

Universidade Federal de Minas Gerais

Escola de Engenharia

Departamento de Engenharia Eletrônica
Laboratório de Informática Industrial

AULA 3 – PROGRAMAÇÃO CONVENCIONAL DE CLP (II): TUTORIAL DO SOFTWARE RSLogix 5000

Objetivos: Familiarização com o ambiente de trabalho do software de programação *RSLogix 5000* da Rockwell Automation.

Atividades Prévias

Leia atentamente, em casa, o texto desta prática, de forma a agilizar os procedimentos a serem executados no laboratório.

Recapitulação: Definições importantes para a linguagem *ladder*

Ponto de Entrada: Corresponde a um sinal recebido pelo CLP. Ex.: contatos de relés e contadores, sensores de posição, botões, microchaves, etc.

Ponto de Saída: Corresponde a um sinal produzido pelo CLP para o acionamento de algum dispositivo. Ex.: lâmpadas, motores, válvulas, contadores, etc.

Endereço: Um valor alfanumérico que identifica unicamente onde um dado é armazenado. Ex.: `Local:1:I.Data.0` refere-se à entrada 00 do primeiro cartão de entradas discretas do CLP; `Local:2:O.Data.2` refere-se à saída 02 do segundo cartão de saídas discretas do CLP.


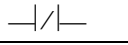
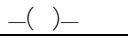
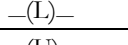
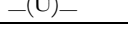
Tabela de estados (*status table*): É a parte da CPU que armazena o *status* de todas as entradas.

Contato Normalmente Aberto: Símbolo lógico que representa continuidade (fluxo), se a entrada de referência for “1” lógico (verdadeiro), quando avaliada.

Contato Normalmente Fechado: Símbolo lógico que representa continuidade (fluxo), se a entrada de referência for “0” lógico (falso), quando avaliada.

Ciclo de Varredura (*Scan time*): Tempo requerido pelo processador para ler todas as entradas, executar o programa de controle e atualizar as saídas. O ciclo de varredura de um CLP é repetido continuamente, enquanto o processador está em modo “Run” (Execução/Operação).

A tabela a seguir apresenta os elementos gráficos disponíveis no editor *ladder* do *RSLogix 5000*:

Contatos		Normalmente Aberto
		Normalmente Fechado
Bobinas		Bobina Simples
		Bobina do tipo L (<i>latched</i>)
		Bobina do tipo U (<i>unlatched</i>)

Parte Prática: Tutorial do Software RSLogix 5000

O software *RSLogix 5000* é utilizado para programar o CLP *CompactLogix* presente no Módulo de Instrumentação, Automação e Controle (MICA). Nele, é possível escrever um programa em *ladder* ou nas outras linguagens da norma IEC 61131-3 (exceto “Lista de Instruções”).

As lições a seguir o guiarão na criação de um projeto, na edição de um programa simples em linguagem *ladder*, na carga deste programa no CLP e na monitoração das variáveis do programa.

LIÇÃO 1: INICIANDO O *RSLOGIX*

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

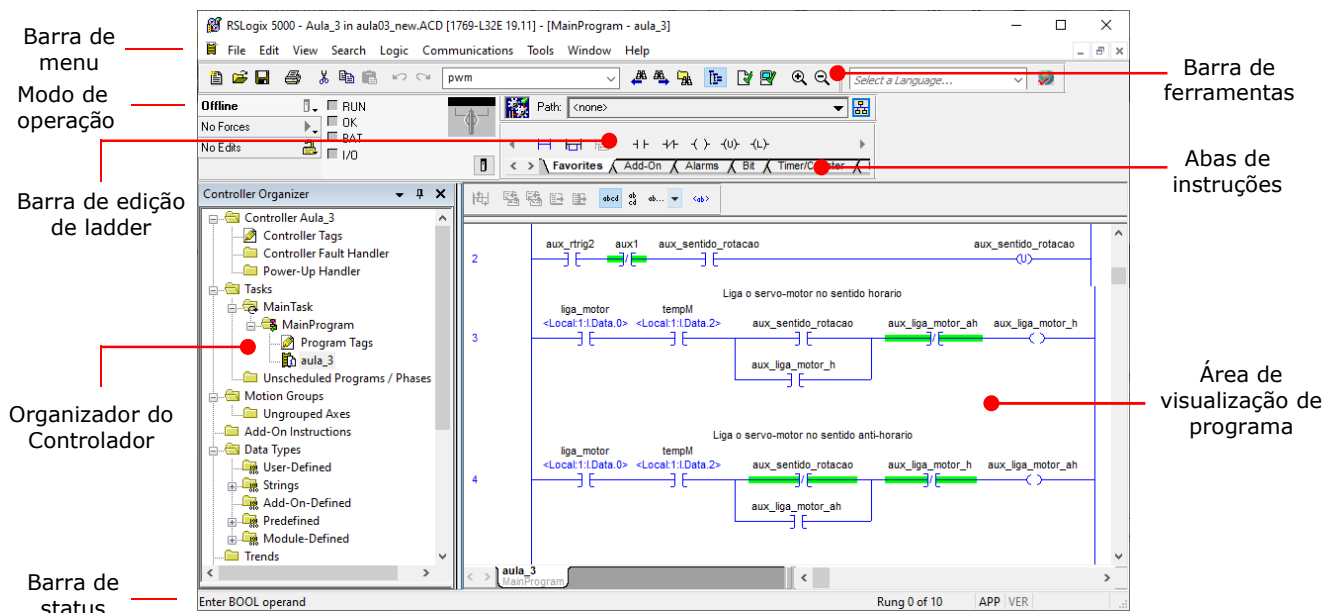


Figura 1: Áreas do ambiente de trabalho do *RSLogix 5000*.

LIÇÃO 2: CRIANDO UM PROJETO NO *RSLOGIX 5000*

Para criar um novo projeto, na janela do *RSLogix*, selecione o menu *File* → *New* (ou use as teclas de atalho CTRL + N). Uma janela de nome *New Controller* (Fig. 2) será aberta.

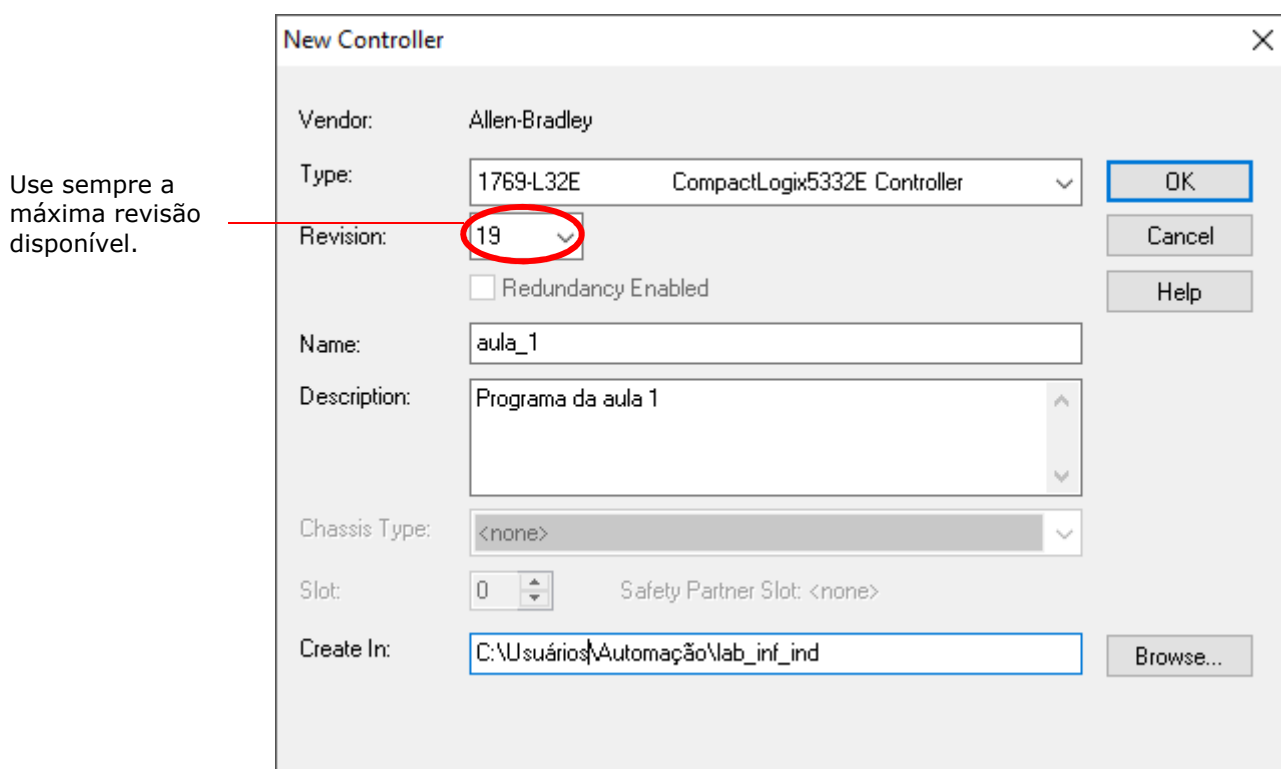


Figura 2: Janela de criação de um novo projeto.

Inicialmente, com o *Windows Explorer*, crie um diretório de trabalho para o seu grupo em **C:\Usuários\Automação\lab_inf_ind**. Em seguida, na janela *New Controller*, selecione o tipo do controlador (*CompactLogix 1769-L32E*), a revisão do software (sempre a máxima revisão disponível), atribua um nome para o projeto, escreva uma descrição (opcional) e defina a pasta onde o projeto será criado como sendo o diretório anteriormente criado.

Observações:

- Ao dar nomes para o controlador, *tags*, rotinas, módulos de E/S etc, deve-se usar apenas letras, números e *underscore* (“_”), sendo que o primeiro caractere não pode ser um número;
- Não há distinção entre letras maiúsculas e minúsculas.

Após a criação de um projeto, a tela do *RSLogix* estará semelhante à Fig. 3. Nessa figura, do lado esquerdo, tem-se o “organizador do controlador”. Ele exibe as características gerais do projeto e deve ser utilizado para navegar entre os vários componentes do projeto. Essa janela pode ser mostrada ou ocultada pelo menu *View* → *Controller Organizer* (ou usando as teclas de atalho ALT + 0).

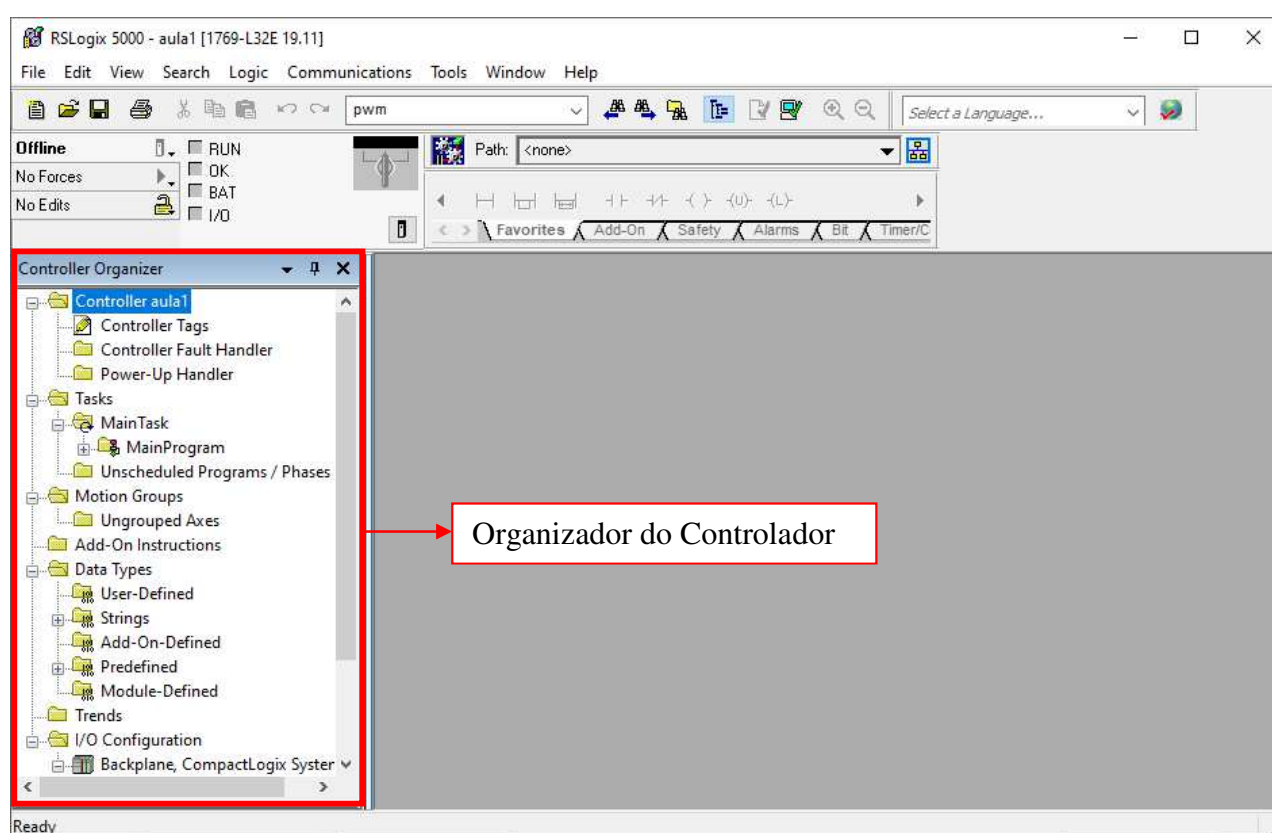


Figura 3: Janela inicial do *RSLogix 5000*

LIÇÃO 3: ADICIONANDO MÓDULOS

Após criado um projeto, é necessário criar também uma configuração de controlador, adicionando-se à mesma os cartões de entrada e saída que estão conectados ao CLP. Como se pode notar no painel do MICA, o CLP possui 4 módulos de E/S, e estes devem ser adicionados ao organizador do controlador.

Para adicionar módulos ao controlador, selecione a opção *CompactBus Local* no organizador do controlador (como mostrado na Fig. 4) e clique sobre a mesma com o botão direito do mouse, selecionando a seguir a opção *New Module* e, em seguida, adicionando os módulos a serem inseridos (Fig. 5). Os módulos do modelo *CompactLogix 1769-L32* são, da esquerda para a direita:

Código	Descrição	Slot
1769 IQ6XOW4	6 entradas digitais 24 VDC e 4 saídas relé AC/DC	1
1769 IQ6XOW4	6 entradas digitais 24 VDC e 4 saídas relé AC/DC	2
1769 OB16	16 saídas digitais 24 VDC	3
1769 IF4XOF2 (MICA com cartão analógico de 8 bits) ou 1769 IF4FXOF2F (MICA com cartão analógico de 12 bits) Atenção! Pergunte ao Professor qual o modelo do MICA em sua bancada!	4 entradas analógicas / 2 saídas analógicas	4

Observações:

- Os módulos devem ser adicionados na ordem que se encontram fisicamente no controlador, da esquerda para a direita.
- No campo *Name*, recomenda-se colocar “cN”, onde N refere-se ao número do *slot* (figura 6).
- Atenção para o cartão de E/S analógicas: este difere entre alguns modelos do MICA! Pergunte ao professor qual é o modelo em sua bancada.

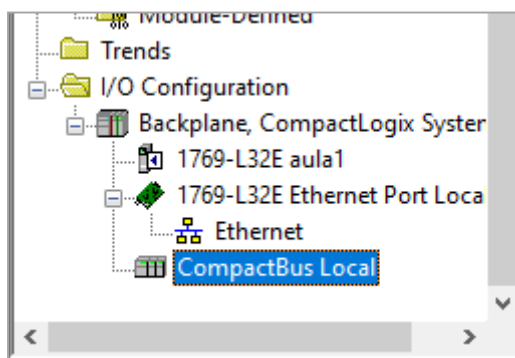


Figura 4: Detalhe do “Organizador do Controlador”, destacando o barramento no qual serão acrescentados os módulos de E/S.

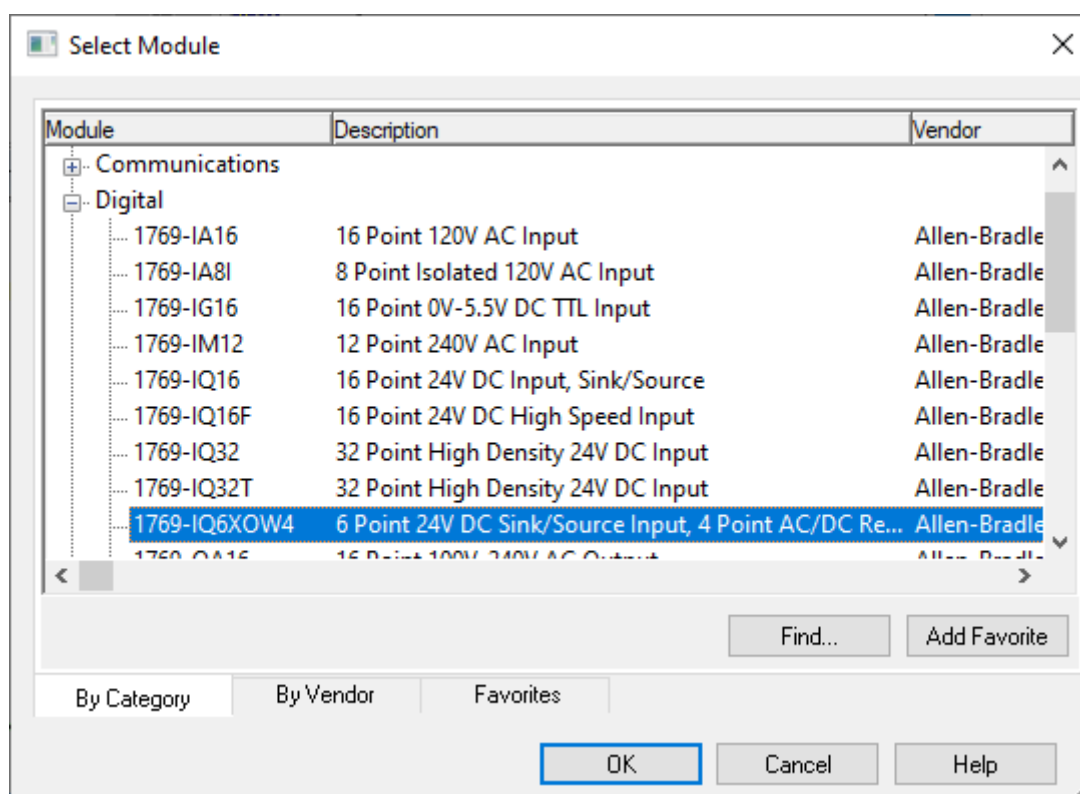


Figura 5: Adição de módulos

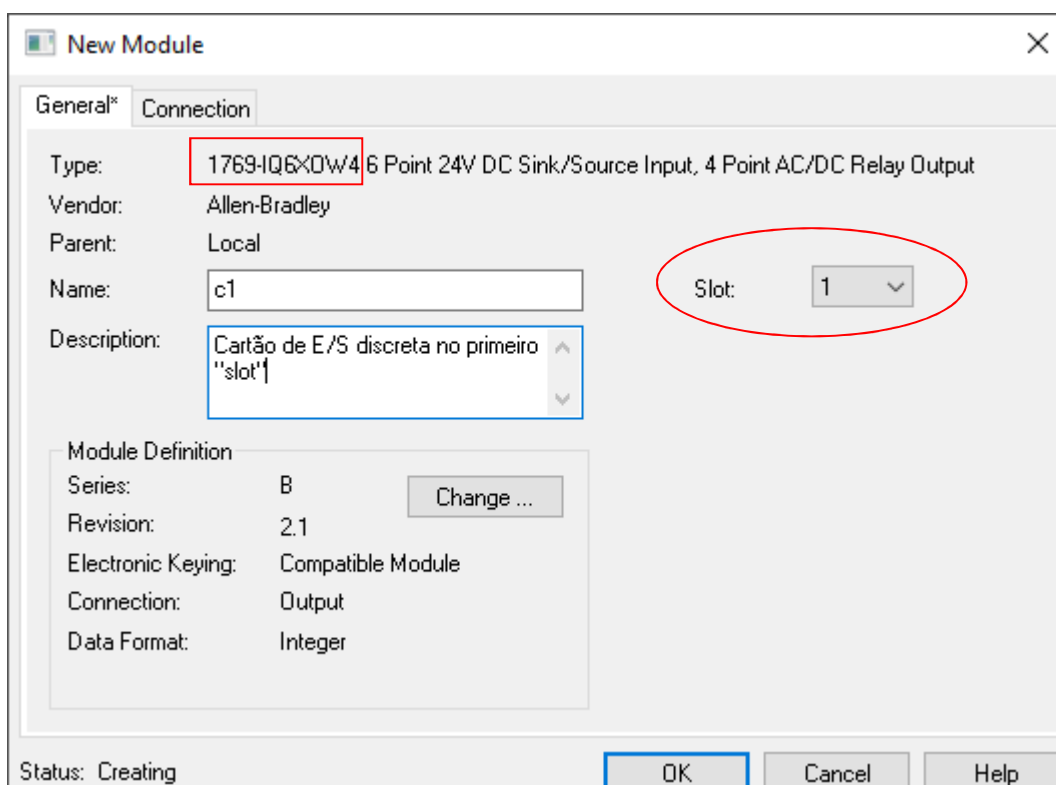


Figura 6: Configuração do Módulo

Após a adição dos quatro módulos, a pasta *I/O Configuration* deverá estar como mostrado na Fig. 7:

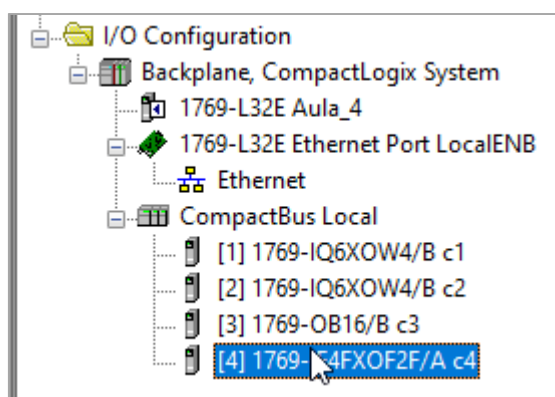


Figura 7: Pasta *I/O Configuration*

LIÇÃO 4: ESCRIVENDO A LÓGICA DE PROGRAMAÇÃO

Para iniciar a programação da lógica, selecione, no organizador do controlador, a rotina principal (fig. 8), com um duplo clique.



Figura 8: Rotina Principal

Ao abrir a rotina de programação, as ferramentas de desenho da lógica *ladder* ficam disponíveis (Fig. 9).

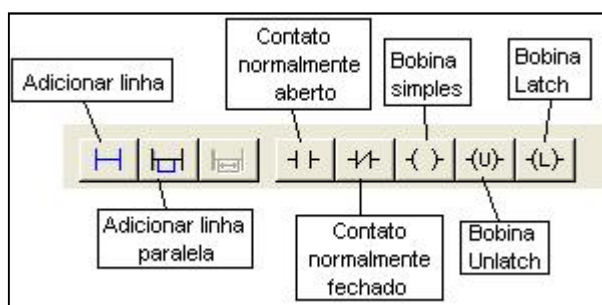


Figura 9: Ferramentas de programação.

Exercício 1: Construindo um diagrama *ladder* simples

Utilizando as ferramentas de programação em *ladder*, construa o diagrama correspondente ao circuito lógico da Fig. 10, empregando contatos para representar as entradas e bobinas para representar as saídas:

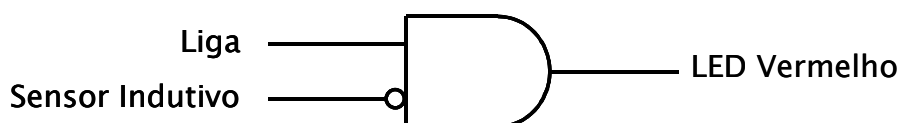


Figura 10: Diagrama lógico referente ao exercício 1

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

No CLP *CompactLogix*, um endereço de E/S segue o formato **Local:Slot:Tipo.Membro.Bit**, conforme a tabela abaixo:

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

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

No endereçamento direto, basta fazer um duplo clique, com o botão esquerdo do *mouse*, no símbolo de interrogação que aparece logo acima do elemento (Fig. 11):

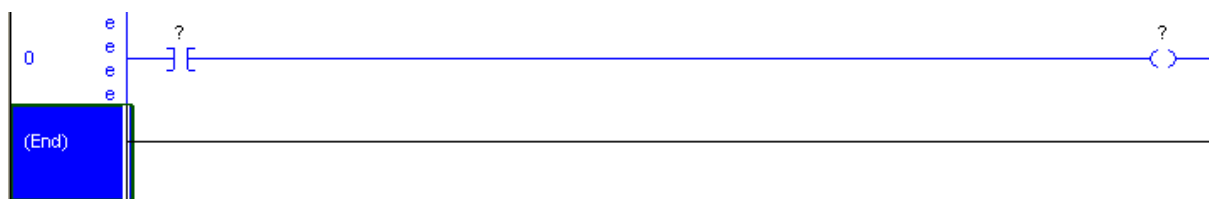


Figura 11: Elementos *ladder* ainda sem definição de endereço.

Dessa forma, um menu é apresentado para se possa escolher o endereço do elemento (Fig. 12).

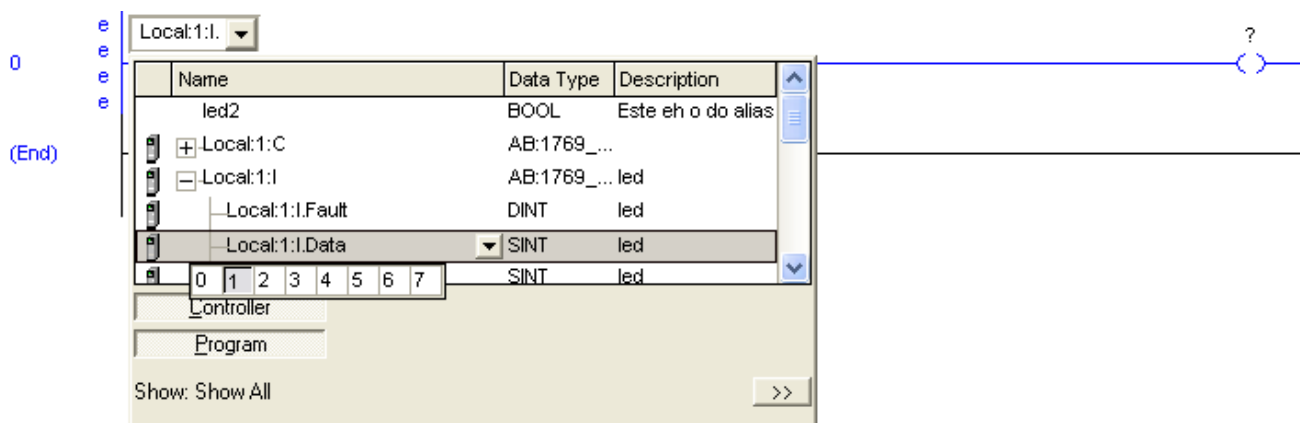


Figura 12: Janela de endereçamento de um elemento *ladder*

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

Variáveis de Entrada			Variáveis de Saída		
Endereço	Descrição	ID no MICA	Endereço	Descrição	ID no MICA
Local:1:I.Data.0	Liga/Desliga	RB01/00	Local:3:O.Data.0	LED Vermelho	RB03/00
Local:1:I.Data.1	Sensor Indutivo	RB01/01			

Tabela 1: Endereçamento de elementos da linguagem *ladder*

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

Exercício 2: Empregando “alias”

A segunda maneira de se endereçar elementos é criando-se um “*alias*”. Com o *alias* definido, pode-se reutilizar mais facilmente este mesmo elemento em outras partes do programa. Para testar essa funcionalidade, insira uma nova linha e os elementos necessários para implementar a lógica a seguir, sem endereçá-los:

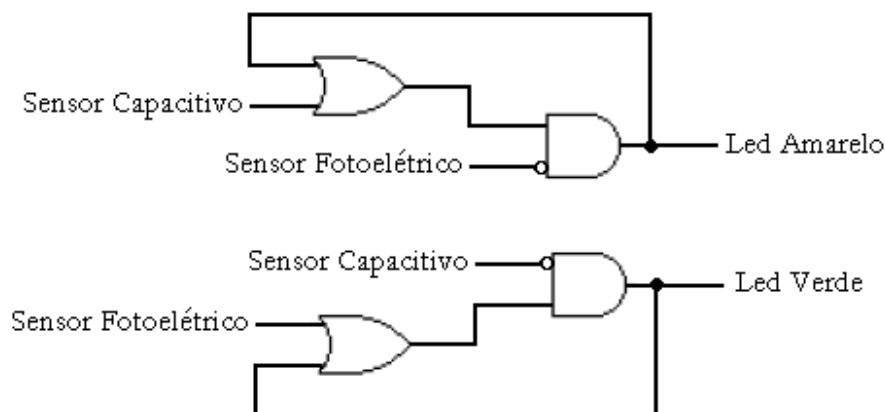


Figura 13: Diagrama lógico referente ao exercício 2.

Para cada elemento inserido, clique na interrogação logo acima do elemento com o botão direito do mouse e selecione a opção *New Tag*. Na janela *New Tag* (Fig. 14), preencha os campos *Name*, *Description*, *Type* e *Alias For* de acordo com a tabela a seguir. Observe que, quando for utilizar elementos já existentes, não é necessário criar um novo *tag*: basta fazer um duplo clique no símbolo “?” do elemento e digitar o *alias*.

Variáveis de Entrada			Variáveis de Saída		
Endereço	Significado	ID no MICA	Endereço	Significado	ID no MICA
Local:1:I.Data.2	Sensor Fotoelétrico	RB01/02	Local:3:O.Data.2	LED Amarelo	RB03/02
Local:1:I.Data.4	Sensor Capacitivo	RB01/04	Local:3:O.Data.3	LED Verde	RB03/03

Tabela 2: Endereçamento de elementos adicionais da linguagem *ladder*

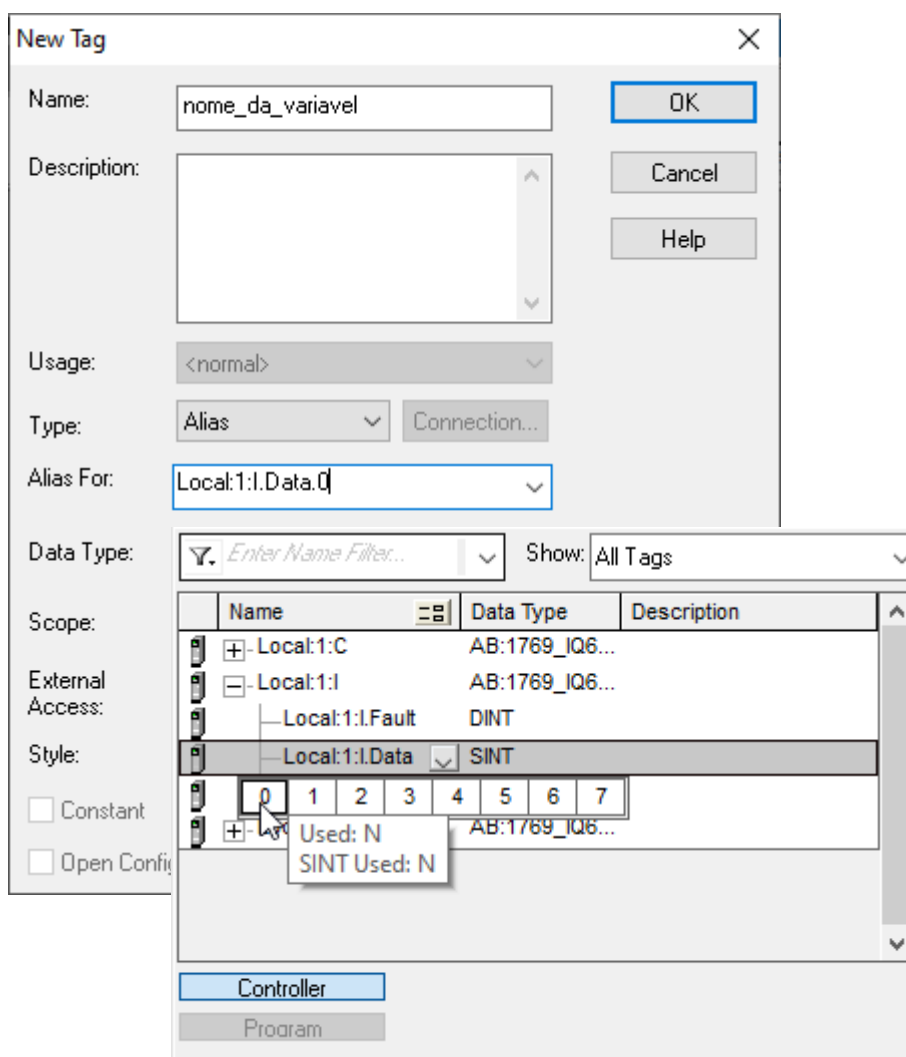


Figura 14: Criando Aliases

LIÇÃO 5: ENTENDENDO OS MODOS DE FUNCIONAMENTO DO CLP

A chave frontal presente no CLP *CompactLogix* (Fig. 15) seleciona a forma de operação desejada:

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

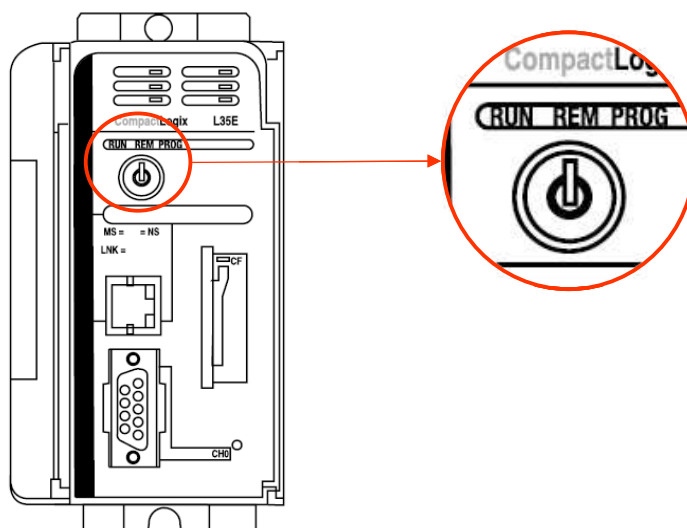


Figura 15: Chave de seleção de modo no painel do CompactLogix.

Nas práticas de laboratório a chave deve estar sempre na posição REM, possibilitando o uso do RSLogix de forma “on-line”. Além disto, após a carga de um programa no CLP CompactLogix, sua execução só inicia após a colocação do CLP no modo REMOTE RUN. Para chavear entre os modos REMOTE PROG e REMOTE RUN, clique no ícone apropriado do RSLogix 5000 como indicado na Fig. 16.

Clique neste ícone para chavear entre os modos REMOTE PROG e REMOTE RUN

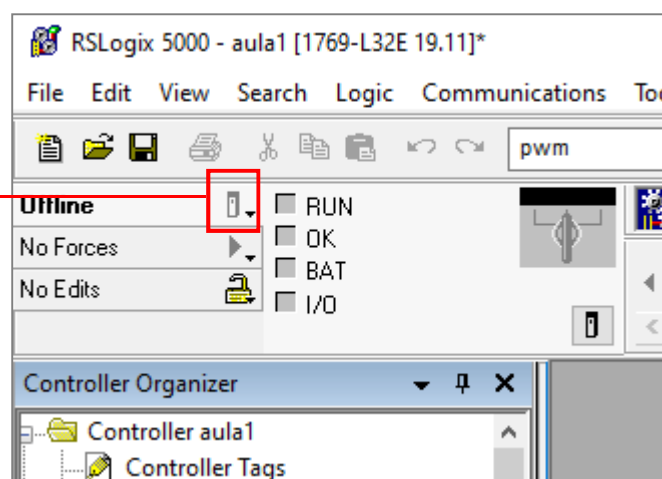



Figura 16: Seleção do modo de operação do CLP

LIÇÃO 6: VERIFICANDO A LÓGICA E CARREGANDO O PROGRAMA NO CLP

A verificação de lógica pode ser feita para todo o projeto ou para apenas a rotina corrente. Para verificar a lógica do projeto, selecione *Logic* → *Verify* → *Controller*, para verificar apenas a lógica da rotina corrente, selecione *Logic* → *Verify* → *Routine*. Certifique-se que a janela de visualização de erros esteja visível, selecionando *View* → *Errors* caso não esteja.

Fique atento para eventuais mensagens de alerta (*warnings*): apesar das mesmas não impedirem o carregamento e a execução do programa, muitas vezes essas mensagens reportam problemas de lógica, sintaxe ou erros semânticos que passaram despercebidos ao programador.

Para realizar a carga da lógica desenvolvida na lição 4 para o *CompactLogix*, selecione o menu *Communications* → *Who Active* ou clique diretamente no ícone *Who Active*. 

Ao realizar um dos comandos acima, será exibida uma janela onde se deve configurar o “caminho” (*path*) do CLP e, em seguida, realizar o *download*. Para isso, expanda o CLP correspondente à sua bancada clicando em “+”, expanda o *Backplane* e selecione o processador (*Compact Logix Processor*). Com o *path* definido, clique no botão *Download*, como mostrado na figura 17.

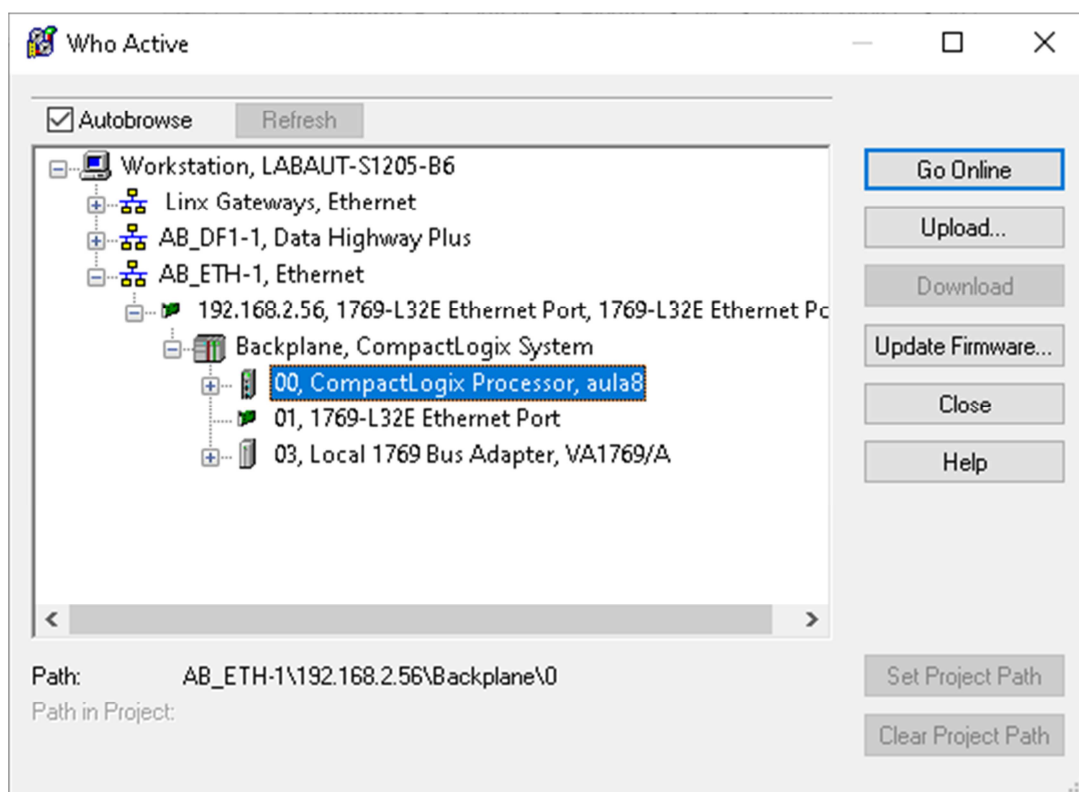


Figura 17: Janela de carregamento do programa no CLP.

Durante a carga do programa, o modo de operação do *CompactLogix* será automaticamente alterado de “modo de execução remota” (*Remote Run Mode*) para “Modo de Programação” (*Program Mode*), de modo a suspender a execução do programa anteriormente carregado no CLP e permitir sua substituição pelo novo programa. Ao final da carga, uma janela de mensagem será exibida solicitando a confirmação de volta ao “modo de execução remota”. Clique em *Yes* para iniciar a execução do programa no CLP.

Instruções de montagem no MICA:

1. Com o MICA energizado, certifique-se que o botão de emergência esteja pressionado.
2. Conecte os bornes de alimentação do contator C1 (situado à direita dos sensores capacitivo e fotoelétrico) à régua RB24 (fonte 24V) e à régua RB00 (terra).
3. Conecte o borne lateral L1 à régua RB24 (fonte 24V) e o borne lateral T1 ao ponto de entrada da variável “Liga/Desliga” na tabela 1, de forma a sinalizar ao CLP a operação de liga/desliga efetuada pelo operador. Estes bornes laterais situam-se na lateral direita do MICA.
4. Alimente os sensores e conecte suas saídas NA às entradas indicadas nas tabelas 1 e 2.
5. Alimente a régua RB03 e conecte as saídas indicadas nas tabelas 1 e 2 aos sinalizadores luminosos (lâmpadas).
6. Confira com atenção a sua montagem antes de rearmar o botão de emergência.

LIÇÃO 7: UTILIZANDO TEMPORIZADORES

Nesta lição você irá utilizar o bloco de temporização *Timer On Delay* (TON). Este bloco de temporização possui entradas IN e PRESET e saídas (entre outras) EN, DN e ACCUM, e funciona do seguinte modo:

1. Ao ter sua entrada IN energizada, o bloco começa sua contagem (em milissegundos). A saída EN é então energizada, indicando que o temporizador está ativo. A saída ACCUM exibe o tempo decorrido desde o início da contagem. A saída DN permanece inicialmente no nível lógico “0”.
2. Quando o tempo de contagem alcançar o valor da entrada PRESET, a saída DN é colocada no estado lógico “1” e assim permanece, enquanto o temporizador estiver energizado.
3. Em qualquer tempo, se a entrada IN for ao nível lógico “0”, as saídas EN e DN também são levadas imediatamente a este nível.

Na Fig. 18, que mostra uma aplicação fictícia, a variável **Local:3:O.Data.3** é energizada 5 segundos após a variável **Local:3:O.Data.0** ser energizada. A variável “timer_1” é do tipo TIMER, e deve ser declarada.

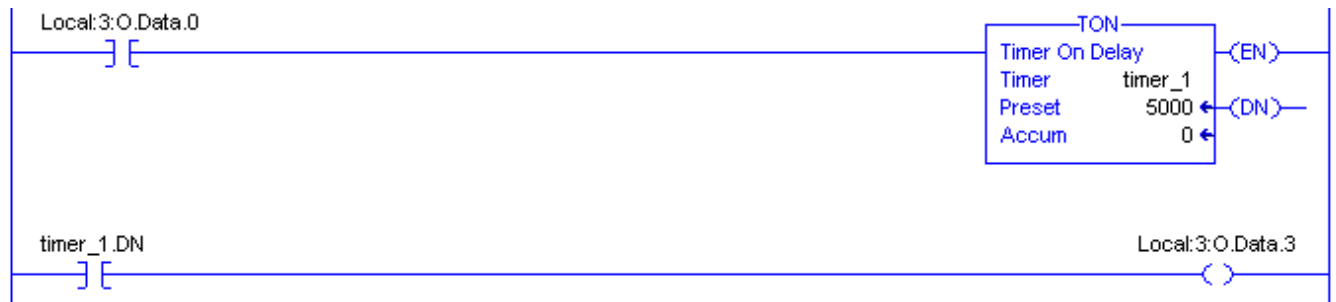


Figura 18: Exemplo de uso do temporizador TON

NOTA: Para ajuda quanto aos parâmetros de qualquer bloco de função do CLP, consulte o help do RSLogix5000 selecionando *Help* → *Contents*.

Insira uma nova linha em seu diagrama *ladder*, correspondente ao diagrama lógico a seguir:

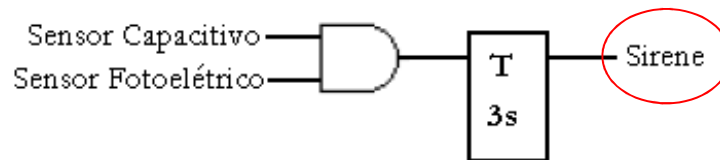


Figura 19: Diagrama lógico demonstrando o uso de temporizador

Para inserir um bloco de temporizador, selecione a aba *Timer/Counter* e, nesta, clique no temporizador desejado (no caso desta prática, TON). Após inseri-lo no programa *ladder*, clique no mesmo com o botão direito do mouse e selecione *New Tag* para definir o nome do temporizador. Escolha o valor “base” para o tipo de endereçamento e “timer” para o tipo do dado.

Observe que os endereços dos sensores já foram definidos anteriormente, bastando definir o endereço da sirene, que deve ser conectada ao RB03/04 e, portanto, deve ter o endereço **Local:3:O.Data.4**, e declarar a variável correspondente ao temporizador.

LIÇÃO 8: MONITORANDO AS VARIÁVEIS - CONTROLLER TAGS

Com o CLP no modo REM e o RSLogix 5000 no modo *Run*, este executa em modo monitoração, permitindo ao usuário verificar *on-line* o status dos contatos e bobinas, usando a cor verde para indicar os estados energizados e azul para indicar os estados desenergizados.

Todas as informações de E/S do CLP são apresentadas como um conjunto de *tags* (Fig. 20).

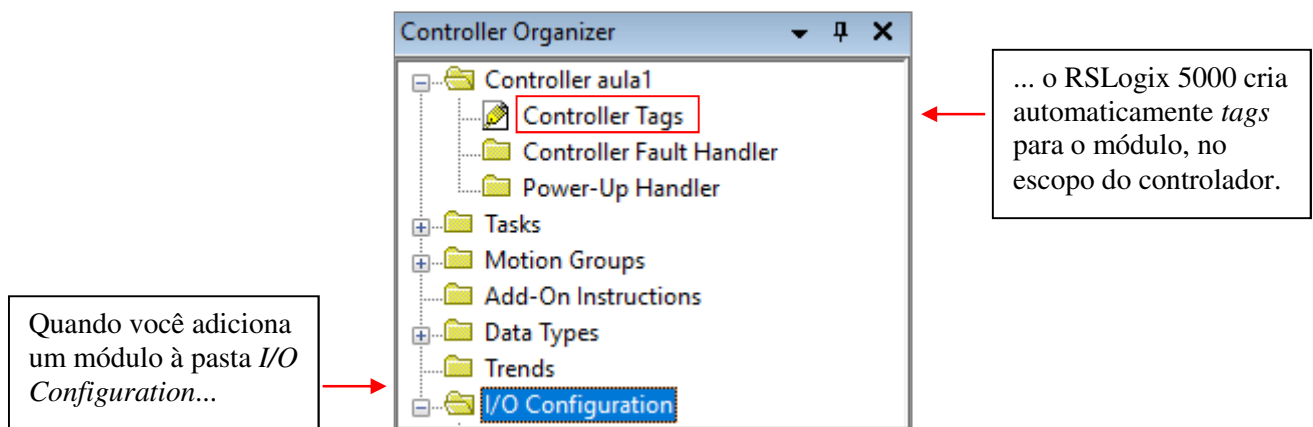


Figura 20: Monitorando Tags

Para monitorar as variáveis do escopo do controlador, clique com o botão direito do mouse em *Controller Tags* e selecione a opção *Monitor Tags*. Será apresentada uma janela como a da Fig. 21, na qual é possível verificar todas as variáveis disponíveis para uso no controlador.

Name	Value	Force Mask	Style	Data Type
Local:1:C	{...}	{...}		AB:1769_IQ6X0...
Local:1:I	{...}	{...}		AB:1769_IQ6X0...
Local:1:I.Fault	2#0000_000...		Binary	DINT
Local:1:I.Data	2#0000_0010		Binary	SINT
Local:1:I.Data.0	0		Decimal	BOOL
Local:1:I.Data.1	1		Decimal	BOOL
Local:1:I.Data.2	0		Decimal	BOOL
Local:1:I.Data.3	0		Decimal	BOOL
Local:1:I.Data.4	0		Decimal	BOOL
Local:1:I.Data.5	0		Decimal	BOOL
Local:1:I.Data.6	0		Decimal	BOOL
Local:1:I.Data.7	0		Decimal	BOOL

Figura 21: Janela de monitoração de variáveis

Para expandir um *tag* e exibir seus membros, clique no sinal “+”.

Os valores das variáveis monitoradas são mostrados com os seguintes estilos:

Estilo	Base	Prefixo
Binário	2	2#
Octal	8	8#
Decimal	10	---
Hexadecimal	16	16#

Tabela 3: Prefixos empregados no RSLogix 5000 para indicar diferentes bases numéricas diferentes.

Para visualizar valores em uma base diferente, selecione o estilo desejado.

Para realizar a monitoração das variáveis de uma determinada rotina, ou de uma ou mais variável à sua escolha, selecione *View* → *Watch* ou então use as teclas de atalho ALT + 3. Existem duas opções de monitoração (Fig. 22): *Current Routine* ou *Quick Watch*.

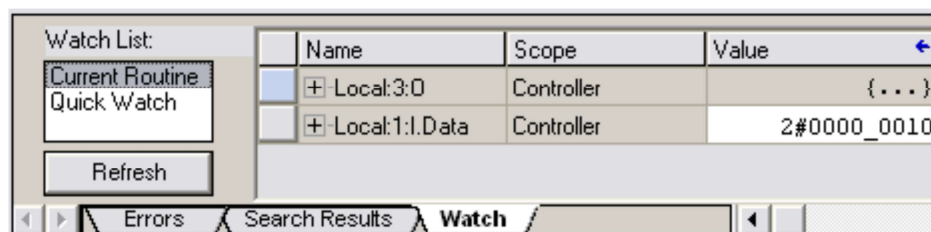


Figura 22: Watch List

- *Current Routine*: Monitora todas as variáveis pertencentes à rotina selecionada no momento.
- *Quick Watch*: Monitora apenas as variáveis desejadas pelo programador. Variáveis pertencentes a escopos distintos podem ser monitoradas, e pode-se ainda monitorar apenas determinado bit de uma variável.

Realize testes no painel do MICA, modificando as entradas e verificando o *status* das variáveis de entrada e saída no *Controller Tags*.