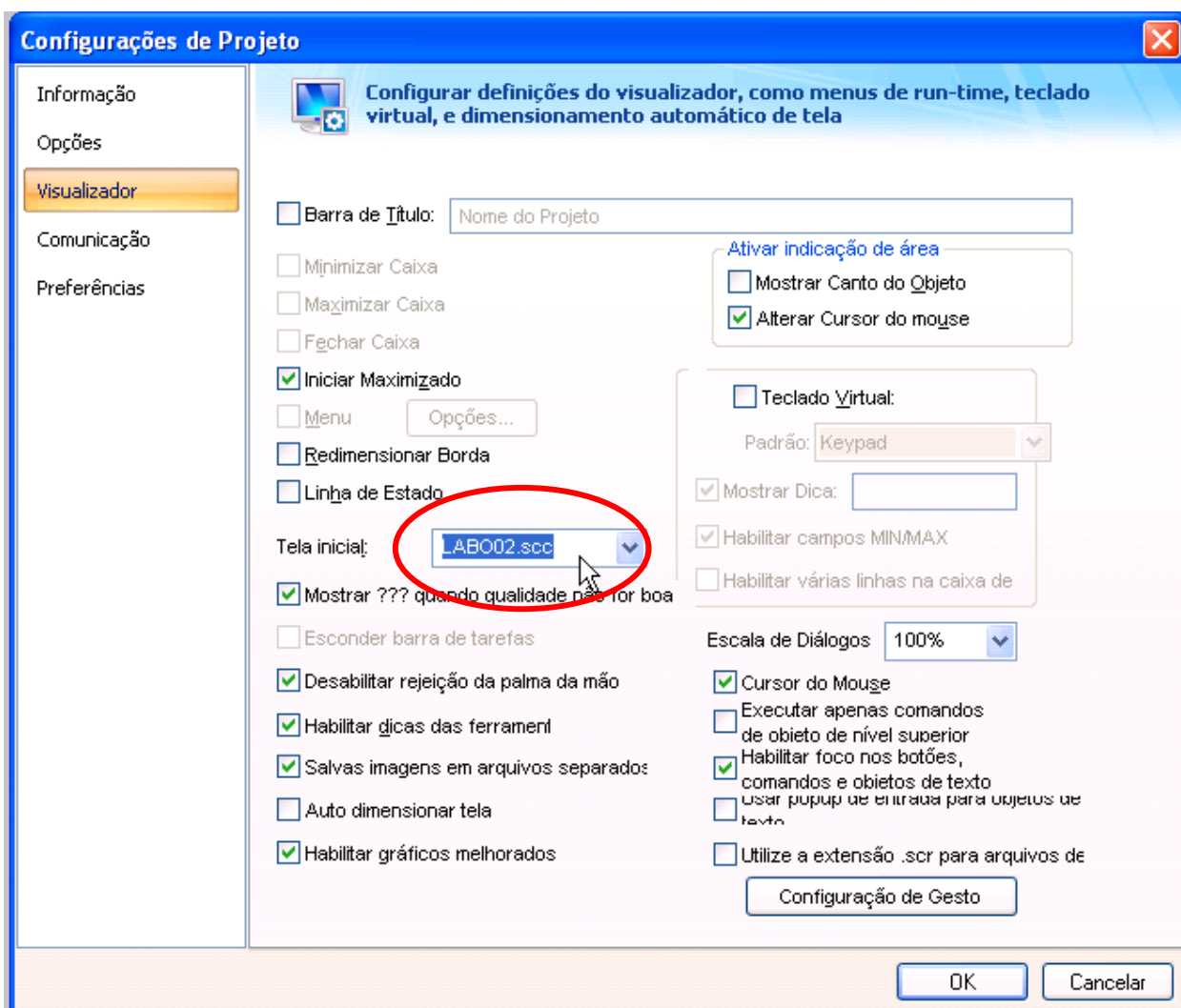


Figura 191.

ETAPA 4. CONFIGURAR COMUNICAÇÃO DO INDUSOFT COM O STARDOM VIA OPC

Selecionar a aba “**Comunicação**” no rodapé do explorador de projeto.



Figura 192.

Clicar com a tecla da direita do mouse na pasta “**OPC DA 2.05**” e inserir uma planilha de comunicação via OPC.

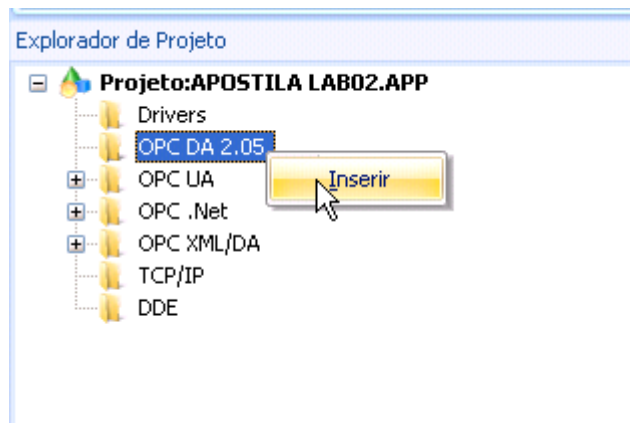


Figura 193.

Selecionar a opção “**Yokogawa.ExaopcDASTARDOMFCX**” no campo “**Identificador do Servidor**” conforme mostrado a seguir:

Descrição:	Identificador do Servidor:	Desativar:
	aopcDASTARDOMFCX	
Taxa de Atual. Leitura (ms):	Porcentagem Banda Morta:	Estado:
Nome do Servidor Remoto:	☐ Ler antes de escrever ☐ Ler depois de escrever ☒ Aceitar Nome de Tag na coluna Item	
	Procurar...	

Figura 194.

Preencher o campo “**Nome do Tag**” com a variável: **BOT_ENCHE[1]**, conforme figura a seguir:

Descrição: Identificador do Servidor: Yokogawa.ExaopcDAS Desativar:

Taxa de Atual. Leitura (ms): Porcentagem Banda Morta: Estado:

Nome do Servidor Remoto: Procurar...
 ☐ Ler antes de escrever
☐ Ler depois de escrever
☒ Aceitar Nome de Tag na coluna Item

	Nome da Tag	Item	Scan	Div	Adicionar
			(Todos)		
1	BOT_ENCHE[1]		Sempre		
*			Sempre		

Figura 195.

Duploclicar no campo em branco “Item” ao lado do “Nome do Tag”: BOT_ENCHE. Abrir-se-á a janela de configuração do **OPC BROWSER**, conforme figura a seguir:

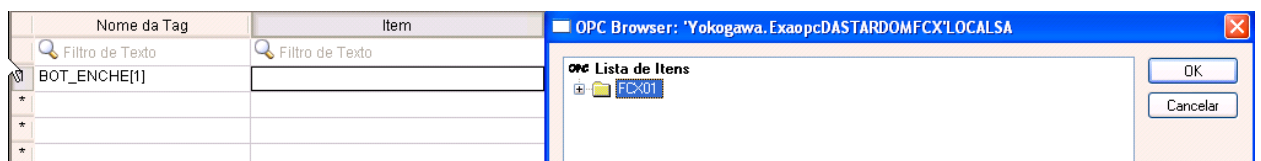


Figura 196.

Abrir a pasta: “@GV” do FCX1, que corresponde a todas variáveis do programa criado no CLP, que poderão ser acessadas via OPV pelo Indusoft Web Studio, conforme figura a seguir:

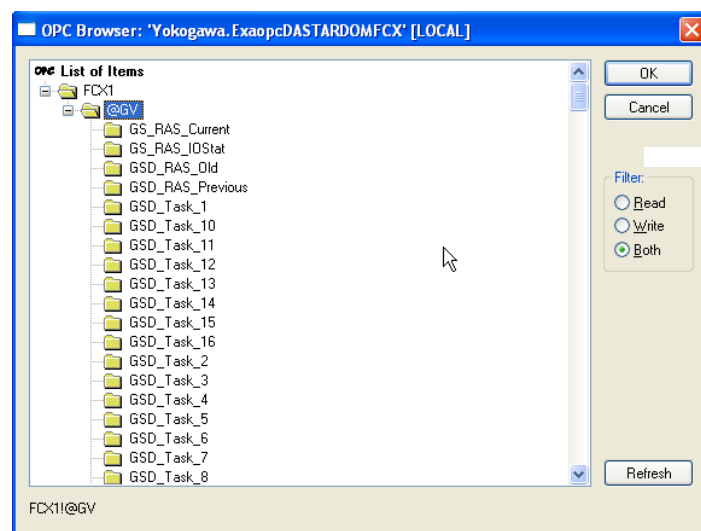


Figura 197.

Selecionar a opção: “BOTA01_BOOL”, conforme figura a seguir. Em seguida clicar no botão “OK”:

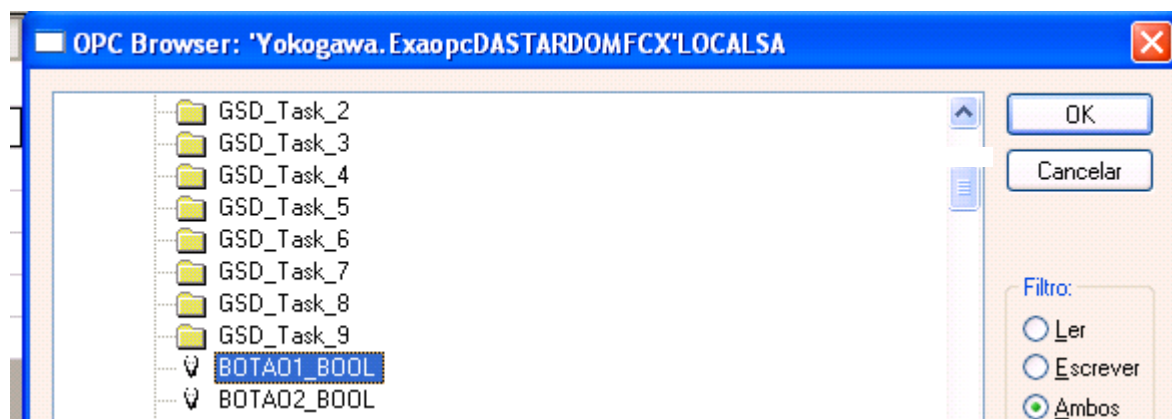


Figura 198.

Preencher o campo “Nome do Tag” com a variável: **BOT_ENCICHE[2]**, conforme figura a seguir:

	Nome da Tag	Item	Scan	Div	Adicionar
	Filtro de Texto	Filtro de Texto	(Todos)	Filtro de Te	Filtro de Te
1	BOT_ENCICHE[1]	FCX011@GV.BOTA01_BOOL	Sempre		
2	BOT_ENCICHE[2]		Sempre		
3	BOT_ESVAZIA[1]		Sempre		
4	BOT_ESVAZIA[2]		Sempre		
5	VALV_ENCICHE		Sempre		
6	VALV_ESVAZIA		Sempre		
*			Sempre		

Figura 199.

Duploclicar no campo em branco “Item” ao lado do “Nome do Tag”: **BOT_ENCICHE[2]**. Abrir a pasta: “@GV” do FCX1, que corresponde a todas variáveis do programa criado no CLP, que poderão ser acessadas via OPV pelo Indusoft Web Studio. Selecionar a opção: “**BOTA02_BOOL**”, conforme figura a seguir. Em seguida clicar no botão “OK”:

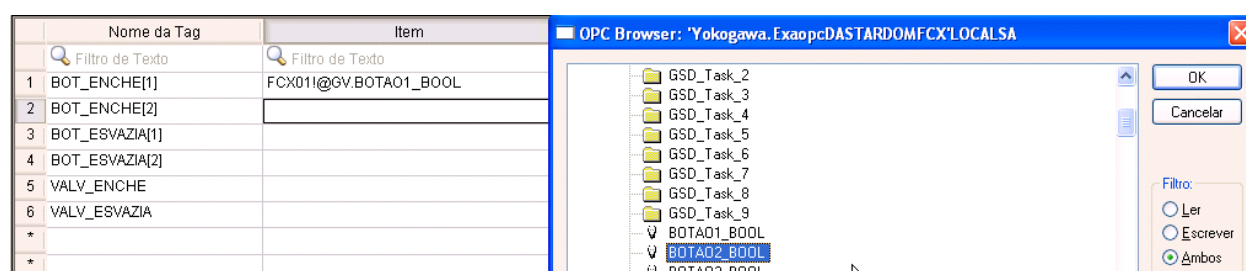


Figura 200.

Repetir o procedimento anterior, para as linhas correspondentes aos Tags: **BOTA0_ESVAZIA[1]**; **BOTA0_ESVAZIA[2]**; **VALV_ENCICHE** e **VALV_ESVAZIA**, conforme figura a seguir. Em seguida, salvar a planilha de configuração do OPCCL001.

	Nome da Tag	Item	Scan	Div	Adicionar
	Filtro de Texto	Filtro de Texto	(Todos)	Filtro de Te	Filtro de Te
1	BOT_ENCHE[1]	FCX01!@GV.BOTA01_BOOL	Sempre		
2	BOT_ENCHE[2]	FCX01!@GV.BOTA02_BOOL	Sempre		
3	BOT_ESVAZIA[1]	FCX01!@GV.BOTA03_BOOL	Sempre		
4	BOT_ESVAZIA[2]	FCX01!@GV.BOTA04_BOOL	Sempre		
5	VALV_ENCHE	FCX01!@GV.LAMP4_BOOL	Sempre		
6	VALV_ESVAZIA	FCX01!@GV.LAMP3_BOOL	Sempre		
*			Sempre		

Figura 201.

É possível observar na Figura 199 que a VALV_ENCHE e a VALV_ESVAZIA estão endereçadas para as lâmpadas 4 e 3, porém isso só foi feito pois as lâmpadas 1 e 2 estavam com defeito durante a produção deste exercício, porém, é permitido que você use quaisquer lâmpadas disponíveis.

Salve a aplicação e rode o sistema.

Verificar se as válvulas e o status dos botões mudam de cor, quando são acionados os botões.

ETAPA 5. ACRESCENTAR FUNCIONALIDADES NO SISTEMA

Alterar o programa do CLP, através do **LOGIC DESIGNER**, de tal forma que a 3ª. saída digital correspondente a LAMP3, seja acionada sempre que os 4 botões conectados nas 4 primeiras entradas digitais sejam acionados simultaneamente. (Faça uma linha de programação em ladder).

Alterar também, o software INDUSOFT, na tela e na planilha de comunicação OPC, para que a nova válvula de saída do tanque, mude de cor quando a 3ª saída digital correspondente a LAMP3 for acionada, conforme figura a seguir.

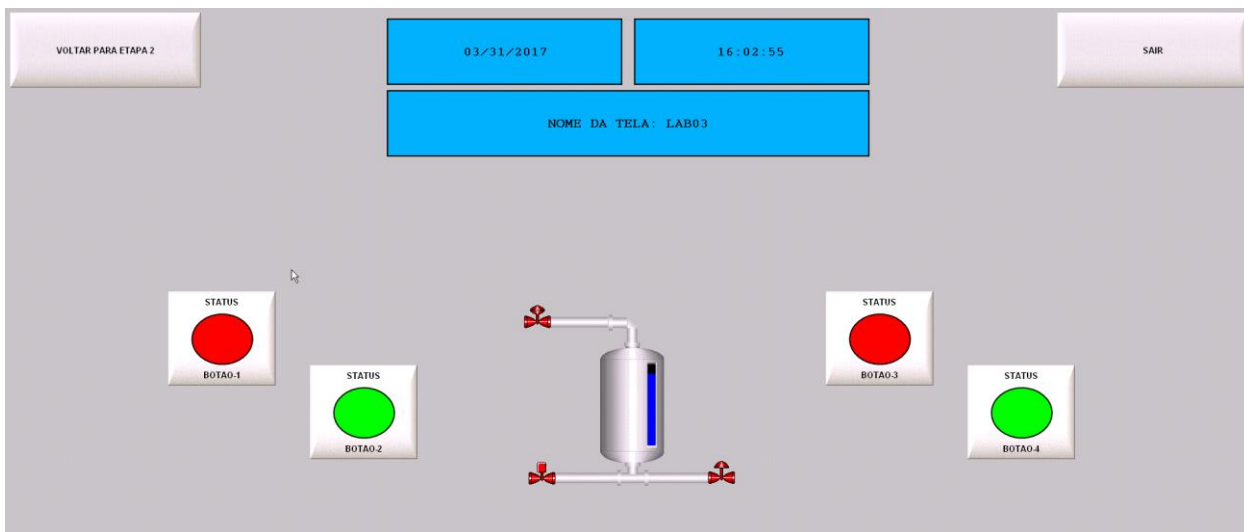


Figura 202.

3º. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA DE CONTROLE COM SOFTWARE INDUSOFT E CLP STARDOM EM ACIONAMENTO DE ENTRADAS E SAÍDAS DIGITAIS

O objetivo deste trabalho prático visa conhecer as ferramentas para programação do CLP Stardom utilizando instruções de contato no software LOGIC DESIGNER e construção de aplicação para supervisão e monitoramento de variáveis digitais utilizando o software SUPERVISÓRIO INDUSOFT.

ETAPA 1. CONFIGURAÇÃO DE HARDWARE

Antes de iniciarmos a aplicação, iremos descrever características de hardware e de configuração de rede de uma FCN/FCJ. Toda a operação que envolva hardware, seja de uma FCN ou de uma FCJ, esta deverá ser executada através de uma ferramenta conhecida como “Resource Configurator”.

INICIANDO O RESOURCE CONFIGURATOR

- Duplo clique no ícone abaixo:



Se o ícone não estiver disponível na área de trabalho, siga o seguinte caminho:

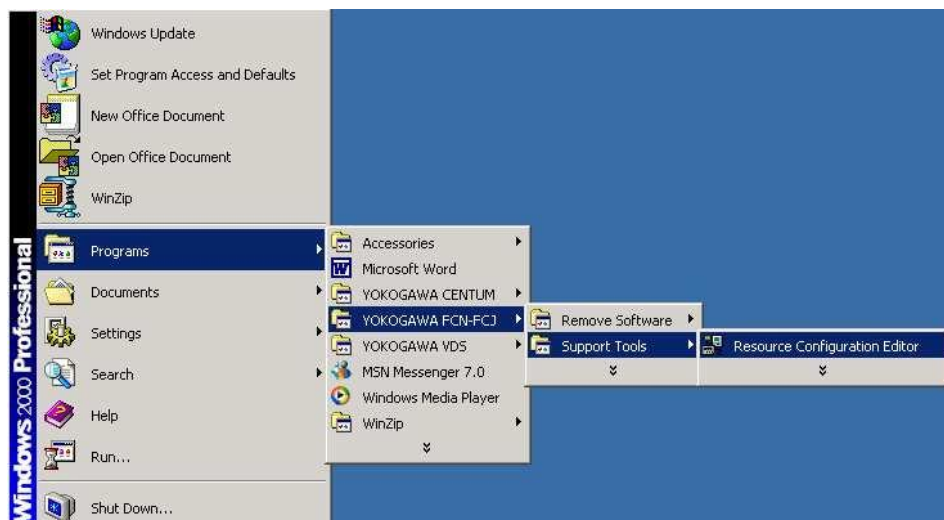


Figura 203.

- Selecionar a opção “Connection” no menu “File”, conforme figura a seguir.

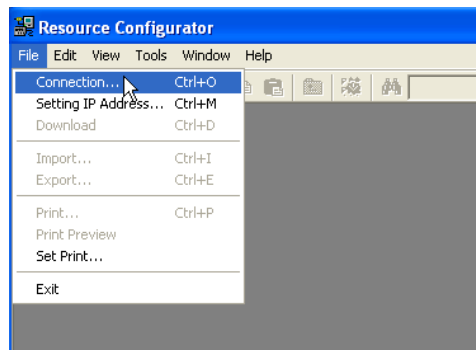


Figura 204.

- Entre com o número de IP “10.111.203.12X” no campo “Host”, “stardom” no campo “User Name” e “YOKOGAWA” no campo “Password” e clique em “OK”. Obs.: XX=Número variável de acordo com a bancada. Vide a figura abaixo exibindo a janela “Connect”.

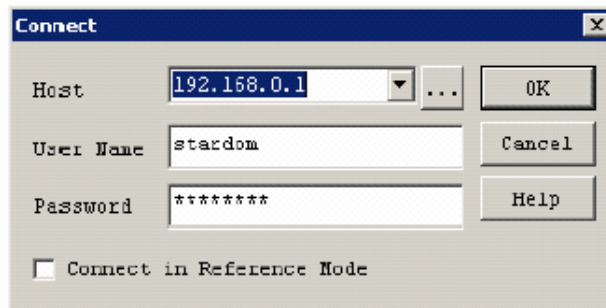


Figura 205.

DEFINDO DEVICE LABELS

O “Device Label” é um rótulo dado às entradas e saídas do sistema. Assim, torna-se mais familiar a manipulação de variáveis de campo, sabendo-se o seu nome de projeto.

Com o resource configurator em modo “tree” clique em “IOM”.

- Duplo-clique em DI/DO.
- Ajuste o device label correspondente a:
 - 1ª. entrada digital para “BOTAO1”;
 - 2ª. entrada digital para “BOTAO2”;
 - 3ª. entrada digital para “BOTAO3”;
 - 4ª. entrada digital para “BOTAO4”.
- Ajuste o device label correspondente a:
 - 1ª. saída digital para “LAMP1”;

- 2ª. saída digital para “LAMP2”;
- 3ª. saída digital para “LAMP3”;
- 4ª. saída digital para “LAMP4”.
- Selecione File-Download
- Quando aparecer uma janela de prompt, clique no OK.

ETAPA 2. CRIAÇÃO DA APLICAÇÃO DE CONTROLE (PROGRAMA DO CLP)

Neste momento será iniciado o desenvolvimento da criação de uma aplicação de controle.

INICIANDO O LOGIC DESIGNER

- Para iniciar o Logic Designer, clique no ícone na área de trabalho como exibido na figura



ao lado:

Ou, siga o caminho como exibido abaixo:

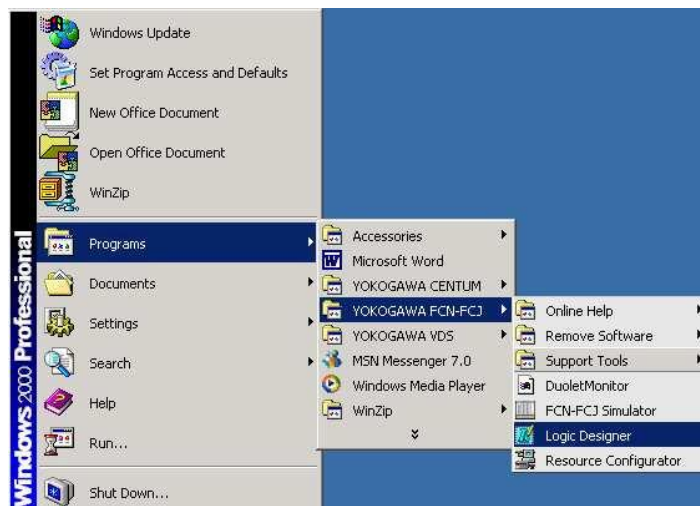


Figura 206.

CRIANDO UM NOVO PROJETO

- Selecione “File-New Project”.
- Selecione o ícone “STARDOM NPAS” e clique “OK”.

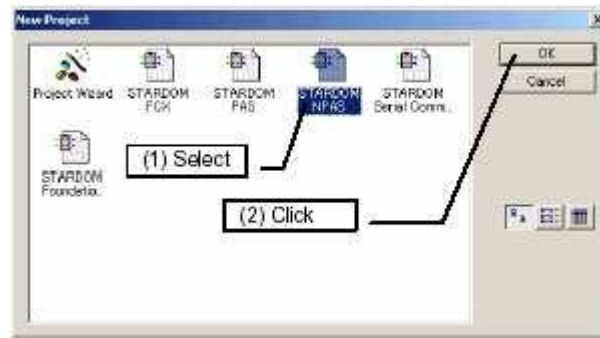


Figura 207.

ESPECIFICANDO O “TARGET” DA FCN/FCJ

- Duplo-clique em “TargetSetting” na árvore do projeto.
- Entre com o número de IP “10.111.203.12X” e clique em “OK”. Obs.: X=Número variável de acordo com a bancada.

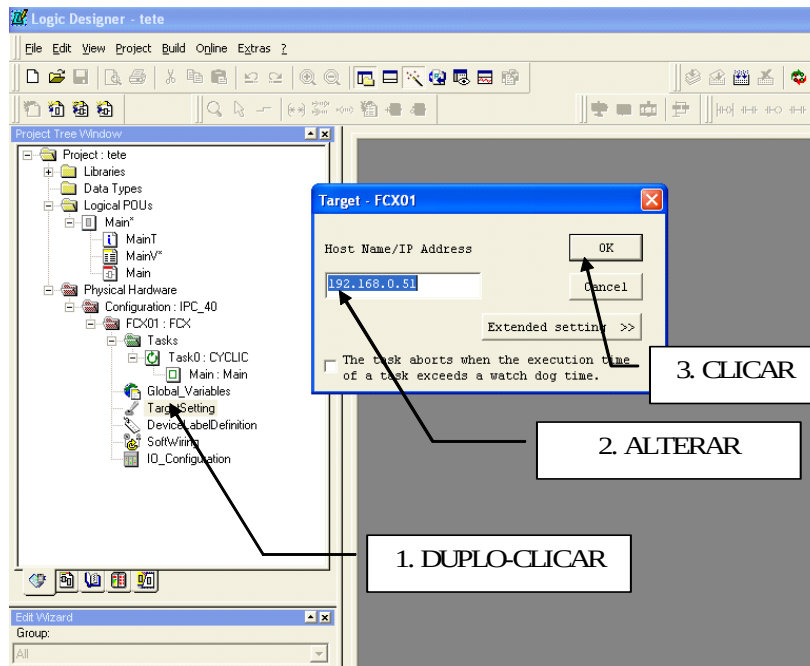


Figura 208.

DEFININDO DEVICE LABEL PARA AS VARIÁVEIS

Nesta aplicação iremos utilizar duas “variáveis globais” que farão as vezes dos sinais de entrada e saída (digitais) deste exercício.

- Duplo-clique sobre “DeviceLabelDefinition” na árvore do projeto.
- Dentro da janela defina as variáveis como descritas abaixo. Após finalizar os ajustes das variáveis, clique em “Ok” e clique em “YES” na janela de diálogo de confirmação.

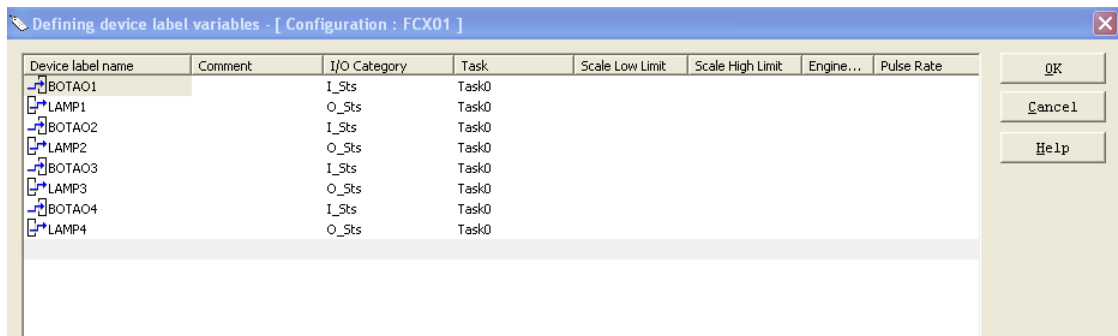


Figura 209.

CRIANDO UM PROGRAMA

Iremos agora construir uma aplicação de controle usando a unidade POU (Program Organization Unit).

- Clique com o botão direito sobre “Logical POU’s” na árvore do projeto e então selecione “Insert – Program” para aparecer a janela exibida abaixo.
- Neste exercício, insira no campo “Name” a informação “COMANDOS” e selecione “LD” (LADDER) no campo “Language” e clique em “Ok”.

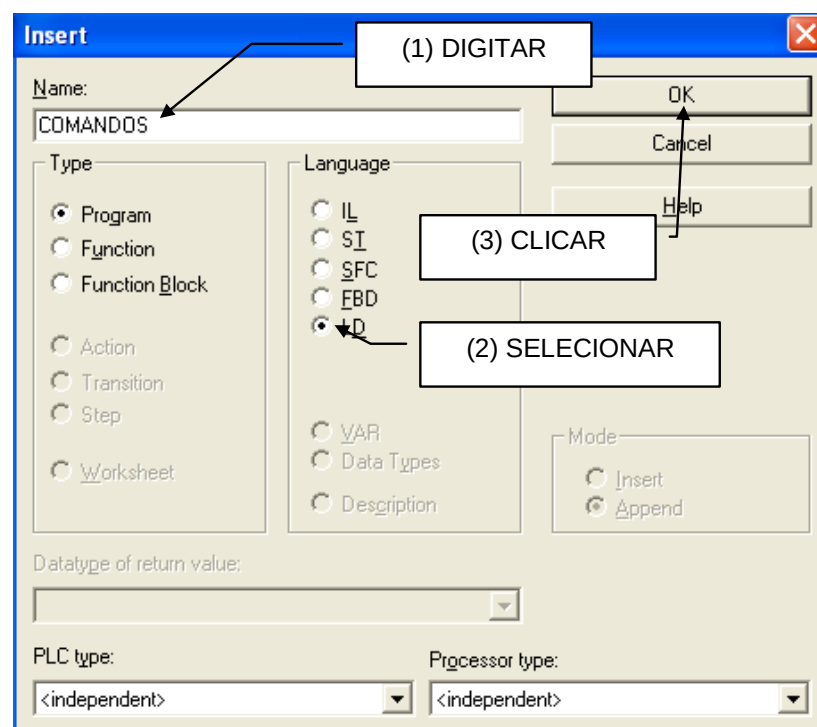


Figura 210.

- Verifique no seu projeto se como na figura abaixo a estrutura com três ícone foram criados:

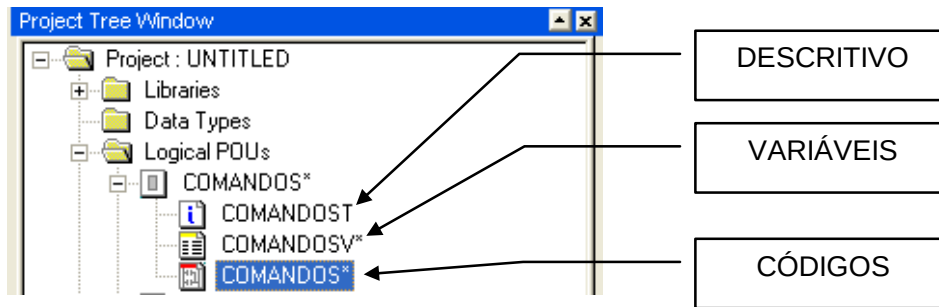


Figura 211.

- Duplo-clique sobre “COMANDOS” (ícone de código) criado no projeto.
- Clique numa posição específica dentro da área de trabalho a fim de determinar onde ficará a linha de programa. Em seguida, clicar sobre o ícone “Contact Network” para iniciar uma linha básica de programa.

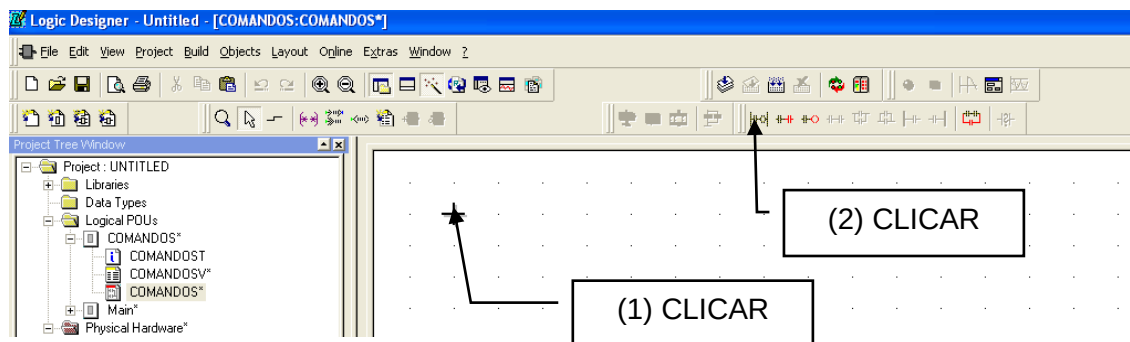


Figura 212.

- Duplo-clique no contato aberto da linha de programa, conforme mostrado na figura a seguir. Em seguida, no campo “Global Variable Groups:”, selecione “Physical Hardware– Configuration –FCX01–Default”. Mude a opção dentro de “Scope” para “Global” e clique na opção “Show all variable of worksheet”. Selecione o tag “BOTAO1_BOOL” na opção “NAME”. Não esqueça de ativar a opção “OPC”, que permitirá a comunicação com o supervisor. Clique em “OK”.

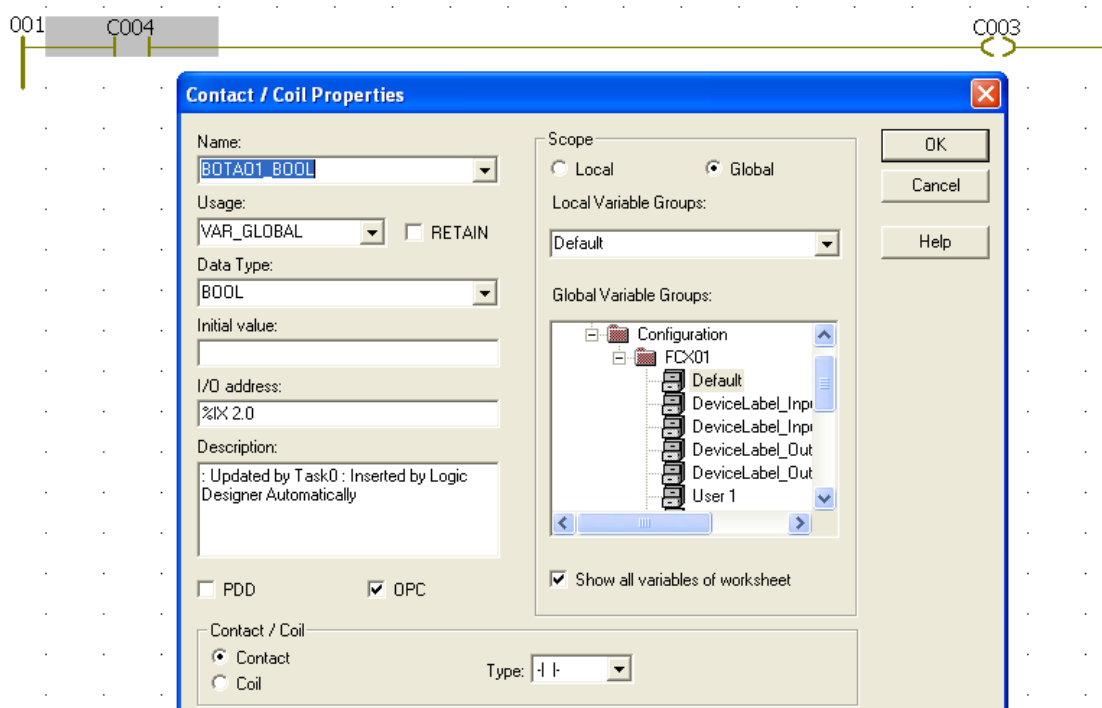


Figura 213.

É importante informar que ao definir o tag “BOTA01_BOOL” como variável vinculada a instrução “Contato-Aberto” automaticamente o sistema cria uma variável de escopo Local com o mesmo nome. Esta variável de escopo local é um espelho da mesma variável externa de escopo Global.

- Duplo-clique no contato aberto da linha de programa, conforme mostrado na figura a seguir. Em seguida, no campo “Global Variable Groups:”, selecione “Physical Hardware– Configuration –FCX01–Default”. Mude a opção dentro de “Scope” para “Global” e clique na opção “Show all variable of worksheet”. Selecione o tag “LAMP1_BOOL” na opção “NAME”. Não esqueça de ativar a opção “OPC”, que permitirá a comunicação com o supervisor. Clique em “OK”.

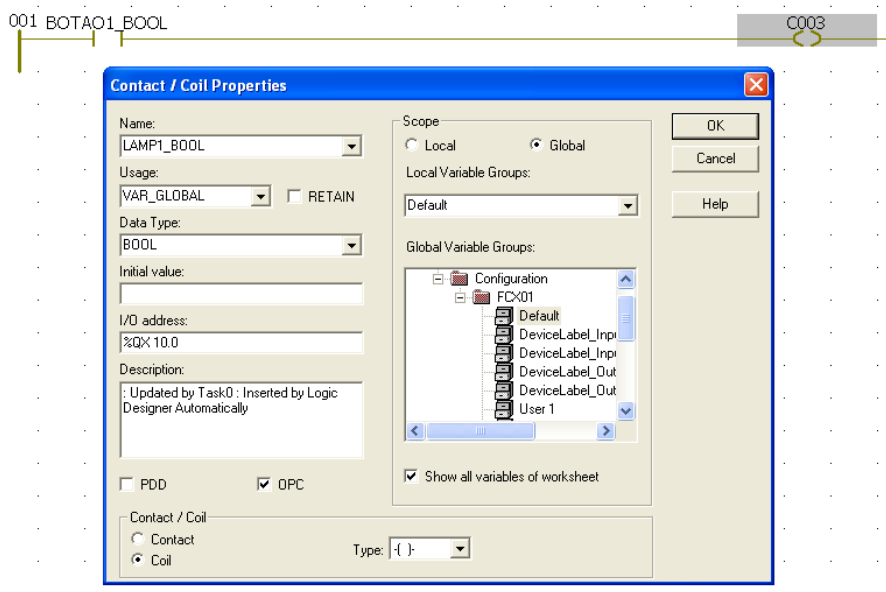


Figura 214.

- Clique numa posição específica dentro da área de trabalho, logo após a primeira linha de programa recém criada. Em seguida, dentro da janela “Edit Wizard”, clicar sobre a opção “all FUs and FBs” para visualizar todos os blocos de instruções. Caso a janela “Edit Wizard” não esteja visível, clicar no menu “View” e selcionar a opção “Edit Wizard”.

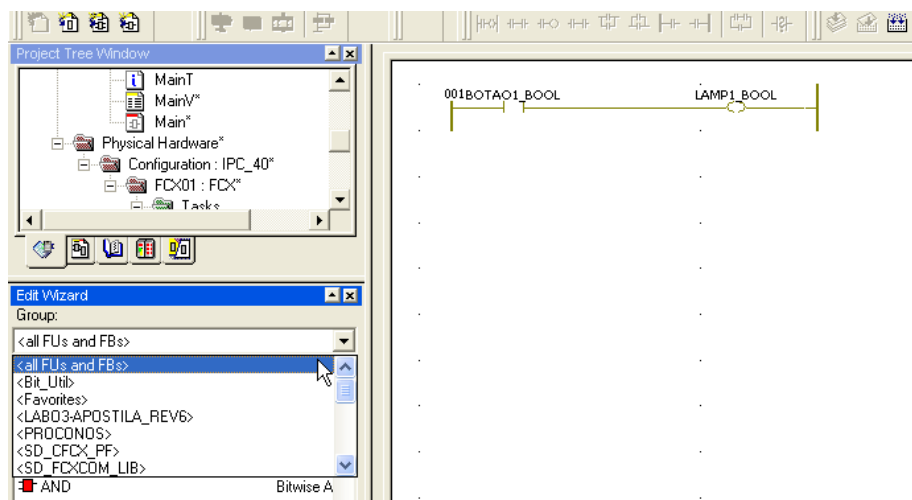


Figura 215.

- Selecionar a opção “CTU”, conforme mostrado na figura a seguir. Em seguida, confirmar o nome CTU_1 como nome desta instrução e clicar em “OK”.

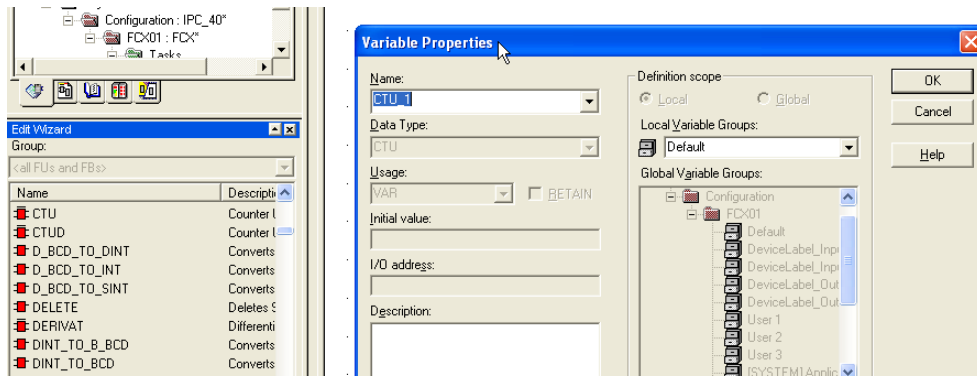


Figura 216.

- Desenvolver o programa mostrado a seguir. Observar que a variável CONTAGEM deverá ser do tipo OPC.
- Ao ligar **PV** e **CV**, basta clicar na bolhinha e digitar “INT#8” e “CONTAGEM”, não é necessário declarar essas tags.

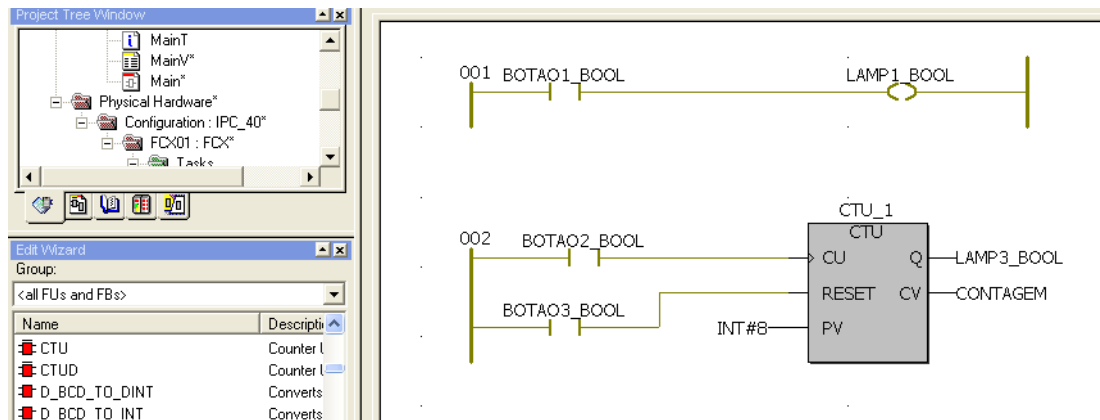


Figura 217.

- Completar o programa com as linhas a seguir. As variáveis “ACIONA_LAMP4” e “SUP_VALV_Y” também deverão ser do tipo OPC, pois serão respectivamente recebidas e enviadas de/para o supervisor.

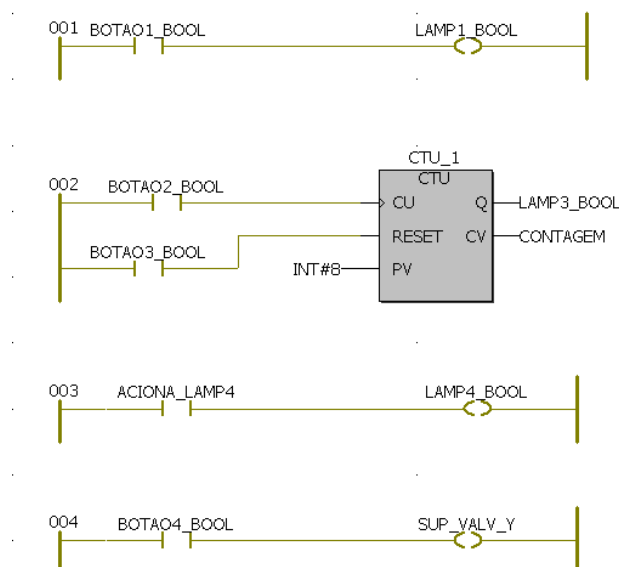


Figura 218.

Não se esqueça de definir as variáveis como OPC no “GLOBAL SETTINGS”

DeviceLabel_Input_A									
DeviceLabel_Input_D									
BOTAO1	DTag_I_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IB 0					
BOTAO2	DTag_I_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IB 8					
BOTAO3	DTag_I_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IB 16					
BOTAO4	DTag_I_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IB 24					
BOTAO1_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IX 2.0					
BOTAO2_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IX 10.0					
BOTAO3_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IX 18.0					
BOTAO4_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IX 26.0					
DeviceLabel_Output_A									
DeviceLabel_Output_D									
LAMP1_RB	DTag_O_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IB 32					
LAMP1_RB_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IX 34.0					
LAMP2_RB	DTag_O_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IB 40					
LAMP2_RB_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IX 42.0					
LAMP3_RB	DTag_O_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IB 48					
LAMP3_RB_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IX 50.0					
LAMP4_RB	DTag_O_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IB 56					
LAMP4_RB_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%IX 58.0					
LAMP1	DTag_O_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%QB 32					
LAMP2	DTag_O_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%QB 40					
LAMP3	DTag_O_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%QB 48					
LAMP4	DTag_O_Sts	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%QB 56					
LAMP1_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%QX 34.0					
LAMP2_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%QX 42.0					
LAMP3_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%QX 50.0					
LAMP4_BOOL	BOOL	VAR_GLOBAL	! Updated by Task0 : Inserted by Logic Designer Automati...	%QX 58.0					

Figura 219.

AJUSTANDO UMA “TASK”

- Selecione “Task0” exibida na árvore do projeto. Clique com o botão direito do mouse e selecione “Properties”. Task0 é do tipo “cyclic”. Após confirmação do tipo da “task”, clique em “Cancel” para fechar esta janela.
- Selecione “Task0” com o botão direito do mouse e selecione “Settings”. Irá surgir uma janela “Task setting for IPC_40”. Nesta tarefa, valores como “interval” e “watchdogtime” serão mantidos.. Após confirmação, clique em “Cancel” para fechar esta janela.

Nota:

◁ “Default task”: É a que possui menor prioridade pois é executada apenas quando nenhuma outra está em execução.

◁ “Cyclical Task”: Executada de acordo com a sua especificação de intervalo (interval). O que a torna prioritária.

◁ “System Task”: Uma task desse tipo é chamada a ser executada apenas pelo sistema quando na ocorrência de um erro durante a operação com a “FCX”.

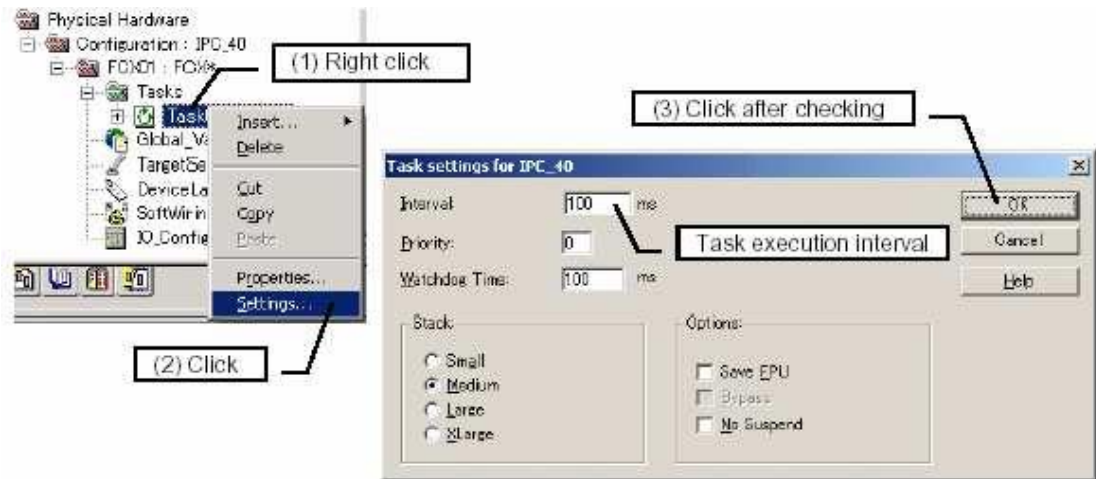


Figura 220.

ASSOCIANDO UMA “TASK” A UM PROGRAMA

- Selecione “Task0” e clique com o botão direito do mouse. Selecione “Insert - Program Instance”, através do menu.
- Entre no campo “Program Instance” e insira “_COMANDOS” e selecione “COMANDOS” no campo “Program Type”. Clique em “OK”.

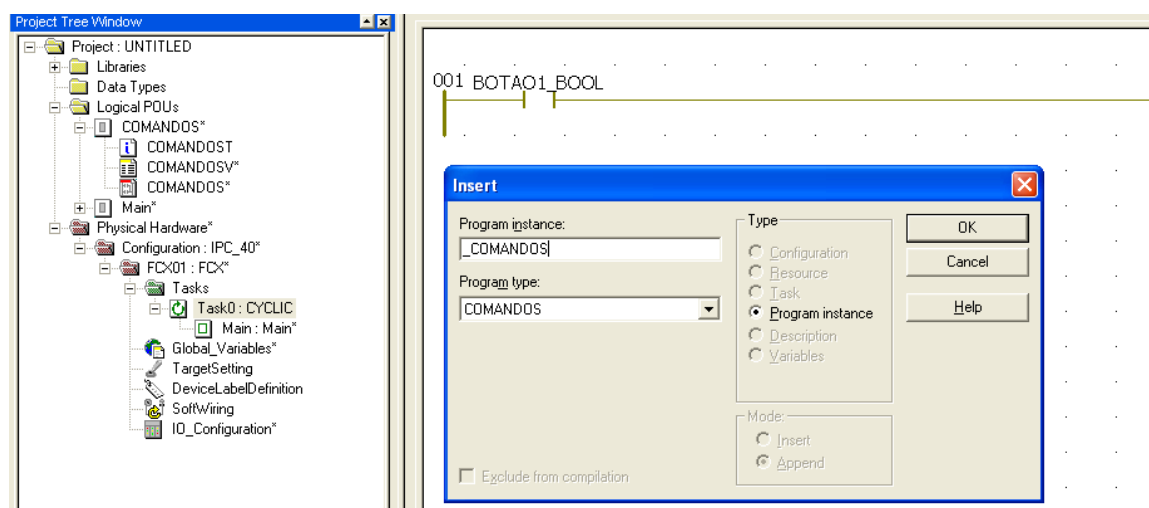


Figura 221.

FAZENDO O “BUILDING” DO PROJETO

Neste ponto, estaremos convertendo para código de máquina toda a aplicação desenvolvida até o momento.

- Selecione “Build – Rebuild Project” através da barra de menu. Uma mensagem aparecerá informando que a compilação está em progresso.

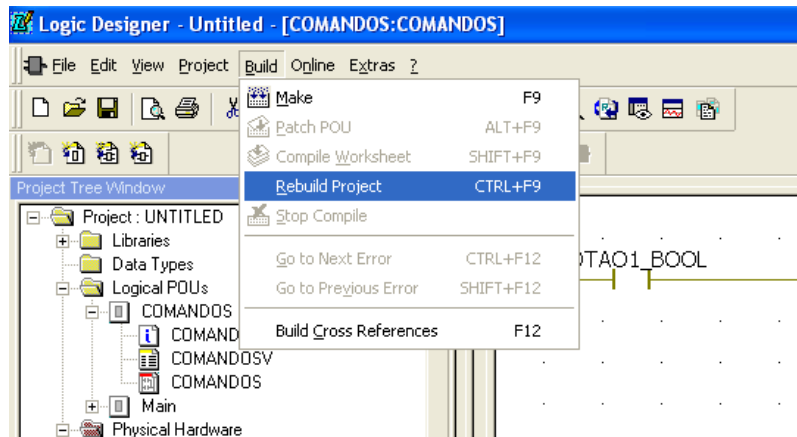


Figura 222.

Quando concluída a compilação, o resultado da mesma será exibida na janela de mensagem como na figura abaixo. Toda mensagem que exibe “0 Error(s)” indica que a compilação foi concluída com sucesso.

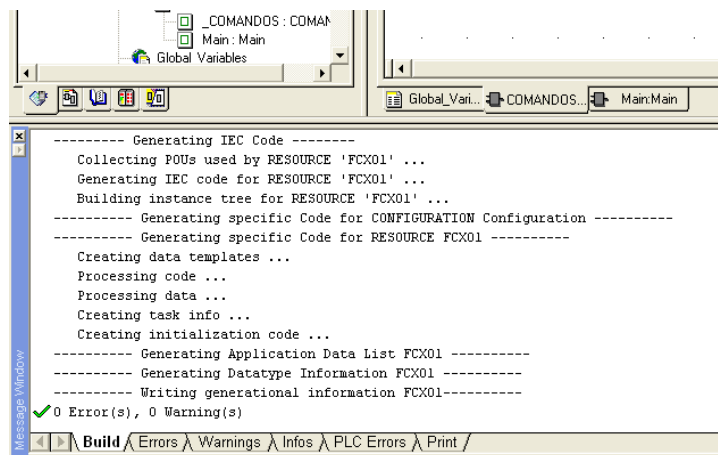


Figura 223.

FAZENDO O “DOWNLOAD” DE UM PROJETO

O que será feito agora é a carga do projeto compilado para a “FCX”.

- Selecione “Online – Project Control” através da barra de menu. O diálogo do “Project Control” aparecerá como na figura abaixo:

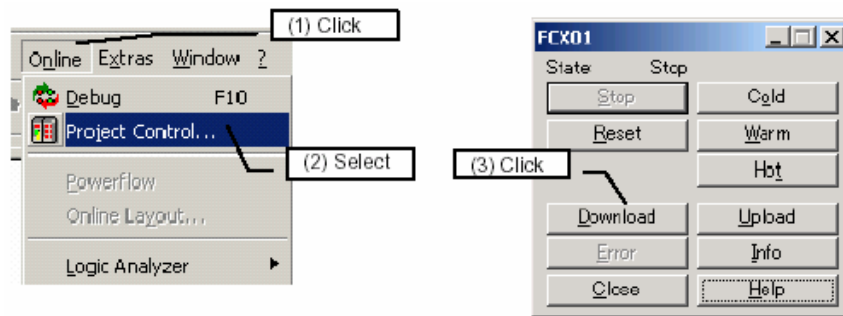


Figura 224.

- Clique em “Download”. O diálogo deste botão será exibido.
- Clique no botão “Download” abaixo do campo “Project”.

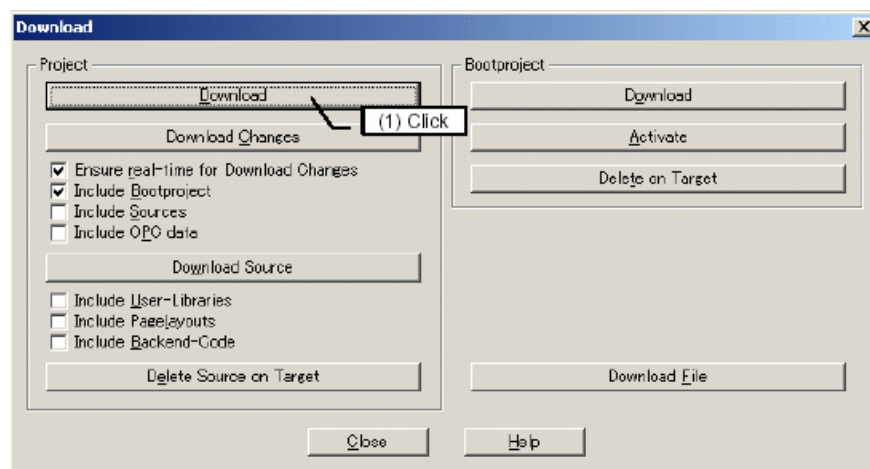


Figura 225.

Se algum outro projeto foi carregado anteriormente, aparecerá uma janela informando que este projeto será sobrescrito. Clique em “YES”.

Na parte inferior do Logic Designer será exibida uma barra de progressão referente ao “download”. Aguarde até que haja a indicação operação realizada.

INICIANDO UM PROJETO

Neste passo, a aplicação ainda não foi executada. Para alterar o estado do CLP, selecione um dos modos proposto na tabela abaixo:

- Clicar no ícone “Project Control Dialog”, conforme figura abaixo.

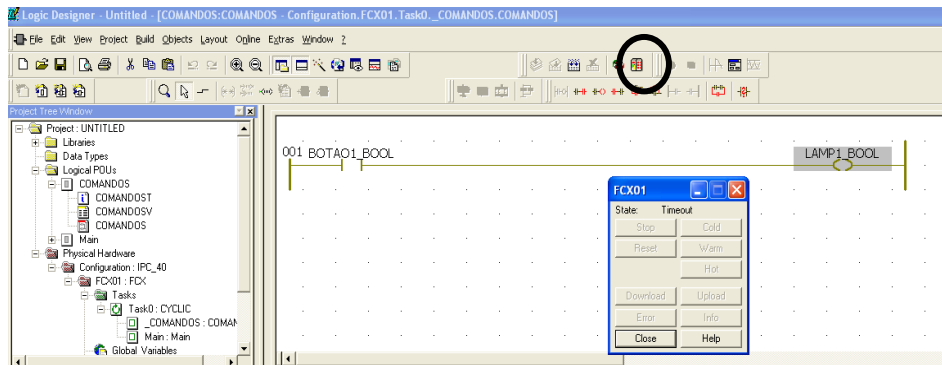


Figura 226.

- Neste exercício, iremos optar pela inicialização a frio (“Cold”). Vide figura abaixo:

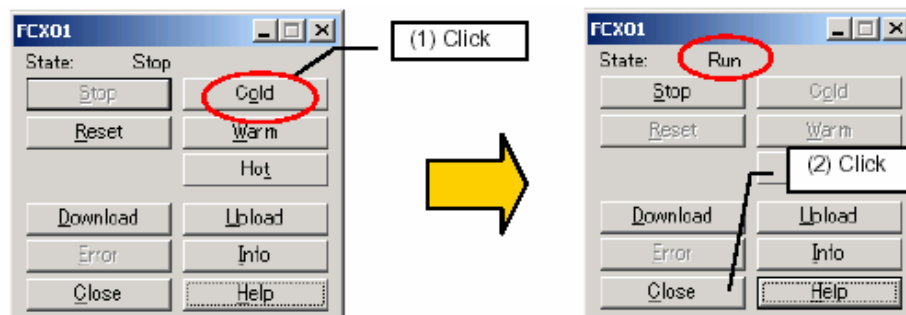


Figura 227.

SALVANDO UM PROJETO

- Selecione “File – Save Project As/Zip Project As” no menu do “Logic Designer”.
- Entre com um nome para o seu projeto (LABO3, por exemplo) e clique em “Save”.

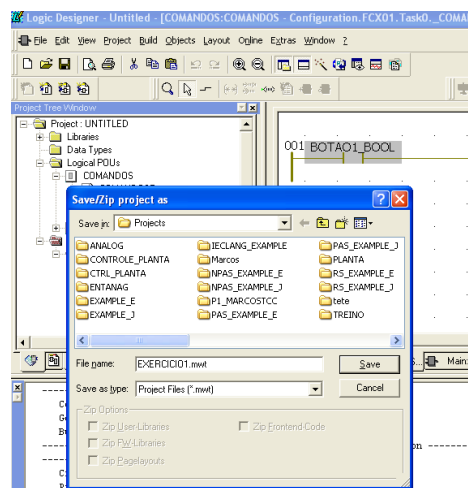


Figura 228.

ETAPA 3. CRIAR TELA NO PROJETO

A figura a seguir mostra a tela a ser criada pelo supervisor.

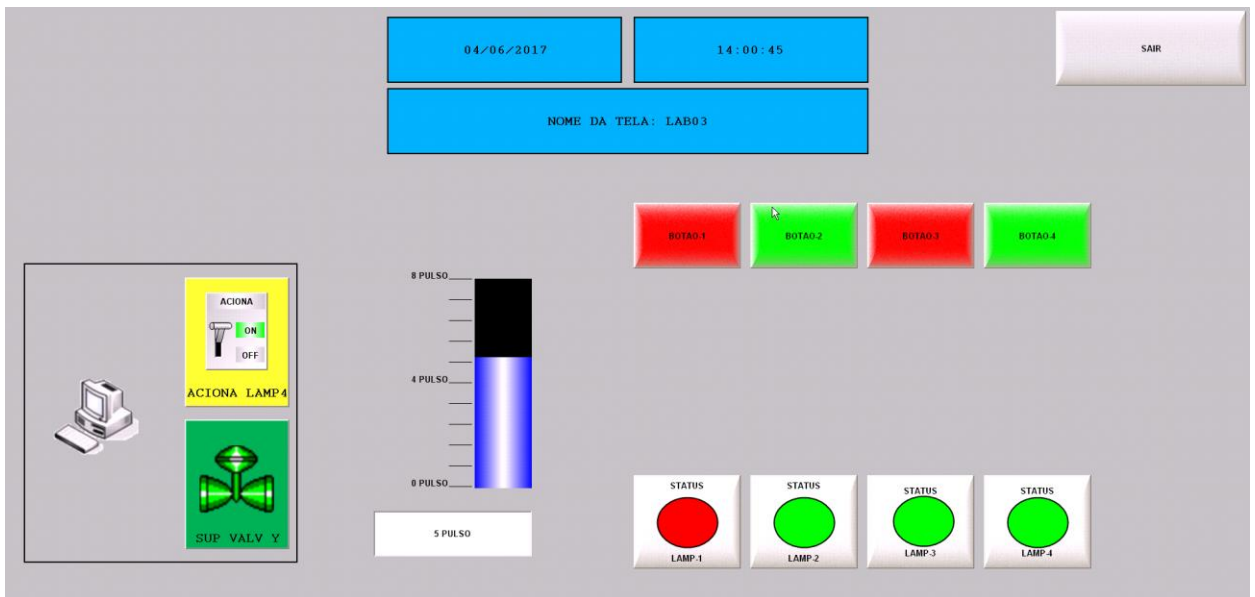


Figura 229.

Selecione a aba "**Gráficos**" na Área de Trabalho e selecione na pasta "**Telas**". Agora, clique com o botão direito na pasta "**Telas**" e clique na opção "**Inserir**"

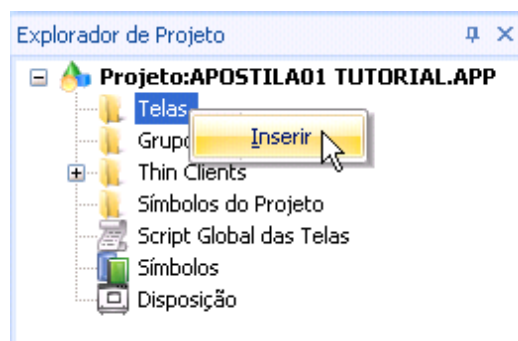


Figura 230.

Na janela "**Atributos de Tela**" devemos configurar alguns atributos gerais sobre a tela que está sendo criada. Primeiro configure a tela **LABO-3** como mostrado na figura abaixo:

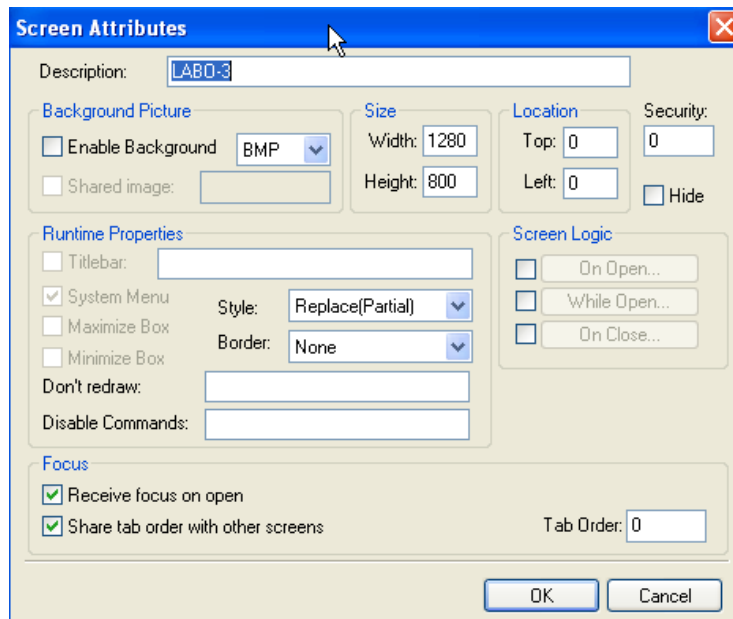


Figura 231.

Pressione o botão OK e uma tela vazia deverá aparecer com os atributos configurados para ela.

Depois de criar a tela, mudemos a cor de fundo usando o “**Cor de Fundo**”.

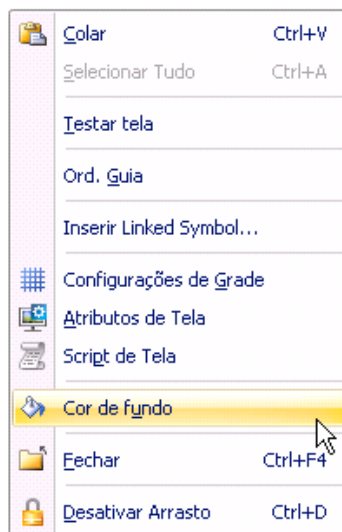


Figura 232.

Selecione o **cinza claro** nas opções de cores da janela aberta e clique em **OK**:

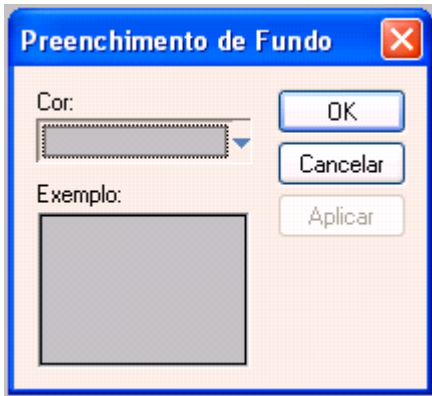


Figura 233.

Salve a tela como LABO3. Para salvar, clique no menu File na barra superior, e no item Save As e digite o nome LABO3 no campo Nome do arquivo, e clique em Salvar.

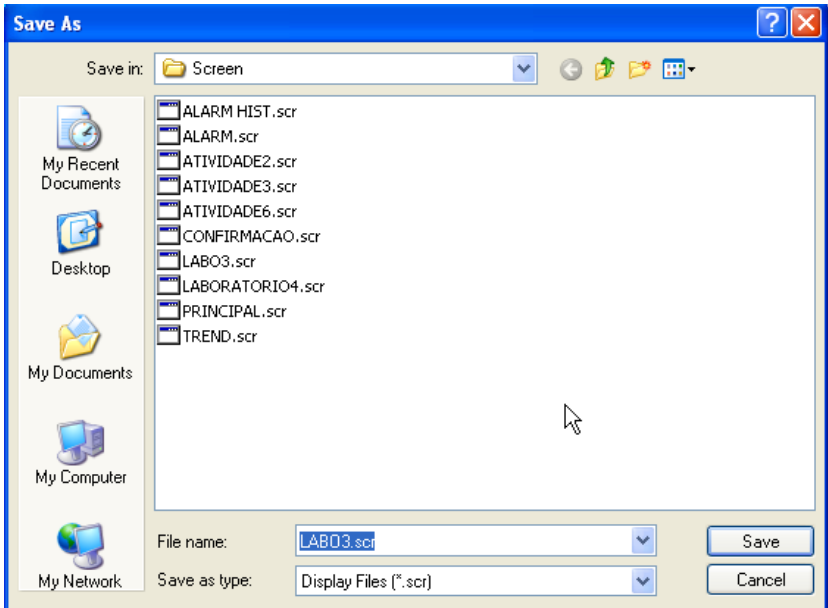


Figura 234.

Vamos configurar a “**Planilha de Dados**”, como mostrado a seguir:

Crie os tags, conforme mostrado a seguir.

1	LAMPADA	4	Booleana		Servidor	
2	BOTAO	4	Booleana		Servidor	
3	CONTAGEM	0	Inteira		Servidor	
4	SUPVALVY	0	Booleana		Servidor	
5	ACIONALAMP4	0	Booleana		Servidor	

Figura 235.

Salve a configuração do banco de dados usando o ícone:

Para criar os botões, lâmpadas e os outros objetos, usaremos objetos da biblioteca. Para acessarmos a biblioteca de símbolos, clique com no ícone mostrado na figura abaixo:

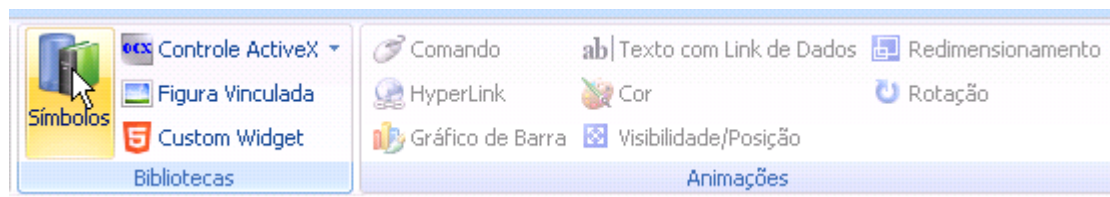


Figura 236.

Para importar um objeto da biblioteca para a tela da aplicação, simplesmente clique no objeto e depois na tela.

Depois de criado os objetos, vamos configurá-los.

Os quatro botões com as propriedades tipo “Cor”, conforme é mostrada na figura a seguir.



Figura 237.

As quatro lâmpadas com as propriedades tipo “Colors”, conforme é mostrada na figura a seguir.

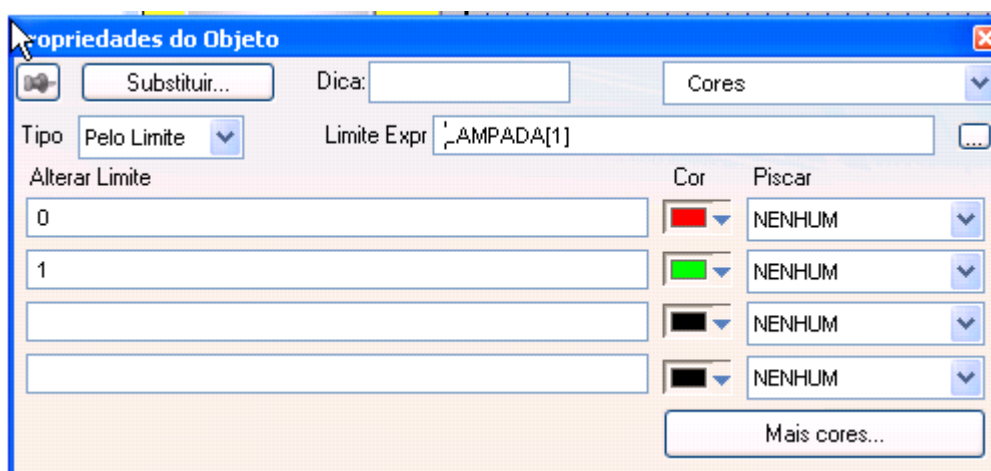


Figura 238.

O “**Bargraph**” para contagem de pulso com as propriedades mostradas na figura a seguir.

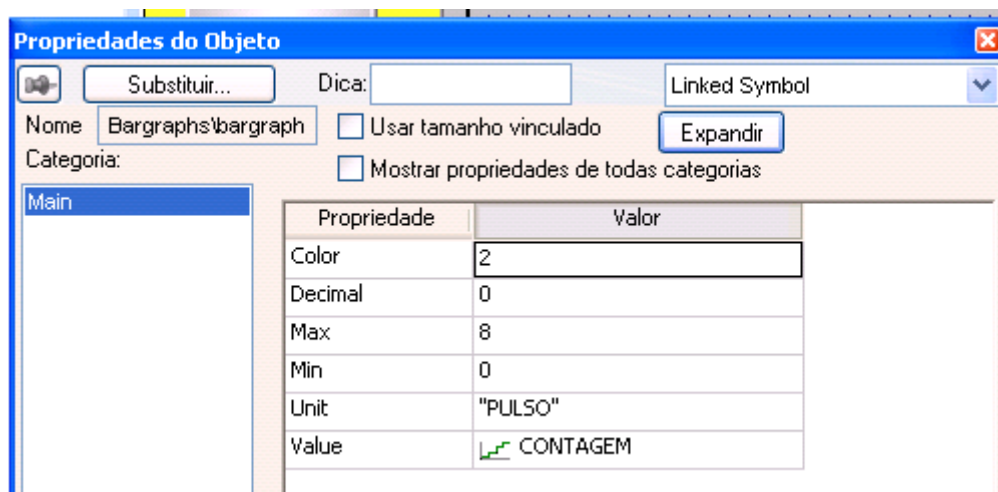


Figura 239.

O “**Comando**” para acionamento da lâmpada 4, com as propriedades mostradas na figura a seguir.

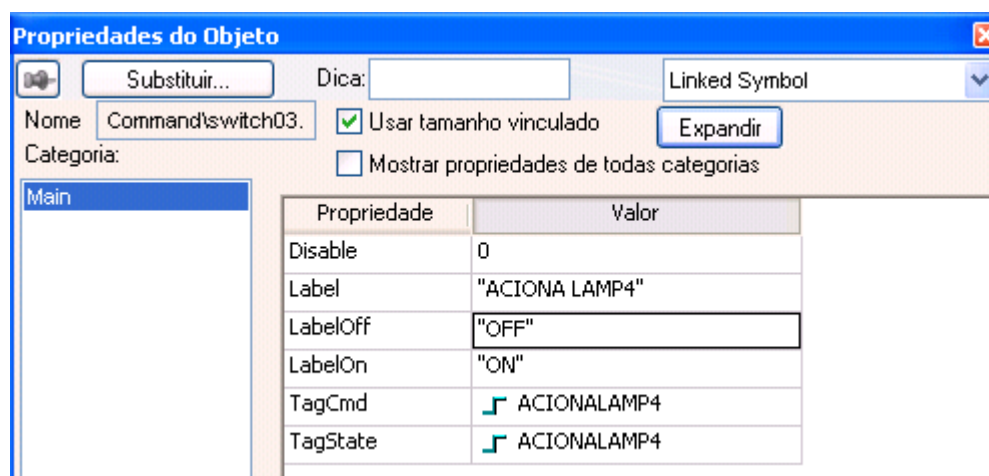


Figura 240.

O “**Válvula**” comandada pelo 4º. Botão do CLP, com as propriedades mostradas na figura a seguir.

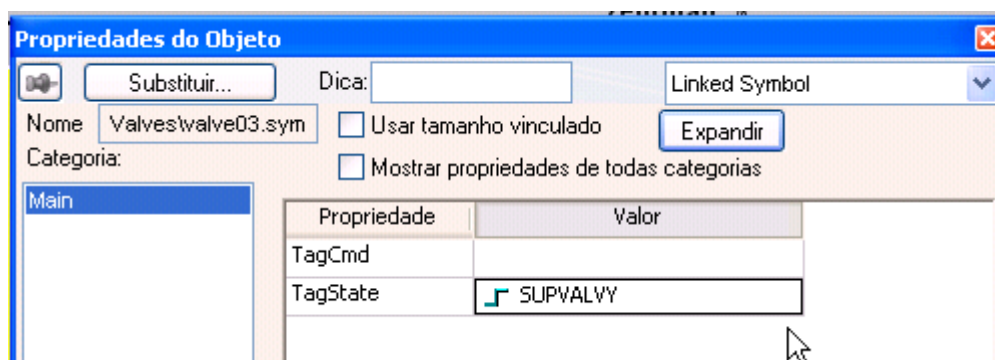


Figura 241.

Salve a tela como LABO3.

ETAPA 4. CONFIGURAR COMUNICAÇÃO DO INDUSOFT COM O STARDOM VIA OPC

Selecionar a aba “**Comunicação**” no rodapé do explorador de projeto.



Figura 242.

Clicar com a tecla da direita do mouse na pasta “**OPC DA 2.05**” e inserir uma planilha de comunicação via OPC.

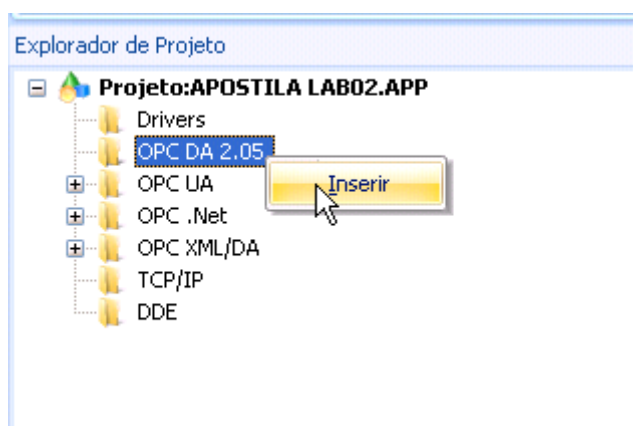


Figura 243.

Selecionar a opção “**Yokogawa.ExaopcDASTARDOMFCX**” no campo “**Identificador do Servidor**” conforme mostrado a seguir:

Descrição:	Identificador do Servidor:	Desativar:
	aopcDASTARDOMFCX	
Taxa de Atual. Leitura (ms):	Porcentagem Banda Morta:	Estado:
Nome do Servidor Remoto:	☐ Ler antes de escrever ☐ Ler depois de escrever ☒ Aceitar Nome de Tag na coluna Item	
	Procurar...	

Figura 244.

Preencher o campo “**Nome do Tag**” com a variável: **BOTAO[1]**, conforme figura a seguir:

	Nome da Tag	Item	Scan	Div	Adicionar
	Filtro de Texto	Filtro de Texto	(Todos)	Filtro de Te	Filtro de Te
1	BOTAO[1]		Sempre		
2	BOTAO[2]		Sempre		
3	BOTAO[3]		Sempre		
4	BOTAO[4]		Sempre		

Figura 245.

Duploclicar no campo em branco “Item” ao lado do “Nome do Tag”: **BOTAO[1]**.
Abri-se-á a janela de configuração do **OPC BROWSER**, conforme figura a seguir:

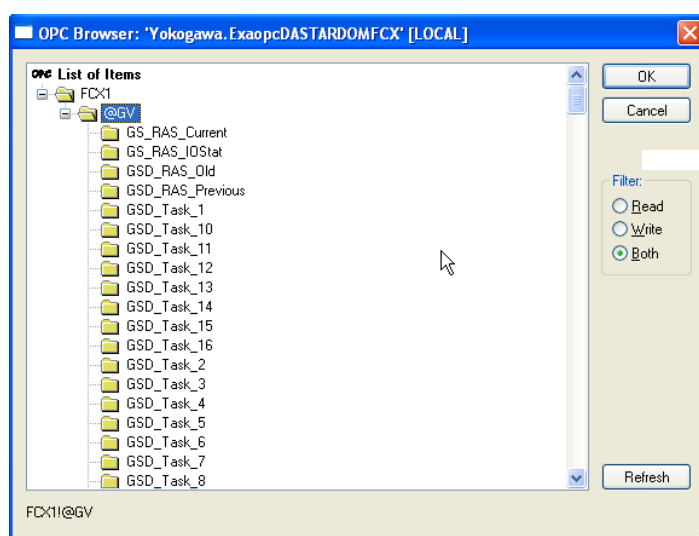


Figura 246.

Abrir a pasta: “@GV” do FCX1, que corresponde a todas variáveis do programa criado no CLP, que poderão ser acessadas via OPV pelo Indusoft Web Studio, e selecionar a opção: “BOTAO1_BOOL”, conforme figura a seguir. Em seguida clicar no botão “OK”:

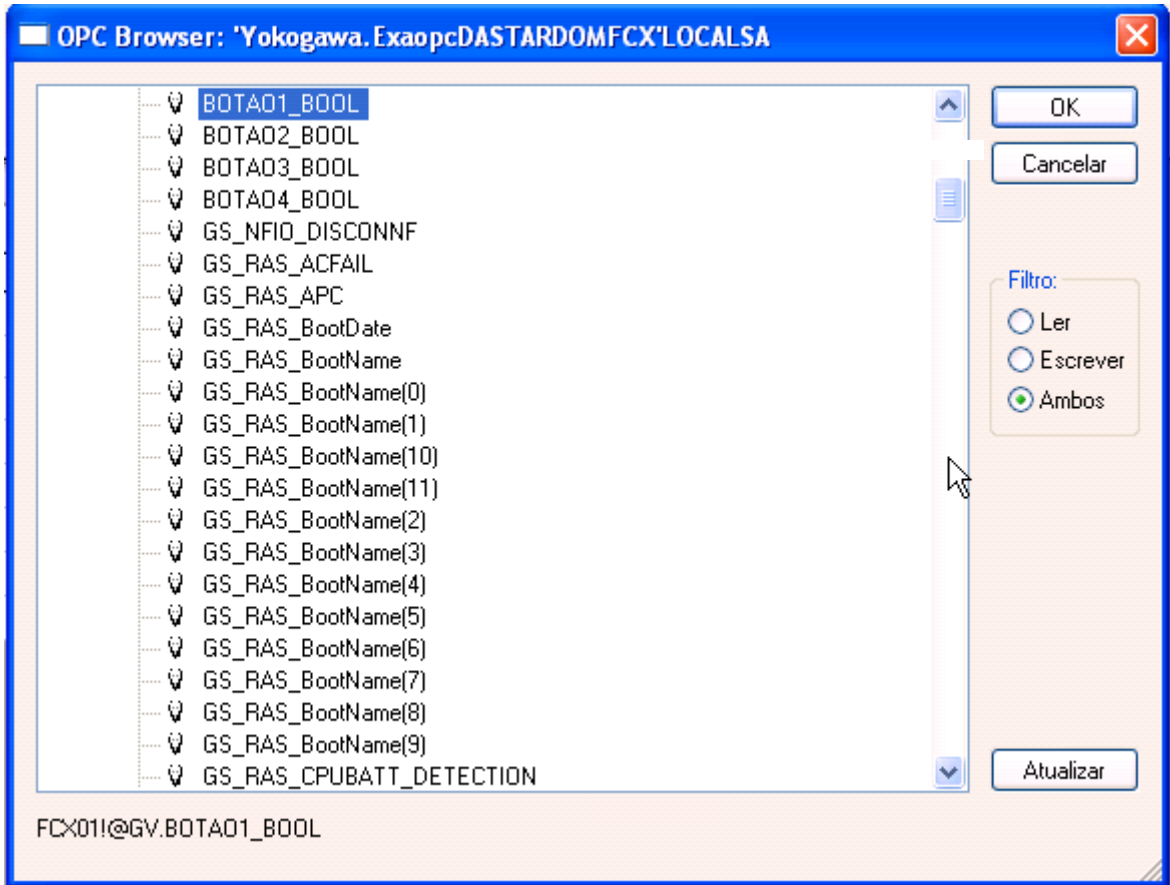


Figura 247.

Repetir o procedimento para os outros três botões, também para as quatro lâmpadas, a variável SUP_VALV_Y e a variável CONTAGEM, conforme figura a seguir. Em seguida, salvar a planilha de configuração do OPCCL001.

	Nome da Tag	Item	Scan	Div	Adicionar
	Filtro de Texto	Filtro de Texto	(Todos)	Filtro de Te	Filtro de Te
1	BOTA0[1]	FCX01!@GV.BOTA01_BOOL	Sempre		
2	BOTA0[2]	FCX01!@GV.BOTA02_BOOL	Sempre		
3	BOTA0[3]	FCX01!@GV.BOTA03_BOOL	Sempre		
4	BOTA0[4]	FCX01!@GV.BOTA04_BOOL	Sempre		
5	LAMPADA[1]	FCX01!@GV.LAMP1_BOOL	Sempre		
6	LAMPADA[2]	FCX01!@GV.LAMP2_BOOL	Sempre		
7	LAMPADA[3]	FCX01!@GV.LAMP3_BOOL	Sempre		
8	LAMPADA[4]	FCX01!@GV.LAMP4_BOOL	Sempre		
9	ACIONALAMP4	FCX01!@GV.ACIONA_LAMP4	Sempre		
10	SUPVALVY	FCX01!@GV.SUP_VALV_Y	Sempre		
11	CONTAGEM	FCX01!@GV.CONTAGEM	Sempre		

Figura 248.

Salve a aplicação e rode o sistema. Verificar se as válvulas e o status dos botões mudam de cor, quando são acionados os botões reais do Kit didático de CLP.

ETAPA 5. ACRESCENTAR FUNCIONALIDADES NO SISTEMA

Alterar o programa do CLP, através do LOGIC DESIGNER, de tal forma que o valor limite de contagem de pulsos passe a ser um valor variável entre 5 e 15 e não mais um valor constante de 8 pulsos.

Alterar também, o software INDUSOFT, na tela e na planilha de comunicação OPC, para que o operador possa ajustar, usando um botão deslizante ou inserindo diretamente pelo teclado, o valor limite de contagem de pulsos. Esta nova ação no supervisório deverá ser criada na região da tela, conforme figura a seguir.

*O LOCAL ONDE SERÁ POSICIONADO ESTE OBJETO É DE SUA ESCOLHA.



Figura 249.

4ª. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA STARDOM + SUPERVISÓRIO COM TEMPORIZADOR

O objetivo deste trabalho prático visa conhecer as ferramentas para programação do CLP Stardom utilizando instruções de contagem e temporização com software LOGIC DESIGNER e construção de aplicação para supervisão, ajustes e monitoramento de valores utilizando os software supervisório INDUSOFT WEB STUDIO.

ETAPA 1. CONFIGURAÇÃO DE HARDWARE



- Duplo clique no ícone abaixo:

O “Device Label” é um rótulo dado às entradas e saídas do sistema. Assim, torna-se mais familiar a manipulação de variáveis de campo, sabendo-se o seu nome de projeto.

- Com o resource configurator em modo “tree” clique em “IOM”.
- Duplo-clique em DI/DO.
- Ajuste o device label correspondente a 1ª. entrada digital para “BOTAO1”.
- Ajuste o device label correspondente a 2ª. entrada digital para “BOTAO2”.
- Ajuste o device label correspondente a 1ª. saída digital para “LAMP1”.
- Ajuste o device label correspondente a 2ª. saída digital para “LAMP2”.
- Selecione File-Download
- Quando aparecer uma janela de prompt, clique no OK.

ETAPA 2. CRIAÇÃO DA APLICAÇÃO DE CONTROLE (PROGRAMA DO CLP)

Neste momento será iniciado o desenvolvimento da criação de uma aplicação de controle.

- Para iniciar o Logic Designer, clique no ícone na área de trabalho como exibido na figura



ao lado:

- Selecione “File-New Project”.
- Selecione o ícone “STARDOM NPAS” e clique “OK”.

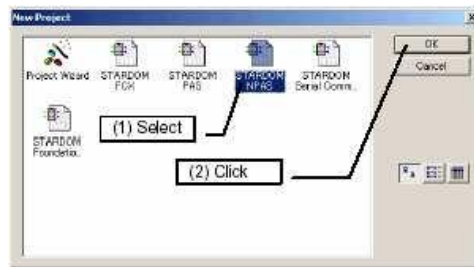


Figura 250.

- Duplo-clique em “TargetSetting” na árvore do projeto.
- Entre com o número de IP “192.168.0.XX” e clique em “OK”. Obs.: XX=Número variável de acordo com a bancada.

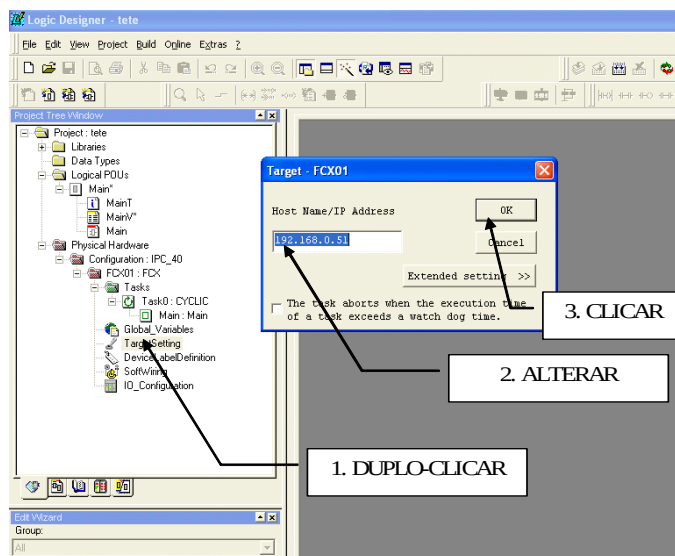


Figura 251.

Nesta aplicação iremos utilizar duas “variáveis globais” que farão as vezes dos sinais de entrada e saída (digitais) deste exercício.

- Duplo-clique sobre “DeviceLabelDefinition” na árvore do projeto.
- Dentro da janela defina as variáveis como descritas abaixo. Após finalizar os ajustes das variáveis, clique em “Ok” e clique em “YES” na janela de diálogo de confirmação.

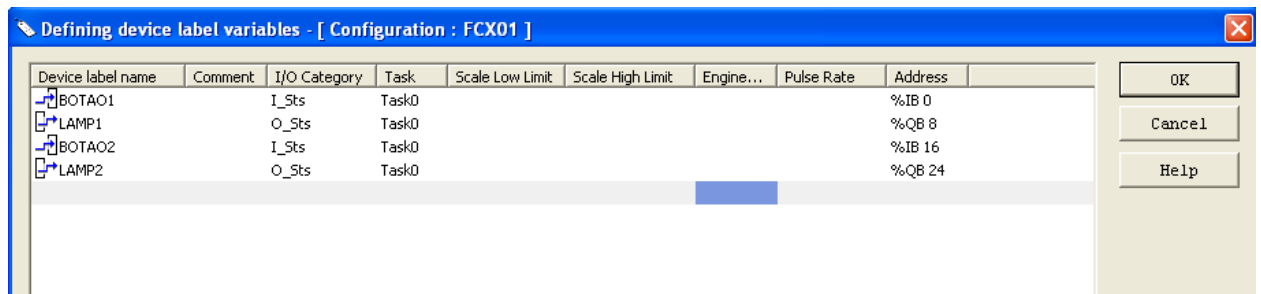


Figura 252.

Iremos agora construir uma aplicação de controle usando a unidade POU (Program Organization Unit).

- Clique com o botão direito sobre “Logical POU’s” na árvore do projeto e então selecione “Insert – Program” para aparecer a janela exibida abaixo.
- Neste exercício, insira no campo “Name” a informação “COMANDOS” e selecione “LD” (LADDER) no campo “Language” e clique em “Ok”.

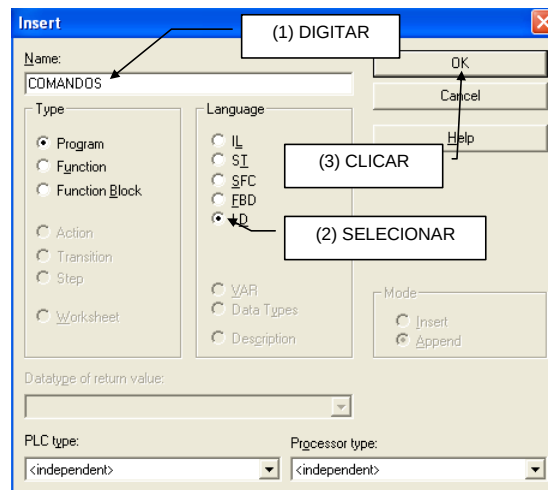


Figura 253.

Verifique no seu projeto se como na figura abaixo a estrutura com três ícone foram criados:

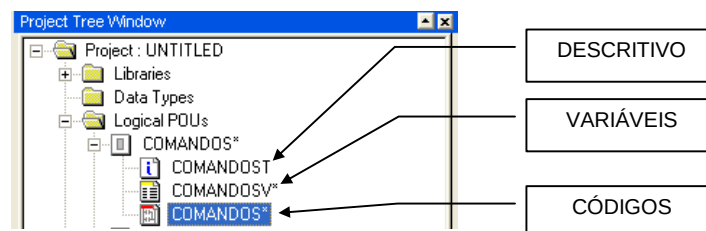


Figura 254.

- Duplo-clique sobre “COMANDOS” (ícone de código) criado no projeto e clique numa posição específica dentro da área de trabalho a fim de determinar onde ficará a linha de programa. Em seguida, clicar sobre o ícone “Contact Network” para iniciar uma linha básica de programa.

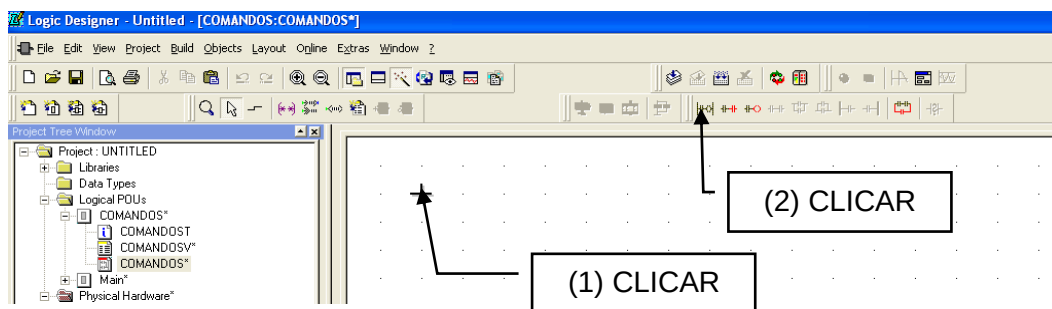


Figura 255.

- Desenvolver o programa conforme mostrado na figura a seguir. A tabela abaixo mostra as propriedades de cada variável a ser utilizada neste programa. Não esqueça de ativar a opção “**OPC**”, em “**Global Variables**”, das seguintes variáveis: BOTAO1_BOOL; BOTAO2_BOOL; LAMP1_BOOL; LAMP2_BOOL; AJUSTA_TEMPO_DINT; CONT_PULSO e CONTA_TEMPO_DINT , que permitirá a comunicação com o supervisor

Nome da Variável	Escopo (“Scope”)	Uso (“Usage”)	Tipo (“Data Type”)	Campo OPC
AUX1	LOCAL	VAR	BOOL	NÃO
CONT_PULSO	LOCAL	VAR	INT	SIM
AJUSTA_TEMPO	LOCAL	VAR	TIME	NÃO
CONTA_TEMPO	LOCAL	VAR	TIME	NÃO
CONTA_TEMPO_DINT	LOCAL	VAR	DINT	SIM
AJUSTA_TEMPO_DINT	LOCAL	VAR	DINT	SIM
ADJ_TEMPO	LOCAL	VAR	REAL	SIM

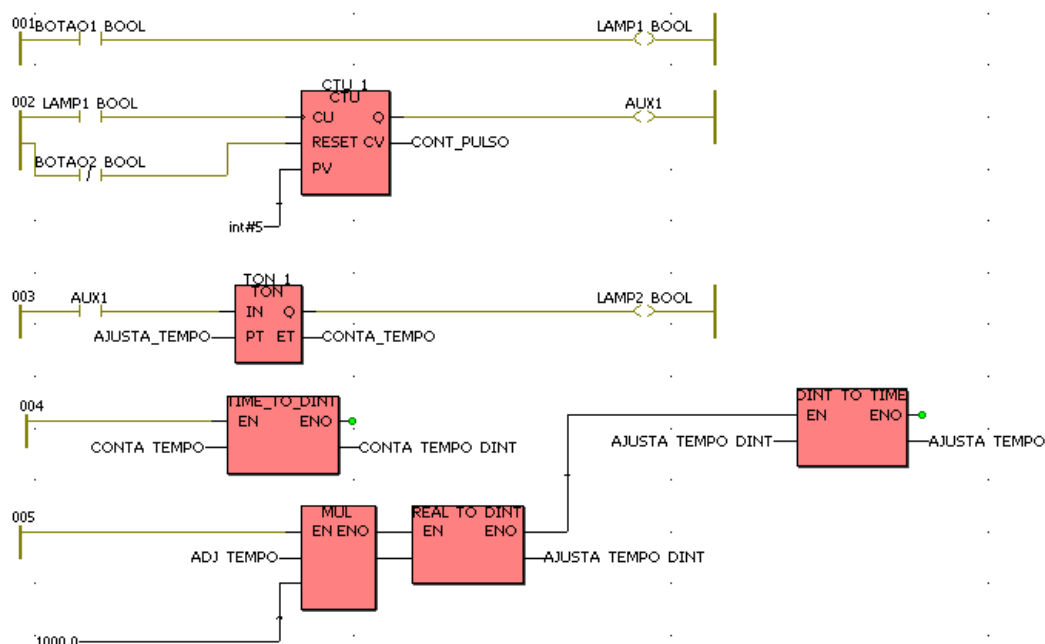


Figura 256.

AJUSTANDO E ASSOCIANDO UMA “TASK”

- Selecione “Task0” exibida na árvore do projeto. Clique com o botão direito do mouse e selecione “Properties”. Task0 é do tipo “cyclic”. Após confirmação do tipo da “task”, clique em “Cancel” para fechar esta janela.
- Selecione “Task0” com o botão direito do mouse e selecione “Settings”. Irá surgir uma janela “Task setting for IPC_40”. Nesta tarefa, valores como “interval” e “watchdogtime” serão mantidos.. Após confirmação, clique em “Cancel” para fechar esta janela.

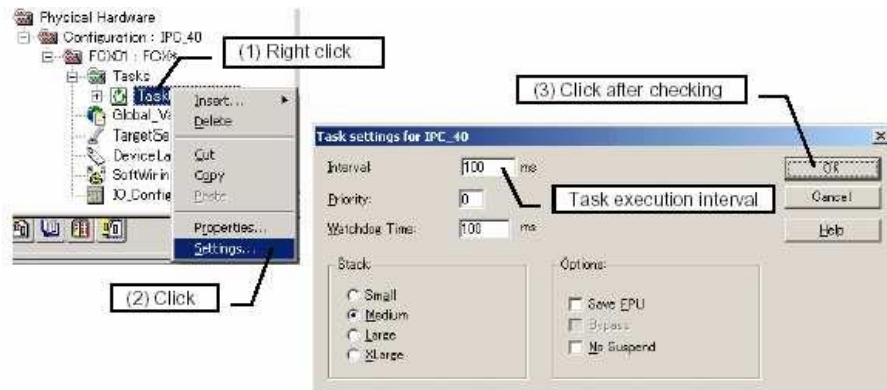


Figura 257.

- Selecione “Task0” e clique com o botão direito do mouse. Selecione “Insert - Program Instance”, através do menu.
- Entre no campo “Program Instance” e insira “_COMANDOS” e selecione “COMANDOS” no campo “Program Type”. Clique em “Ok”.

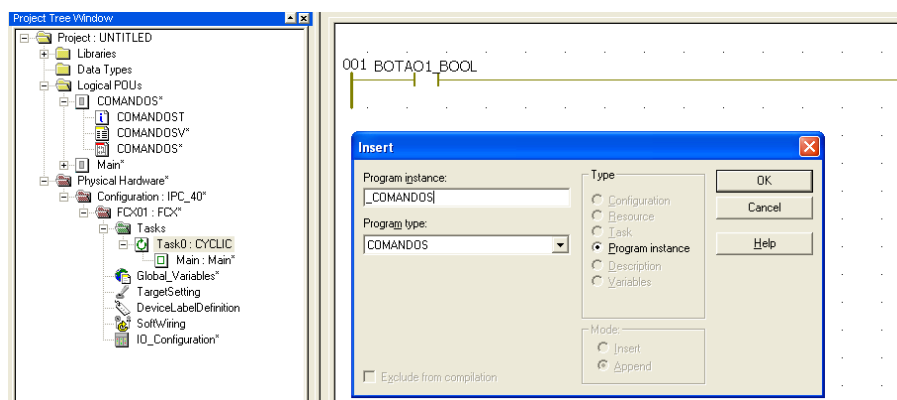


Figura 258.

FAZENDO O “BUILDING” E “DOWNLOAD” DO PROJETO

Neste ponto, estaremos convertendo para código de máquina toda a aplicação desenvolvida até o momento.

- Selecione “Build – Rebuild Project” através da barra de menu. Uma mensagem aparecerá informando que a compilação está em progresso.

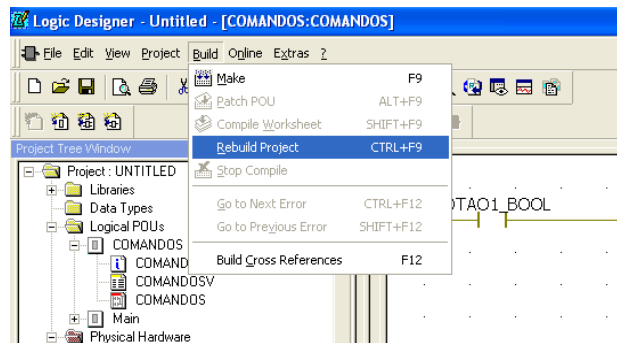


Figura 259.

- Selecione “Online – Project Control” através da barra de menu. O diálogo do “Project Control” aparecerá como na figura abaixo:

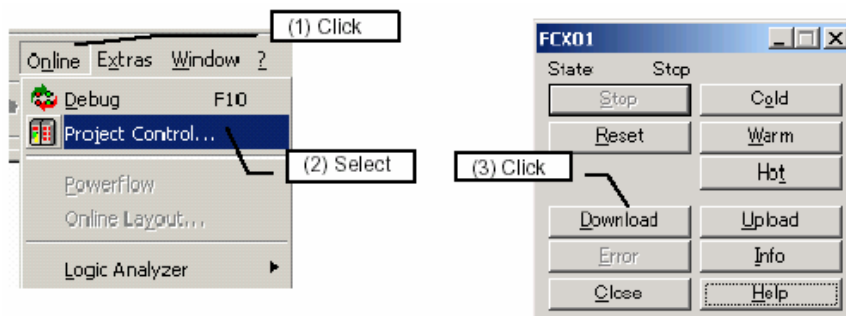


Figura 260.

- Clique em “Download”. O diálogo deste botão será exibido.
- Clique no botão “Download” abaixo do campo “Project”.

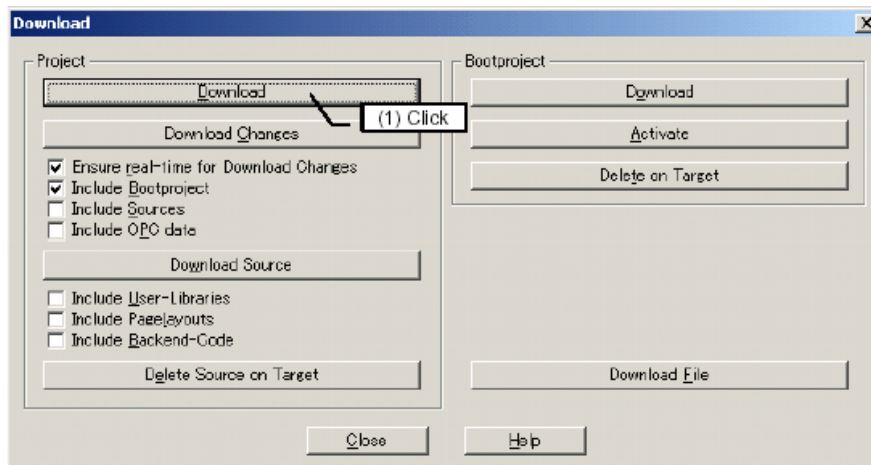


Figura 261.

- Se algum outro projeto foi carregado anteriormente, aparecerá uma janela informando que este projeto será sobrescrito. Clique em “YES”.

Na parte inferior do Logic Designer será exibida uma barra de progressão referente ao “download”. Aguarde até que haja a indicação operação realizada.

INICIANDO UM PROJETO

Neste passo, a aplicação ainda não foi executada. Para alterar o estado do CLP, selecione um dos modos proposto na tabela abaixo:

- Clicar no ícone "Project Control Dialog", conforme figura abaixo.

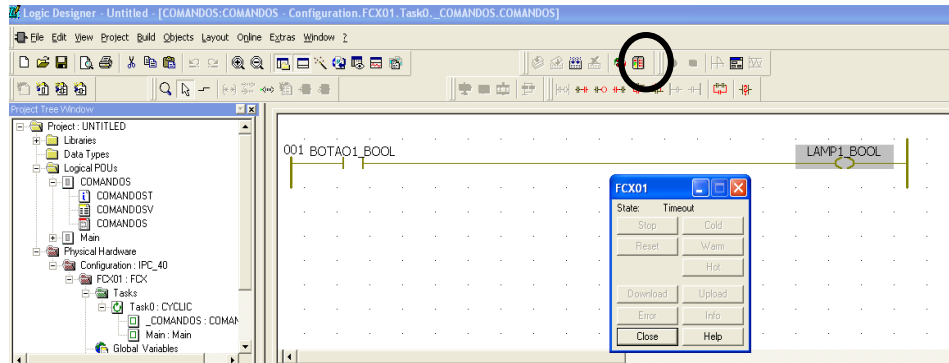


Figura 262.

Neste exercício, iremos optar pela inicialização a frio ("Cold"). Vide figura abaixo:

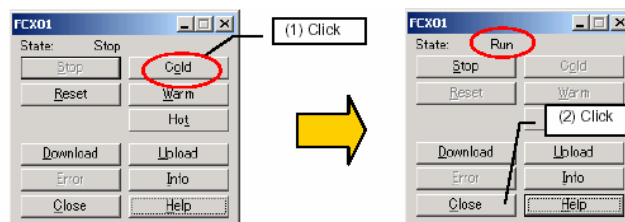


Figura 263.

SALVANDO UM PROJETO

- Selecione "File – Save Project As/Zip Project As" no menu do "Logic Designer".
- Entre com um nome para o seu projeto (EXERCICIO2, por exemplo) e clique em "Save".

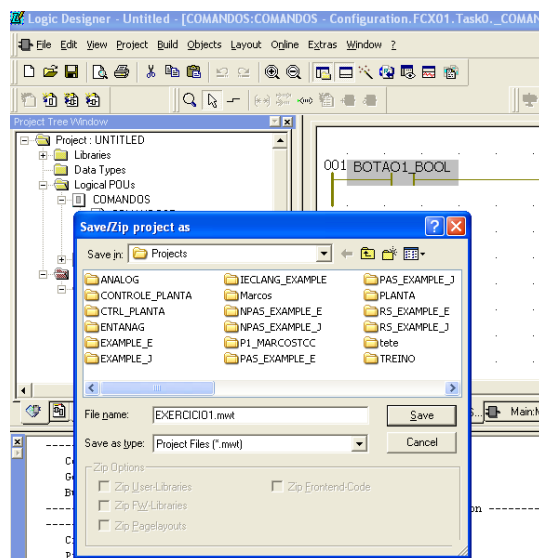


Figura 264.

ETAPA 3. CRIAR TELA NO PROJETO

A figura a seguir mostra a tela a ser criada pelo supervisor.

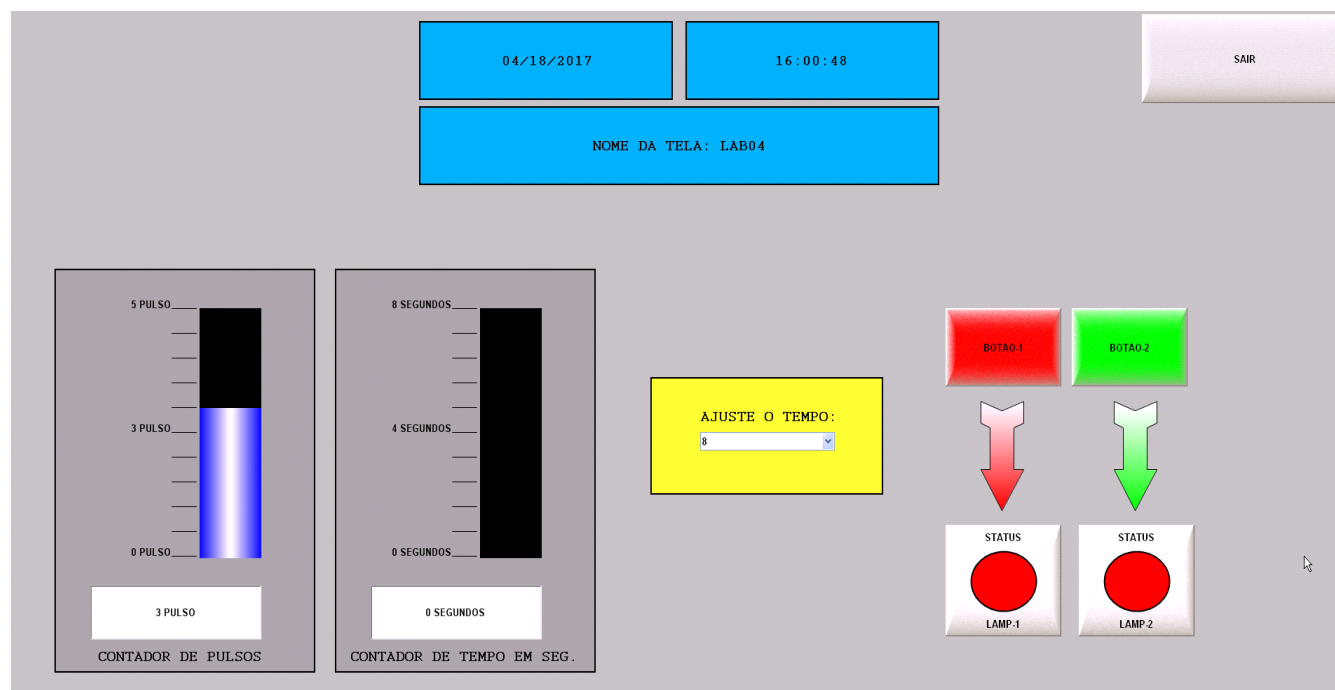


Figura 265.

Vamos configurar a “**Planilha de Dados**”, como mostrado a seguir:

Crie os tags, conforme mostrado a seguir.

	Nome	Vetor	Tipo	Descrição	Escopo
	Filtro de Texto	= ...	(Todos)	Filtro de Texto	(T...
1	LAMPADA	4	Booleana		Servidor
2	BOTAO	4	Booleana		Servidor
3	CONTAGEM	0	Real		Servidor
4	ADJ_TEMPO	0	Real		Servidor
5	CONTA_TEMPO	0	Real		Servidor

Figura 266.

Salve a configuração do banco de dados usando o ícone:

Salve a tela como LABO4.

ETAPA 4. CONFIGURAR COMUNICAÇÃO DO INDUSOFT COM O STARDOM VIA OPC

Selecionar a aba “**Comunicação**” no rodapé da árvore de projeto.



Figura 267.

Clicar com a tecla da direita do mouse na pasta “**OPC DA 2.05**” e inserir uma planilha de comunicação via OPC.

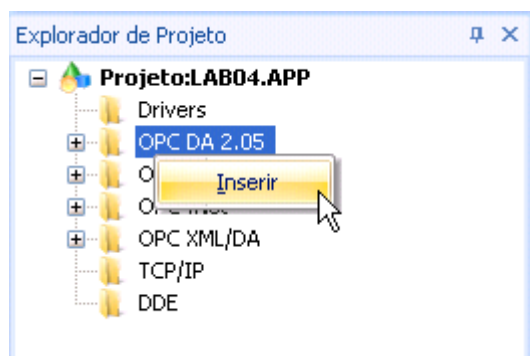


Figura 268.

Selecionar a opção “**Yokogawa.ExaopcDASTARDOMFCX**” no campo “**Identificador do Servidor**” conforme mostrado a seguir:

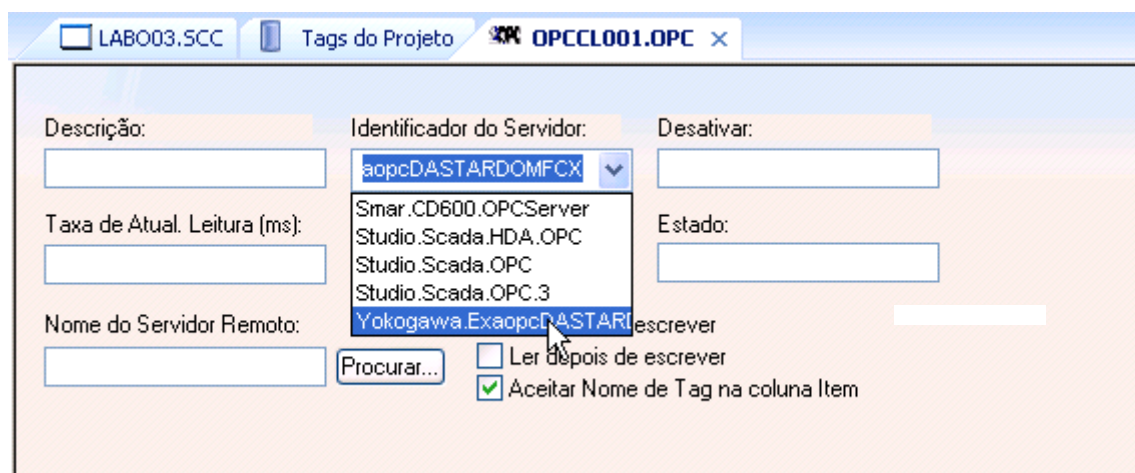


Figura 269.

Preencher planilha, conforme figura a seguir:

	Nome da Tag	Item	Scan	Div	Adicionar
	Filtro de Texto	Filtro de Texto	(Todos)	Filtro de Te	Filtro de Te
1			Sempre		
2	BOTAO[1]	FCX01!@GV.BOTAO1_BOOL	Sempre		
3	BOTAO[2]	FCX01!@GV.BOTAO2_BOOL	Sempre		
4	LAMPADA[1]	FCX01!@GV.LAMP1_BOOL	Sempre		
5	LAMPADA[2]	FCX01!@GV.LAMP2_BOOL	Sempre		
6	CONTAGEM	FCX01!_COMANDOS.CONT_PULSO	Sempre		
7	CONTA_TEMPO	FCX01!_COMANDOS.CONTA_TEMPO_DINT	Sempre	1000.000000	
8	ADJ_TEMPO	FCX01!_COMANDOS.ADJ_TEMPO	Sempre		

Figura 270.

A partir deste momento, quando o aplicativo for iniciado, a comunicação via OPC do Indusoft automaticamente será executada.

Salve a aplicação e rode o sistema. Verificar se as válvulas e o status dos botões mudam de cor, quando são acionados os botões reais do Kit didático de CLP.

ETAPA 5. ALTERAR SISTEMA

Alterar o sistema supervisorio e o programa do CLP, de tal forma que:

- O número de pulsos a serem contados seja alterado em uma tela do supervisorio.
- Considerando que cada vez que se completa uma contagem, signifique um ciclo completo, criar na tela do supervisorio, um bargraph com indicação do número de ciclos realizados pelo sistema.

5ª. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA YOKOGAWA STARDOM + SUPERVISÓRIO COM ANALÓGICAS

O objetivo deste trabalho prático visa conhecer as ferramentas para programação do CLP Stardom utilizando instruções e/ou blocos de leitura de sinal analógico com software LOGIC DESIGNER e construção de aplicação para supervisão, ajustes e monitoramento de valores analógicos, utilizando o software SUPERVISÓRIO INDUSOFT.

ETAPA 1. CONFIGURAÇÃO DE HARDWARE



- Duplo clique no ícone abaixo:

O “Device Label” é um rótulo dado às entradas e saídas do sistema. Assim, torna-se mais familiar a manipulação de variáveis de campo, sabendo-se o seu nome de projeto.

- Com o resource configurator em modo “tree” clique em “IOM”.
- Duplo-clique em DI/DO.
- Ajuste o device label correspondente a 1ª. entrada digital para “BOTAO1”.
- Ajuste o device label correspondente a 2ª. entrada digital para “BOTAO2”.
- Ajuste o device label correspondente a 1ª. saída digital para “LAMP1”.
- Ajuste o device label correspondente a 2ª. saída digital para “LAMP2”.
- Ajuste o device label correspondente a 1ª. entrada analógica para “INANAG1”.
- Ajuste o device label correspondente a 1ª. saída analógica para “OUTANAG1”.
- Selecione File-Download
- Quando aparecer uma janela de prompt, clique no OK.

ETAPA 2. CONEXÃO DE TRANSMISSOR DE TEMPERATURA AO CLP

Neste momento será iniciado o procedimento para ligação do transmissor de temperatura ao CLP da YOKOGAWA.

- Faça a ligação como mostrada abaixo na Figura 272; os terminais 1A e 1B tratam-se de borneiras.

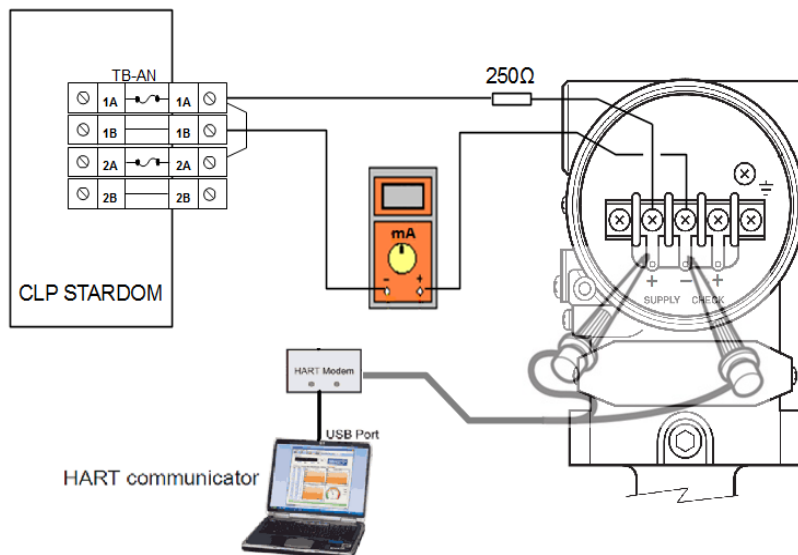
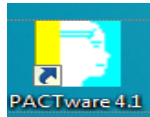


Figura 271.



- Abra o software “PACTware 4.1”.
- Clicar “PC HOST” e selecionar a opção “Adicionar instrumento”, conforme figura a seguir:

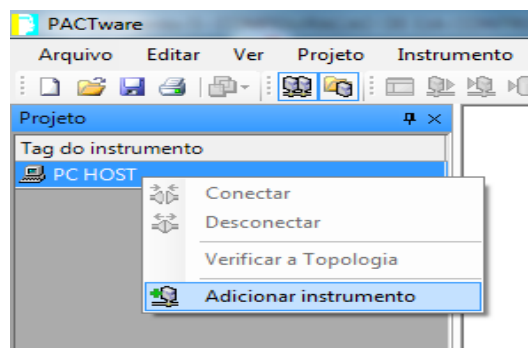


Figura 272.

- Selecionar a opção *HART Communication* e em seguida clicar “OK”, conforme figura a seguir.

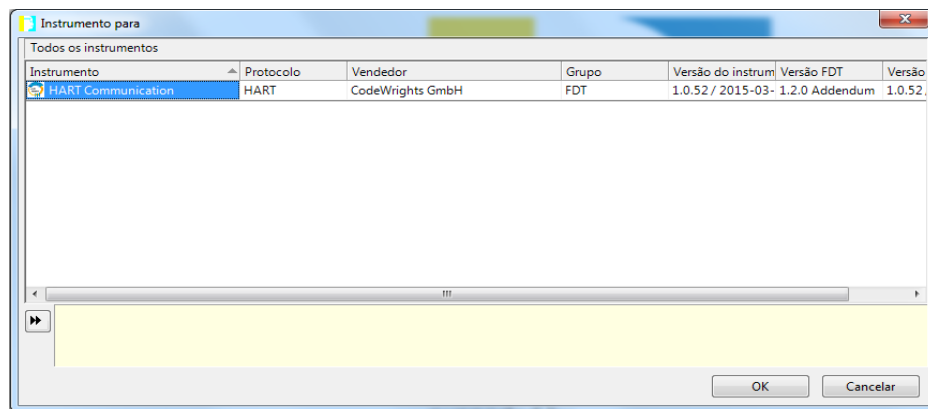


Figura 273.

- Em seguida iremos configurar a porta USB correspondente a qual ela foi ligada no computador. Clique com botão direito em COM1 e vá em “parâmetro”.

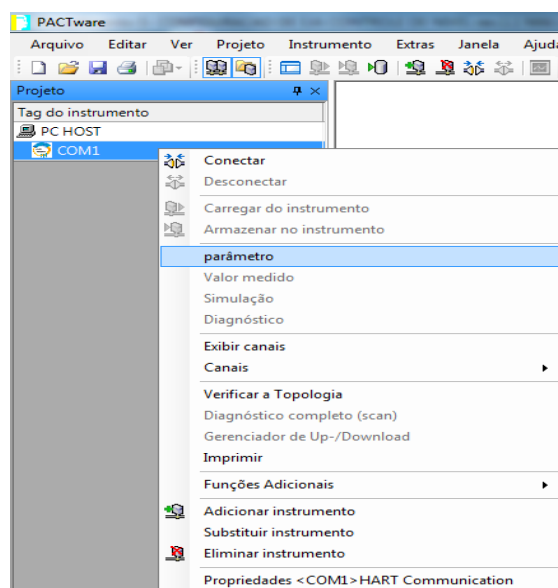


Figura 274.

- Verifique a “Serial interface”, selecione a nova porta COM e desmarque a caixa localizada na parte inferior desta tela. Pressione OK.

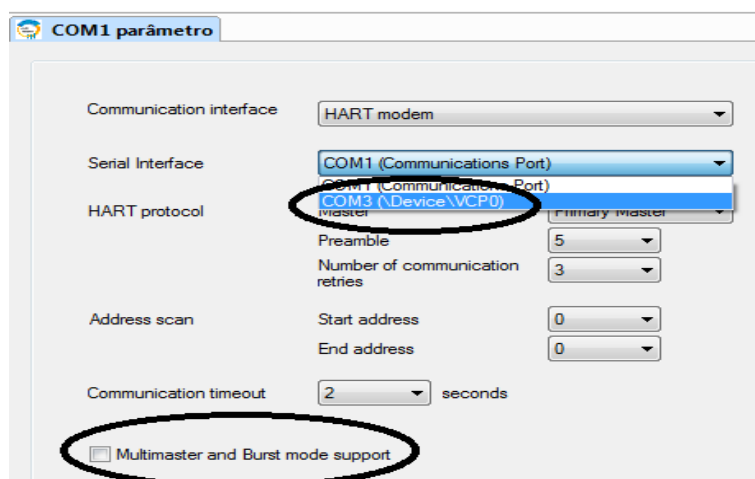


Figura 275.

- Após selecionar a porta correta, clique com o botão direito sobre ela e clique em “Conectar”.

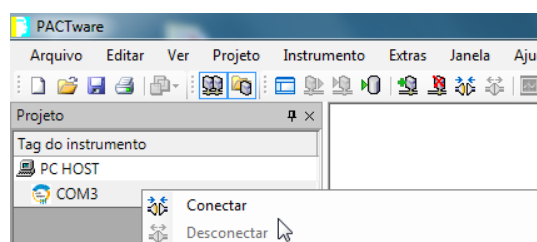


Figura 276.

- Depois de se conectar como mostrado na figura anterior, clique em “Adicionar instrumento”.

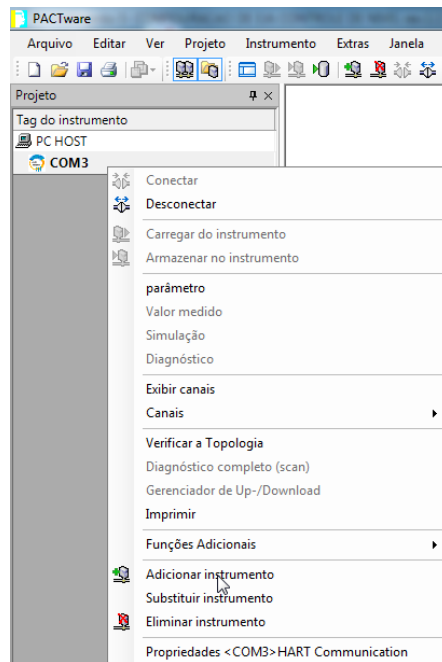


Figura 277.

- Na janela de “Instrumentos para” procure YTA 2.1 para a configuração do transmissor de temperatura.

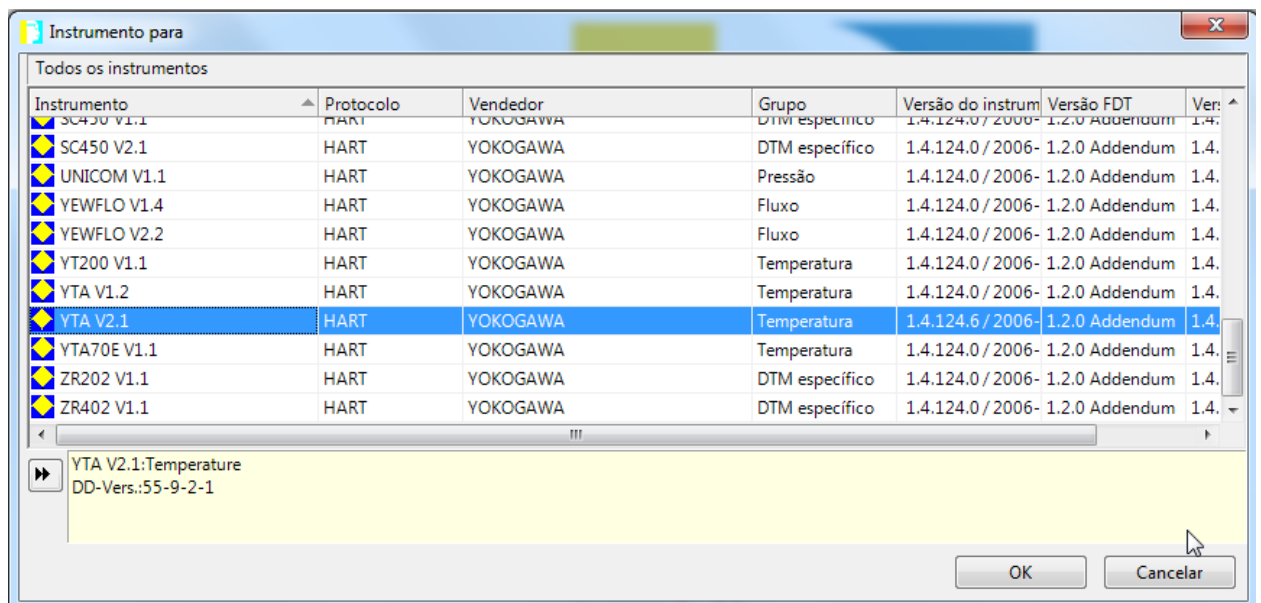


Figura 278.

- Após escolher o instrumento, clique com botão direito sobre ele e clique em “Conectar” e depois em “Parametrização Online” como vemos nas duas imagens a seguir.

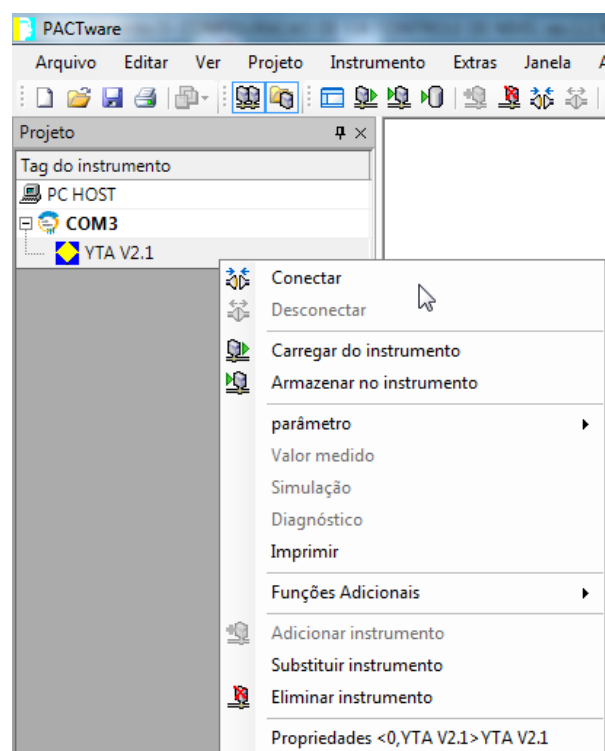


Figura 279.

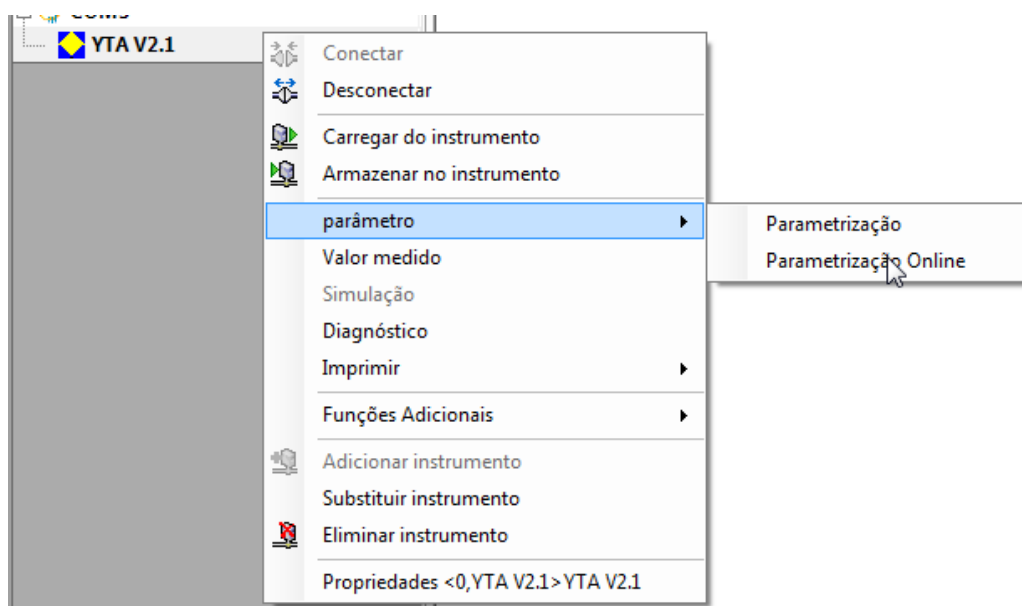


Figura 280.

- Já nas propriedades de configuração do transmissor, clique em “Easy Setup”.

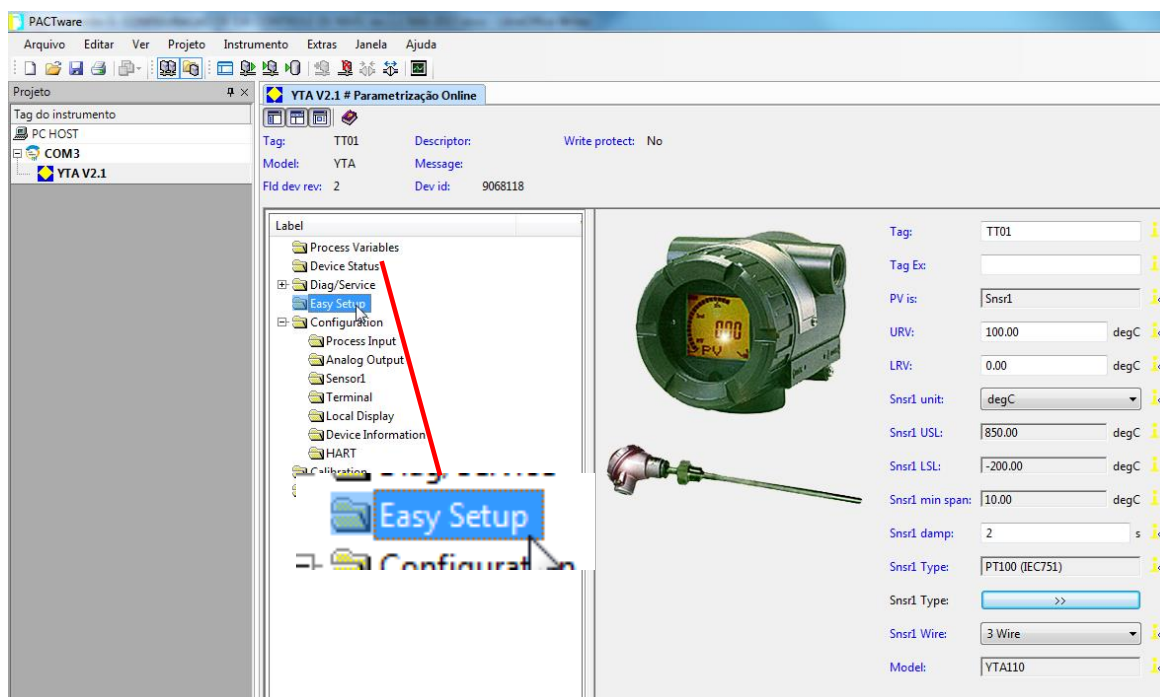


Figura 281.

Depois disso basta configurar o transmissor com as especificações desejadas. Neste caso, verificaremos o URV, LRV e o tipo de sensor de temperatura que estaremos usando. Podemos ver a localização destes ajustes na figura a seguir.

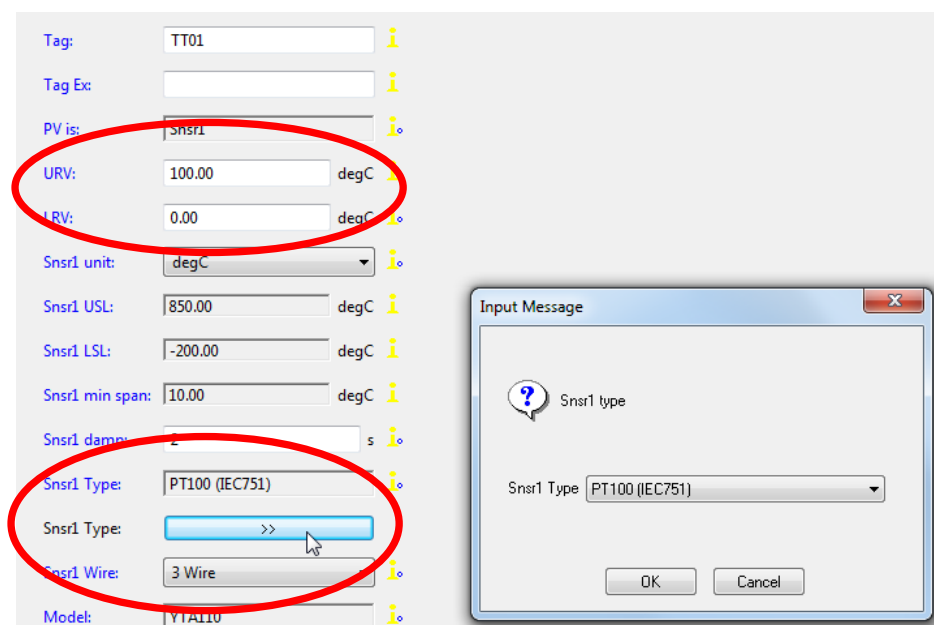


Figura 282.

ETAPA 3. CRIAÇÃO DA APLICAÇÃO DE CONTROLE (PROGRAMA DO CLP)

Neste momento será iniciado o desenvolvimento da criação de uma aplicação de controle.

- Para iniciar o Logic Designer, clique no ícone na área de trabalho como exibido na figura



ao lado:

- Selecione “File-New Project”.
- Selecione o ícone “STARDOM NPAS” e clique “OK”.

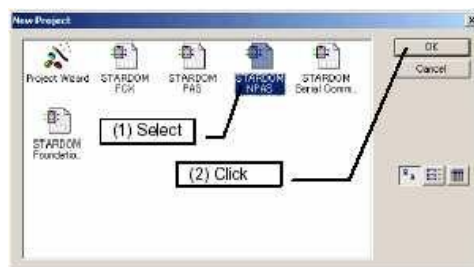


Figura 283.

- Duplo-clique em “TargetSetting” na árvore do projeto.
- Entre com o número de IP “192.168.0.XX” e clique em “OK”. Obs.: XX=Número variável de acordo com a bancada.

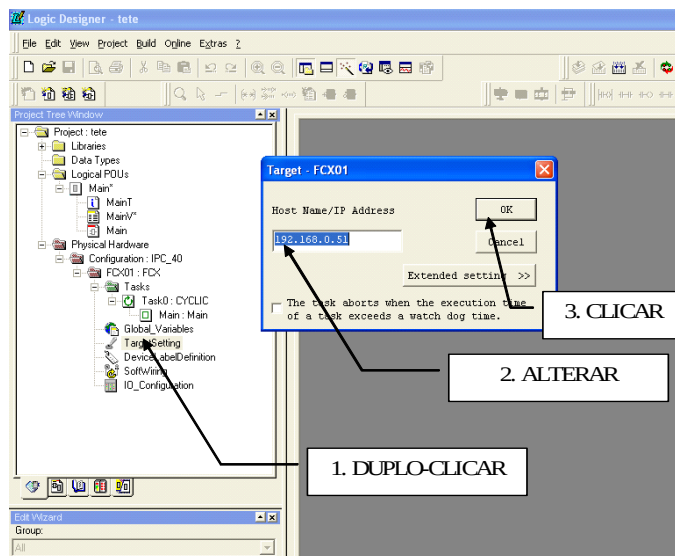
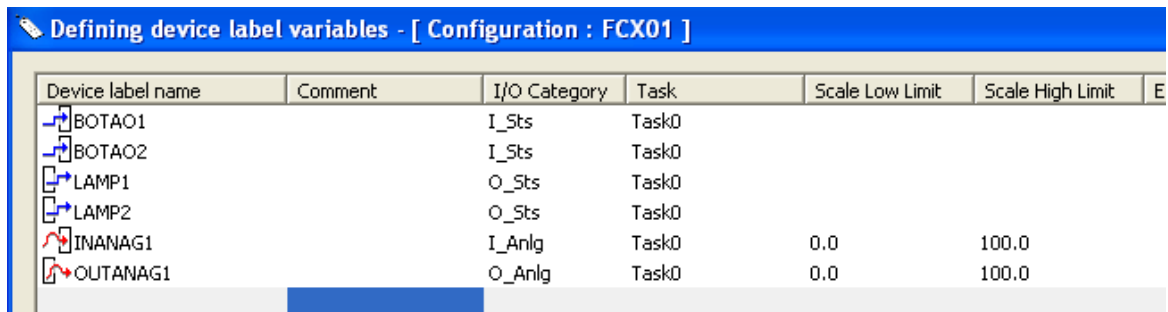


Figura 284.

- Nesta aplicação iremos utilizar duas “variáveis globais” que farão as vezes dos sinais de entrada e saída (digitais) deste exercício.

- Duplo-clique sobre “DeviceLabelDefinition” na árvore do projeto.
- Dentro da janela defina as variáveis como descritas abaixo. Após finalizar os ajustes das variáveis, clique em “Ok” e clique em “YES” na janela de diálogo de confirmação.



Device label name	Comment	I/O Category	Task	Scale Low Limit	Scale High Limit	E
BOTAO1		I_Sts	Task0			
BOTAO2		I_Sts	Task0			
LAMP1		O_Sts	Task0			
LAMP2		O_Sts	Task0			
INANAG1		I_Anlg	Task0	0.0	100.0	
OUTANAG1		O_Anlg	Task0	0.0	100.0	

Figura 285.

Iremos agora construir uma aplicação de controle usando a unidade POU (Program Organization Unit).

- Clique com o botão direito sobre “Logical POU” na árvore do projeto e então selecione “Insert – Program” para aparecer a janela exibida abaixo.
- Neste exercício, insira no campo “Name” a informação “ANALOG” e selecione “LD” (LADDER) no campo “Language” e clique em “Ok”.

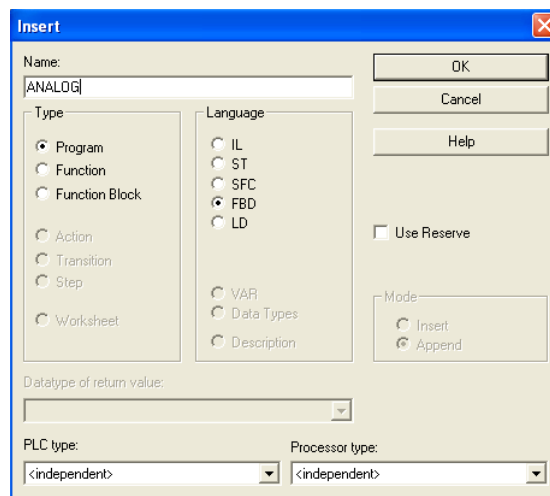


Figura 286.

- Duplo-clique sobre “ANALOG” (ícone de código) criado no projeto e clique numa posição específica dentro da área de trabalho a fim de determinar onde ficará a linha de programa. Em seguida, clicar sobre o ícone “Contact Network” para iniciar uma linha básica de programa.

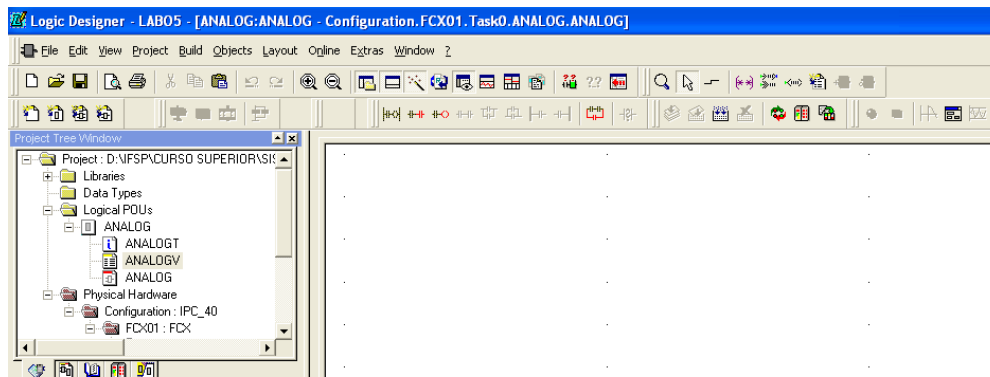


Figura 287.

- Clique numa posição específica dentro da área de trabalho, logo após a primeira linha de programa recém criada. Em seguida, dentro da janela “Edit Wizard”, clicar sobre a opção “all FUs and FBs” para visualizar todos os blocos de instruções. Caso a janela “Edit Wizard” não esteja visível, clicar no menu “View” e selecionar a opção “Edit Wizard”.

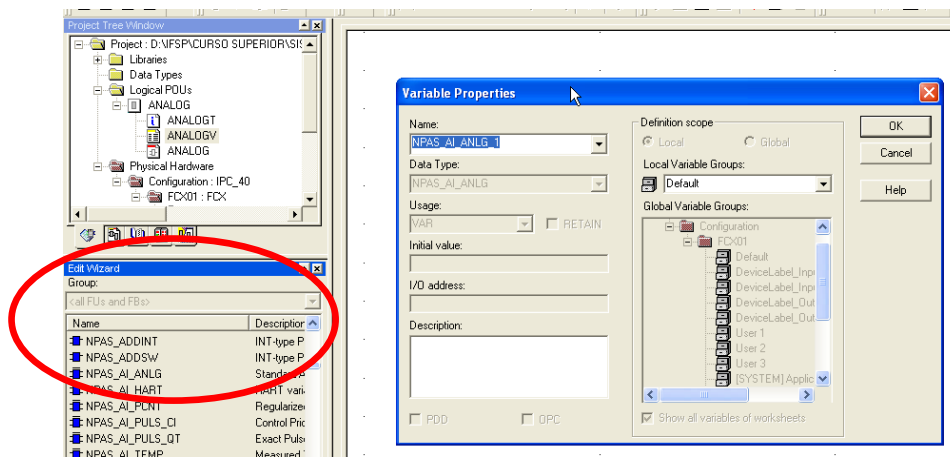


Figura 288.

- Selecionar a opção “NPAS_AI_ANLG”, conforme mostrado na figura a seguir. Em seguida, confirmar o nome “NPAS_AI_ANLG_1” como nome desta função e clicar em “OK”. Aparecerá o bloco na área de trabalho.

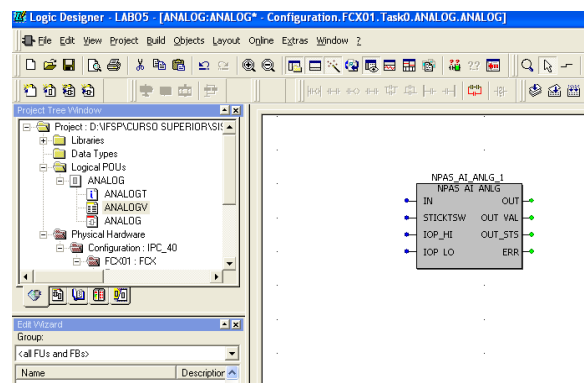


Figura 289.

- Clicar na entrada “IN” do novo bloco e selecionar na janela “Variable Properties” a variável “INANAG1”, conforme mostrado na figura a seguir. Esta variável é do escopo “Global”

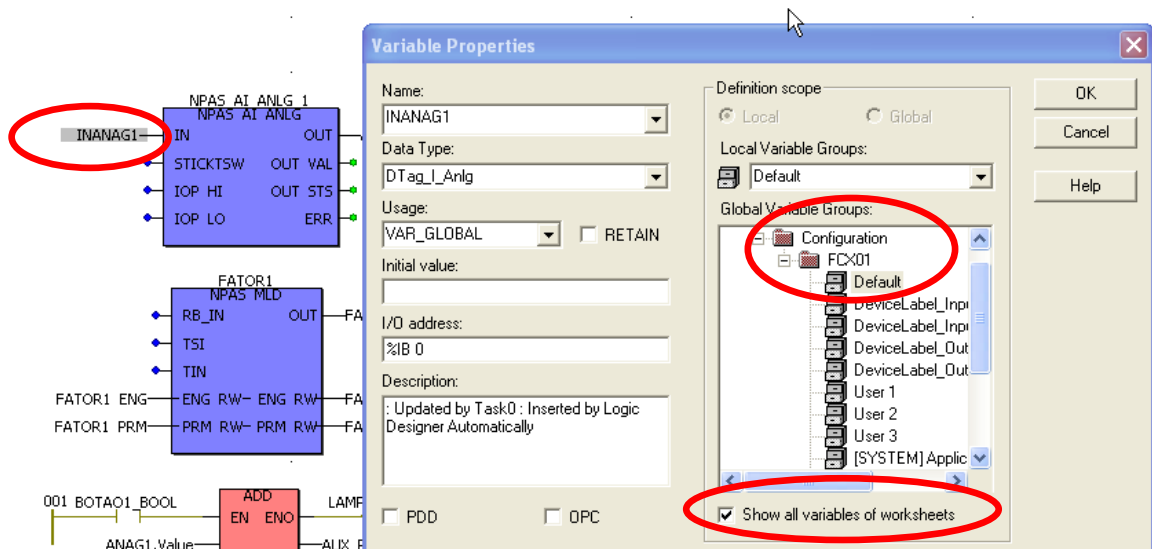


Figura 290.

- Clicar na saída “OUT” do novo bloco e selecionar na janela “Variable Properties” o escopo “Local”. No campo “Name” digitar “ANAG1”, pois está será uma nova variável.

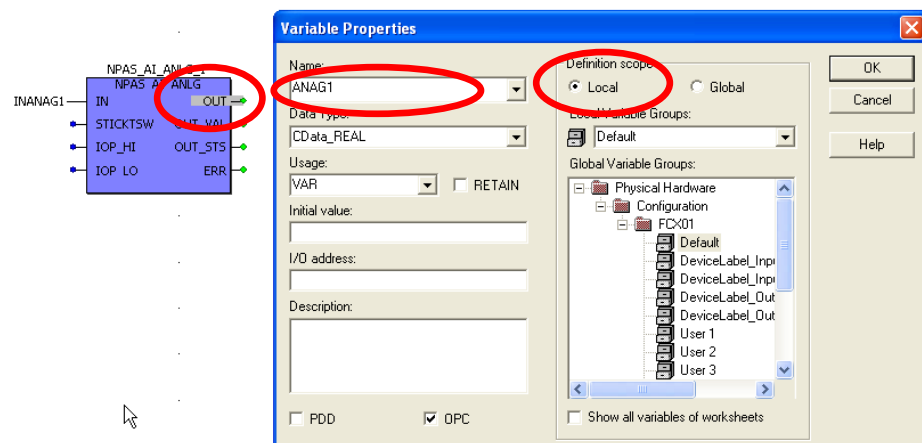


Figura 291.

- Desenvolver o programa conforme mostrado na figura a seguir. Observar que todas as variáveis dentro dos círculos em vermelho serão variáveis do escopo Local.

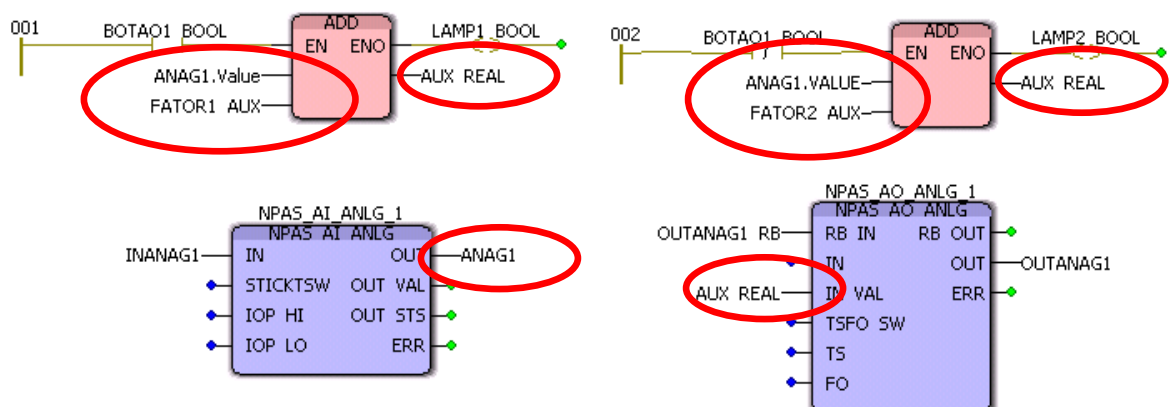


Figura 292.

- A tabela abaixo mostra as propriedades de cada variável a ser utilizada neste programa. Não esqueça de ativar a opção “OPC” das variáveis: ANAG1, BOTAO1_BOOL, AUXREAL, FATOR1 e FATOR2 que permitirá a comunicação com o supervisor.

AJUSTANDO E ASSOCIANDO UMA “TASK”

- Selecione “Task0” exibida na árvore do projeto. Clique com o botão direito do mouse e selecione “Properties”. Task0 é do tipo “cyclic”. Após confirmação do tipo da “task”, clique em “Cancel” para fechar esta janela.
- Selecione “Task0” com o botão direito do mouse e selecione “Settings”. Irá surgir uma janela “Task setting for IPC_40”. Nesta tarefa, valores como “interval” e “watchdogtime” serão mantidos.. Após confirmação, clique em “Cancel” para fechar esta janela.
- Selecione “Task0” e clique com o botão direito do mouse. Selecione “Insert - Program Instance”, através do menu.
- Entre no campo “Program Instance” e insira “_ANALOG” e selecione “ANALOG” no campo “Program Type”. Clique em “Ok”.

FAZENDO O “BUILDING” E “DOWNLOAD” DO PROJETO

- Neste ponto, estaremos convertendo para código de máquina toda a aplicação desenvolvida até o momento.
- Selecione “Build – Rebuild Project” através da barra de menu. Uma mensagem aparecerá informando que a compilação está em progresso.
- Selecione “Online – Project Control” através da barra de menu. O diálogo do “Project Control” aparecerá e em seguida, clique em “Download”. O diálogo deste botão será exibido.
- Clique no botão “Download” abaixo do campo “Project”.

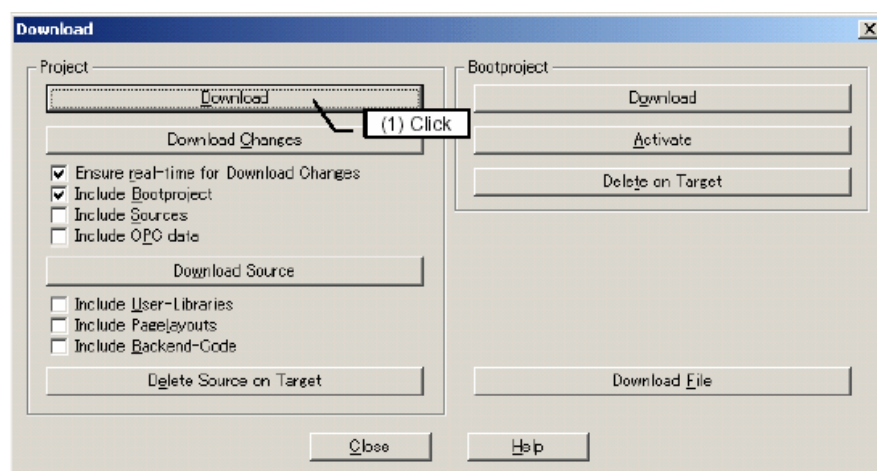


Figura 293.

- Se algum outro projeto foi carregado anteriormente, aparecerá uma janela informando que este projeto será sobrescrito. Clique em “YES”.

Na parte inferior do Logic Designer será exibida uma barra de progressão referente ao “download”. Aguarde até que haja a indicação operação realizada.

INICIANDO UM PROJETO

Neste passo, a aplicação ainda não foi executada. Para alterar o estado do CLP, selecione um dos modos proposto na tabela abaixo:

- Clicar no ícone “Project Control Dialog”, conforme figura abaixo.

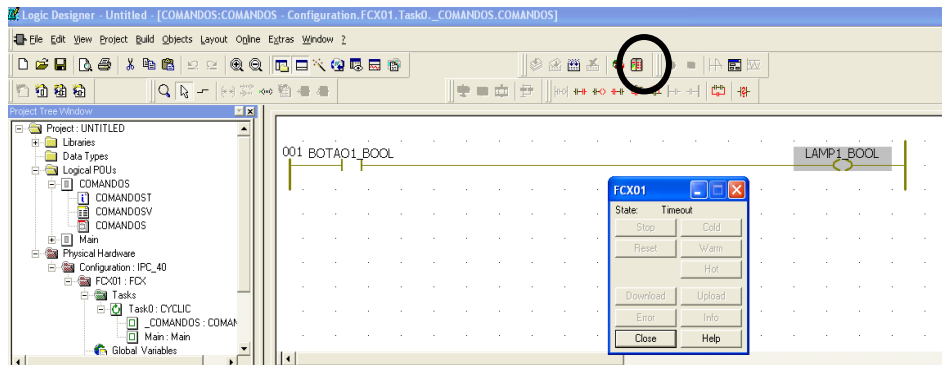


Figura 294.

Neste exercício, iremos optar pela inicialização a frio (“Cold”).

SALVANDO UM PROJETO

- Selecione “File – Save Project As/Zip Project As” no menu do “Logic Designer”.
- Entre com um nome para o seu projeto (EXERCICIO3, por exemplo) e clique em “Save”.

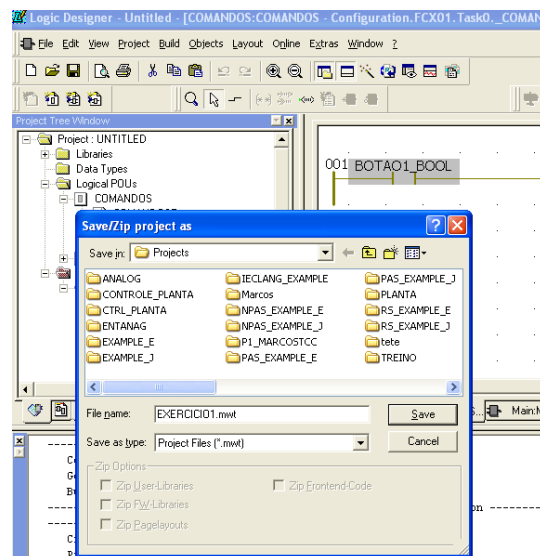


Figura 295.

ETAPA 4. CRIAR TELA NO PROJETO

A figura a seguir mostra a tela a ser criada pelo supervisor.

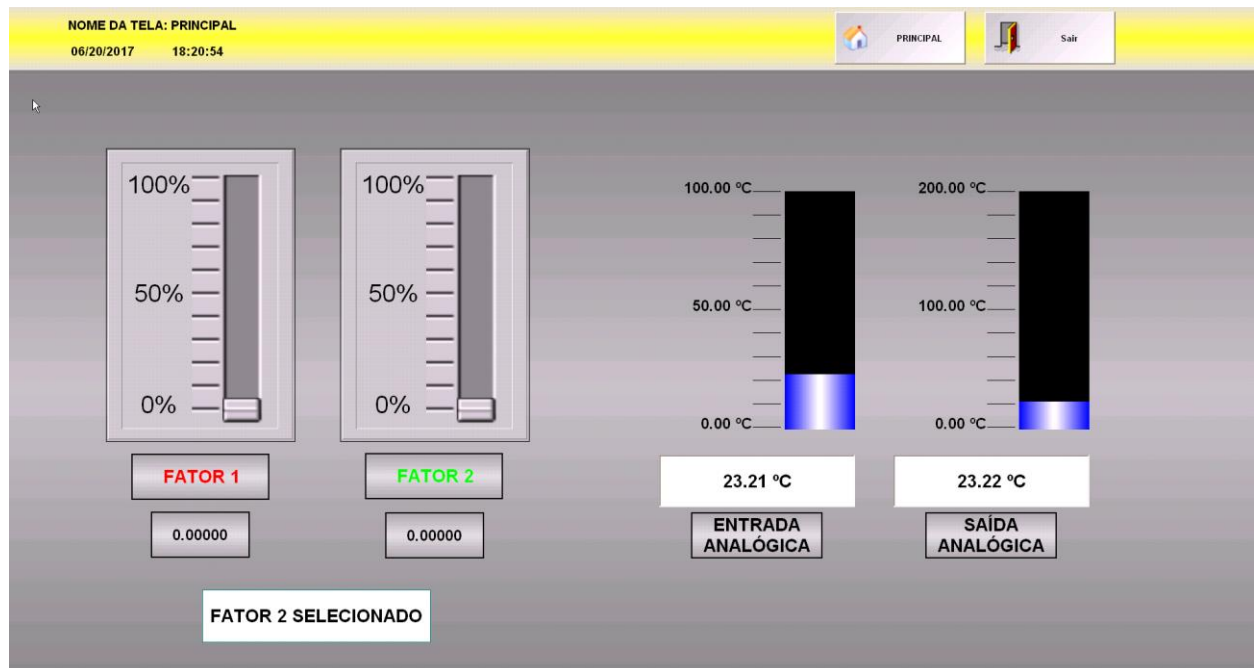


Figura 296.

Vamos configurar as "**Tags do Projeto**" como mostrado a seguir:

- Crie os tags, conforme mostrado a seguir.

	Nome	Vetor	Tipo	Descrição	Escopo
	Filtro de Texto	=...	(Todos)	Filtro de Texto	(T...
1	FATOR1	0	Real		Servidor
2	FATOR2	0	Real		Servidor
3	INANAG	0	Real		Servidor
4	OUTANAG	0	Real		Servidor
5	FAT1	0	Booleana		Servidor

Figura 297.

- Salve a configuração dos tags
- Crie uma tela chamada "LABO5"
- Inserir nesta tela, dois objetos: **Slider02** (proveniente da biblioteca) e um "**Texto com Link de Dados**" associados ao tag "**FATOR1**", conforme mostrado na figura a seguir. Fazer as mesmas etapas para o segundo fator, o **FATOR2**.

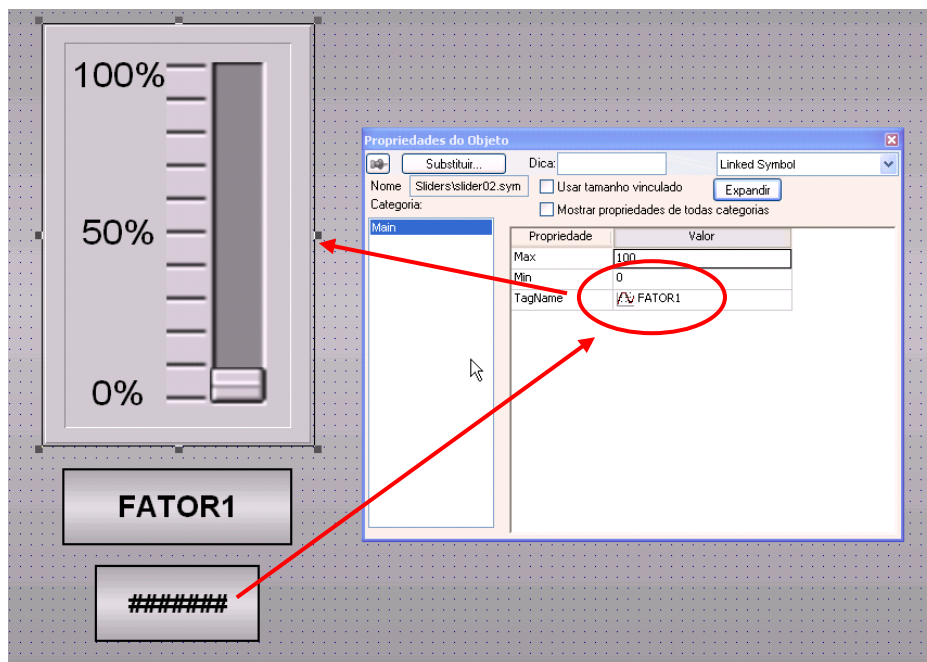


Figura 298.

- Inserir na tela “LABO5”, o objeto: **Bargraph05** (proveniente da biblioteca) associado ao tag “**INANAG**”, conforme mostrado na figura a seguir. Adicionar também uma caixa de texto identificando qual *bargraph* é este, no caso, “**Entrada Analógica**”.
- Fazer as mesmas etapas para o segundo *bargraph*, a “**Saída Analógica**”.

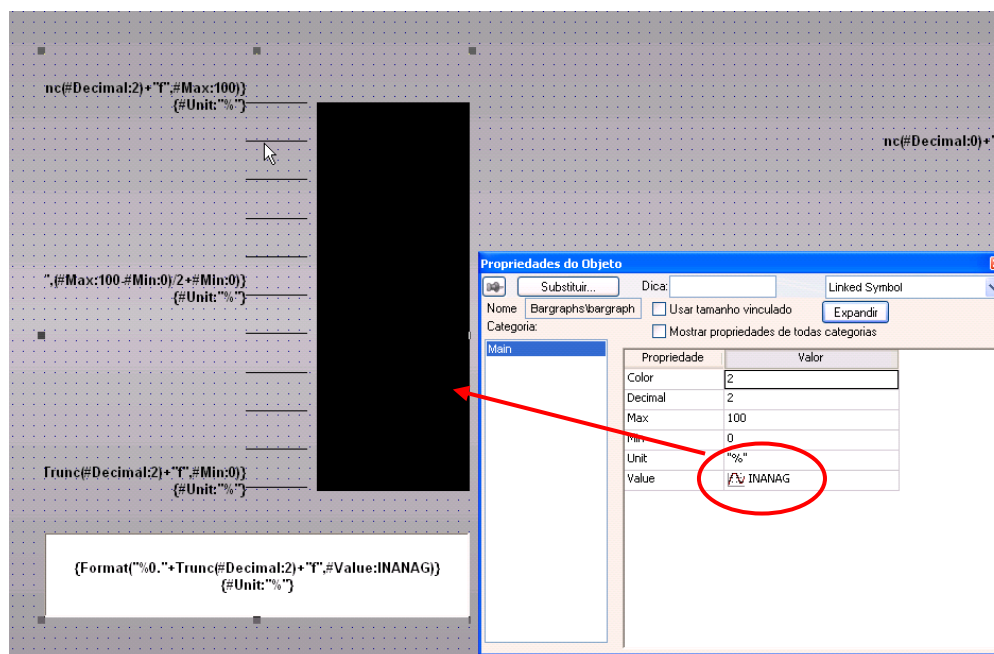


Figura 299.

- Salve a tela como **LABO5**.

ETAPA 5. CONFIGURAR COMUNICAÇÃO DO INDUSOFT COM O STARDOM VIA OPC

- Selecionar a aba “Comm” no rodapé da árvore de projeto e uma nova planilha clicando com o botão direito em “OPC DA 2.05”

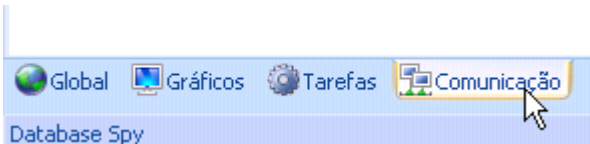


Figura 300.

- Selecionar a opção “Yokogawa.ExaopcDASTARDOMFCX” no campo “Identificador do Servidor” conforme mostrado a seguir:

A screenshot of the OPC configuration dialog box. The 'Identificador do Servidor' dropdown menu is open, showing a list of options: 'Smar.CD600.OPCServer', 'Studio.Scada.HDA.OPC', 'Studio.Scada.OPC', 'Studio.Scada.OPC.3', and 'Yokogawa.ExaopcDASTARDOMFCX'. The 'Yokogawa.ExaopcDASTARDOMFCX' option is selected. Other fields include 'Descrição:', 'Taxa de Atual. Leitura (ms):', 'Nome do Servidor Remoto:', 'Desativar:', 'Estado:', and checkboxes for 'Ler depois de escrever' and 'Aceitar Nome de Tag na coluna Item'. A 'Procurar...' button is also present.

Figura 301.

- Preencher planilha, conforme figura a seguir:

	Nome da Tag	Item	Scan	Div	Adicionar
			(Todos)		
*			Sempre		
*			Sempre		
*			Sempre		
*			Sempre		
*			Sempre		

Figura 302.

- Salve a aplicação e rode o sistema. Verificar se as válvulas e o status dos botões mudam de cor, quando são acionados os botões reais do Kit didático de CLP.

6ª. LABORATÓRIO PARA CONFIGURAÇÃO DE SISTEMA DE CONTROLE REGULATÓRIO COM SOFTWARE INDUSOFT E CLP STARDOM

O objetivo deste trabalho prático visa conhecer as ferramentas para programação do CLP Stardom utilizando blocos de leitura de sinal analógico e controle PID com software LOGIC DESIGNER e construção de aplicação para supervisão, ajustes e monitoramento de valores analógicos, utilizando os softwares do sistema INDUSOFT.

ETAPA 1. CONFIGURAÇÃO DE HARDWARE



- Duplo clique no ícone abaixo:
- O “Device Label” é um rótulo dado às entradas e saídas do sistema. Assim, torna-se mais familiar a manipulação de variáveis de campo, sabendo-se o seu nome de projeto.
- Com o resource configurator em modo “tree” clique em “IOM”.
- Duplo-clique em DI/DO.
- Ajuste o device label correspondente a 1ª. entrada digital para “BOTA01”.
- Ajuste o device label correspondente a 2ª. entrada digital para “BOTA02”.
- Ajuste o device label correspondente a 1ª. saída digital para “LAMP1”.
- Ajuste o device label correspondente a 2ª. saída digital para “LAMP2”.
- Duplo-clique em DI/DO.
- Ajuste o device label correspondente a 1ª. entrada analógica para “AI001”.
- Ajuste o device label correspondente a 1ª. saída analógica para “AO001”.
- Selecione File-Download
- Quando aparecer uma janela de prompt, clique no OK.

ETAPA 2. CRIAÇÃO DA APLICAÇÃO DE CONTROLE (PROGRAMA DO CLP)

Neste momento será iniciado o desenvolvimento da criação de uma aplicação de controle.

- Para iniciar o Logic Designer, clique no ícone na área de trabalho como exibido na figura



ao lado:

- Selecione “File-New Project”.
- Selecione o ícone “STARDOM NPAS” e clique “OK”.

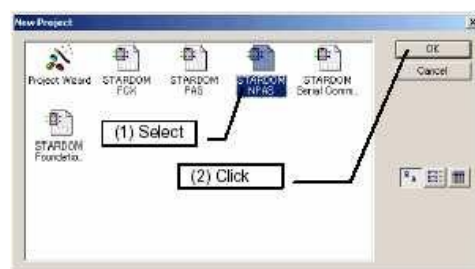


Figura 303.

- Duplo-clique em “TargetSetting” na árvore do projeto.
- Entre com o número de IP “192.168.0.XX” e clique em “OK”. Obs.: XX=Número variável de acordo com a bancada.

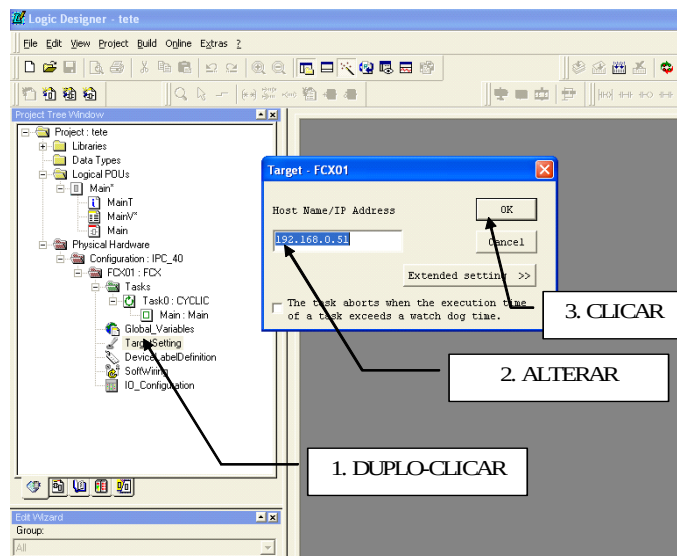


Figura 304.

- Nesta aplicação iremos utilizar duas “variáveis globais” que farão as vezes dos sinais de entrada e saída (digitais) deste exercício.
- Duplo-clique sobre “DeviceLabelDefinition” na árvore do projeto.

- Dentro da janela defina as variáveis como descritas abaixo. Após finalizar os ajustes das variáveis, clique em “Ok” e clique em “YES” na janela de diálogo de confirmação.

[Analog input] Device label name: AI001 I/O Category: I_Anlg Task: Task0 Scale Low Limit: 0.0 Scale High Limit: 100.0 Engineering Unit: I/h	[Analog output] Device label name: AO001 I/O Category: O_Anlg Task: Task0 Scale Low Limit: 0.0 Scale High Limit: 100.0 Engineering Unit: %
--	---

Figura 305.

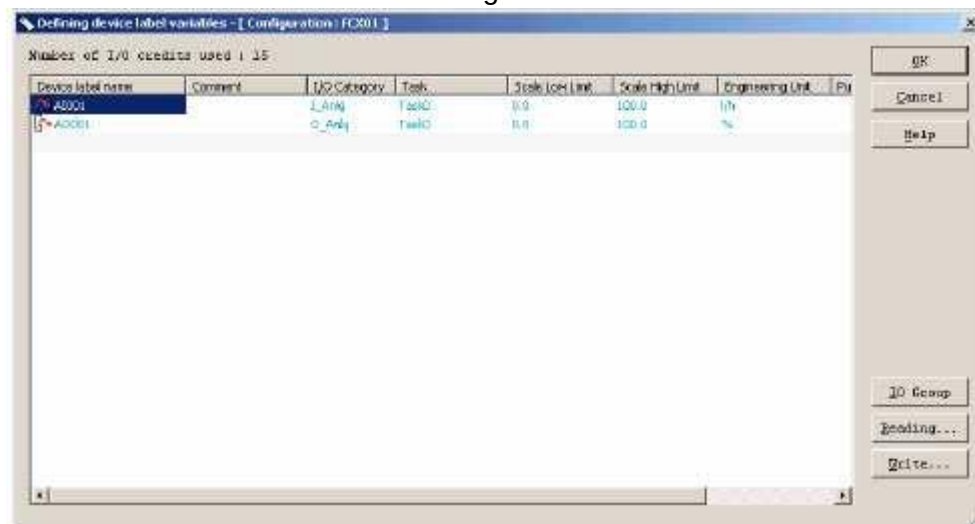


Figura 306.

Iremos agora construir uma aplicação de controle usando a unidade POU (Program Organization Unit).

- Clique com o botão direito sobre “Logical POUs” na árvore do projeto e então selecione “Insert – Program” para aparecer a janela exibida abaixo.
- Neste exercício, insira no campo “Name” a informação “SINGLE_LOOP” e selecione “LD” (LADDER) no campo “Language” e clique em “Ok”.

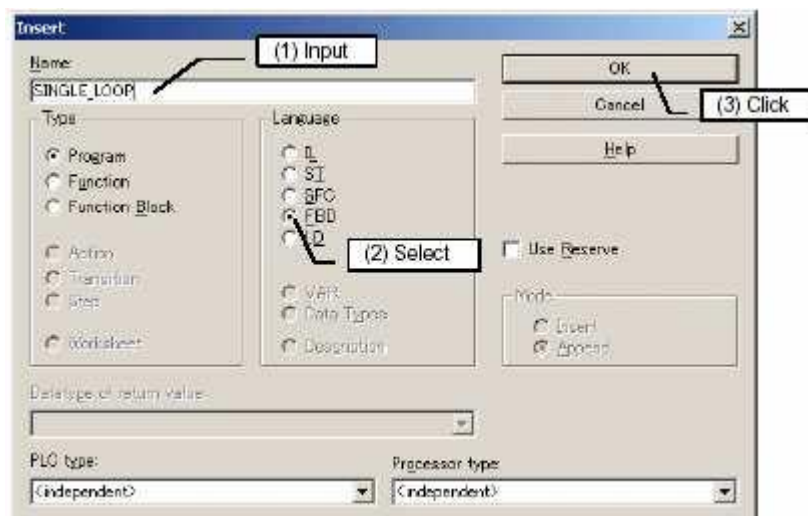


Figura 307.

- Duplo-clique sobre “SINGLE_LOOP” (ícone de código) criado no projeto e clique numa posição específica dentro da área de trabalho a fim de determinar onde ficará a linha de programa.

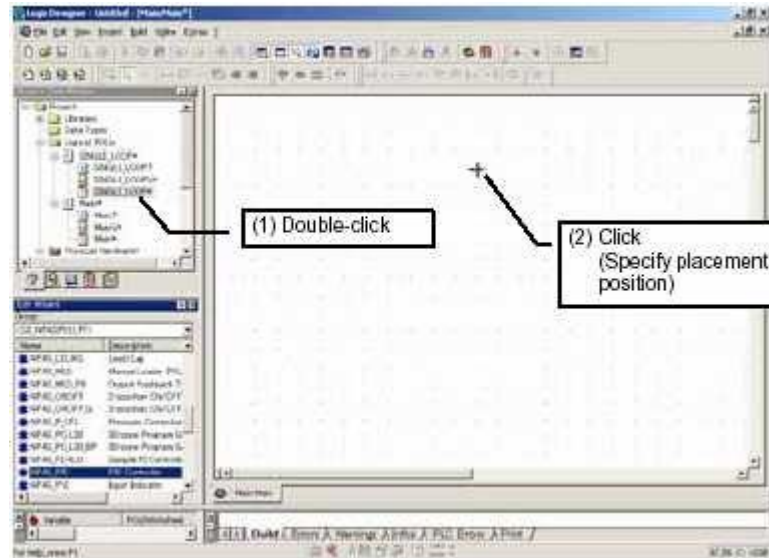


Figura 308.

- Duplo-clique sobre “SINGLE_LOOP” (code worksheet) criado no projeto.
- Clique numa posição específica dentro da área de trabalho a fim de determinar onde ficará alocado o bloco.
- Selecione <SD_NPASPOU_PF> através do campo “Group”. Para isto, habilite o “Edit WizardDesenvolver o programa conforme mostrado na figura a seguir. A tabela abaixo mostra as propriedades de cada variável a ser utilizada neste programa.
- Caso ele não esteja habilitado, clique em “View – Edit Wizard” através do menu do Logic Designer.

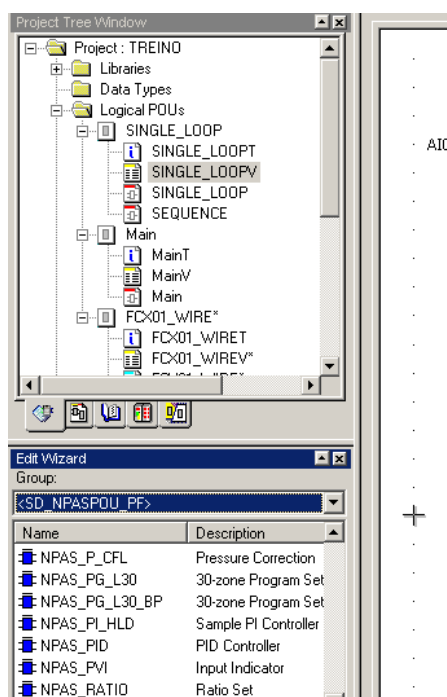


Figura 309.

- Duplo-clique em “NPAS_PID” (PID Controller) na lista exibida através do “Wizard”.
- Digite “FIC001” (sem aspas) no campo “Name” e clique em “Ok”.



Figura 310.

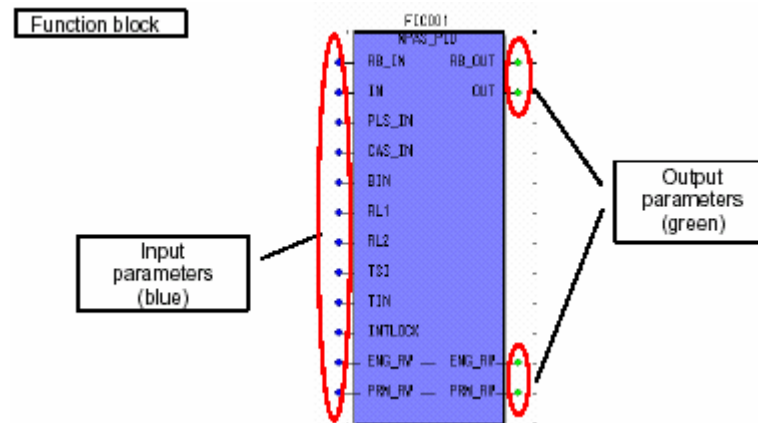


Figura 311.

- Clique em uma área à esquerda do bloco “FIC001” para definir a posição deste bloco.
- Duplo-clique em “NPAS_AI_ANLG(Standard Analog Input)” na lista.
- Entre com o nome “FIC001_AI001” e clique em “Ok”. Assim, o bloco de função será inserido na sua área de trabalho.

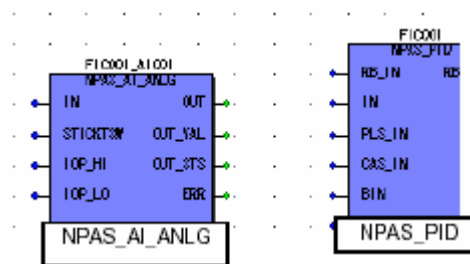


Figura 312.

- Clique em uma área à direita do bloco “FIC001” para definir a posição deste bloco.
- Duplo-clique em “NPAS_AO_ANLG(Standard Analog Output)” na lista. Entre com o nome “FIC001_AO001” e clique em “Ok”. Assim, o bloco de função será inserido na sua área de trabalho.

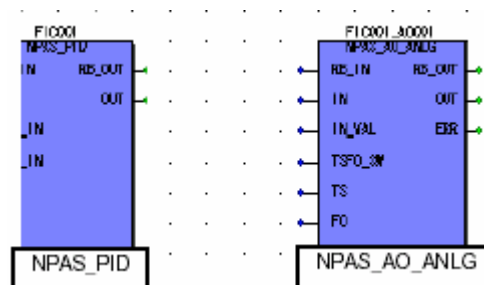


Figura 313.

- Selecione o ícone “Connect Objects” na barra de ferramentas.
- Clique no terminal em verde (OUT) do “FIC001_AI001” e em seguir clique no terminal em azul (IN) do bloco “FIC001”.

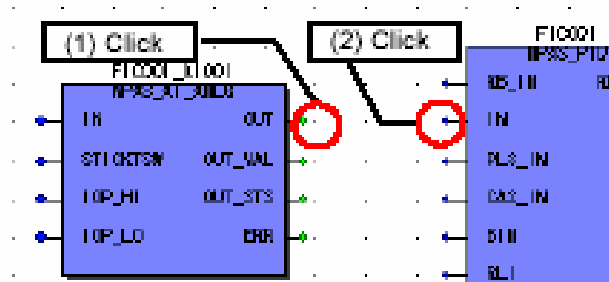


Figura 314.

Conecte o terminal (OUT) do “FIC001” com o terminal (IN) do “FIC001_AO001” da mesma forma que no passo anterior.

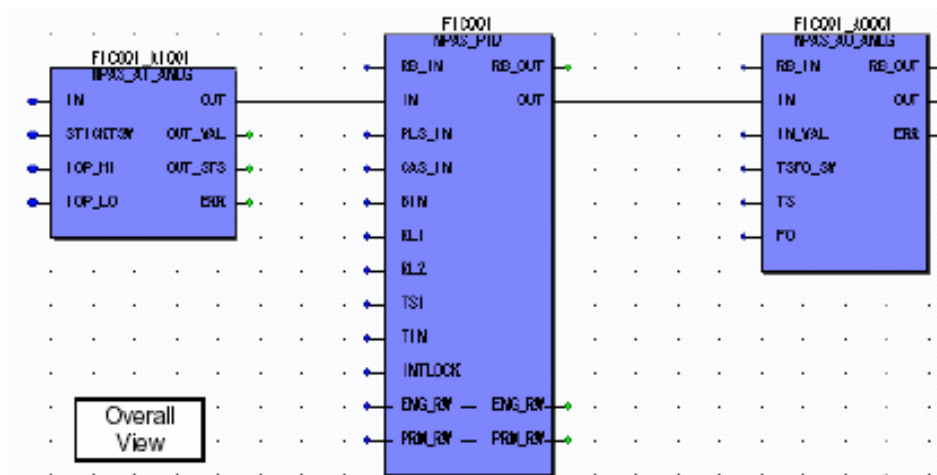


Figura 315.

- Duplo-clique no terminal (IN) do “FIC001_AI”. Aparecerá uma janela “Variable Properties”.
- Variáveis definidas no “Device Label” são do tipo globais. Para visualizar a variável “AI001”, deve-se especificar o “resource” (FCN/FCJ). No campo “Global Variable Groups:”, selecione “Physical hardware – Configuration – FCX01 – Default”. Mude a opção dentro de “Scope” para “Global” e clique na opção “Show all variable of worksheet”.
- Clique no “combo box”, localize e selecione a variável “AI001” que será exibida na lista de variáveis. Clique em “Ok”..

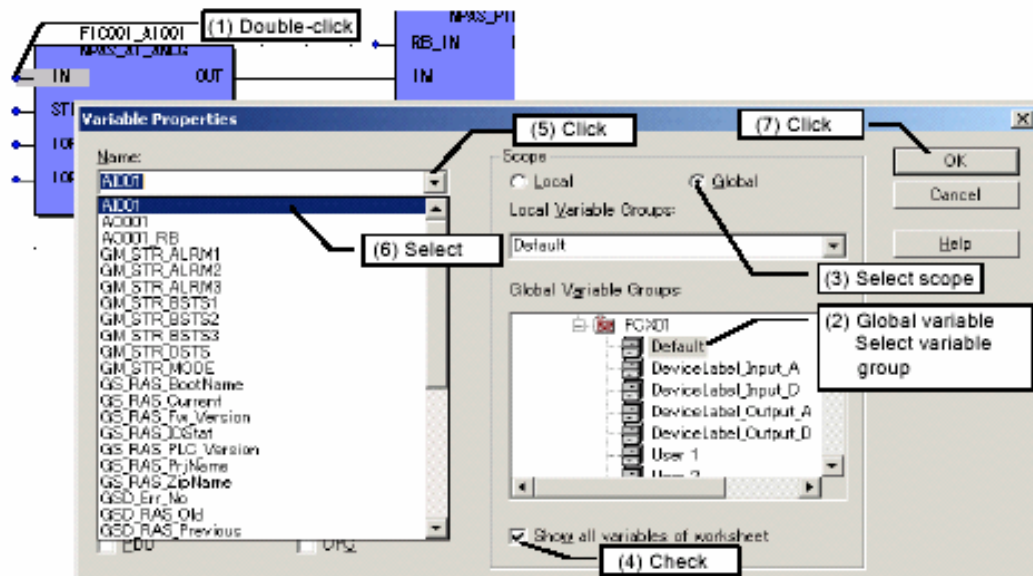


Figura 316.

- À seguir, conecte o device label “AO001” como saída (OUT) do “FIC001”. Duplo-clique no terminal (OUT) do “FIC001_AO001”. A janela “Variable Properties” irá aparecer.
- Da mesma forma que foi selecionado o “AI001”, no passo anterior, selecione o “AO001” e clique em “Ok”.

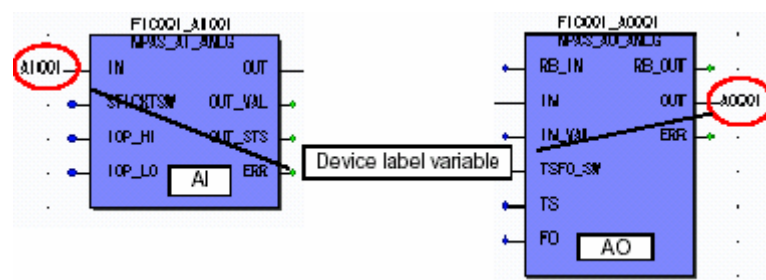


Figura 317.

- Duplo-clique no terminal azul (RB_IN) deste bloco.
- Na janela “Variable Properties”, mude o campo “Scope” para “Global”. Clique no “combo box” ao lado do campo “Name” e selecione a variável “AO001_RB” que irá surgir da lista de variáveis globais. Clique em “Ok”.

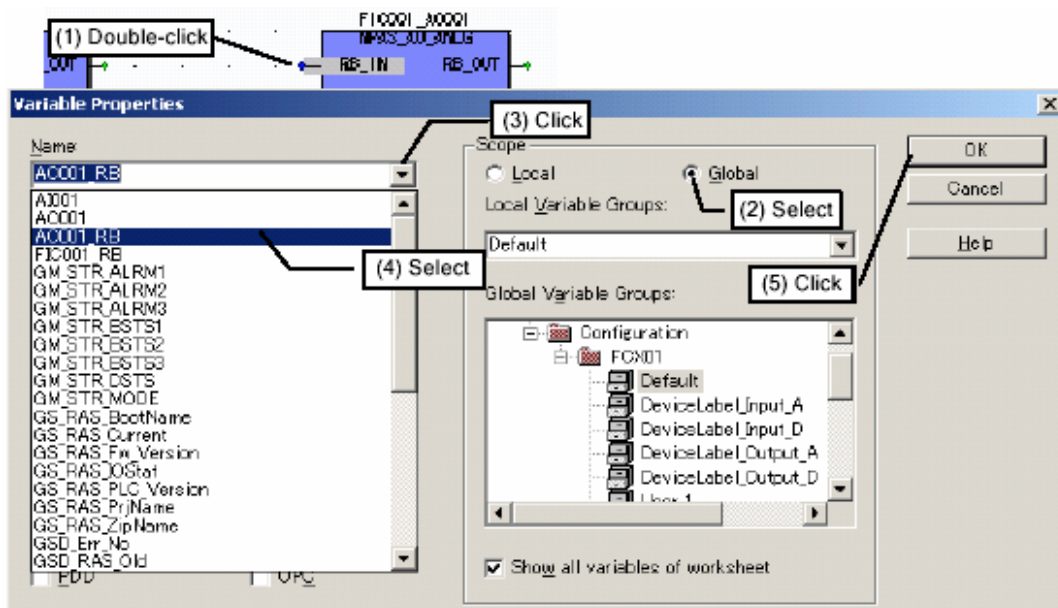


Figura 318.

- Desta forma, definimos o “read back” da variável “AO001” em relação ao bloco “FIC001_AO001”. Vamos agora definir o “read back” do bloco “FIC001_AO001” em relação ao “FIC001”.
- Duplo-clique no terminal (RB_OUT) do “FIC001_AO001”.
- Selecione no campo “Scope” a opção “Local”. Digite “FIC001_RB” (sem aspas) no campo nome, selecione o “Data Type” como “CData_Real” e clique em “Ok” para concluir a adição desta variável.”.

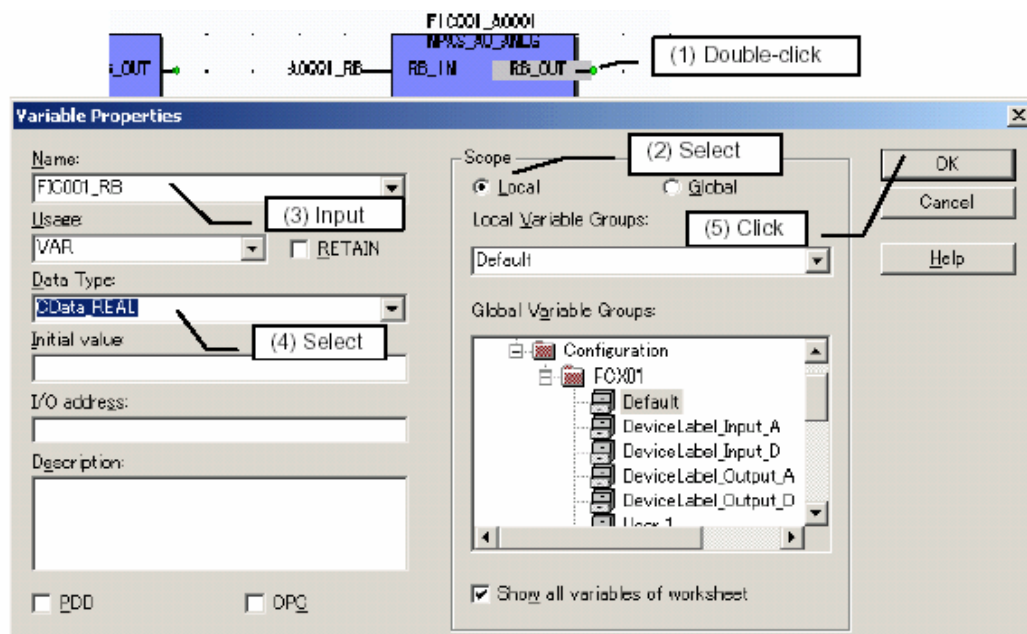


Figura 319.

- Duplo-clique em (RB_IN) do “FIC001”.
- Selecione “FIC001_RB” no campo “Name” e clique em “Ok”.

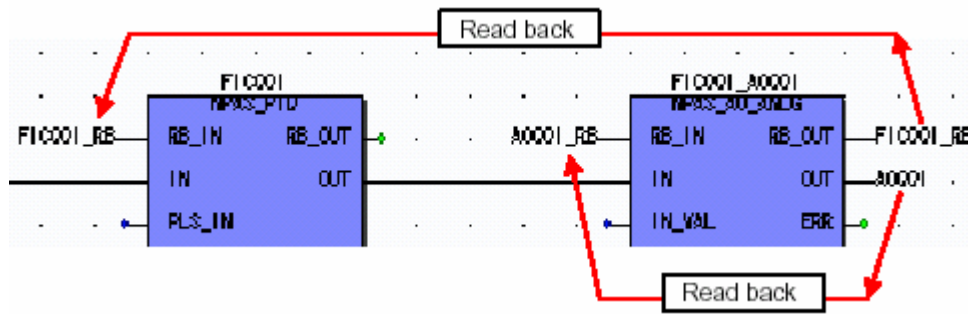


Figura 320.

- Duplo-clique no terminal (ENG_RW) do “FIC001”.

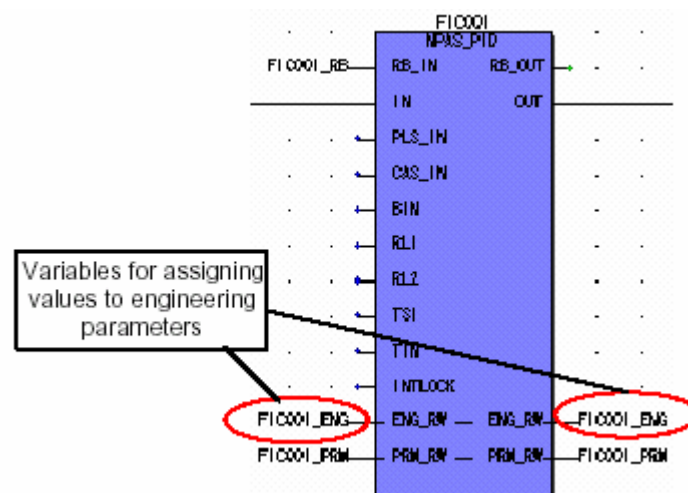


Figura 321.

- Confirme que o “Scope” está como “Local”. Digite “FIC001_ENG” no campo “Name” e clique em “Ok”. O “Data Type” automaticamente é ajustado para “SD_NPENG_PID”.

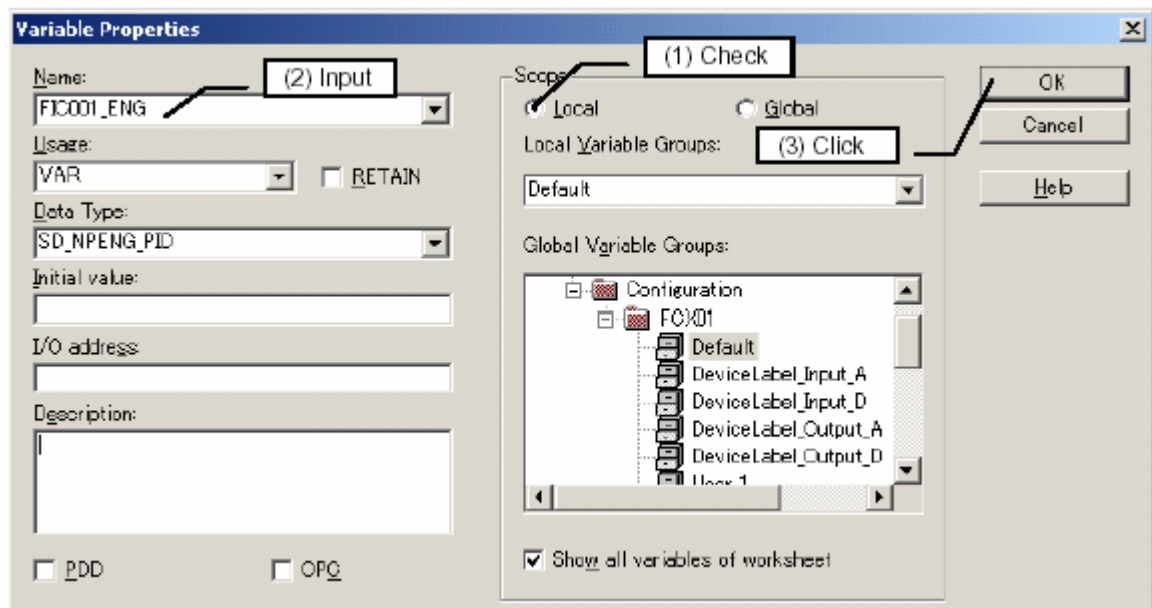


Figura 322.

- Duplo-clique no terminal (PRM_RW) do “FIC001”.

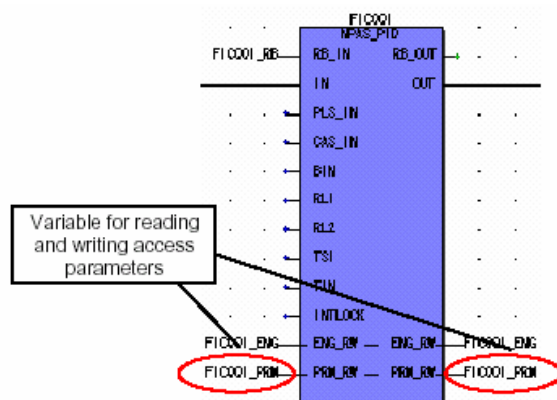


Figura 323.

- Certifique-se de que o “Scope” está em “Local”. No campo “Name” entre com o “FIC001_PRM” e clique em “Ok”. Veja que o “Data Type” é ajustado automaticamente para “SD_NPPRM_PID”.

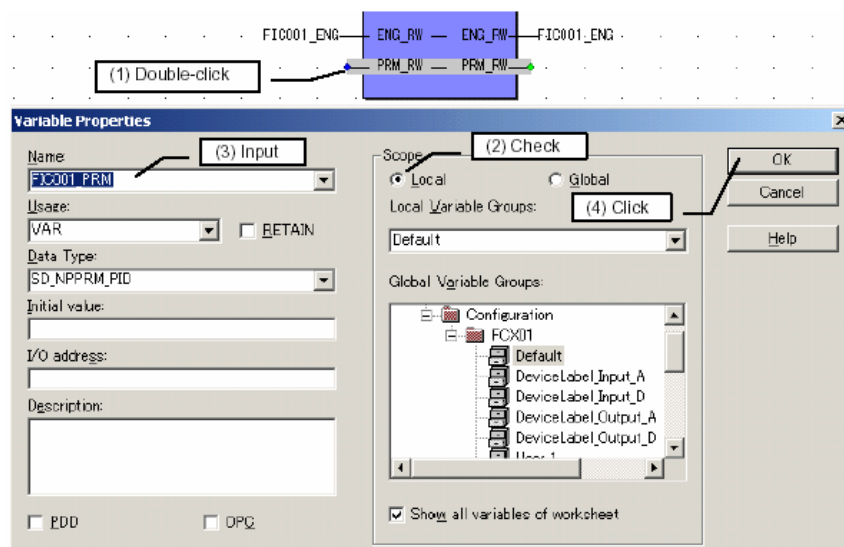


Figura 324.

O programa final deverá apresentar-se como mostrado na figura a seguir

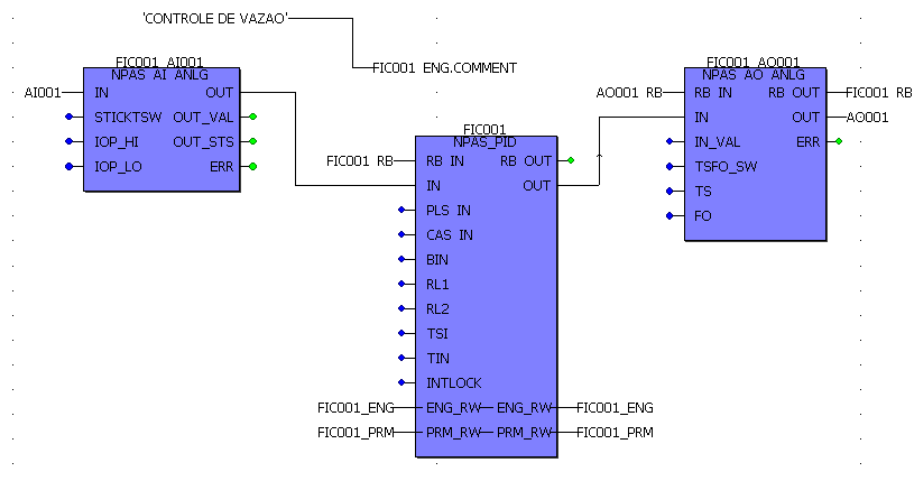


Figura 325.

- Selecione “File – Save Project As/Zip Project As” no menu do “Logic Designer”.
- Entre com um nome para o seu projeto (TREINO, por exemplo) e clique em “Save”.

AJUSTANDO E ASSOCIANDO UMA “TASK”

- Selecione “Task0” exibida na árvore do projeto. Clique com o botão direito do mouse e selecione “Properties”. Task0 é do tipo “cyclic”. Após confirmação do tipo da “task”, clique em “Cancel” para fechar esta janela.
- Selecione “Task0” com o botão direito do mouse e selecione “Settings”. Irá surgir uma janela “Task setting for IPC_40”. Nesta tarefa, valores como “interval” e “watchdogtime” serão mantidos.. Após confirmação, clique em “Cancel” para fechar esta janela.

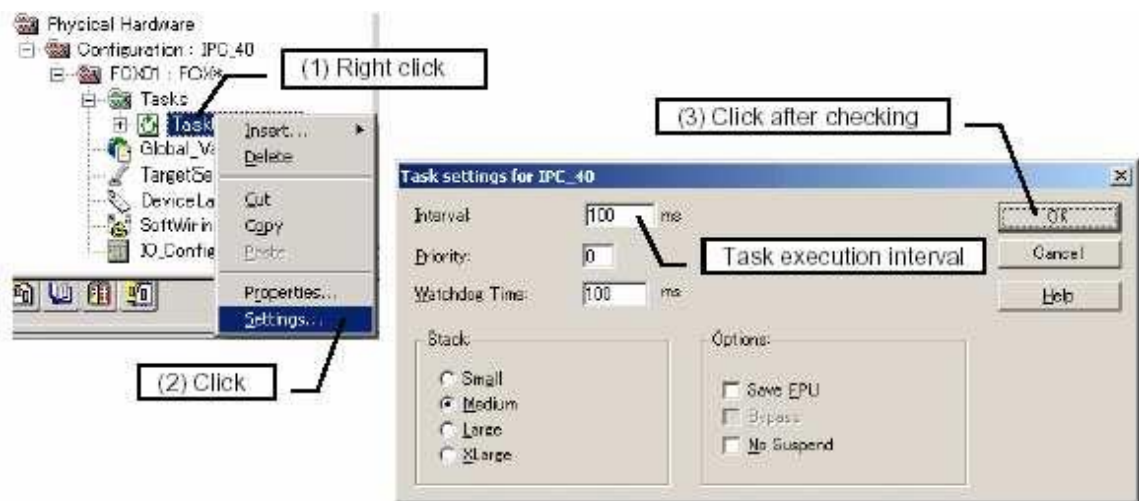


Figura 326.

- Selecione “Task0” e clique com o botão direito do mouse. Selecione “Insert - Program Instance”, através do menu.
- Entre no campo “Program Instance” e insira “_SINGLE_LOOP” e selecione “SINGLE_LOOP” no campo “Program Type”. Clique em “Ok”.

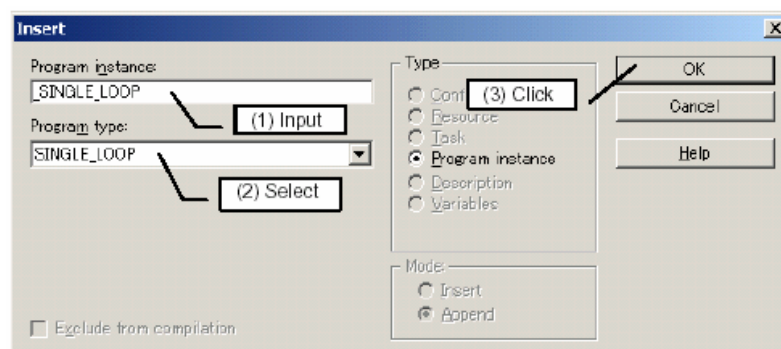


Figura 327.

DEFININDO “SOFTWARE WIRING”

- Duplo-clique em “Softwiring” na árvore do projeto.
- Clique em “Wizard”.

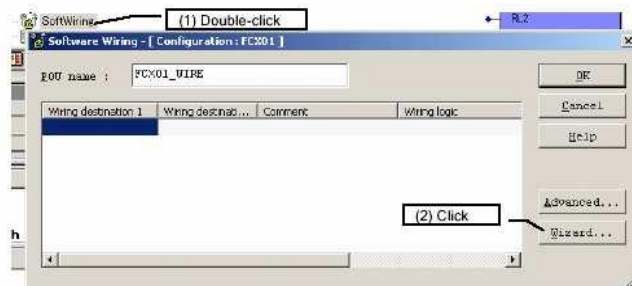


Figura 328.

- Selecione “AI001” para “Wiring Destination1”. Clique em “Next”.
- Selecione “Close Loop” para “Input form”. Clique em “Next”.
- Selecione “AO001” para “Wiring Source1”. Clique em “Next”.
- Selecione “SD_WIRE_DTANLG” para “Wiring logic”. Clique em “Next”.
- Entre com “t#5s” (5 segundos) em “First-Order Lag time Constant”. Deixe os outros campos com os seus valores “default”. Clique em “Next”.

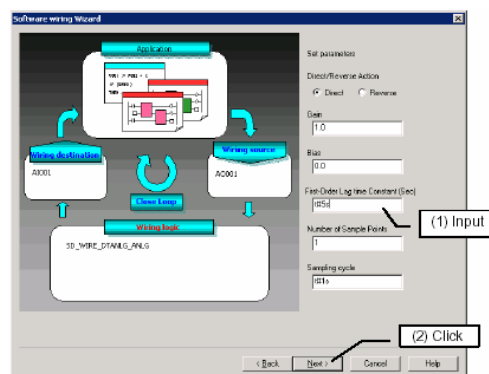


Figura 329.

- Clique em “Finish” para sair do modo “Wizard”.
- Clique em “Ok” para fechar o diálogo “Software Wiring”. Verifique que “FCX01_WIRE” acaba de ser adicionado abaixo de “Logical POU’s” no seu projeto.

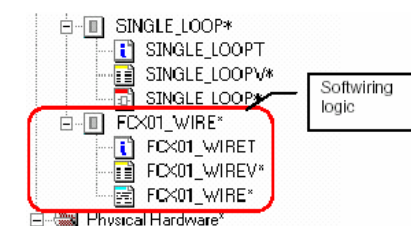


Figura 330.

- Selecione “Task0” e clique com o botão direito do mouse. Selecione “Insert - Program Instance” através do menu.
- Na janela que será exibida, entre com “FCX_WIRE” no campo “Program Instance” e “FCX01_WIRE” para o tipo de programa. Clique em “Ok”.
- Duplo-clique em “Global Variables” na árvore do projeto.
- Mude o valor inicial de “GS_NFIO_DISCONN” de “FALSE” para “TRUE”.

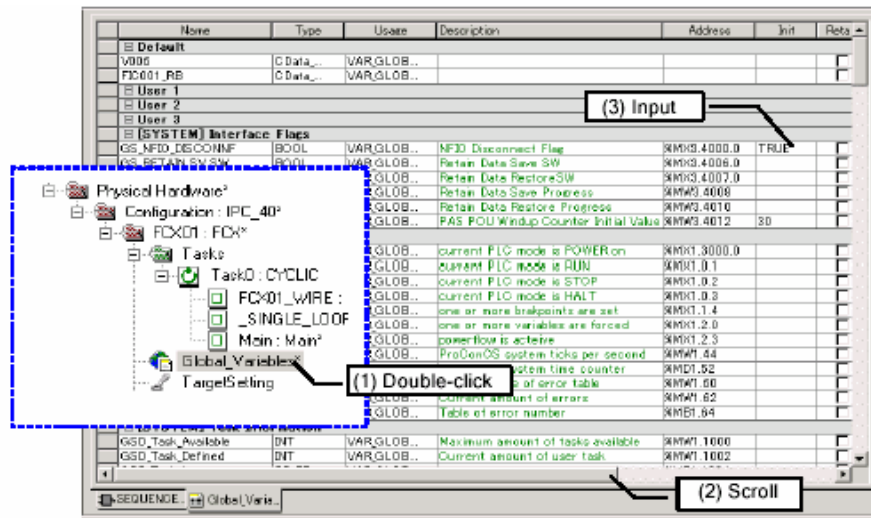


Figura 331.

FAZENDO O “BUILDING” E “DOWNLOAD” DO PROJETO

Neste ponto, estaremos convertendo para código de máquina toda a aplicação desenvolvida até o momento.

- Selecione “Build – Rebuild Project” através da barra de menu. Uma mensagem aparecerá informando que a compilação está em progresso. Quando concluída a compilação, o resultado da mesma será exibida na janela de mensagem como na figura abaixo. Toda mensagem que exibe “0 Error(s)” indica que a compilação foi concluída com sucesso.

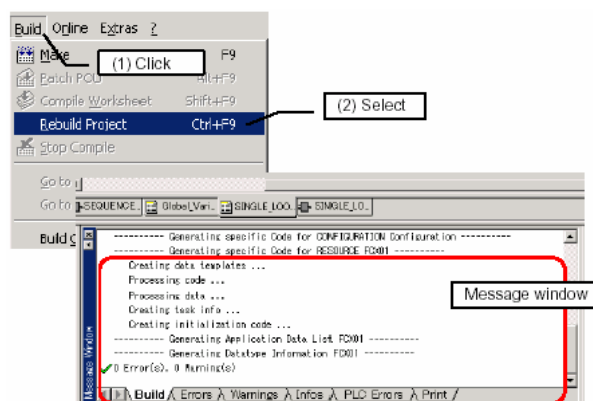


Figura 332.

Na figura abaixo é exibido um exemplo na qual a compilação apresenta erro.

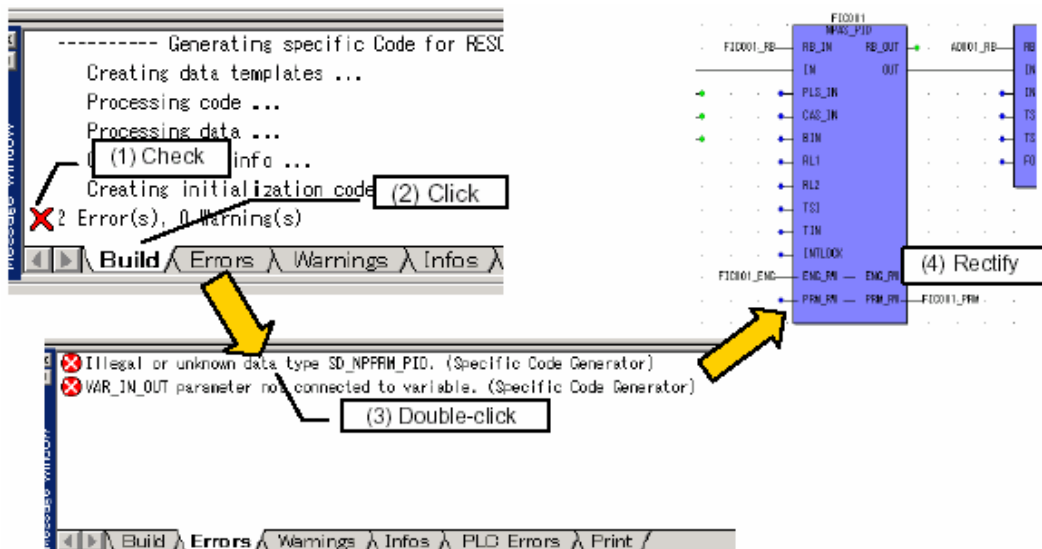


Figura 333.

- Selecione “Online – Project Control” através da barra de menu. O diálogo do “Project Control” aparecerá e em seguida,.

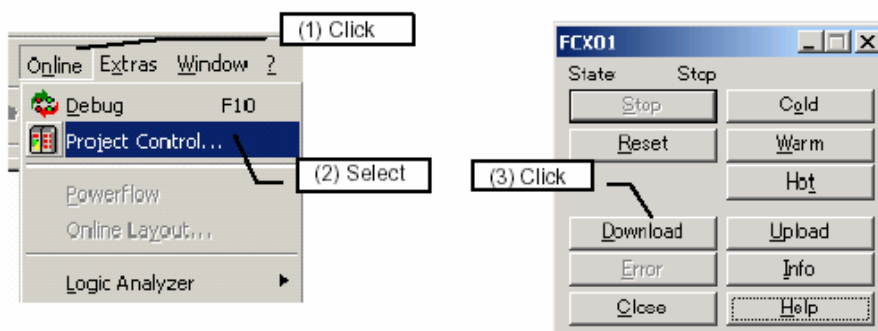


Figura 334.

- Clique em “Download”. O diálogo deste botão será exibido.
- Clique no botão “Download” abaixo do campo “Project”.

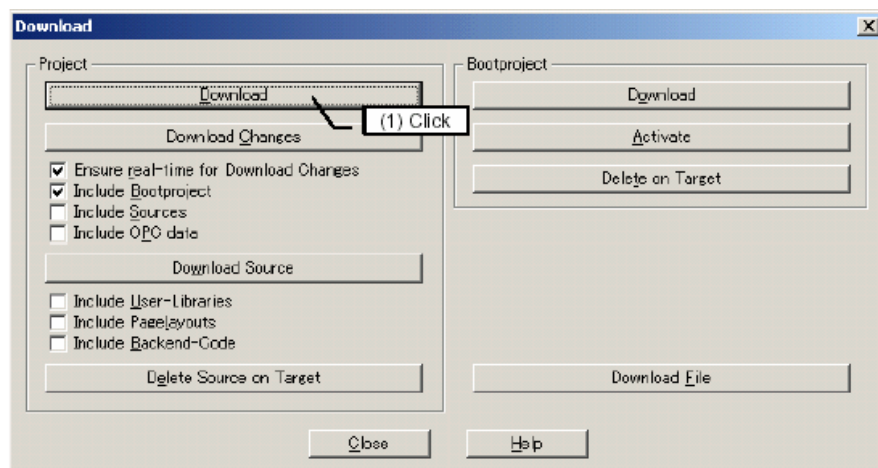


Figura 335.

- Se algum outro projeto foi carregado anteriormente, aparecerá uma janela informando que este projeto será sobrescrito. Clique em “YES”.

Na parte inferior do Logic Designer será exibida uma barra de progressão referente ao “download”. Aguarde até que haja a indicação operação realizada.

INICIANDO UM PROJETO

- Clicar no ícone “Project Control Dialog”, conforme figura abaixo.

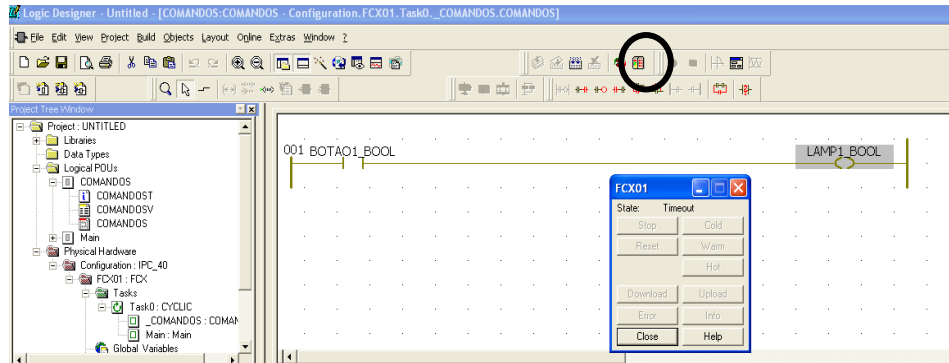


Figura 336.

- Clicar na opção “Cold” para iniciar o projeto, conforme figura abaixo.

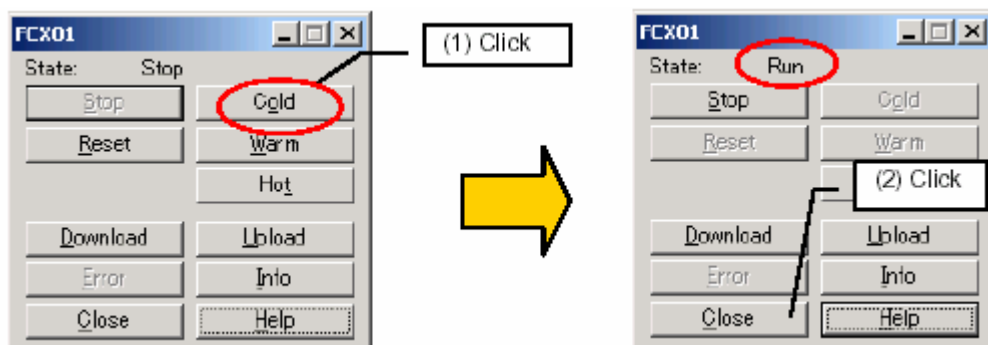


Figura 337.

ETAPA 4. VERIFICANDO A APLICAÇÃO

Veremos agora se a tarefa proposta está realmente de acordo com nossa necessidade. Para isto, através do “Logic Designer”, iremos coloca-la em modo “Debug”.

ALTERANDO O “LOGIC DESIGNER” PARA MODO “DEBUG”

- Clique sobre a planilha “SINGLE_LOOP” para que seja possível visualizar a lógica.
- Clique sobre o ícone “Debug on/off” na barra de ferramentas.

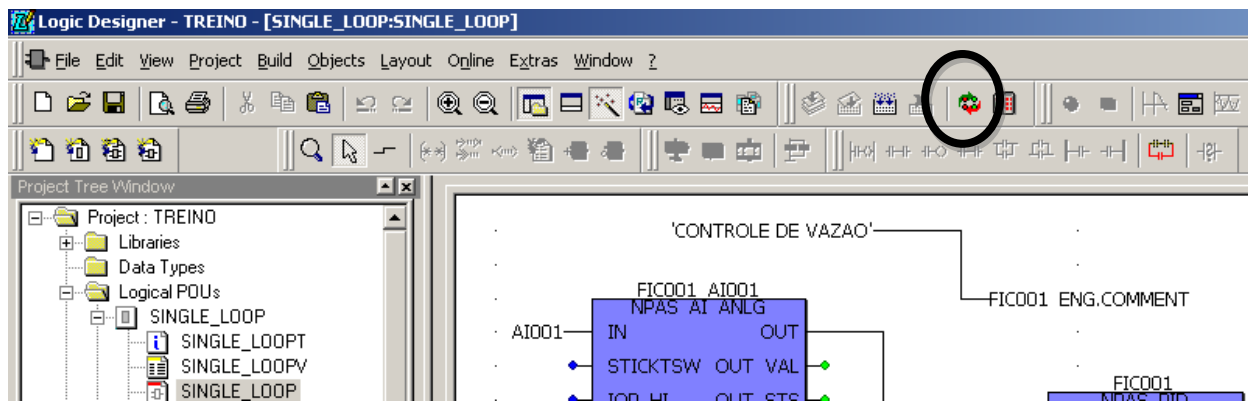


Figura 338.

No modo “Debug” o rodapé do Logic Designer será exibido na cor verde. Caso haja falha, o mesmo será exibido em vermelho. No “Logic Designer” existe uma ferramenta chamada de “Watch Window”. Através dela é possível visualizar valores como PV, SV e MV.

- Selecione “View - Watch Window” através da barra de ferramenta.
- Clique com o botão direito do mouse sobre a variável “FIC001_PRM”, presente na planilha e clique em “Add to Watch Window”.

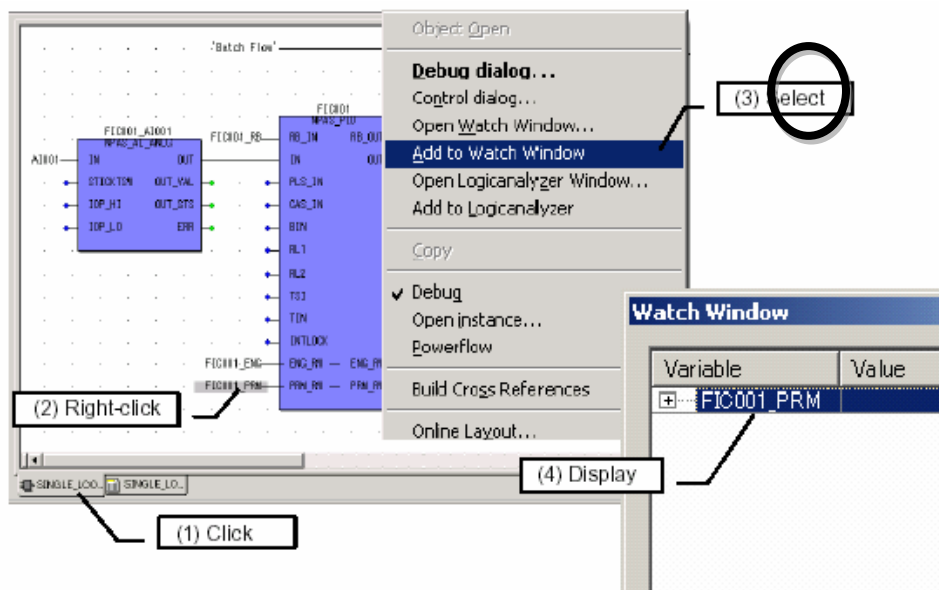


Figura 339.

- Clicando no símbolo de (+) da variável “FIC001_PRM” é possível acessar os parâmetros existentes no bloco de função “FIC001”.

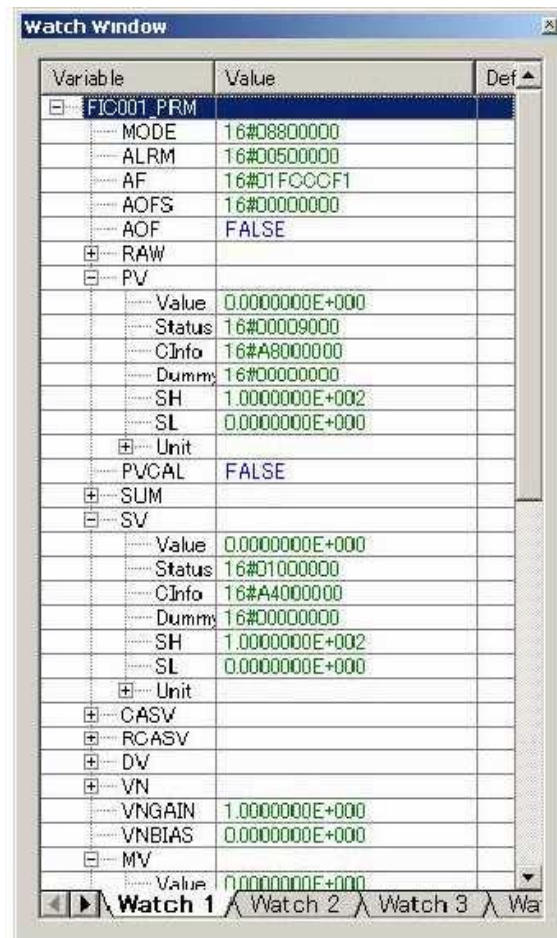


Figura 340.

Nota: É possível através deste recurso, visualizar e alterar o modo do bloco “FIC001”. O modo é uma “DWORD” e seu valor é exibido em hexadecimal.

MAN: 16#00800000

AUT: 16#00400000

CAS: 16#00200000

VERIFICANDO A OPERAÇÃO DO “LOOP” DE CONTROLE

Através dos passos abaixo iremos verificar se o controlador está operando normalmente.

MUDANDO VALOR DE “SV” (SETPOINT VALUE)

- Duplo-clique em “FIC001_PRM – SV –Value” na janela “Watch Window”. A janela “Debug:FCX01” é exibida.
- Insira o valor “45” no campo “Value” e clique em “Overwrite”.

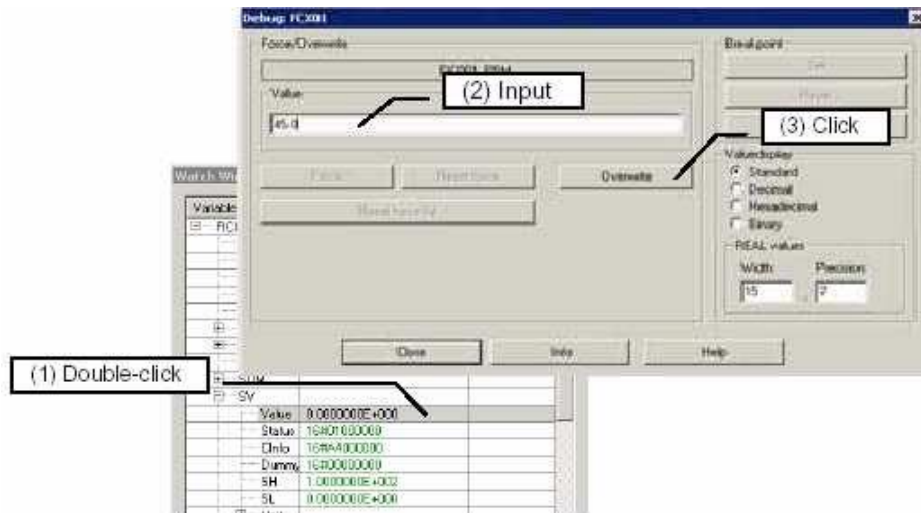


Figura 341.

A figura abaixo exibe o valor alterado.

Variable	Value
SV	4.500000E+001
Status	16#01000000
CInfo	16#A4000000
Dummy	16#00000000
SH	1.000000E+002
SL	0.000000E+000
Unit	

Figura 342.

ALTERANDO O MODO DO BLOCO

- Duplo-clique sobre a variável “FIC001_PRM – MODE” .
- Insira o valor “16#00400000” e clique no botão “Overwrite”.

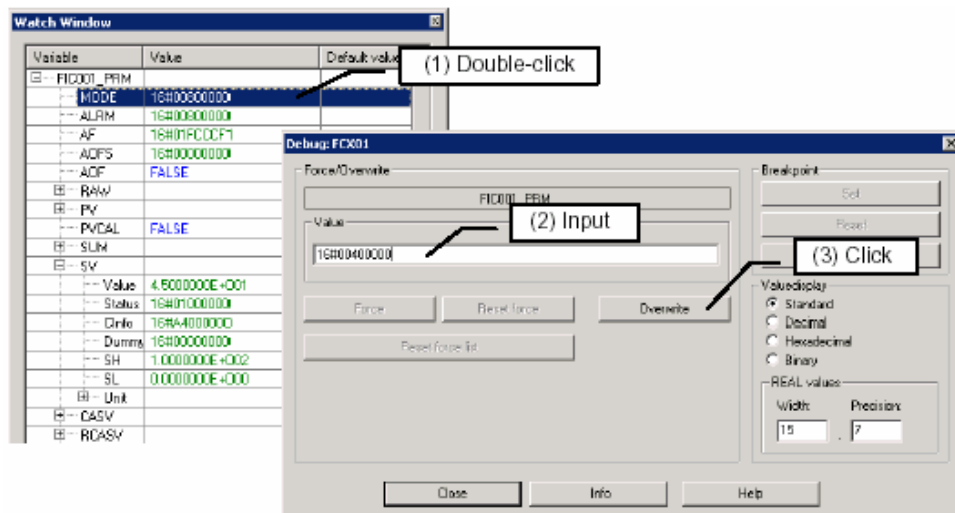


Figura 343.

Como foi realizado o procedimento de “Software wiring”, logo que o controlador for para modo “AUT”, as variáveis “MV” e “PV” terão que sofrer alterações.

ETAPA 5. CRIAR O PROJETO DO SUPERVISÓRIO

A figura a seguir mostra a tela a ser criada pelo supervisório.

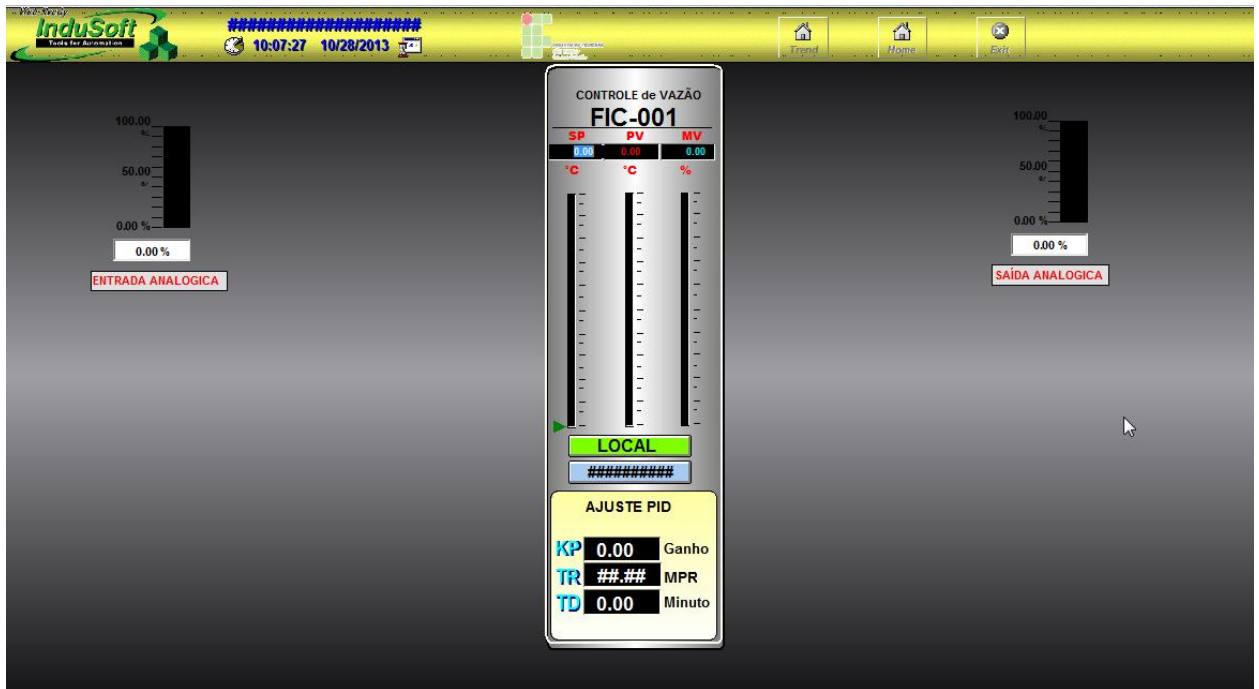


Figura 344.

Para iniciar o trabalho de configuração, solicite ao Professor, a cópia do projeto básico do INDUSOFT contendo as telas e configurações básicas. Este projeto tem o nome de "LABO_6_REV9". Copie a pasta completa para o local de trabalho do seu computador.

- Para abrir este projeto usando o InduSoft Web Studio, rodar o **InduSoft Web Studio** clicando no ícone , ou usando o botão "Iniciar", "Programas" -> "InduSoft Web Studio Tools" -> "InduSoft Web Studio".
- Já no ambiente do InduSoft Web Studio, selecionar **File** e clicar em **Open Project**.
- Selecionar o caminho correto onde foi copiado o pré-projeto e escolher o nome da aplicação "LABO_6_REV9.APP".

Vamos configurar o "Application Datasheet" como mostrado a seguir:

- Criar os tags, conforme mostrado a seguir.

Vamos criar agora a Classe que conterà os tags que manipularão as variáveis do bloco PID.

- Para criar a CLASSE, clicar com o botão direito na pasta Classes na Aba Database e escolher a opção **Insert**.
- Na caixa de dialogo, digitar o nome da classe, "PID" e clicar em OK.

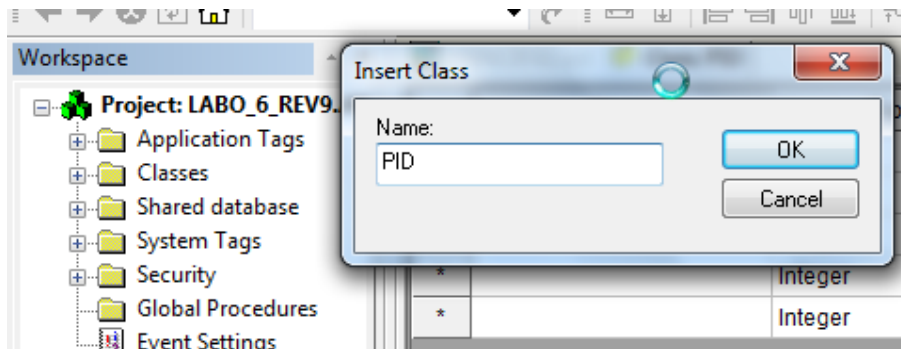


Figura 345.

Na planilha da classe que acabou de surgir, criar os nomes dos Membros da Classe.

Class: PID

	Name	Type	Description
1	PV	Real	
2	MV	Real	
3	SP	Real	
4	AM	Boolean	
5	P	Real	
6	TI	Real	
7	TD	Real	
*		Integer	
*		Integer	
*		Integer	

Figura 346.

- Fechar a planilha, ela pode ser fechada sem ser salva.

Com isso teremos a classe criada, no entanto, temos de associá-la a algum tag;

- No Workspace, selecionar o tab Database. Clicar duas vezes na pasta Application Tags para expandi-la.
- Clicar duas vezes em Datasheet View.

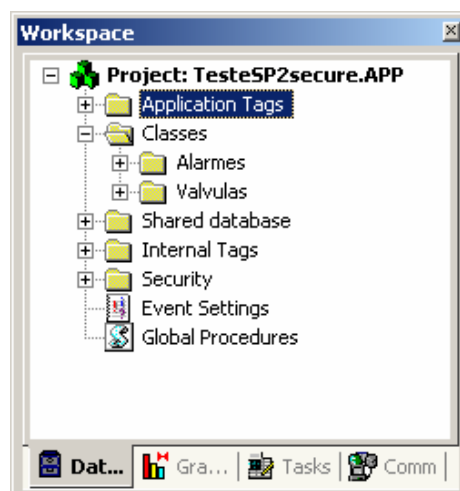
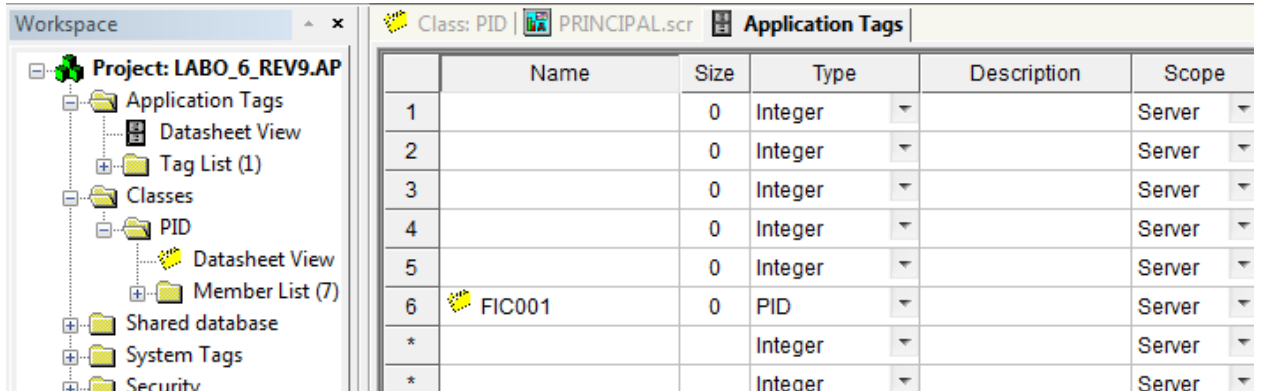


Figura 347.

- Configurar a janela Application Tags como mostrado a seguir:



	Name	Size	Type	Description	Scope
1		0	Integer		Server
2		0	Integer		Server
3		0	Integer		Server
4		0	Integer		Server
5		0	Integer		Server
6	FIC001	0	PID		Server
*			Integer		Server
*			Integer		Server

Figura 348.

Desta forma você estará criando o tag FIC001, que servirá de meio de associação entre o Application Tags e a planilha de Classe.

Feche a janela Application Tags, não é necessário salvar.

- Salvar a configuração do banco de dados usando o ícone: 
- Abrir uma tela chamada “PRINCIPAL”

As principais configurações desta tela já foram realizadas no pré-projeto, conforme pode ser observado na figura a seguir. Porém, alguns objetos ainda dependem de configuração, como por exemplo, as variáveis de atuação nos objetos “Bargraph” de ENTRADA ANALÓGICA e SAÍDA ANALÓGICA.

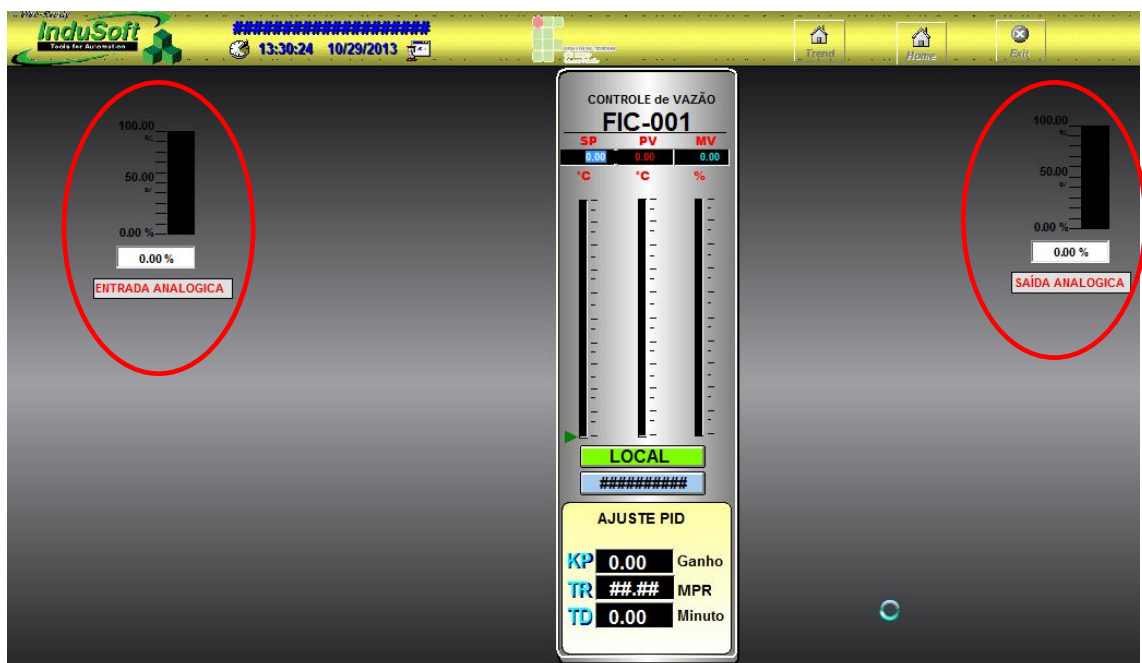


Figura 349.

- Em seguida configurar a partida no menu Project. Para tal, clicar na opção Settings.



Figura 350.

- No tab Runtime Desktop, digitar dentro da janela Startup: PRINCIPAL.
- Abrir uma tela chamada “TREND”
- Duplo-clicar sobre o objeto “Trend Control” e ao aparecer a janela de propriedades clicar na opção “Points”, conforme mostrado na figura a seguir.

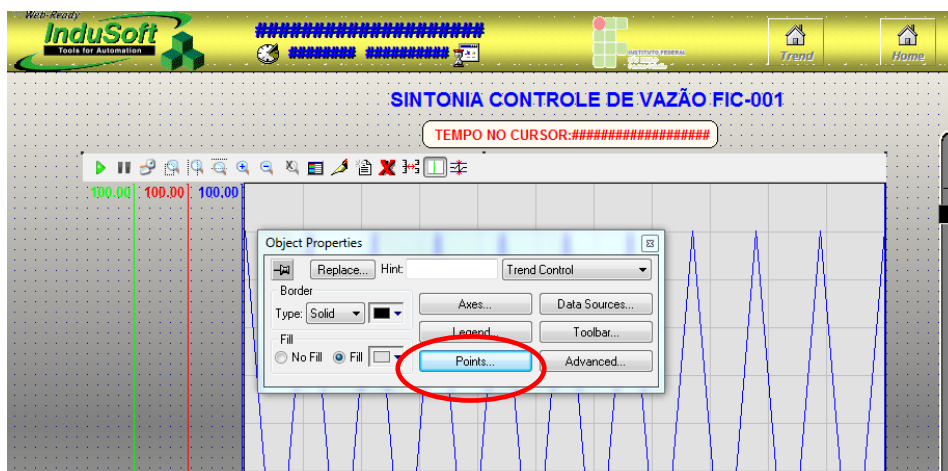


Figura 351.

- Digitar no campo “Tag/Field” os tag’s correspondentes ao SP, PV e MV do controlador FIC001 , conforme mostrado na figura a seguir.

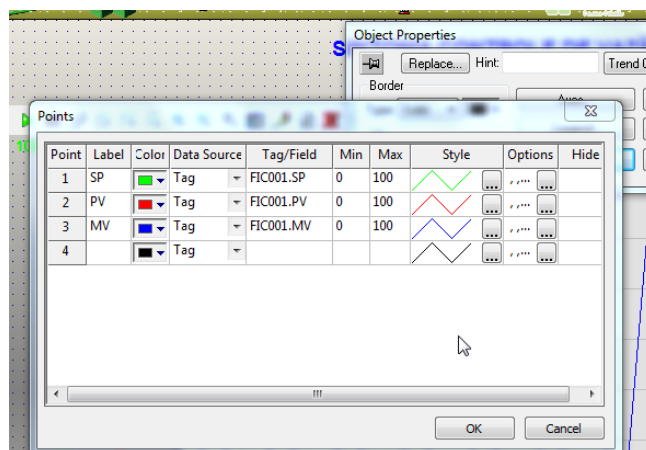


Figura 352.

ETAPA 6. CONFIGURAR COMUNICAÇÃO DO INDUSOFT COM O STARDOM VIA OPC

- Selecionar a aba “Comm” no rodapé da árvore de projeto.



Figura 353.

- Clicar com a tecla da direita do mouse na pasta “OPC” e inserir uma planilha de comunicação via OPC.

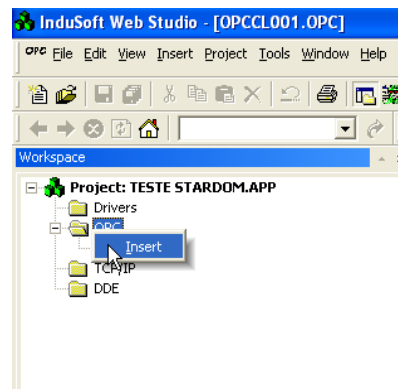


Figura 354.

- Selecionar a opção “Yokogawa.ExaopcDASTARDOMFCX” no campo “Server Identifier” conforme mostrado a seguir:

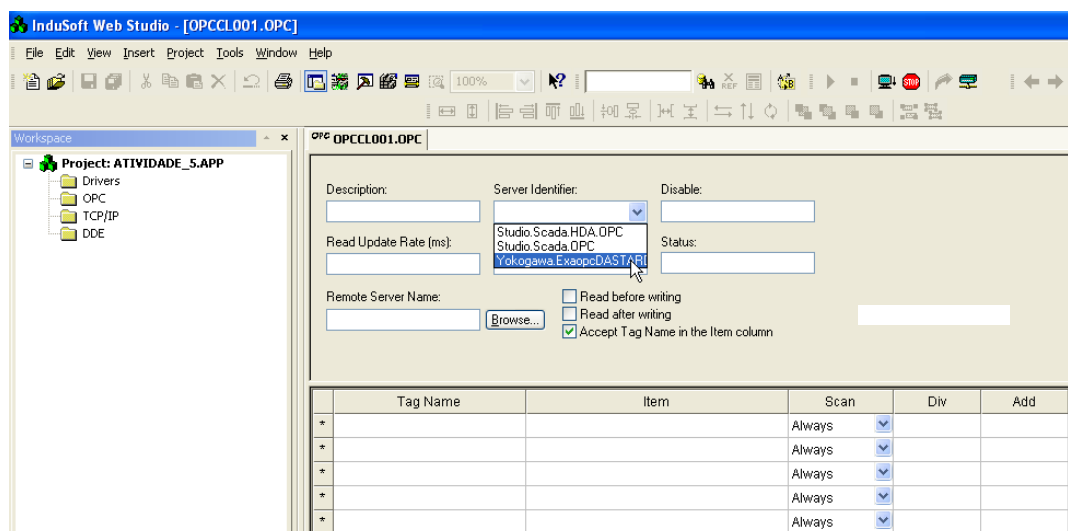


Figura 355.

- Preencher esta planilha de comunicação, conforme as necessidades do seu projeto. Para tal, duplo-clicar na primeira linha da coluna “Item” e identificar na janela “OPC BROWSER” o item que corresponde ao tag “FIC001_PRM.???”. Onde “???” corresponde ao parâmetro a ser configurado para troca de dados entre o CLP e o supervisor Indusoft.

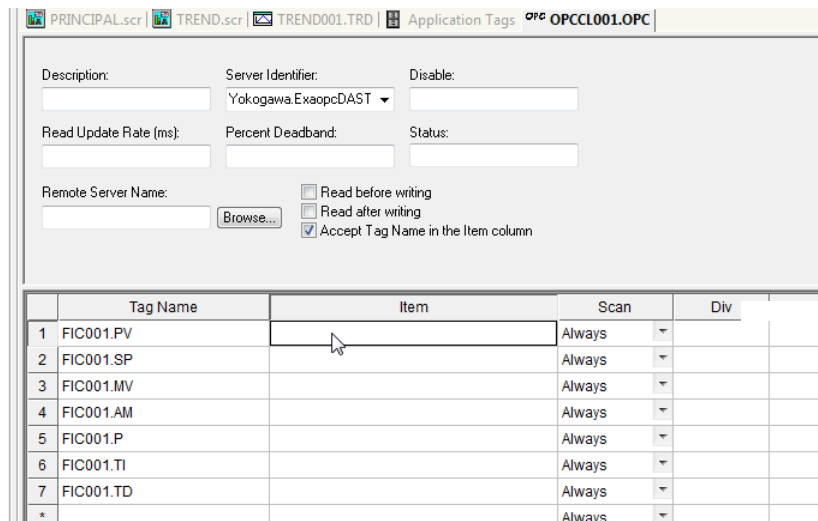


Figura 356.

Observar que o valor de ganho do controlador FIC001 selecionado pelo supervisor será enviado/lido pelo CLP como banda proporcional. Portanto, o parâmetro “Div” nesta planilha de comunicação deverá ser atribuído em função da relação: $BP = \frac{100}{G}$.

- Selecionar a opção “Status” no menu “Project”, conforme figura a seguir.

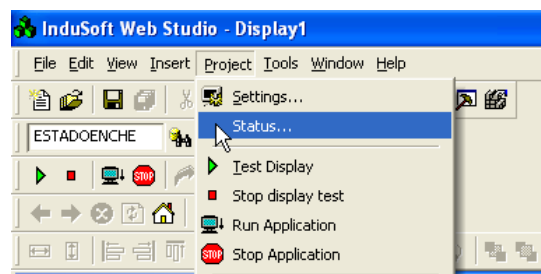


Figura 357.

- Selecionar a opção “OPC Client Runtime” e em seguida clicar no botão “Startup...”

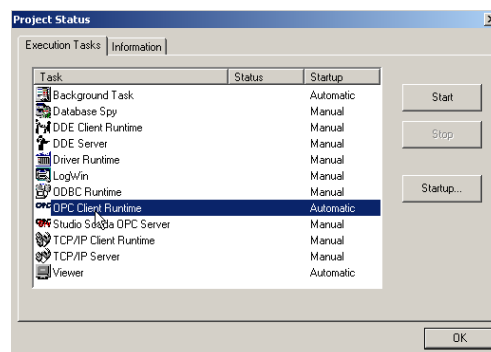


Figura 358.

- Selecionar a opção “Automatic” e em seguida clicar no botão “OK”

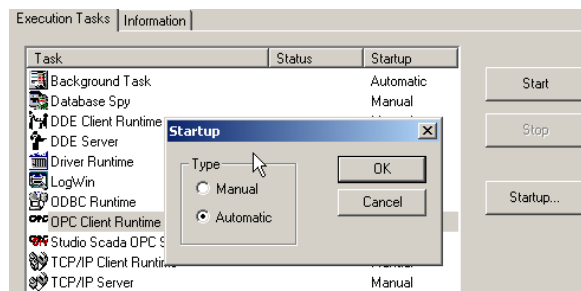


Figura 359.

A partir deste momento, quando o aplicativo for iniciado, a comunicação via OPC do Indusoft automaticamente será executada.

ETAPA 7. TESTAR CONTROLE PID DA MALHA DE VAZÃO

- Salvar a aplicação e rodar o sistema.
- Verificar se o controlador pode ser manipulado pelas telas do supervisão, testando as seguintes funções:
 - possibilidade para transferir o modo do controlador de automático para manual e vice-versa;
 - possibilidade de alteração do valor do SP do controlador, tanto em modo manual quanto em modo automático, através da digitação do valor na parte superior do *face plate*;
 - possibilidade de alteração do valor do SP do controlador, tanto em modo manual quanto em modo automático, através do deslocamento vertical da seta ao lado do *bargraph*, operação a ser realizada pelo cursor do *mouse*;
 - possibilidade de alteração do valor da MV do controlador, somente quando o controlador estiver em modo manual, através da digitação do valor na parte superior do *face plate*;
 - possibilidade de alteração do valor da MV do controlador, somente quando o controlador estiver em modo manual, através do deslocamento vertical da seta ao lado do *bargraph*, operação a ser realizada pelo cursor do *mouse*;
 - possibilidade de alteração do ganho, integral e derivativa, tanto em modo manual quanto em modo automático, através da digitação de seus valores na parte inferior do *face plate*;
- Verificar se na tela TREND é possível a visualização gráfica dos parâmetros SP, PV e MV do controlador FIC001.