



HI tecnologia
Indústria e Comércio Ltda

Notas de Aplicação

Utilização da MMI800 – Módulo I

HI Tecnologia

Documento de acesso público

ENA.00032

Versão 1.02

outubro-2013

Apresentação

Esta nota de aplicação foi elaborada pela **HI Tecnologia Indústria e Comércio Ltda.** O departamento de suporte a clientes, está disponível, através do telefone **(19) 2139-1700** ou do e-mail suporte@hitecnologia.com.br, para esclarecimentos sobre as informações contidas neste documento, ou para dirimir quaisquer dúvidas a respeito de nossos produtos. Favor mencionar as informações a seguir para que possamos identificar os dados relativos a este documento.

ID da Nota de Aplicação: ENA.00032
Versão Documento: 1.0

HI Tecnologia Indústria e Comércio Ltda.

Endereço: Av. Dr. Armando de Sales Oliveira, 445

Cidade: Campinas – SP
CEP: 13076-015

Fone: +55 (19) 2139-1700
Fax: +55 (19) 2139-1710

E-mail: hi@hitecnologia.com.br

Web site: www.hitecnologia.com.br

Referência: ENA.00032
 Arquivo : ENA0003200.doc

Revisão: 2
 Atualizado em: 17/04/2008

Índice

Apresentação da Interface Homem-Máquina MMI800.....	5
Referências.....	6
Introdução.....	7
Definindo um processo para supervisão	7
O que é uma autoclave?	7
Capítulo 1	10
1.1 Criação de um novo projeto	10
1.2 Telas.....	12
1.2.1 Criando a primeira tela.....	12
1.3 Tags	14
1.3.1 Criando um tag para M14	15
1.3.2 Criando um tag para D0.....	15
1.4 Campos.....	16
1.4.1 Criando um campo na tela “Principal” para o Tempo de Vácuo.....	17
1.4.2 Criando um campo na tela “Principal” para a Pressão Interna.....	18
1.5 Comandos.....	19
1.5.1 Configurando o comando “Open Screen”	19
1.6 Macros.....	21
1.6.1 Criando Macro Abre Tela Principal.....	21
1.7 Ajustes Finais.....	22
1.8 Carga do Programa.....	22
Capítulo 2	25
2.1 Criação da tela para programação do tempo de vácuo.....	25
2.2 Criação do campo para a tela de tempo de vácuo	26
2.3 Criação do comando da abertura da tela de programação	26
2.4 Criação de macro para tela de tempo de vácuo	27
2.5 Associação das teclas de função da IHM	27
2.5.1 Programando a Tecla F1	27
2.5.2 Programando a Tecla F2	28
2.6 Carga do programa	28
Capítulo 3	29
3.1 Tag para o operando R100.....	29
3.2 Comandos Liga/Desliga Bomba de Vácuo	29
3.2.1 Cria comando Liga Bomba de Vácuo	30
3.2.2 Cria comando Desliga Bomba de Vácuo.....	30
3.3 Macros.....	30
3.3.1 Cria macro Liga Bomba de Vácuo.....	30
3.3.2 Cria macro Desliga Bomba de Vácuo.....	31
3.4 Teclas da IHM	31
3.4.1 Associação das teclas	31
3.5 Alterando a tela “Principal”	32
3.5.1 Alteração da tela.....	32



Utilização da MMI800 – Módulo I

Tipo de Doc.: Notas de Aplicação
Referência: ENA.00032

Revisão: 2
Atualizado em: 17/04/2008

3.5.2	Criando um campo para o motor	32
3.6	Lista de texto	33
3.6.1	Criando a lista de texto	33
3.6.2	Associando a lista de texto a um campo	34
3.7	Carga do programa	34
Capítulo 4	35
4.1	Telas.....	35
4.1.1	Criando tela de alarme.....	35
4.2	Tag	36
	Criando <i>tag</i> para R14	36
4.3	Campos	36
	Criação do campo oculto nas telas	36
4.4	Comandos	37
4.4.1	Configuração do comando “ <i>OpenScreen</i> ” para a tela de alarme.....	37
4.4.2	Configuração do comando “ <i>ProgramBuzzer</i> ”	38
4.4.3	Configuração do comando “ <i>ProgramLed</i> ”	38
4.4.4	Configuração do comando “ <i>ProgramBacklight</i> ”	39
4.5	Macros.....	40
4.5.1	Criação da macro ativa alarme	40
4.5.2	Criação da macro reconhece alarme.....	40
4.6	Associação das Teclas da IHM.....	41
4.7	Ajustes Finais	41
4.8	Carga do programa	42
Controle do Documento.....		43
Considerações gerais		43
Responsabilidades pelo documento		43

Apresentação da Interface Homem-Máquina MMI800



O Painel de Operação MMI800 é uma IHM (Interface Homem Máquina) para ambientes industriais desenvolvida para integração com CLP's (Controladores Lógicos Programáveis) e equipamentos de supervisão e controle. Com um *display* de 4 linhas e 20 caracteres por linha, possui caracteres grandes (4.84 x 9.22mm) de fácil visualização e *backlight* programável.

Dotado de um teclado funcional e numérico, o usuário pode definir teclas de função, associar variáveis para supervisão e programação, protegidas ou não com senhas dotadas de quatro níveis de privilégio.

Toda esta funcionalidade é definida através do ambiente de programação OPPE, que possui as ferramentas necessárias para criação, programação, teste e carga das telas desenvolvidas.

A comunicação entre a MMI800 e os CLP's é realizada através de um canal serial RS232-C (opcionalmente pode ser utilizado um canal RS485). O protocolo de comunicação entre a MMI800 e o equipamento remoto é o SCP-HI ou Modbus-RTU (a partir da versão 2.xx)

A MMI800 apresenta uma ruptura com o método anterior de programação das outras IHMs da HI Tecnologia, uma vez que as telas não são mais programadas dentro do ambiente de programação SPDSW, mas sim por meio de um ambiente chamado OPPE ("Operator Panel Program Environment"). O *software* OPPE está disponível para *download* gratuito no site da HI Tecnologia, (<http://www.hitecnologia.com.br>) na seção de "downloads".



Utilização da MMI800 – Módulo I

Tipo de Doc.: Notas de Aplicação
Referência: ENA.00032

Revisão: 2
Atualizado em: 17/04/2008

Referências

Controlador ZAP900/901

- PET.108.001 Folha de Especificação Técnica do ZAP900/901 (formato PDF).

Interface Homem Máquina Remota (MMI)

- PET.405.001 Folha de Especificação Técnica da MMI650 (formato PDF).
- PET.403.001 Folha de Especificação Técnica da MMI800 (formato PDF).

Notas de Aplicação

- ENA.00004 Programação da IHM no SPDSW (formato PDF)
- ENA.00047 Introdução a MMI - Geração GII (formato PDF)
- ENA.00048 Tipos de dados da MMI GII (formato PDF)
- ENA.00039 Navegação de telas nas MMI Geração II - Módulo 1 (formato PDF)
- ENA.00040 Navegação de telas nas MMI Geração II - Módulo 2 (formato PDF)
- ENA.00041 Navegação de telas nas MMI Geração II - Módulo 3 (formato PDF)



Utilização da MMI800 – Módulo I

Tipo de Doc.: Notas de Aplicação
Referência: ENA.00032

Revisão: 2
Atualizado em: 17/04/2008

Introdução

Este documento tem como objetivo introduzir e exemplificar ao cliente o funcionamento da IHM (Interface Homem Máquina) MMI800. Esta é a primeira de um conjunto de notas de aplicação que apresentam aos usuários o ambiente OPPE e o processo de criação de telas para a MMI800 e outras interfaces homem máquina compatíveis com este ambiente.

Esta nota será de nível básico e após a sua leitura e compreensão fornecerá ao usuário as seguintes competências:

- Conhecer as informações básicas sobre a MMI800 e suas características técnicas
- Criação e programação de telas.
- Utilização de “**Tags**” e de “**Campos**”, para a visualização e alteração de variáveis do tipo R (contatos), M (memórias inteiras) e D (memórias reais).
- Programação e utilização das teclas de função.
- Utilização das funções “**open screen**”, “**send direct data**”, “**program led**”, “**program buzzer**” e “**program backlight**”, que lidam com as telas, envio de comandos para CLP, *leds* das teclas de função, sinalizador sonoro e a iluminação do *display*.

Definindo um processo para supervisão

Para apresentação dos conceitos necessários para utilização do ambiente de programação, será definido um processo de exemplo. Utilizando esta aplicação serão desenvolvidas telas para supervisão e controle da mesma.

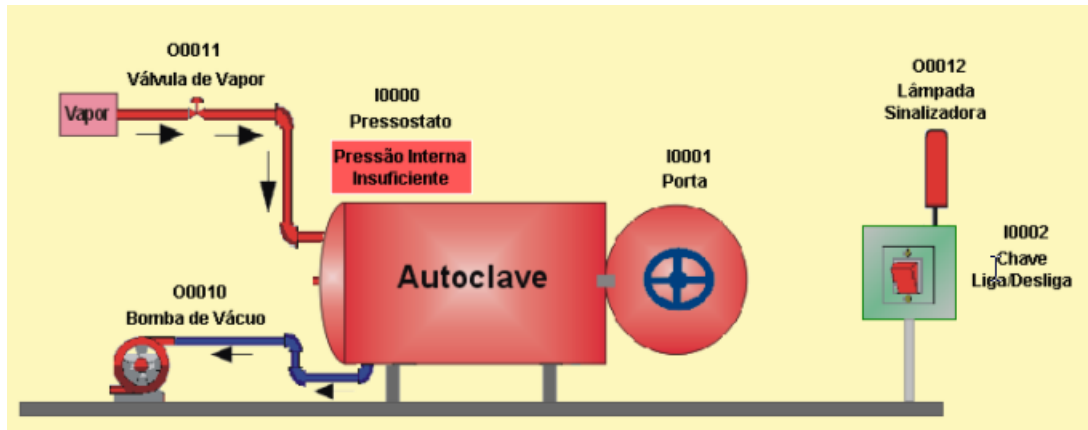
A aplicação escolhida foi uma autoclave simples controlada por um programa de PLC da HI Tecnologia. Assim será apresentada a máquina, definidos os requisitos de controle com a sequência de operação e os recursos necessários. O programa de PLC que executa este controle esta disponível junto com esta documentação e poderá ser carregado em um PLC para que os exercícios apresentados possam ser repetidos pelo leitor.

O que é uma autoclave?

É um equipamento utilizado para esterilização de produtos. A esterilização é efetuada inserindo o produto em uma câmara fechada, retirando-se o ar por meio de uma bomba vácuo, e em seguida aquecendo-se a câmara através de uma linha de vapor.

O sistema de controle mantém o produto em uma temperatura controlada por um tempo programado e em seguida desliga o vapor, fazendo com que a temperatura retorne as condições ambientais para retirada do produto.

A ilustração abaixo mostra uma autoclave e seus componentes:



Para iniciar o funcionamento da autoclave deste programa de exemplo, é necessário que a porta seja fechada (**Entrada digital I1 deve ser ativada**), pois do contrário o sistema irá indicar falha (**Saída digital O12 vai oscilar indicando alarme**). Após a ativação da chave Liga/Desliga (**Entrada digital I2**), o sistema irá ativar a bomba de vácuo (**Saída digital O10**) por um tempo de 15 segundos, para que todo o ar da câmara interna seja retirado.

Após a retirada do ar, o sistema irá injetar vapor na câmara interna (**Saída digital O11 será ativada**), até que o pressostato indique que a pressão interna está no valor desejado (**Entrada digital I0 deve ser ativada**). O pressostato é um instrumento elétrico que fecha um contato quando o valor da pressão na câmara interna atinge um valor pré-programado.

Quando esse valor de pressão é atingido, o sistema entra em uma fase conhecida como “esterilização”, sendo que a câmara interna fica submetida a este valor de pressão sob um determinado tempo e esta é também a fase na qual ocorre efetivamente a eliminação dos germes e bactérias. Após transcorrer o tempo de esterilização, a linha de vapor é desligada sinalizando o final do ciclo de esterilização. A partir daí o sistema estará pronto para um novo ciclo.

Existe também um transdutor de temperatura analógico, que mede a temperatura da câmara interna e utiliza a entrada analógica **E0** para interface com o CLP. O valor lido pelo transdutor de temperatura é movido para a memória real **D0**, que será a memória cujo valor indica a pressão interna da câmara.

O valor do tempo de vácuo (em segundos) é inserido na memória inteira **M14**.

Utilizando o exemplo descrito acima, serão desenvolvidos programas para a MMI800 que permitirão ao usuário supervisionar e programar a autoclave apresentada. Em cada um dos programas serão apresentados conceitos com nível de complexidade crescente. Os conceitos utilizados foram distribuídos da seguinte forma:



Utilização da MMI800 – Módulo I

Tipo de Doc.: Notas de Aplicação
Referência: ENA.00032

Revisão: 2
Atualizado em: 17/04/2008

Capítulo 1

- Criação de uma tela.
- Criação de um Tag.
- Utilização de “escala” para conversão de valores.
- Supervisão de variáveis do tipo M e D.
- Introdução ao conceito de “campo”.

Capítulo 2

- Programação e supervisão de variáveis do tipo M
- Criação de uma tela para programação de tempo de vácuo.
- Utilização das teclas de função para visualização de diferentes telas

Capítulo 3

- Programação e supervisão de variável do tipo R.
- Alteração de tela para a utilização de “lista de texto”
- Utilização de tecla de função para ligar e desligar um operando.

Capítulo 4

- Utilização de campo oculto
- Criação de uma tela de alarme
- Utilização dos leds das teclas de função, sinalizador sonoro e iluminação do display (Backlight)

Observação:

Os arquivos gerados pelo OPPE (ambiente de programação da MMI800) para cada capítulo são fornecidos em conjunto com esta nota de aplicação, sendo que a sua nomenclatura é a seguinte:

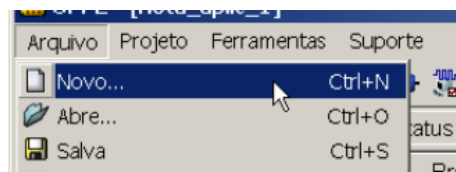
- Capítulo 1.OPJ – Arquivo gerado para as funcionalidades do Capítulo 1.
- Capítulo 2.OPJ – Arquivo gerado para as funcionalidades do Capítulo 2.
- Capítulo 3.OPJ – Arquivo gerado para as funcionalidades do Capítulo 3.
- Capítulo 4.OPJ – Arquivo que abrange todas as funcionalidades desta nota de aplicação.

Capítulo 1

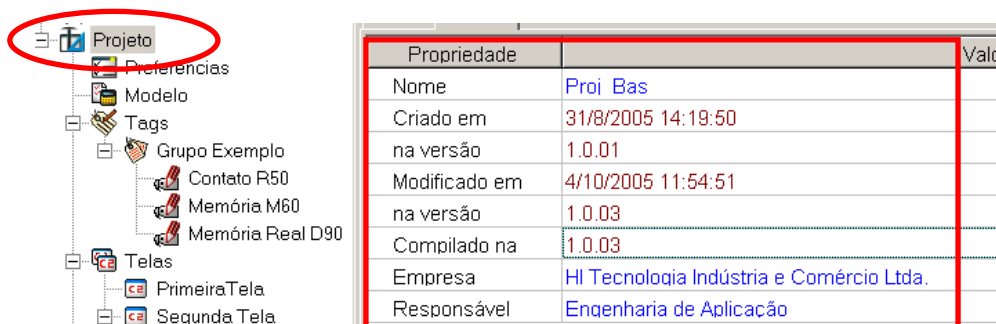
Neste capítulo serão utilizadas algumas funções elementares de criação de telas e supervisão de variáveis do CLP, sendo que os conceitos aqui aplicados terão como base a autoclave que foi descrita na introdução. Inicialmente serão definidos os passos para a criação de um projeto novo.

1.1 Criação de um novo projeto

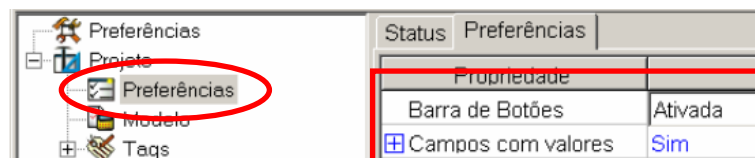
Com o OPPE sendo executado, vá até o menu “**Arquivo**” e selecione “**Novo**”.



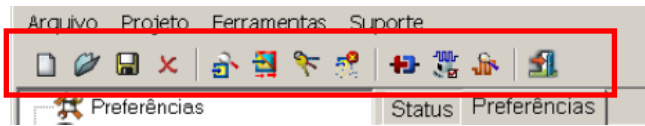
Na janela que se abrirá, especifique o nome e o diretório no qual o arquivo será criado. Em seguida, será preciso especificar o nome do projeto e o nome do responsável pelo mesmo. Para isto, clique no item “**Projeto**” no menu à esquerda e especifique o nome do projeto, o nome do responsável e o nome da empresa nos campos especificados.



Ao clicar no item “**Preferências**”, existem dois itens a serem configurados: “**Barra de Botões**” e “**Campos com Valores**”.

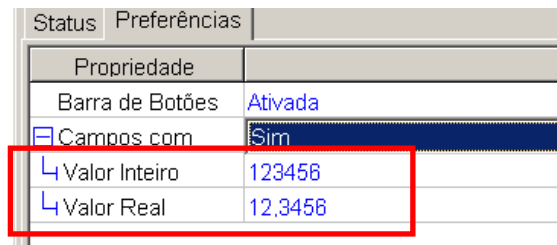


O item “**Barra de Botões**” permite configurar a exibição da barra que contém os ícones para algumas operações do OPPE. Se esta opção receber o valor “**Desativado**”, a barra com os botões não será exibida, e ao selecionar a opção “**Ativado**”, os botões serão exibidos. A “**Barra de botões**” está ilustrada na figura a seguir:

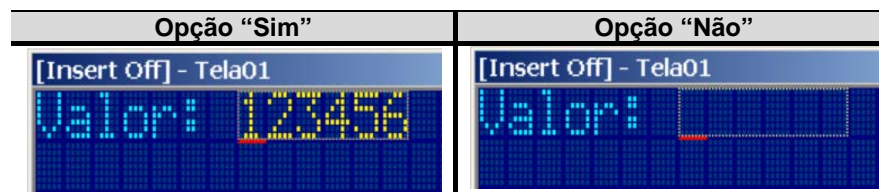


Para este exemplo, deixe a opção inicial da “Barra de botões” que é “Ativada”.

Existe também a opção “Campos com valores”, que aceita as opções “Sim” e “Não”. Isto é, esta opção determinará se algum caractere será exibido no momento da criação de um campo. Entretanto, os caracteres serão exibidos **somente no momento de edição da tela e criação do campo dentro do OPPE**, com o objetivo de auxiliar o usuário no posicionamento do campo. Ao clicar com o *mouse* no botão “+” na opção “Campos com valores” será possível especificar os valores que serão mostrados no caso de um campo de valor inteiro e um campo de valor real, conforme ilustração abaixo:

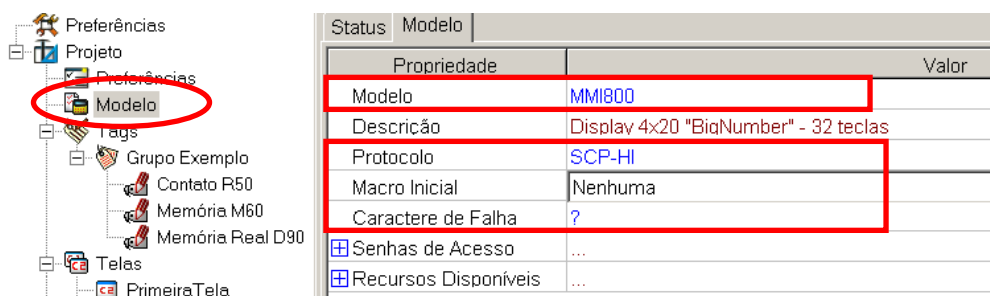


A tabela a seguir mostra a diferença entre as opções “Sim” e “Não” do item “Campos com valores”, para um campo do tipo valor inteiro:



Para este exemplo, deixe a opção inicial do item “Campos com valores” que é “sim”.

O próximo passo é especificar o modelo da IHM que será programado. Para isso, clique no item “Modelo”, onde existirá uma caixa de seleção para determinar o modelo de IHM. Após a seleção do modelo (**MMI800**), os dados e opções relativas a este modelo serão mostrados.



Para auxiliar o usuário, um campo de descrição da IHM surgirá para que este tenha certeza de que especificou o modelo corretamente.

No campo **"Protocolo"** determina-se qual será o protocolo de comunicação usado. As opções disponíveis são protocolo **"SCP-HI"** e protocolo **"MODBUS RTU"**. Selecione a opção **"SCP-HI"**

Em seguida, no campo **"Macro Inicial"**, especifica-se qual será a macro executada no momento da ativação da IHM. Maiores informações sobre macros serão fornecidas nas seções posteriores. Por enquanto, não altere este campo, deixe para alterá-lo nas seções posteriores.

O próximo item é **"Caractere de Falha"**. Neste campo determina-se qual será o caractere que será mostrado em caso de falha de comunicação com o CLP. Exemplo: Se o caractere de falha for "?" e estiver monitorando um campo qualquer, seja o valor de um contato, ou de uma memória M ou de uma memória D, o caractere "?" será mostrado no campo monitorado em caso de perda de comunicação com o CLP.

Abaixo de **"Caractere de Falha"**, tem o campo **"Senhas de Acesso"**, que especifica as senhas para cada nível e os respectivos tempos de *logout*. O *logout* funciona da seguinte maneira: Ao entrar com sua senha, o usuário habilita o seu nível de senha. Após o tempo especificado de *logout*, o nível de senha será desabilitado e o usuário terá que re-digitar a senha. Se o valor do tempo de logout for zero, a função de logout automático estará desabilitada. Maiores informações sobre **"Senhas de Acesso"** serão fornecidas nos próximos documentos.

Protocolo	SCP-HI
Macro Inicial	Nenhuma
Caractere de Falha	?
<input type="checkbox"/> Senhas de Acesso	...
↳ Senha Administrador	1
↳ Logout Admin. (min.)	2
↳ Senha Gerente	2
↳ Logout Gerente (min.)	5
↳ Senha Supervisor	3
↳ Logout Superv. (min.)	15
↳ Senha Operador	4
↳ Logout Oper. (min.)	60
<input type="checkbox"/> Recursos Disponíveis	...

Agora que o modelo já foi especificado e as preferências do projeto já foram determinadas, pode-se passar à fase de criação de telas.

1.2 Telas

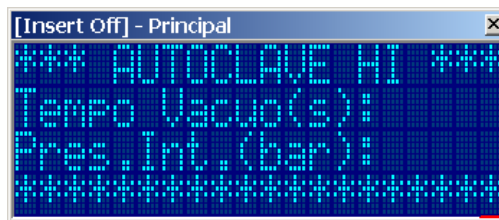
1.2.1 Criando a primeira tela

A tela criada neste capítulo deverá supervisionar o tempo de vácuo e o valor da pressão lido pelo transdutor de pressão da autoclave. Portanto, irá executar a supervisão de uma memória do tipo M (inteira) - tempo de vácuo e do tipo D (real) - pressão.

Para criar uma tela, clique com o botão esquerdo do *mouse* no item **"Telas"** e depois clique com o botão direito do *mouse*. Uma opção **"Adiciona Tela"** será mostrada. Clique com o botão esquerdo do *mouse* para adicionar uma nova tela.



Uma janela com fundo azul surgirá e ela será a tela da IHM a ser editada. Digite na tela o texto a seguir:



Note que ao ser criada a primeira tela, surgirá mais um nível abaixo do item “**Telas**”. Um item com o nome de cada tela criada surgirá, onde ao clicar em cada um destes itens será possível configurar os demais parâmetros de cada tela, como, por exemplo, o Nome da Tela, tempo de *refresh*, e as macros associadas a cada tecla.

Deve-se mudar o nome da tela. Para isso, clique em “**Tela01**”, abaixo do item “**Telas**” e veja na parte direita do OPPE os parâmetros relativos a cada tela. Mude o nome para “**Principal**”



Clique no campo “**Descrição**” e digite “**Tela principal da Autoclave**”. Em seguida, mude também o “**Tempo de Refresh**” para o valor 500. Esse tempo é dado em milissegundos (ms) e é o intervalo de tempo em que a tela é atualizada. Mude este parâmetro sempre que desejar que a atualização dos valores na tela seja mais rápida do que o normal (2000 ms.).

Propriedade	
Nome	Principal
Descrição	Tela principal da autoclave
Identificador da Tela	0
Tempo de Refresh	500
Macros	...
Número de	0

1.3 Tags

O que é um Tag?

O *Tag* é uma “declaração de variável” para a IHM, isto é, determina-se o tipo de variável a ser lida, o número dela, o endereço do controlador associado a ela (a IHM pode estar conectada em rede com vários controladores) e se a variável é apenas de leitura ou se pode ser escrita também.

No exemplo dado é preciso criar *Tags* para duas variáveis. As variáveis lidas são as seguintes:

- M14 – Memória do tipo inteiro cujo valor determina o tempo de vácuo.
- D0 – Memória real cujo valor indica a pressão da câmara interna.

Para criá-los:

Clique com o botão esquerdo do *mouse* em “**Tags**”. Surgirá a opção “**Adiciona Grupo**”. Clique com o botão esquerdo do *mouse* nesta opção e um “**Grupo de Tags**” será criado.



Grupo de Tags: Muitas vezes, será interessante agrupar os *Tags* em grupos, pois isso facilitará a identificação, ou mesmo a edição destes, sendo que o critério para agrupamento será definido pelo bom senso do usuário. Pode-se inclusive alocar o endereço específico do CLP no qual as variáveis serão manipuladas.

Deve-se renomear o grupo dos *Tags* para “**Grupo Exemplo**”. Para isso, clique em “**Grupo01**”, abaixo do item “**Tags**”. Na parte direita da tela clique em “**Nome**” e digite “**Grupo Exemplo**”. Logo abaixo do nome existe a opção “**Endereço Remoto**”, sendo que o número ali digitado deve ser o endereço físico do CLP (para o caso de a IHM estar conectada a uma rede que possui vários CLP’s). Para este exemplo, deixe a opção inicial do “**Endereço Remoto**” que é “**1**”.



Após clicar com o botão esquerdo do *mouse* em “**Grupo01**”, são apresentadas as propriedades do grupo, como na figura abaixo. Em “**Nome**” mude para “**Grupo Exemplo**”

Propriedade	
Nome	Grupo Exemplo
Endereço Remoto	1

1.3.1 Criando um *tag* para M14

Após criar o grupo de *Tags*, clique com o botão esquerdo do *mouse* sobre o item “**Grupo Exemplo**” e em seguida clique com o botão direito do *mouse*. A opção “**Adiciona Tag**” surgirá. Clique com o botão esquerdo do *mouse* sobre esta opção.



Note que o “**Tag01**” será criado. Em seguida, deve-se renomeá-lo para facilitar o trabalho e deixar o programa organizado.



Clique no campo “**Nome**” e digite “**M Tempo Vácuo**”. No campo “**Descrição**” digite “**Memória do tempo de Vácuo**”. A seguir, no tipo de variável selecione “[**M**] Variável Inteira”

No campo “**Endereço da Var.**” digite o endereço da variável sendo programada, isto é, no caso está trabalhando com memória M14, portanto o seu endereço é “**14**”.

Status	Tag
Propriedade	
Nome	M Tempo Vácuo
Descrição	Memória do Tempo de Vácuo
End. de	1
Tipo	[M] Variável Inteira
Endereço da Var.	14
Quantidade	1
Escala	Não

Pronto. O *Tag* para a variável M14 já foi criado. Agora será criado os *Tags* para as outras variáveis.

1.3.2 Criando um *Tag* para D0

Clique com o botão esquerdo do *mouse* sobre o item “**Grupo Exemplo**”. Em seguida, clique com o botão direito do *mouse*. No menu que surge, selecione a opção “**Adiciona Tag**”. O “**Tag01**” será criado. Mude o nome para “**D Pressão Interna**” e a descrição para “**Pressão da Câmara Interna**”. No tipo da variável selecione “[**D**] Variável Real” e no endereço coloque “**0**”.

Neste exemplo, a memória do tipo real (D) lida pelo CLP irá variar entre o valor 0 e 4095. Entretanto, para o usuário esses valores não possuem um significado intuitivo. É aqui onde entra o papel da escala, que irá fazer uma conversão dos valores lidos para o valor de -0,5 a 2 bar, que é a escala real de leitura do sensor. Isto é, internamente a IHM faz uma regra para conversão, sendo que o valor 0 na memória equivale a -0,5 bar e o valor 4095 na memória equivale ao valor de pressão igual a 2 bar.

Em resumo: Com a utilização da escala, o usuário não necessita se preocupar em inserir uma lógica de conversão no programa *ladder*, porque a IHM se encarrega de fazer esta conversão com a utilização da escala.

Para habilitar a escala, selecione a opção **“Sim”** no item **“Escala”**. A seguir, é necessário configurar os valores limites da Unidade Fonte (Valor lido pelo CLP) e os limites para a Unidade de Engenharia (Valor mostrado ao usuário).

Inicialmente é preciso configurar os limites para o valor lido pelo CLP. O item **“Unid. Fonte Inicial”** deverá ser configurado com o menor valor que será lido pelo CLP. O item **“Unid. Fonte Final”** deverá ser configurado com o valor máximo do valor lido pelo CLP.

A seguir será necessário configurar os limites dos valores que serão mostrados ao usuário. Portanto, deve-se configurar o item **“Unid. Eng. Inicial”** com o valor associado à Unid. Fonte Inicial e o item **“Unid. Eng. Final”** com o valor associado à Unid. Fonte Final.

Para o exemplo, configure o item **“Unid. Fonte Inicial”** com o valor **“0”**, o item **“Unidade Fonte Final”** com o valor **“4095”**, o item **“Unid. Eng. Inicial”** com o valor **“-0,5”** e o item **“Unid. Eng. Final”** com o valor **“2”**.

A figura abaixo mostra como ficarão os parâmetros para o *Tag* D0 após a configuração dos itens acima.

Status	Tag
Propriedade	
Nome	D Pressão Interna
Descrição	Pressão da Câmara Interna
End. de	1
Tipo	[D] Variável Real
Endereço da Var.	0
Quantidade	1
Escala	Sim
Unid. Fonte Inicial	0
Unid. Fonte Final	4095
Unid. Eng. Inicial	-0,5
Unid. Eng. Final	2

O processo de criação de *Tags* já foi concluído. O próximo passo é criar os **“campos”** nas diversas telas, cuja definição é mostrada no item a seguir.

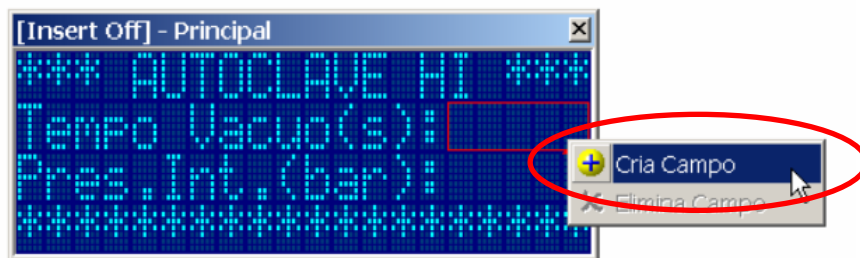
1.4 Campos

O “campo” é o recurso utilizado para visualização e edição das variáveis do CLP. O “campo” criado está relacionado com a tela na qual ele será mostrado. Isto é, o “campo” é o espaço na tela da IHM através do qual será possível visualizar ou alterar o valor da variável relacionada a ele. Como o campo possui uma informação de processo, este deve estar sempre associado a um *“Tag”*.

1.4.1 Criando um campo na tela “Principal” para o Tempo de Vácuo

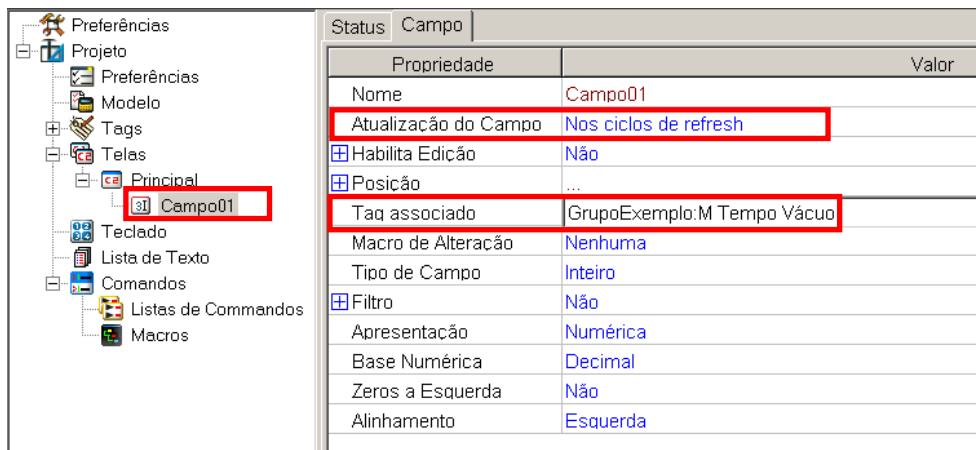
Para criar o campo, clique com o botão esquerdo do *mouse* na tela desejada, que no nosso caso será a tela “Principal”. Quando a janela com a tela se abrir, clique com o botão esquerdo do *mouse* na posição onde o campo será inserido, isto é, no primeiro caractere após o final da linha “Tempo Vácuo(s)”: A seguir, com o botão esquerdo do *mouse* apertado, arraste o cursor até que o campo tenha a largura desejada.

No momento em que a janela possuir a largura desejada, clique com o botão direito do *mouse* sobre o retângulo vermelho e selecione a opção “Cria Campo”.



Agora que foi criada a área onde a variável será mostrada, faz-se necessário atrelar essa variável ao campo. Para isso, clique com o botão esquerdo do *mouse* em “Campo01”, logo abaixo da do item “Principal”, em “Telas”. É possível observar à direita as propriedades do campo criado.

Veja abaixo como ficará configurado o “Campo01” e os valores que deverão ser modificados:



O “Nome” permanece o mesmo, “Campo01” é criado automaticamente.

No item “Atualização do Campo”, pode-se especificar qual será o momento da atualização do valor do campo. Se a opção “somente na abertura da tela” for selecionada, o *refresh* ocorrerá apenas no momento em que a tela for chamada pelo sistema. Se a opção “Nos ciclos de *refresh*” for selecionada, a atualização dos dados ocorrerá a cada ciclo de *refresh*, sendo que este tempo pode ser alterado na configuração da tela. Se deixar os valores padrão, as variáveis serão alteradas a cada dois segundos (2000 ms). Para este exemplo, selecione a opção “Nos Ciclos de *Refresh*”.

O item a seguir é o “**Habilita Edição**”. Se este item for deixado como “**Não**”, a variável será apenas visualizada, sem possibilidade de edição pelo usuário. Se a opção “**Sim**” for selecionada, a variável poderá ser editada pelo usuário através do teclado numérico da IHM. Além disso, é possível também especificar um nível de senha para essa alteração ou mesmo liberar a alteração para qualquer usuário, sem a necessidade de senha. Se no “**Nível de senha**” a opção “**Sem restrição**” for selecionada, a variável pode ser alterada livremente. Por enquanto não será utilizada a edição do campo, logo, deixe a opção padrão “**Não**”.

O próximo passo é associar o *Tag* com o campo que está sendo configurando. No item “**Tag Associado**”, selecione a opção “**Grupo Exemplo: M Tempo Vacuo**”. Isto é, está declarando para a IHM que o operando do CLP a ser manipulado neste campo seja a memória M14.

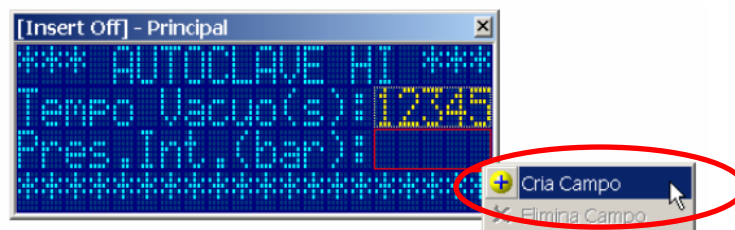
No item “**Macro de Alteração**” é possível especificar uma macro de comandos para ser executada no momento em que a variável for alterada. Neste caso não será utilizada nenhuma macro de alteração. Para maiores informações sobre macros, consulte as seções seguintes.

No “**Tipo de Campo**”, pode-se especificar o tipo de dado que será mostrado. Sendo que as opções são: “**Inteiro**”, “**Real**” e “**Oculto**”. No caso, deixe a opção padrão “**Inteiro**”.

Existem outras opções para configurar o campo, entretanto não é necessário alterar estas outras opções neste momento.

1.4.2 Criando um campo na tela “Principal” para a Pressão Interna

Para criar o campo, clique com o botão esquerdo do *mouse* na tela desejada, que no caso será a tela “**Principal**”. Na linha do texto “**Pres.Int.(bar):**”, clique com o botão esquerdo do *mouse* e arraste-o com o botão esquerdo pressionado. Quando a caixa formada estiver com a largura desejada, solte o botão esquerdo e com o cursor posicionado sobre o campo, clique no botão direito do *mouse* e selecione a opção “**Cria Campo**” no menu que se abrirá.

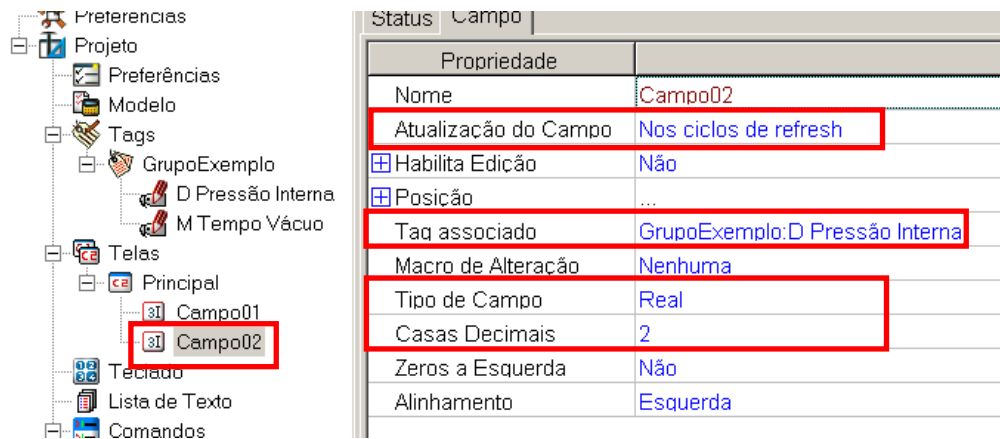


Em seguida, note que o “**Campo02**” aparecerá logo abaixo do item “**Principal**”, em “**Telas**”. Para configurar este campo, clique com o botão esquerdo do *mouse* sobre “**Campo02**”.



Seguindo os passos semelhantes para o item anterior, configure o campo criado para a memória D0 na tela “**Principal**”. Para isso, utilize a figura abaixo como modelo para a configuração. Atente para o fato de que desta vez

o “**Tag Associado**” é o *Tag* da memória D0, que é “**Grupo Exemplo: D Pressão Interna**” . Os retângulos em vermelho mostram os locais onde os dados devem ser alterados.



The image shows two parts of the MMI800 software interface. On the left is a project tree with a red box around 'Campo02'. On the right is a table with the following data:

Propriedade	
Nome	Campo02
Atualização do Campo	Nos ciclos de refresh
Habilita Edição	Não
Posição	...
Tag associado	GrupoExemplo:D Pressão Interna
Macro de Alteração	Nenhuma
Tipo de Campo	Real
Casas Decimais	2
Zeros a Esquerda	Não
Alinhamento	Esquerda

Note que o “**Tipo de Campo**” deve ser configurado como “**Real**”. Quando esta opção é escolhida, imediatamente abre-se na tela a opção “**Casas Decimais**”, onde é possível escolher quantas casas após a vírgula serão mostradas. Para este exemplo, determine que sejam mostradas duas casas após a vírgula.

O próximo passo é configurar os comandos, cuja definição é explicada no item a seguir.

1.5 Comandos

O que é um Comando?

O comando é o ente através do qual uma ação é executada pela MMI800. Isto é, não basta os programas possuírem apenas as telas e seus respectivos “*Tags*” e “campos”, sendo que comandos são necessários para efetuar operações sobre estes elementos.

Um exemplo básico de comando é o “**Open Screen**”, que serve para carregar uma tela cujo número é determinado no seu argumento. Para uma descrição das funções disponíveis consulte a nota de aplicação **ENA.00047**

A seguir será configurado alguns comandos para este programa de exemplo.

1.5.1 Configurando o comando “**Open Screen**”

O comando “**Open Screen**” serve para abrir telas. Isto é, sua ação é executar a abertura de uma tela na IHM.

Ao configurar um comando com “**Open Screen**”, é necessário fornecer o “**Identificador da Tela**”, isto é, o número da tela que será chamada quando o comando for executado. Este valor é mostrado no item “**Identificador da Tela**”, mostrado na parte de definição e especificação das telas.

1.5.1.1 **Open Screen** para a tela “Principal”

Para iniciar a configuração de um comando, o usuário deverá inicialmente clicar com o botão esquerdo do *mouse* sobre o sinal “+”, ao lado do item “**Comandos**”.



Os níveis abaixo de “**Comandos**” serão abertos.



A seguir, clique com o botão direito do *mouse* sobre o item “**Adiciona Comando**”:



Em seguida o “**Comando01**” será criado logo abaixo de “**Listas de Comandos**”. Para configurá-lo, clique com o botão direito sobre “**Comando01**”. Veja que a configuração do comando é efetuada na parte direita da tela.

O primeiro passo será renomeá-lo. **Uma boa dica é nomear os comandos utilizando rótulos que lembrem a operação que será efetuada e o elemento que é referenciado a este comando.** À medida que os programas forem aumentando de tamanho, esta operação de utilizar nomes familiares aos comandos evita que o usuário tenha o trabalho de clicar sobre cada um dos comandos para que ele saiba sua respectiva função.

Renomeando o “**Comando 01**” para “**Abre Tela Principal**”. Em seguida, no campo “**Função**”, clique com o botão esquerdo do *mouse* e escolha a função “**[01]:OpenScreen**”

No campo “**Tela**”, clique com o botão esquerdo do *mouse* e selecione a tela que será aberta. Para este exemplo, selecione a “**Principal**”.

O item “**Campo**” é utilizado somente quando há campos na tela que será aberta. O número contido neste item será o número do campo que será o primeiro a receber o cursor em caso de pressionamento da tecla “**Enter**” da IHM para a alteração de um valor. No exemplo, deixe o valor padrão “**1**”.

Após estas configurações, o comando “**Abre Tela Principal**” estará configurado do seguinte modo:

Propriedade	
Nome	Abre Tela Principal
Função	[01]: OpenScreen
Tela	Principal
Campo	1

O comando já foi configurado, e o próximo passo é configurar a macro, cuja explicação vem a seguir.

1.6 Macros

O que é uma “Macro”?

A “Macro” é o recurso por meio do qual um comando ou um conjunto de comandos pode ser executado de uma só vez. Não é possível executar algum comando sem que o mesmo esteja associado a uma macro.

Uma macro pode conter apenas um comando ou vários comandos. Caso ela contenha vários comandos, todos eles serão executados ao mesmo tempo quando a macro for executada.

Um exemplo interessante para ilustrar a funcionalidade de uma macro é o seguinte: Pode-se criar uma macro que tenha um comando para chamar uma tela (**open screen**), acender um *led* da tecla de função (**program led**), soar o sinalizador sonoro com bips (**program buzzer**) e acender o *backlight* (**program backlight**). Ao final da criação desta macro, pode-se associá-la a uma tecla de função da IHM, e toda vez que esta tecla for pressionada, pode-se chamar uma tela, acender o *led* da tecla, acender o *display* e ativar os bips do sinalizador sonoro.

Para esta primeira macro, será criado o comando “**Open Screen**”, e ela servirá somente para abrir a tela “**Principal**”.

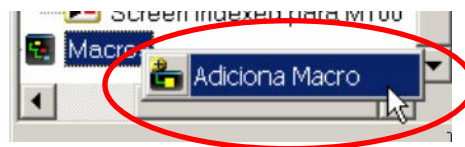
A seguir estão descritos os procedimentos para a criação de cada macro. Veja que a dica de nomear macros com nomes familiares às funções que ela executa também é válida, uma vez que evita ao usuário ter que verificar o conteúdo de cada macro para saber quais funções ela executa.

1.6.1 Criando Macro Abre Tela Principal

Esta macro deve executar a seguinte função:

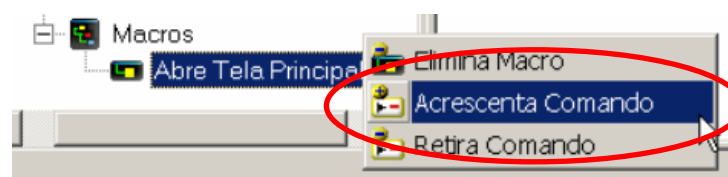
- Executar comando “**Abre Tela Principal**”

Para criar uma macro, clique com o botão esquerdo do *mouse* sobre “**Macros**” e em seguida clique com o botão direito do *mouse*. No menu que aparecerá, selecione a opção “**Adiciona Macro**”.



Em seguida, você verá que uma macro chamada “**Macro01**” foi criada. Mude seu nome para “**Abre Tela Principal**”.

O próximo passo é adicionar o comando que será executado quando a macro for ativada. Para adicionar um comando, clique com o botão esquerdo do *mouse* sobre a macro “**Abre Tela Principal**”. A seguir, no menu que se abrirá, selecione a opção “**Acrescenta Comando**”, conforme ilustra a figura a seguir:



À direita, você verá que um item “**Comando01**” será criado na parte direita da tela. Neste campo selecione o comando “**Abre Tela Principal**”

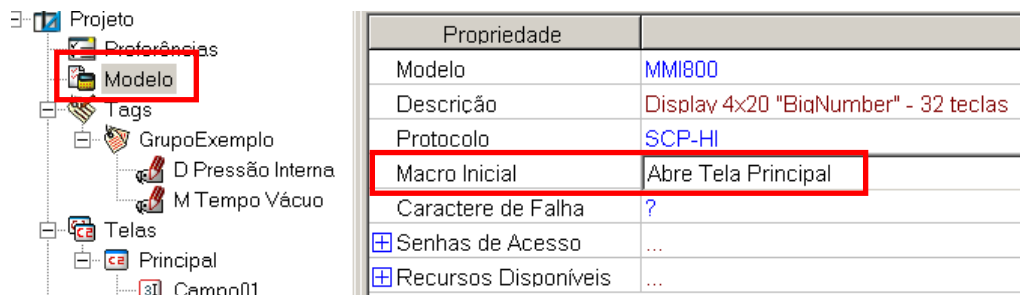
Após adicionar o comando, a macro “**Abre Tela Principal**” terá a seguinte aparência:

Propriedade	
Nome	Abre Tela Principal
Identif. da Macro	0
Número de	1
Comando 01	Abre Tela Principal

1.7 Ajustes Finais

O programa de exemplo já está quase pronto. Para finalizar sua edição, é necessário determinar qual será a macro inicial, isto é, qual será a macro que será executada no momento em que a IHM for inicializada. Como só possuímos uma macro, que é a que chama a tela “**Principal**”, ela será definida como a macro inicial.

Para especificar qual vai ser a macro inicial, clique no item “**Projeto**” e depois em “**Modelo**”. No lado direito da tela clique em “**Macro Inicial**” e especifique a macro “**Abre Tela Principal**”. Após este ajuste, o item “**Modelo**” terá o seguinte aspecto:



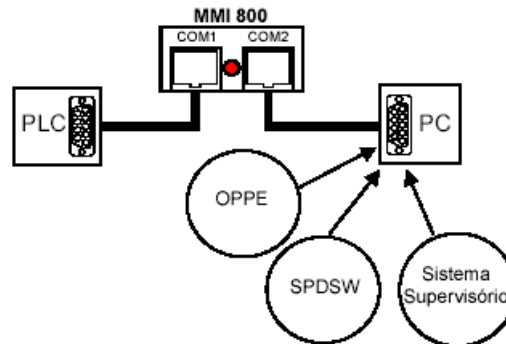
Propriedade	
Modelo	MMI800
Descrição	Display 4x20 "BigNumber" - 32 teclas
Protocolo	SCP-HI
Macro Inicial	Abre Tela Principal
Caractere de Falha	?
Senhas de Acesso	...
Recursos Disponíveis	...

Em seguida, deve-se efetuar a carga do programa. A seção a seguir mostra como deve ser este procedimento.

1.8 Carga do Programa

A partir de agora, basta carregar o *software* na IHM e executá-lo. Para carregar o *software*, certifique-se de o cabo esteja ligado entre o PC e a IHM.

A figura abaixo mostra como deve ser efetuada a conexão entre a IHM, o CLP e o PC:

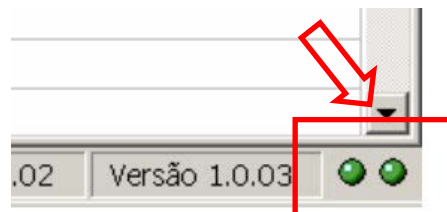


Observe que o **PC** deverá estar ligado na porta **COM2** da IHM, sendo que esta conexão deverá ser efetuada por meio do cabo de comunicação **PC-IHM**.

Além disso, o **CLP** deverá estar conectado na porta de comunicação **COM1** da IHM, sendo que o cabo a ser utilizado é o cabo de comunicação **PLC-IHM**. Para esta conexão entre o CLP e a IHM qualquer uma das portas de comunicação do CLP que estiver configurada para operação com o protocolo SCP-HI em modo escravo poderá ser utilizada. A porta da IHM a ser utilizada deverá ser obrigatoriamente a COM1.

Para verificar se a conexão entre o PC e a IHM está estabelecida, *verifique a cor dos dois leds localizados no canto inferior direito da tela, ao lado do número da versão do software OPPE.*

O primeiro *led* indica que o *driver* de comunicação está instalado, e o segundo *led* indica que a conexão entre o computador e entre a IHM foi estabelecida.



O próximo passo é carregar o programa na IHM. Para que isto ocorra, clique no botão "**Carrega**" localizado na barra de botões do OPPE:



Uma barra de progressão vai indicar o estado de carga do programa, isto é, quanto do programa já foi carregado na IHM:



Quando o programa estiver carregado, execute-o e veja seu funcionamento. Isto é, verifique a carga da tela principal e varie a entrada analógica no CLP para observar a variação do valor mostrado na IHM com a utilização da escala.

Observação:

Para que os programas de exemplo funcionem corretamente, é necessário que o CLP utilizado em conjunto com a IHM esteja com a conexão estabelecida e também deve estar carregado por meio do SPDSW com o seguinte programa.

- *Autoclave_OPPE_MCI.PJT* (Caso o CLP seja da série **MCI02-QC**).
- *Autoclave_OPPE_ZAP.PJT* (Caso o CLP seja da série **ZAP**).

Estes dois programas do SPDSW são fornecidos em anexo a esta nota de aplicação, juntamente com os arquivos do OPPE referente a cada capítulo.

Observação: Neste ponto, salve uma cópia do programa gerado pelo OPPE como “Capítulo1-EX.OPJ”. Em seguida, continue utilizando o arquivo original para a edição das funções que serão incorporadas neste capítulo.

Capítulo 2

Agora que já foi construída uma tela básica e apresentado alguns conceitos iniciais para construir um primeiro programa, deve-se incrementar o programa existente com a adição de novos recursos.

Continue a utilizar como base a autoclave de exemplo, que neste capítulo receberá novos recursos. Os recursos deste capítulo serão:

- Criação de uma nova tela, onde será programado o tempo de vácuo.
- Utilização das teclas de função para chaveamento entre as diferentes telas.

Comece a configurar os novos recursos para este capítulo.

2.1 Criação da tela para programação do tempo de vácuo

Deve ser criada uma tela onde será possível programar o tempo de vácuo.

Utilizando o procedimento para criação de telas, já efetuado, crie uma nova tela. Em seguida, digite o seguinte texto:



Na tela que foi criada, mude seu “Nome” para “Tempo Vácuo”, a descrição para “Programação do Tempo de Vácuo”. Mude também o “Tempo de Refresh” para o valor “500”.

Após a configuração da tela, seus parâmetros deverão estar como no modelo abaixo:

Status	Tela
Propriedade	
Nome	TempoVácuo
Descrição	Programação do Tempo de Vácuo
Identificador da Tela	1
Tempo de Refresh	500
Macros	...
Número de	0

O próximo passo é configurar o campo, onde poderá ser programado o valor do tempo de vácuo.

2.2 Criação do campo para a tela de tempo de vácuo

Agora que a tela já foi criada, é necessário especificar o campo onde o tempo de vácuo será programado. Para criar um campo, utilize o procedimento já feito anteriormente, isto é, com o botão esquerdo do *mouse* crie uma caixa e em seguida pressione o botão direito do *mouse* sobre esta caixa, selecionando a opção “**Cria Campo**”.

Note que o item “**Atualização do Campo**” deverá ser selecionado como “**Nos ciclos de refresh**” e o item “**Habilita Edição**” deverá estar marcado como “**Sim**”. Após este passo, é necessário especificar o “**Tag Associado**” para este campo. Neste caso, selecione o *Tag* “**Grupo Exemplo: M Tempo Vácuo**”

Após estas configurações, o campo deverá ter a seguinte aparência:

Status	Campo
Propriedade	
Nome	Campo01
Atualização do Campo	Nos ciclos de refresh
<input checked="" type="checkbox"/> Habilita Edição	Sim
<input type="checkbox"/> Nível de Senha	Sem restrição
<input checked="" type="checkbox"/> Posição	
Tag associado	GrupoExemplo:M Tempo Vácuo
Macro de Alteração	Nenhuma
Tipo de Campo	Inteiro
<input checked="" type="checkbox"/> Filtro	Não
Apresentação	Numérica
Base Numérica	Decimal
Zeros a Esquerda	Não
Alinhamento	Esquerda

2.3 Criação do comando da abertura da tela de programação

É necessário criar um comando “**Open Screen**” para a abertura da tela. Este comando irá solicitar ao CLP que carregue a tela para a abertura da tela para a programação do tempo de vácuo.

No item “**Comandos**”, clique com o botão direito do *mouse* e selecione “**Adiciona Comando**”. Após a adição do comando, renomeie a tela para “**Abre Tela Tempo de Vácuo**” e em seguida selecione o comando “[01]: **OpenScreen**”. Em seguida, selecione a tela “**Tempo de Vácuo**”.

Após a configuração destes parâmetros, o comando deverá ter a seguinte aparência:

Status		Lista de Commandos
Propriedade		
Nome	Abre Tela Tempo de Vácuo	
Função	[01]: OpenScreen	
Tela	Tempo de Vacuo	
Campo		

2.4 Criação de macro para tela de tempo de vácuo

Configuraremos agora a macro que irá executar o comando “**Abre Tela Tempo de Vácuo**”

Conforme já efetuado anteriormente, crie uma nova macro utilizando o botão direito do *mouse*. Após a criação da macro, nomeie-a como “**Abre Tela Tempo de Vácuo**” e em seguida adicione o comando “**Abre Tela Tempo de Vácuo**”.

Após a configuração destes parâmetros, eles deverão ter a seguinte aparência:

Status		Macro
Propriedade		
Nome	Abre Tela Tempo de Vácuo	
Identif. da Macro	1	
Número de	1	
Comando 01	Abre Tela Tempo de Vácuo	

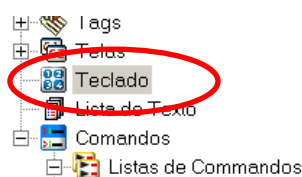
2.5 Associação das teclas de função da IHM

As teclas da IHM permitem ao usuário a execução de macros. Isto é, para ser executado algum comando quando uma tecla da IHM for pressionada, é necessário associar uma macro para esta tecla.

O *software* OPPE permite que uma macro seja executada no momento em que uma tecla for pressionada e outra macro seja executada no momento em que a tecla for liberada.

2.5.1 Programando a Tecla F1

Para a tecla F1 vamos associar como “**Macro tecla pressionada**” a macro “**Abre Tela Principal**”. O primeiro passo é clicar no item “**Teclado**”:



As teclas da IHM permitem ao usuário a execução de macros, isto é, para ser executado algum comando quando uma tecla da IHM for pressionada, é necessário associar uma macro para esta tecla. Em seguida, clique no sinal “+”, ao lado de “**Tecla F1**”:

+	Tecla F1: Nível de acesso	Sem restrição
+	Tecla F2: Nível de acesso	Sem restrição

A partir daí, selecione a macro “**Abre Tela Principal**” para o item “**Macro tecla pressionada**”. Após a configuração da macro, a tecla F1 terá o seguinte aspecto:

+	Tecla F1: Nível de acesso	Sem restrição
+	Macro tecla pressionada	Abre Tela Principal

2.5.2 Programando a Tecla F2

Para a tecla F2 vamos associar como “**Macro tecla pressionada**” a macro “**Abre Tela Tempo de Vácuo**”. O primeiro passo é clicar no item “**Teclado**”:



Em seguida, clique no sinal “+”, ao lado de “**Tecla F2**”:

+	Tecla F2: Nível de acesso	Sem restrição
---	---------------------------	---------------

A partir daí, selecione a macro “**Abre Tela Tempo de Vácuo**” para o item “**Macro tecla pressionada**”. Após a configuração da macro, a tecla F2 terá o seguinte aspecto:

+	Tecla F2: Nível de acesso	Sem restrição
+	Macro tecla pressionada	Abre Tela Tempo de Vácuo
+	Macro tecla liberada	Nenhuma

2.6 Carga do programa

O próximo passo é efetuar a carga do programa. Carregue o programa e pressione as teclas de função. Pressione as teclas de função F1 e F2 e veja como as telas são alternadas de acordo com a tecla que é pressionada.

Observação: Neste ponto, salve uma cópia do programa gerado pelo OPPE como “Capítulo2-EX.OPJ”. Em seguida, continue utilizando o arquivo original para a edição das funções que serão incorporadas neste capítulo.

Capítulo 3

O objetivo deste capítulo é utilizar a IHM para ativar o motor da bomba de vácuo da autoclave por meio de teclas de função e também através do recurso conhecido como “**Lista de texto**”.

Para que isto seja possível, serão apresentados mais dois novos recursos:

- Utilização do comando “**Direct Command**”
- Utilização da lista de texto para supervisionar e alterar o estado de uma variável

3.1 Tag para o operando R100

Para que seja possível a alteração do estado de um motor, é necessário que seja criado um *Tag* referente ao elemento que controla este motor no programa *ladder*. Para o programa de exemplo da autoclave, o operando que controla o motor é o R100.

Como já feito nas seções anteriores, crie um *Tag* para o operando “R100”. Nomeie-o para “**R Motor da Bomba de Vácuo**” e coloque sua descrição como “**Tag do motor da bomba de vácuo**”.

Após estes passos, o *Tag* para R100 deverá estar configurado da seguinte maneira:

Status	Tag
Propriedade	
Nome	R Motor Bomba Vácuo
Descrição	Tag do Motor da Bomba de Vácuo
End. de	1
Tipo	[R] Variável Lógica
Endereço da Var.	100
Quantidade	1

3.2 Comandos Liga/Desliga Bomba de Vácuo

Agora que o *Tag* já foi criado, o próximo passo é criar os comandos que executarão as ações de ligar e desligar o motor da bomba de vácuo.

O comando a ser usado para ligar e desligar o motor será o “**SendDirectData**”. Este comando necessita de dois parâmetros: Um deles é o “**Tag Destino**” e o outro é o “**Valor**”. Quando o comando for executado, o “**Valor**” será enviado para o “**Tag Destino**”.

O comando será o mesmo para as duas operações, entretanto, no comando para ligar o motor o valor a ser enviado é o “**1**” e no comando para desligar o valor a ser enviado será o “**0**”.

3.2.1 Cria comando Liga Bomba de Vácuo

Criando um comando para ligar o motor. Crie um novo comando como já feito anteriormente, nomeie-o para “**Liga Bomba de Vácuo**”, selecione a função “[04]: **SendDirectData**” e escolha como “**Tag Destino**” o “**Grupo Exemplo: R Motor Bomba Vácuo**”. Por fim, selecione o valor a ser enviado como “**1**”.

Após a configuração, o comando deverá ter o seguinte aspecto:

Status		Lista de Commandos
Propriedade		
Nome		Liga Bomba de Vácuo
Função		[04]: SendDirectData
Tag Destino		GrupoExemplo:R Motor Bomba Vácuo
Valor		1

3.2.2 Cria comando Desliga Bomba de Vácuo

Configurando o comando para desligar o motor. Deve-se criar este comando e configurá-lo da mesma maneira que o item anterior, entretanto, este será nomeado como “**Desliga Bomba de Vácuo**” e seu “**Valor**” será “**0**”.

Veja como ele ficará configurado:

Status		Lista de Commandos
Propriedade		
Nome		Desliga Bomba de Vácuo
Função		[04]: SendDirectData
Tag Destino		GrupoExemplo:R Motor Bomba Vácuo
Valor		0

3.3 Macros

O passo seguinte é criar as macros que executarão os comandos de ligar e desligar o motor, sendo que serão criadas duas macros, uma que irá ligar e outra para desligar o motor.

3.3.1 Cria macro Liga Bomba de Vácuo

Esta macro será a responsável por ligar o motor da bomba de vácuo. Como já feito nas seções anteriores, crie uma nova macro e nomeie-a como “**Liga Bomba de Vácuo**”. Em seguida adicione o comando “**Liga Bomba de Vácuo**”.

Veja como ficará configurada esta macro:

Status	Macro
Propriedade	
Nome	Liga Bomba de Vácuo
Identif. da Macro	2
Número de	1
Comando 01	Liga Bomba de Vácuo

3.3.2 Cria macro Desliga Bomba de Vácuo

Esta macro será a responsável por desligar o motor. Como já feito no item anterior, crie uma nova macro e nomeie-a como “**Desliga Bomba de Vácuo**”. Em seguida adicione o comando “**Desliga Bomba de Vácuo**”.

Veja como ficará configurada esta macro:

Status	Macro
Propriedade	
Nome	Desliga Bomba de Vácuo
Identif. da Macro	3
Número de	1
Comando 01	Desliga Bomba de Vácuo

3.4 Teclas da IHM

O objetivo é ligar e desligar o motor da bomba de vácuo, sendo que este deverá funcionar enquanto a tecla de função F3 estiver pressionada e deverá parar quando a mesma for liberada.

3.4.1 Associação das teclas

A tecla F3 será utilizada para ligar e desligar o motor da bomba de vácuo. Sendo que as macros “**Liga Bomba de Vácuo**” e “**Desliga Bomba de Vácuo**” serão utilizadas para tal.

Como já feito, selecione a tecla de programação F3 e selecione a macro “**Liga Bomba de Vácuo**” para o item “**Macro Tecla Pressionada**” e selecione a macro “**Desliga Bomba de Vácuo**” para o item “**Macro Tecla Liberada**”.

Veja a seguir como ficará a configuração da tecla F3:

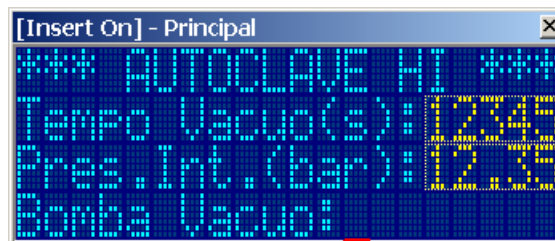
<input type="checkbox"/> Tecla F3: Nível de acesso	Sem restrição
<input type="checkbox"/> Macro tecla pressionada	Liga Bomba de Vácuo
<input type="checkbox"/> Macro tecla liberada	Desliga Bomba de Vácuo

3.5 Alterando a tela “Principal”

Além do controle do motor da bomba de vácuo por meio da tecla F3, deseja-se também visualizar e alterar seu *status* por meio da tela “Principal”.

3.5.1 Alteração da tela

A tela “Principal” já está criada, mas será necessário alterá-la para incluir o *status* do motor. Vá em “Telas” e selecione a tela “Principal”. Altere a terceira linha da tela baseando-se no modelo a seguir.



3.5.2 Criando um campo para o motor

O próximo passo é criar um campo na tela “Principal” onde possa ser efetuada a visualização do *status* do motor. Como já feito anteriormente, crie um campo logo após a palavra “Vácuo:”. Configure o campo segundo o modelo abaixo, atentando para que o “Tag Associado” seja “Grupo Exemplo: R Motor Bomba Vácuo”, o “Habilita Edição” esteja como “Sim” e que a “Apresentação” esteja como “Lista de Texto”.

Veja como deverá ser configurado este “Campo”:

Status	Campo
Propriedade	
Nome	Campo03
Atualização do Campo	Nos ciclos de refresh
<input checked="" type="checkbox"/> Habilita Edição	Sim
<input type="checkbox"/> Nível de Senha	Sem restrição
<input checked="" type="checkbox"/> Posição	
Tag associado	GrupoExemplo:R Motor Bomba Vácuo
Macro de Alteração	Nenhuma
Tipo de Campo	Inteiro
<input checked="" type="checkbox"/> Filtro	Não
Apresentação	Lista de Texto
Alinhamento	Esquerda
Lista de Texto	

3.6 Lista de texto

A novidade deste capítulo é a lista de texto. Seu funcionamento está baseado em textos que estão associados a cada valor da variável em questão.

Isto é, suponha que se tem um operando do tipo R. Pode-se associar um texto para quando o operando possuir o valor 1 e outro texto quando o operando possuir o valor zero. Um bom exemplo é: Associar o texto “Ligado” quando o operando possuir o valor “1” e “Desligado” quando o mesmo possuir o valor “0”.

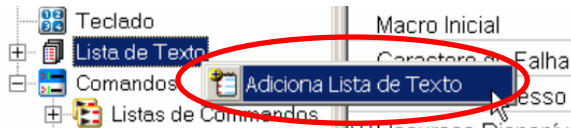
A seguir, será criada uma lista de texto para utilizar neste programa.

3.6.1 Criando a lista de texto

Para criar uma lista de texto, clique em “Lista de Texto”, logo abaixo do item “Teclado”.



Em seguida, clique com o botão direito do *mouse* sobre “Lista de Texto” e selecione a opção “Adiciona Lista de Texto”.



Na lista que acabou de ser criada, modifique seu nome para “**Lista Motor da Bomba**”. Na sequência, digite o “**Texto Padrão**” como “**Nao sei**”. Isto é, esta será a mensagem mostrada caso o operando tenha algum valor diferente dos que já foram programados na lista.

Em seguida, clique novamente com o botão direito do *mouse* e selecione a opção “**Adiciona Novo Texto**”. No “**Texto Nro:01**”, especifique o texto “**Ligada**” e como “**Valor associado**” selecione o valor “**1**”. Repita este procedimento, desta vez criando o “**Texto Nro:02**” e especificando o texto “**Deslig.**”, tendo como valor associado “**0**”.

Veja abaixo o modelo da lista de texto para a “**Lista Motor da Bomba**”:

Status		Lista de Textos
Propriedade		
Nome		Lista Motor da Bomba
Nro. de Textos		2
Texto Padrão		Nao sei
<input type="checkbox"/> Texto Nro: 01		Liqada
↳ Valor associado		1
<input type="checkbox"/> Texto Nro: 02		Desliq.
↳ Valor associado		0

3.6.2 Associando a lista de texto a um campo

Após a criação da lista de texto, é necessário associá-la ao campo na qual ela será mostrada. Para isso, volte à tela “Principal” e selecione o “Campo03”. No item “Apresentação”, selecione o modo “Lista de Texto” e em seguida, no item “Lista de Texto”, selecione a opção “Lista motor bomba”.

Veja abaixo como deverá ser configurado o “Campo03”:

Status		Campo
Propriedade		
Nome		Campo03
Atualização do Campo		Nos ciclos de refresh
<input checked="" type="checkbox"/> Habilita Edição		Sim
↳ Nível de Senha		Sem restrição
<input checked="" type="checkbox"/> Posição		...
Taq associado		GrupoExemplo:R Motor Bomba Vácuo
Macro de Alteração		Nenhuma
Tipo de Campo		Inteiro
<input checked="" type="checkbox"/> Filtro		Não
Apresentação		Lista de Texto
Alinhamento		Esquerda
Lista de Texto		Lista Motor da Bomba

3.7 Carga do programa

O próximo passo é efetuar a carga do programa. Carregue o programa utilizando o procedimento já efetuado e verifique na tela de supervisão da bomba de vácuo as mensagens programadas. Liga e desliga a bomba de vácuo através da tecla de função F3.

Observação: Neste ponto, salve uma cópia do programa gerado pelo OPPE como “Capítulo3-EX.OPJ”. Em seguida, continue utilizando o arquivo original para a edição das funções que serão incorporadas neste capítulo.

Capítulo 4

O objetivo deste capítulo é incorporar um recurso simples para geração de alarmes e também o reconhecimento deste. Serão apresentados os recursos a seguir:

- Utilização do “**campo oculto**” para detecção da ocorrência de uma falha.
- Utilização do comando “**ProgramBuzzer**” para utilização do *buzzer*.
- Utilização do comando “**ProgramLed**” para ativação do *led* da tecla de função.
- Utilização do comando “**Program Backlight**” para alteração do estado do *backlight* da IHM.

4.1 Telas

Será criada uma tela que informará ao usuário a geração de um alarme.

4.1.1 Criando tela de alarme

Utilizando o procedimento para criação de telas, já efetuado, crie uma nova tela. Em seguida, digite o seguinte texto:



Na tela que foi criada, mude seu “**Nome**” para “**Alarme**”, a descrição para “**Tela de Alarme**”. Mude também o “**Tempo de Refresh**” para o valor “**500**”. Após a configuração da tela, seus parâmetros deverão estar como no modelo abaixo:

Propriedade	
Nome	Alarme
Descrição	Tela de alarme
Identificador da Tela	1
Tempo de Refresh (ms)	500
Macros	...
Número de Campos	0

4.2 Tag

No programa de exemplo da autoclave, o operando R14 é o responsável pela sinalização de alarme, isto é, quando ocorrer um pulso em R14, isto significa que o sistema entrou em uma condição de falha.

Criando Tag para R14

Criando um *Tag* para o operando R14. Como já feito anteriormente, crie um novo *Tag* utilizando o procedimento já conhecido. Após a criação do *Tag*, mude seu “Nome” para “**R Falha no processo**”, sua “Descrição” para “**Operando que verifica a falha no processo**”, seu “Tipo” para “[R] Variável Lógica” e o “Endereço da Var.” para “14”, conforme o modelo a seguir:

Status	Tag
Propriedade	
Nome	R Falha no processo
Descrição	Operando que verifica a falha no processo
End. de	1
Tipo	[R] Variável Lógica
Endereço da Var.	14
Quantidade	1

4.3 Campos

Neste capítulo será utilizado o “**campo oculto**” para a detecção de uma condição de falha.

O que é um “Campo Oculto” ?

O “**campo oculto**” é um campo semelhante aos outros campos existentes, diferenciando-se pela capacidade de executar uma macro quando o *Tag* associado a este campo muda de valor.

Baseado nesta funcionalidade será utilizado um campo oculto em conjunto com o operando R14, isto é, quando houver um pulso de transição em R14, será programada a execução de uma macro, chamada “**Ativa Alarme**” que executará os comandos para a sinalização de alarme.

O campo oculto possui uma abrangência de detecção somente na tela local, isto é, ele só consegue detectar a alteração do *Tag* associado a ele na tela em que ele foi configurado. Para que este tenha uma abrangência em todas as telas, é necessário criar instâncias dele nas outras telas.

É isso que será feito agora: configurando um “**campo oculto**” para as duas telas que já existem.

Criação do campo oculto nas telas

É necessário que voltar às duas telas já existentes para criar o “campo oculto”. Primeiramente, voltar à tela “**Principal**” e criar um campo oculto nesta tela.

Quando a tela “**Principal**” se abrir, clique com o botão esquerdo do *mouse* em qualquer posição e efetue o procedimento para a criação de um campo. Após a criação do campo, configure-o segundo o modelo abaixo,

atentando para que o “**Tag associado**” seja “**Grupo Exemplo: R Falha no processo**”. Não se esqueça de especificar o “**Tipo de Campo**” como “**Oculto**”.

Status	Campo
Propriedade	
Nome	Campos04
Atualização do Campo	Nos ciclos de refresh
Posição	...
Tag associado	GrupoExemplo:R Falha no processo
Macro de Alteração	Nenhuma
Tipo de Campo	Oculto

Vá também à tela “**Tempo de Vácuo**” e repita o procedimento para a criação do campo. O procedimento é o mesmo que foi efetuado no item acima. Siga a figura abaixo como exemplo:

Status	Campo
Propriedade	
Nome	Campos02
Atualização do Campo	Nos ciclos de refresh
Posição	...
Tag associado	GrupoExemplo:R Falha no processo
Macro de Alteração	Nenhuma
Tipo de Campo	Oculto

4.4 Comandos

O próximo passo é criar os comandos para a sinalização de alarme, utilização dos *leds* das teclas de função e iluminação do *display* (*backlight*).

4.4.1 Configuração do comando “**OpenScreen**” para a tela de alarme

Configurando agora o comando “**OpenScreen**”, que irá abrir a tela de alarme. Como já feito anteriormente, crie um novo comando. Em seguida, mude seu “**Nome**” para “**Abre Tela de Alarme**”, escolha a “**Função**” como “[01]: **OpenScreen**” e configure o item “**Tela**” como “**Alarme**”
Veja o modelo abaixo como referência:

Status		Lista de Commandos	
Propriedade			
Nome		Abre Tela de Alarme	
Função		[01]: OpenScreen	
Tela		Alarme	
Campo			

4.4.2 Configuração do comando “ProgramBuzzer”

Configurando agora o comando “**ProgramBuzzer**”, que irá ativar o sinalizador sonoro.

Este comando permite ao usuário programar o funcionamento do sinalizador sonoro. O comando permite três opções de funcionamento do sinalizador: **Ligado**, **Desligado** ou **Bips**, sendo que nesta última opção é possível programar o tempo de cada bip.

Neste exemplo, será configurado o comando “**ProgramBuzzer**” para trabalhar com os bips.

Como já feito anteriormente, crie um novo comando. Em seguida, mude seu “**Nome**” para “**Sonorização Alarme**”, escolha a “**Função**” como “[11]: **ProgramBuzzer**”. Configure o “**Estado**” como “**Bip**” e depois selecione o “**Número de Bips**” como “**5**” e por fim digite “**200**” para o “**Tempo de Bip**”

Veja o modelo abaixo como referência:

Status		Lista de Commandos	
Propriedade			
Nome		Sonorização Alarme	
Função		[11]: ProgramBuzzer	
Estado		Bip	
Número de Bips		5	
Tempo de Bip		200	

4.4.3 Configuração do comando “ProgramLed”

O comando “**ProgramLed**” é o comando responsável pela programação dos *leds* das teclas de função. O comando permite também a programação individual do *led* de cada tecla de função.

Seu modo de funcionamento é semelhante ao comando “**ProgramBuzzer**”, sendo que é possível ligar, desligar ou especificar uma quantidade de piscadas do *led*.

Para este exemplo, vamos utilizar 10 piscadas do *led* da tecla F4, sendo que cada piscada terá a duração de 10 segundos.

Como já feito anteriormente, crie um novo comando. Em seguida, mude seu “**Nome**” para “**Pisca Leds tecla F4**”, escolha a “**Função**” como “[12]: **ProgramLed**”. Configure o “**Estado**” como “**Pisca**” e depois selecione o “**Nro. de Piscadas**” como “**10**” e por fim digite “**200**” para o “**Tempo piscada**”.

Veja abaixo como deverá ser configurado o comando:

Propriedade	
Nome	Pisca Leds tecla F4
Função	[12]: ProgramLed
Nro. do Led	[03]: Led da Tecla F4
Estado	Pisca
Nro. de Piscadas	10
Tempo piscada	200

4.4.4 Configuração do comando “ProgramBacklight”

Configurando a ativação e desativação do *backlight*.

O comando “**ProgramBacklight**” é o responsável por configurar o funcionamento da iluminação do *display* (*backlight*). Suas opções de funcionamento são: ligado, desligado e autodesliga, isto é, após um tempo determinado a iluminação será desligada.

A seguir, será criado dois comandos, um que irá ligar e o outro que irá desligar o *backlight*.

Como já feito anteriormente, crie um novo comando. Em seguida, renomeie-o como “**Liga o Backlight**”, escolha a “**Função**” como “[10]: ProgramBacklight” e selecione seu estado como “**Ligado**”.

Utilize como referência a figura abaixo:

Propriedade	
Nome	Liga o Backlight
Função	[10]: ProgramBacklight
Estado	Ligado

Em seguida, crie mais um comando, que desta vez terá o nome de “**Desliga o Backlight**”. Escolha a “**Função**” como “[10]: Program Backlight” e selecione seu estado como “**Desligado**”, tal como foi feito na figura abaixo:

Propriedade	
Nome	Desliga o Backlight
Função	[10]: ProgramBacklight
Estado	Desligado

Pronto, os comandos deste capítulo já foram criados.

4.5 Macros

4.5.1 Criação da macro ativa alarme

A macro “**ativa alarme**” serve para sinalizar ao operador a condição de alarme. Quando um alarme for gerado, o sistema deverá abrir a tela de alarme, desligar o *backlight*, piscar os *leds* da tecla F4 e ativar o sinalizador sonoro.

Como já feito anteriormente, crie uma nova macro. Após a criação da macro, mude seu “**Nome**” para “**Ativa Alarme**” e em seguida adicione os seguintes comandos: “**Abre Tela de Alarme**”, “**Desliga o Backlight**”, “**Pisca Leds tecla F4**” e “**Sonorização Alarme**”.

Para auxiliar nesta tarefa utilize a figura abaixo como referência:

Status	Macro
Propriedade	
Nome	Ativa Alarme
Identif. da Macro	2
Número de	4
Comando 01	Abre Tela de Alarme
Comando 02	Desliga o Backlight
Comando 03	Pisca Leds tecla F4
Comando 04	Sonorização Alarme

4.5.2 Criação da macro reconhece alarme

Após o reconhecimento do alarme, o sistema deverá acender o *backlight* e carregar a tela “**Principal**”. Portanto, deve-se criar uma macro que execute estes comandos.

Assim como realizado no item acima, crie uma nova macro e renomeie-a como “**Reconhece Alarme**”. Em seguida, adicione os comandos “**Liga Backlight**” e “**Abre Tela Principal**”.

Veja abaixo como esta macro deverá ser configurada:

Status	Macro
Propriedade	
Nome	Reconhece Alarme
Identif. da Macro	5
Número de	2
Comando 01	Liga o Backlight
Comando 02	Abre Tela Principal

4.6 Associação das Teclas da IHM

Como já efetuado anteriormente, clique em **“Teclado”** e abra as configurações para a tecla F4. Associe a macro **“Reconhece Alarme”** como **“Macro tecla pressionada”**.

Veja abaixo como deverá ficar a configuração para a tecla F4:

Tecla F4: Nível de acesso	Sem restrição
↳ Macro tecla pressionada	Reconhece Alarme
↳ Macro tecla liberada	Nenhuma

4.7 Ajustes Finais

Para que o programa OPPE esteja pronto, é necessário efetuar um ajuste final, isto é, especificar nas ocorrências de **“Campo Oculto”** qual deverá ser a macro de alteração.

Para isso, volte à tela **“Principal”** e clique em **“Campo04”**, que é o campo oculto. No item **“Macro de Alteração”**, selecione a macro **“Ativa Alarme”**.

Veja na figura abaixo como deverá ser configurado este campo:

Status	Campo
Propriedade	
Nome	Campo04
Atualização do Campo	Nos ciclos de refresh
Posição	...
Taq associado	GrupoExemplo:R Falha no processo
Macro de Alteração	Ativa Alarme
Tipo de Campo	Oculto

Por fim, execute este procedimento também na tela **“Tempo de Vácuo”**. Vá até esta tela e selecione o **“Campo02”**, que é o campo oculto. Em seguida, selecione a macro **“Ativa Alarme”** para o item **“Macro de Alteração”**, assim como foi feito na figura abaixo:

Status	Campo
Propriedade	
Nome	Campo02
Atualização do Campo	Nos ciclos de refresh
Posição	...
Taq associado	GrupoExemplo:R Falha no processo
Macro de Alteração	Ativa Alarme
Tipo de Campo	Oculto



Utilização da MMI800 – Módulo I

Tipo de Doc.: Notas de Aplicação
Referência: ENA.00032

Revisão: 2
Atualizado em: 17/04/2008

Por fim, carregue o programa na IHM utilizando o procedimento já efetuado e verifique o seu funcionamento. Utilize em conjunto o programa *ladder* de exemplo para a autoclave.

4.8 Carga do programa

O próximo passo é efetuar a carga do programa. Carregue o programa utilizando o procedimento já efetuado e simule uma condição de falha. Verifique o funcionamento do programa, conferindo a apresentação da tela de alarme e reconheça o mesmo através da tecla de função F4.

Observação: Neste ponto, salve uma cópia do programa gerado pelo OPPE como “Capítulo4-EX.OPJ”. Em seguida, continue utilizando o arquivo original para a edição das funções que serão incorporadas neste capítulo.



Utilização da MMI800 – Módulo I

Tipo de Doc.: Notas de Aplicação
Referência: ENA.00032

Revisão: 2
Atualizado em: 17/04/2008

Controle do Documento

Considerações gerais

1. Este documento é dinâmico, estando sujeito a revisões, comentários e sugestões. Toda e qualquer sugestão para seu aprimoramento deve ser encaminhada ao departamento de suporte ao cliente da **HI Tecnologia**, especificado na “Apresentação” deste documento.
2. Os direitos autorais deste documento são de propriedade da **HI Tecnologia**.

Responsabilidades pelo documento

	Data	Responsável	
Elaboração	24/10/2005	Ricardo L. Oliveira	
Revisão	17/04/2008	Paulo C M Inazumi	<i>Revisado em mídia</i>
Aprovação	17/04/2008	Isaiás M. C. Ribeiro	<i>Aprovado em mídia</i>

Histórico de Revisões

17/04/2008	2	Inclusão da referência ENA.00048, exclusão da sessão download do OPPE
03/01/2008	1	Atualização das imagens referentes ao site da HI Tecnologia e adesivo da MMI800
31/10/2005	0	Documento original
Data	Rev	Descrição