

MIP

Módulo Programável



JNA Componentes Eletrônicos
www.jnaeletronicos.com
jna@jnaeletronicos.com
(47) 3467 5569



Recursos

4 Entradas Analógicas;
4 Entradas Digitais;
1 Entrada de Temperatura;
4 Saídas Digitais;

Comunicação RS485 para expansão em até 100 pontos para E/S Digitais e 50 pontos para Entradas Analógicas.

LEDs indicativos nas E/S e na alimentação.

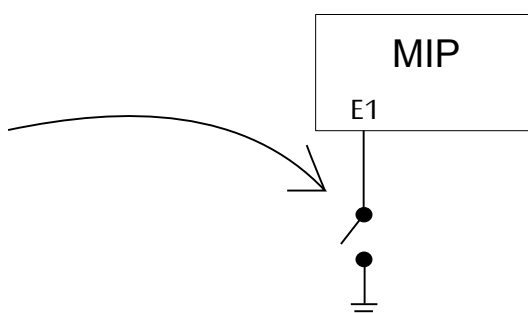
Especificações Elétricas

Alimentação 12/24 Vcc;

Entradas digitais acionadas com contato no negativo;

Saídas digitais a Relê;

Entradas analógicas de 0 a 50 Vcc.



Guia de Instalação e Utilização - Software

- 1 Instale o programa "Beremiz" que se encontra no CD do MIP;
- 2 Copie a pasta MDK do CD para um local de sua escolha no seu computador;
- 3 Renomeie a pasta copiada para um nome de sua escolha;

Repita os passos 2 e 3 sempre que iniciar um novo projeto com os módulos MIP.

- 4 Abra o programa PLCOpenEditor
em Iniciar > Programas > Beremiz > Tools > PLCOpenEditor;
- 5 Em Arquivo > Abrir, Abra o arquivo "programa";

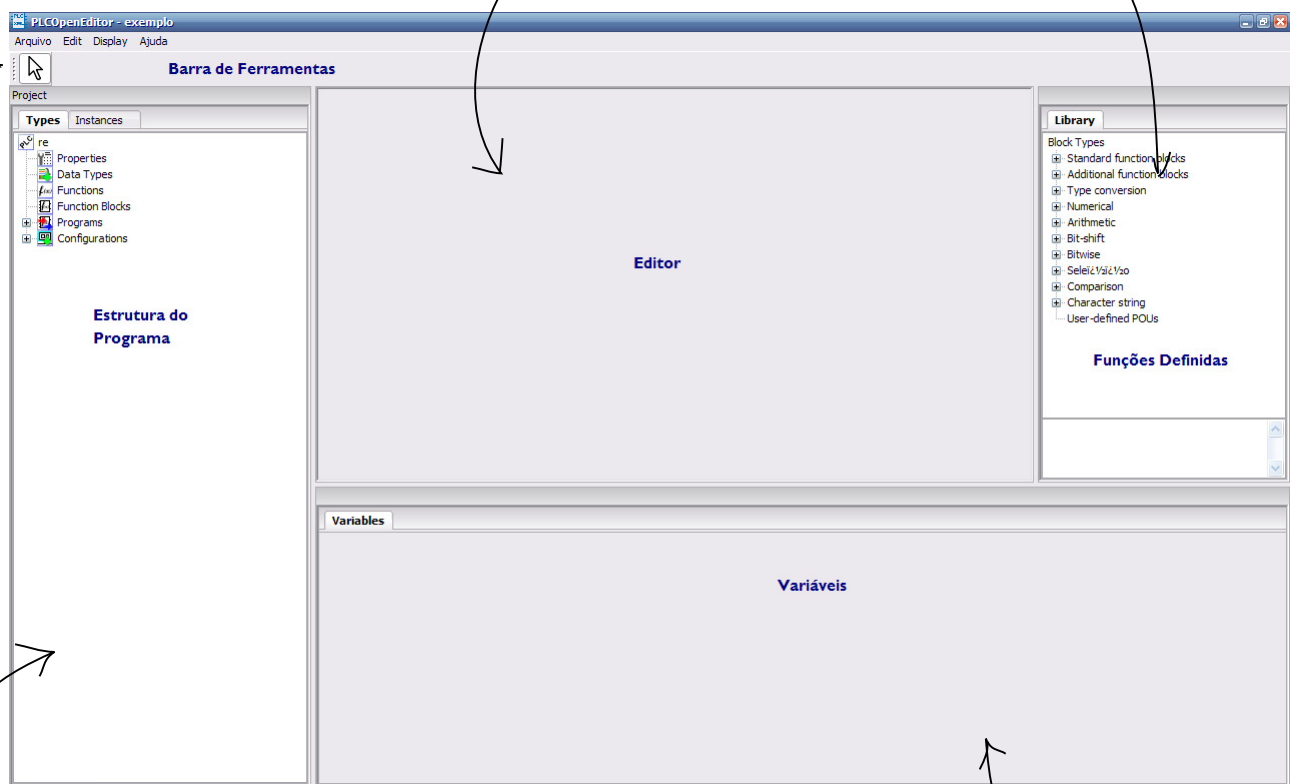
Pronto, agora você pode criar o seu programa.

PLCOpenEditor - Interface

Barra de Ferramentas

Editor Gráfico

Funções Pré-Definidas

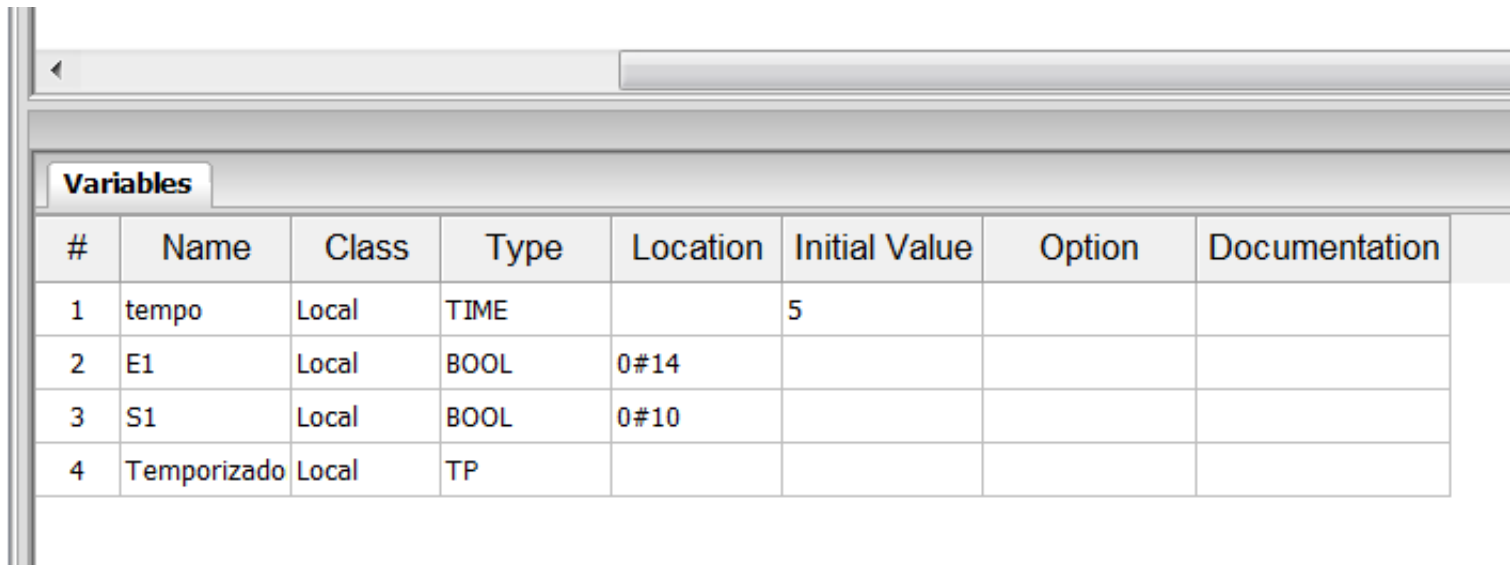


Estrutura do Programa

Variáveis

PLCOpenEditor - Variáveis

As variáveis são utilizadas para representar entradas e saídas do módulo MIP e armazenar valores.



#	Name	Class	Type	Location	Initial Value	Option	Documentation
1	tempo	Local	TIME		5		
2	E1	Local	BOOL	0#14			
3	S1	Local	BOOL	0#10			
4	Temporizado	Local	TP				

Uma variável possui:

Nome

Identifica a variável. Um nome de variável pode conter apenas letras (sem nenhum caracter especial como 'ç' ou assentos), números e o caracter '_'. Ela não pode começar com um número como 9minha_var.

Exemplos: minha_var, var1, E1.

Tipo

O tipo da variável diz o tipo de valor que ela irá armazenar.

Localização

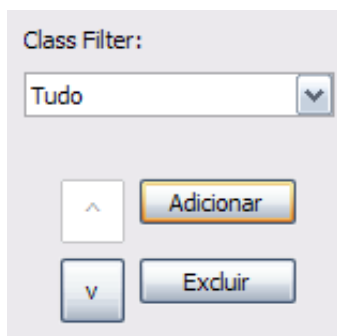
Se for indicada uma localização, a variável irá representar a E/S correspondente. Caso contrário, a variável será apenas armazenada na memória do módulo.

Valor Inicial

Se preenchido, quando o programa iniciar a variável terá este valor.

PLCOpenEditor - Adicionando Variáveis

As variáveis são utilizadas para representar entradas e saídas do módulo MIP e armazenar valores.



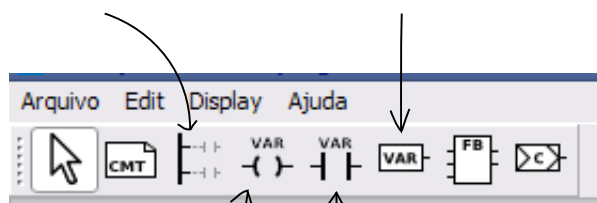
1. Clique em "Adicionar" ao lado da tabela de variáveis.

*Uma nova linha na tabela de variáveis irá aparecer.
Preencha os campos necessários.*

2. Em seguida podemos inserir a variável no programa.

Abrir ou Fechar um circuito em LADDER

Inserir uma E/S em um diagrama FBD

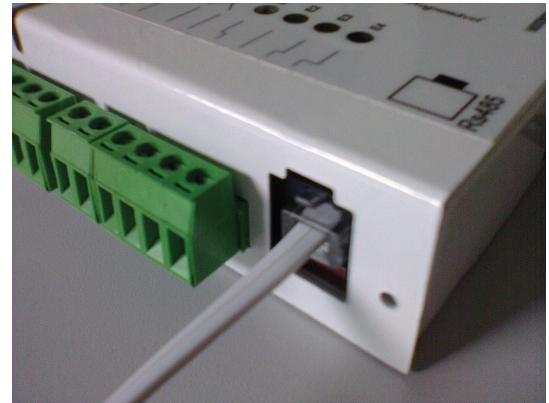
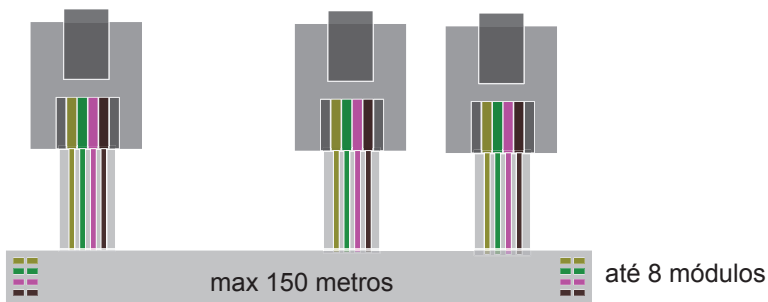


Inserir uma saída em um diagrama LADDER

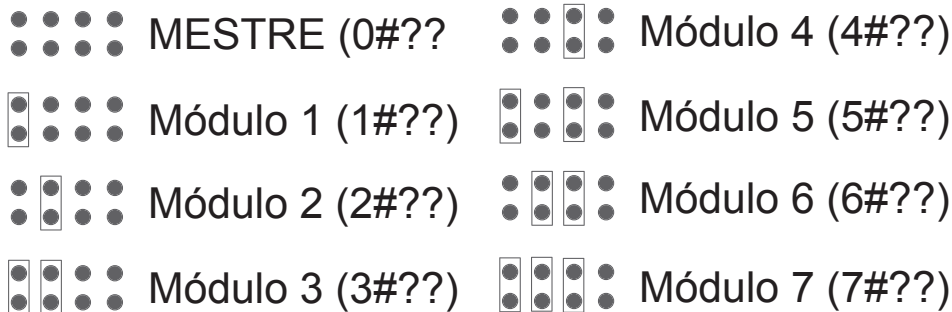
Inserir uma entrada em um diagrama LADDER

Siga estes passos para colocar os módulos MIP em Rede.

1. Conecte os módulos com o cabo de rede.



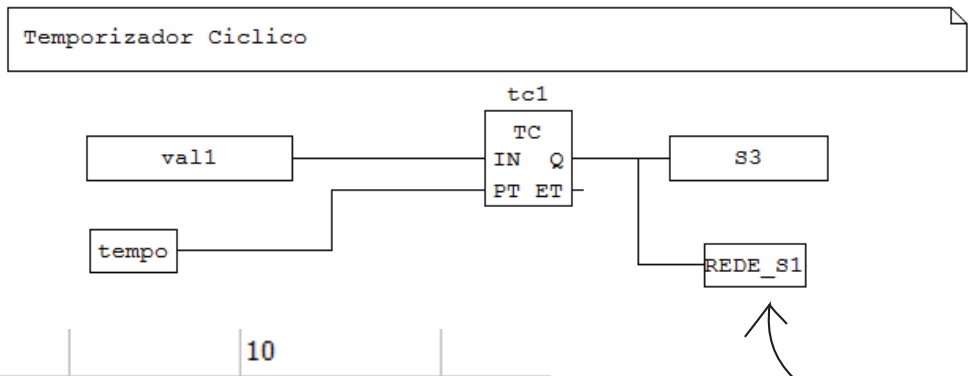
2. Configure os pinos de endereçamento de cada módulo.



"??" é o número da E/S que se deseja acessar.



3. Utilize normalmente no PLCOpenEditor.



17	count	Local	INT		10
18	val1	Local	BOOL		1
19	REDE_S1	Local	BOOL	1#0	

Endereço configurado nos pinos do módulo



Siga esses passos para utilizar a IHM.

1. Defina as mensagens que irão aparecer na IHM no arquivo MENSAGENS.txt dentro da pasta MDK-IHM que já está na sua pasta MDK que foi copiada.

Número da Mensagem * A Mensagem MS 1 será a tela inicial da IHM.

```

MENSAGENS.txt - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
MS 1 :      "      JNA
           "      ELETRONICA
MS 9 :      "Parametros"
MS 2 :      "Temperatura Max."
MS 3 :      18 #  "-> %F gC"
           20 #  "-> %I gC"
MS 4 :      "Temperatura "
           12 #  "NTC: %F gC"
MS 5 :      14 #  "A1: %FV"
           12 #  "NTC: %FgC"
MS 6 :      16 #  "CONTADOR: %I"
MS 7 :      "Ver Variaveis"
MS 8 :      22 #  "Ligar S4"
           " %B"

MENU
> MS 1
> MS 9
Submenu -> >> MS 2 set_num 18 F
          >> MS 3 set_num 20 I
          > MS 7
          >> MS 4
          >> MS 6
          >> MS 5
          > MS 8 set_num 22 B
END_MENU
    
```

Mensagem entre aspas ("")
 Primeira linha do display
 Segunda linha do display

Se a mensagem mostrar um valor, deve-se informar o tipo do valor

Bloco em que se encontra o valor a ser mostrado

Número da mensagem para o primeiro item do menu

2. Defina a estrutura do menu desejado para a IHM. Utilize '>' para menu principal e '>>' para submenu.

Use o comando:
 set_num [BLOCO] [TIPO]
 para possibilitar que o valor de algum bloco seja alterado pelo menu da IHM.

Em Bloco escolha uma posição entre 10 e 27.

O Tipo deve ser: B para alterar um valor do tipo BOOL, I para um do tipo INT e F para um valor decimal.

3. Conecte o cabo serial na IHM;
 4. Execute o programa COMPILAR E GRAVAR IHM;

5. Desligue e ligue a IHM. A gravação deve iniciar imediatamente, ao terminar, desligue e ligue novamente a IHM;

A IHM contém 27 blocos em sua memória. Os blocos de 0 a 10 são blocos especiais e os demais são de uso geral.

Alterando o valor desses blocos é possível:

BLOCO 0: Selecionar a mensagem que aparecerá no display, se não for alterado ou permanecer em '0' a IHM mostrará o menu de navegação definido no arquivo MENSAGENS.txt;

BLOCO 1: Controlar o BackLight do Display LCD, 0 para desligado e 1 para ligado;
 Com os blocos de uso geral o usuário pode copiar valores de qualquer variável em seu programa e enviar para o display.

6. Utilize normalmente no PLCOpenEditor.

COMO ACESSAR OS BLOCOS A PARTIR DO PLCOpenEditor?

				Número do bloco
18	IHM_12	Local	INT	12#12
19	IHM_14	Local	INT	12#14
20	IHM_16	Local	INT	12#16
21	IHM_22	Local	BOOL	12#22

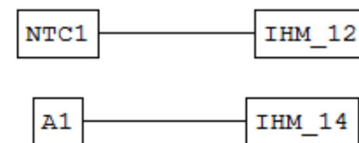
O endereço da IHM é sempre 12



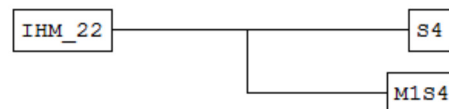
CARACTERÍSTICAS

- Alimentação 12/24 Volts;
- Dimensões: 16,0x5,5 Profundidade: 5,5
- 2 Saídas a Relés;
- Conexão RS485;
- Conexão RS232;
- Teclado com 4 teclas;
- Display de Caracteres LCD.

Copiando valores para os blocos

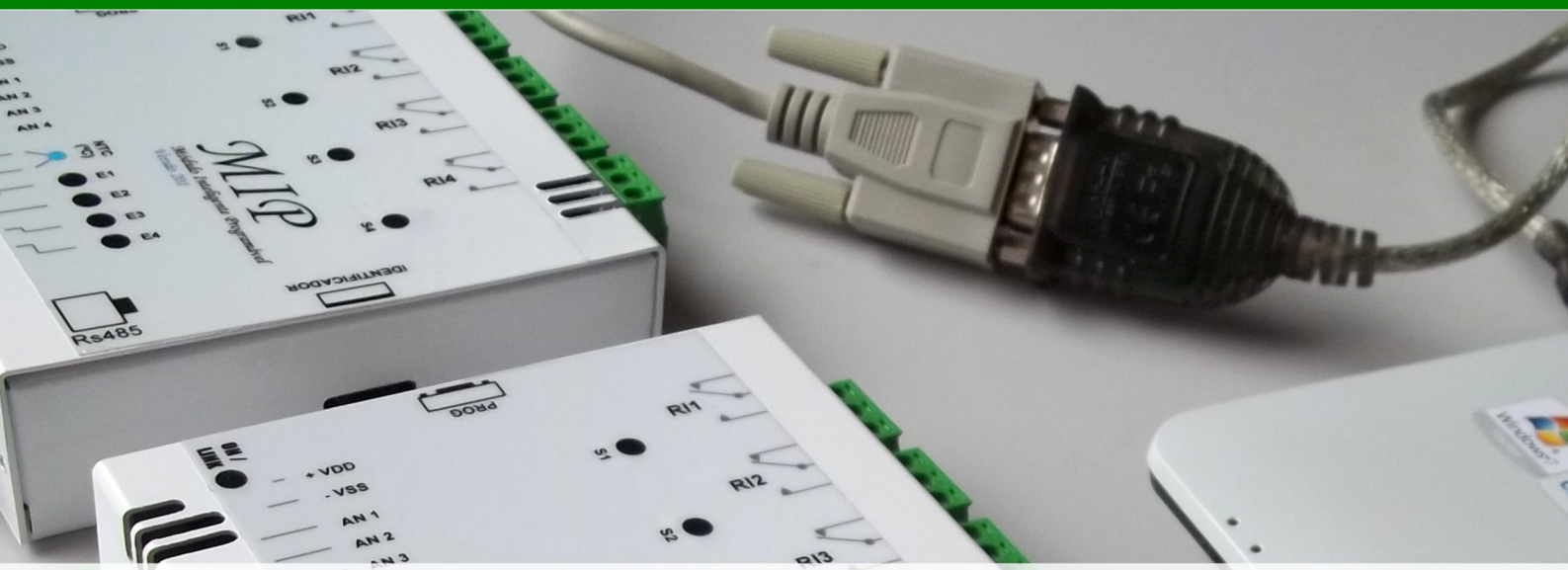


Buscando um valor que foi setado pelo usuário na IHM

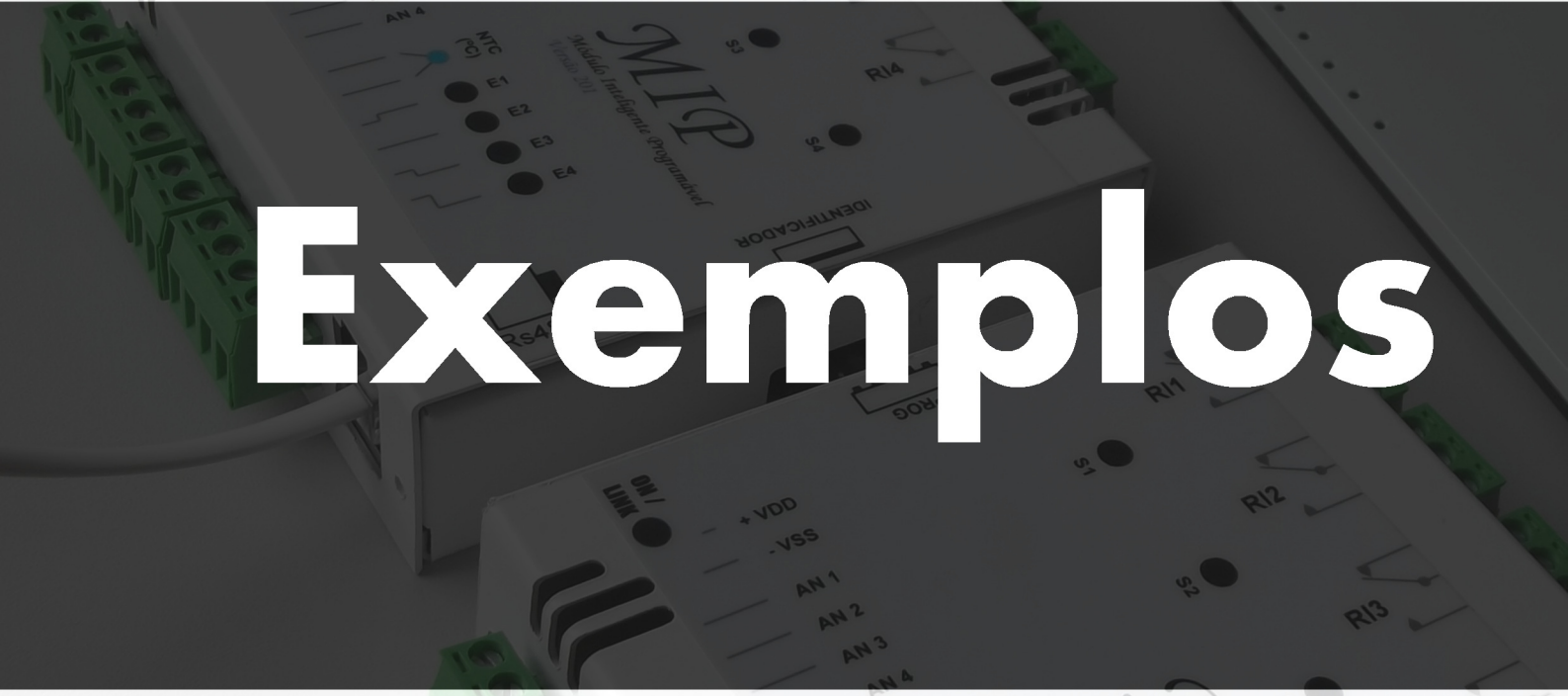


MIP

Módulo Programável



Exemplos

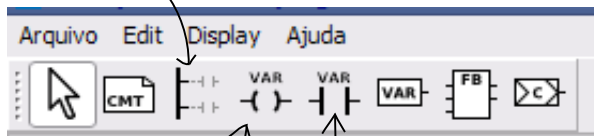


Insira as variáveis na tabela de variáveis como a seguir. Não é necessário manter a ordem de inserção.

Variables							
#	Name	Class	Type	Location	Initial Value	Option	Documentation
2	E1	Local	BOOL	0#14			
3	S1	Local	BOOL	0#10			

Abrir ou Fechar um circuito em LADDER (POWER RAIL)

Abra um circuito ladder clicando no ícone "Power Rail". Em seguida clique na área de desenho para inserir o "Power Rail".



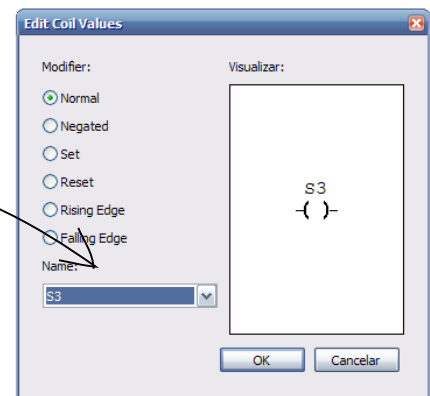
Inserir uma saída em um diagrama LADDER (COIL)

Inserir uma entrada em um diagrama LADDER (CONTACT)

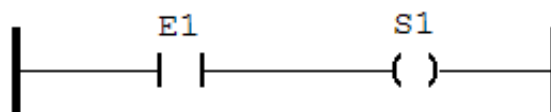
Clique no ícone "CONTACT" e em seguida sobre a área de desenho. Clique duas vezes no contato inserido e selecione "E1" na opção "Name"

Clique no ícone "COIL" e em seguida sobre a área de desenho. Clique duas vezes no coil inserido e selecione "S1" na opção "Name"

Em seguida insira outro "Power Rail" na área de desenho, clique duas vezes sobre ele e selecione "Right".



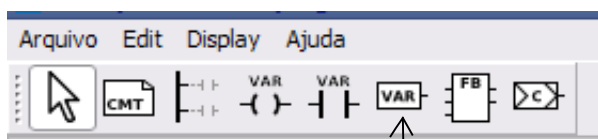
O programa deverá ficar assim:



PLCOpenEditor - FBD (Function Block Diagram)

Insira as variáveis na tabela de variáveis como a seguir. Não é necessário manter a ordem de inserção.

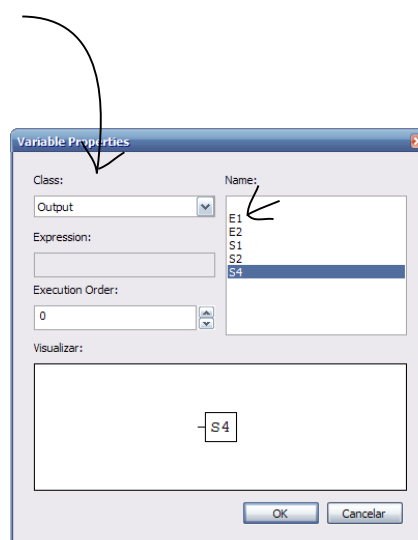
Variables							
#	Name	Class	Type	Location	Initial Value	Option	Documentation
2	E1	Local	BOOL	0#14			
3	S1	Local	BOOL	0#10			



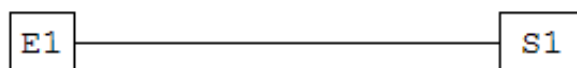
Inserir uma E/S em um diagrama FBD (VAR)

Clique no ícone "VAR" e em seguida sobre a área de desenho. Clique duas vezes no item inserido e selecione "E1" na opção "Name" e em "Class" selecione Input, clique em OK.

Insira outra "VAR" no diagrama dessa vez selecionando S1 em "Name" e Output em "Class". Clique em OK.



O programa deverá ficar assim:



Esse programa faz a mesma função que o anterior mas foi construído em FBD.

Grave o programa para testar.

Gravando o Programa no Módulo

Após criar o programa no PLCOpenEditor, vá em *Arquivo > Generate Program*. Na tela que aparecer salve sobre o arquivo "meuprograma.st" da sua pasta MDK de trabalho.

Vá no CD e abra o programa "MIP PROG".

*Você pode copiar o MIP PROG para o seu computador se desejar. Clique em "Abrir Arquivo para Compilar" e selecione o arquivo que foi gerado "meuprograma.st". Clique em Gravar.

*O módulo MIP deve estar conectado e ligado para que possa gravar um programa.



Quando a gravação terminar, o módulo o executará imediatamente.

*Para gravar novamente o mesmo programa, basta clicar novamente em "Gravar", o MIP PROG irá carregar novamente o arquivo gerado.