

CONTROLADOR LÓGICO PROGRAMÁVEL NANOLOGIX



MANUAL DO USUÁRIO NANOLOGIX

Série: NANO – 002

Ref. 0405.002.N04

Ver. 0411.002.001

Todos os direitos reservados. Proibida a reprodução total ou parcial deste manual sem autorização por escrito da Keylogix.

As informações contidas neste documento apresentam caráter exclusivamente técnico/informativo e a Keylogix, sem qualquer aviso prévio, estará no direito de alterar as informações deste documento caso necessite.

Sumário

1.	Introdução	8
1.1.	Requisitos Mínimos da Ferramenta KeyProgram	9
1.2.	Configurações Recomendadas	9
2.	Especificações do Equipamento	10
2.1.	Dimensões.....	10
2.2.	Alimentação	10
2.3.	LED's de Sinalizações	11
2.3.1.	LED ON – Indicador de equipamento ligado	11
2.3.2.	LED RUN – Indicador de Lógicas sendo processadas .	11
2.3.3.	LED PG – Indicador de Programação sendo enviada...	12
2.4.	Entradas Digitais	13
2.4.1.	Entrada Tipo N (24Vcc).....	13
2.4.2.	Características das Entradas	14
2.4.3.	Endereçamento das Entradas	14
2.5.	Saídas Digitais	16
2.5.1.	Saídas a Relé	16
2.5.2.	Características das Saídas	17
2.5.3.	Endereçamento das Saídas	18
2.6.	Comunicação.....	18
2.6.1.	Modo de Comunicação:	18
3.	Endereçamento.....	21
4.	Instruções de Programação Ladder	24
4.1.	Contatos	24
4.1.1.	Contato Normalmente Aberto	24
4.1.2.	Contato Normalmente Fechado	25
4.1.3.	Contato por borda positiva.....	26
4.1.4.	Contato por borda negativa.....	27
4.1.5.	Parâmetro dos Contatos:	28
4.2.	Bobinas	28
4.2.1.	Saída Simples.....	28
4.2.2.	Saída Complementar	29
4.2.3.	Saída Set e Reset.....	31
4.2.4.	Parâmetro das Saídas:	32
4.3.	Timers (Temporizadores).....	32

4.3.1.	Timer Tipo TON	32
4.3.2.	Parâmetro do Timer ON:	35
4.4.	Contadores	37
4.4.1.	Contador UP	37
4.4.2.	Parâmetro do Contador UP:	37
4.4.3.	Tipos de operandos aceitos:	39
4.5.	Instruções de Comparação	40
4.5.1.	IGUAL (= =).....	40
4.5.2.	MAIOR QUE (>).....	40
4.5.3.	MENOR QUE (<).....	41
4.5.4.	MAIOR OU IGUAL QUE (> =).....	41
4.5.5.	MENOR OU IGUAL QUE (< =).....	42
4.5.6.	NÃO IGUAL (< >)	42
4.6.	Operações Matemáticas	43
4.6.1.	Soma	43
4.6.2.	Subtração	43
4.6.3.	Multiplicação	43
4.6.4.	Divisão	44
4.6.5.	Move	44
5.	Ferramenta KeyProgram	45
5.1.	Visão Geral do Keyprogram	45
5.2.	Instalando o Keyprogram.....	45
5.3.	Iniciando o Keyprogram	49
5.4.	Tela Inicial	50
5.4.1.	Janela de Configuração Inicial.....	51
5.4.2.	Criando um Projeto	52
5.4.3.	Abrindo um Projeto.....	53
5.4.4.	Salvando um Projeto	54
5.5.	Área de Trabalho do Keyprogram	55
5.5.1.	Barra de Menus	56
5.5.1.A.	Menu Projeto.....	57
5.5.1.B.	Menu Edição	58
5.5.1.C.	Menu Exibir	59
5.5.1.D.	Menu Ferramenta:.....	61
5.5.1.E.	Menu CLP:.....	61
5.5.1.F.	Menu Configurações:.....	62
5.5.2.	Barra de Ferramentas	63

5.6.	Criando e executando uma Lógica	65
5.7.	Áreas de Programação do Keyprogram	69
5.7.1.	Área de Programação Ladder.....	69
•	Descrição da Barra de Ferramentas (Ladder).....	70
	Menu de Edição	70
✓	Grupo Geral	70
✓	Grupo de Contatos	70
✓	Grupo de Bobinas	71
✓	Grupo de Temporizadores.....	71
✓	Grupo de Operações Matemáticas	72
✓	Grupo de Comparadores	72
5.8.	Tabelas de Símbolos	73
5.9.	Sub-rotinas	74
5.10.	Configuração da Comunicação	77
5.11.	Instruções Válidas para o NanoLogix.....	78
6.	Glossário	79

Este capítulo apresenta informações sobre o conteúdo deste manual e suas funcionalidades, descreve as principais características do PLC NanoLogix.

Este manual está dividido em 6 capítulos, que seguem uma seqüência lógica com o propósito de facilitar o usuário na instalação, operação e programação de seu equipamento NanoLogix.

Capítulo1 - **Introdução:** Apresenta as informações gerais do produto e sua utilidade no Ambiente Industrial.

Capítulo2 - **Especificações:** Descreve as informações sucintas por meio de figuras, sobre as demais instalações do PLC e suas características principais.

Capítulo3 - **Endereçamento:** fornece a tabela geral do endereçamento utilizado na Lógica de Programação LADDER.

Capítulo4 - **Programação:** Exibe as Instruções de Programação utilizadas na Lógica de Programação LADDER.

Capítulo5 - **Keyprogram:** Apresenta a Ferramenta Keyprogram, descrevendo a instalação, os menus, etc., para a correta utilização do Software.

Capítulo6 - **Glossário:** Descreve alguns dos termos utilizados na programação e utilização deste tipo de sistema.

Suporte Técnico

Dúvidas, críticas ou sugestões sobre este manual poderão ser relatadas ao nosso Suporte Técnico através de um de nossos representantes.

Versões do Manual

As versões deste manual são referenciadas a partir do modelo do equipamento e da versão do software KEYPROGRAM indicadas na contra-capa deste manual como ref. AAMM.EEE.SSS. Onde AA significa o ano do manual; MM, o mês do manual; EEE o código do equipamento; SSS a versão do software.

Das versões do software (m04) até as mais atuais, poucas modificações sofrerão no manual, já que estas atualizações são referentes às correções do software, o que não caracteriza nenhuma alteração física no equipamento. Modificações relevantes serão relatadas em um arquivo anexo disponibilizado na Internet com a data de sua atualização.

1. Introdução

O PLC NANOLOGIX é um dos mais viáveis em relação custo benefício da nossa linha de PLC's Industriais fabricados atualmente.

Possui 10 entradas digitais (tipo N), 6 saídas digitais (a relé), memória RAM de 32KB, EEPROM de 2KB e FLASH de 8KB, Circuito de Watch-Dog, um canal de comunicação serial RS232; ou seja, um equipamento compacto ideal para suas aplicações Industriais de pequeno porte.

Construído com o objetivo de ser inserido no ambiente industrial de Processos de Produção para aumentar a velocidade e controlar suas produções o PLC NANOLOGIX adaptou o dinamismo, controle e robustez em um equipamento compacto ideal para pequenas aplicações. Nosso equipamento veio para substituir os sistemas a Relés (painéis eletromecânicos) trazendo as vantagens de:

- . Menor espaço na instalação;
- . Menor custo na implementação do sistema;
- . Menor consumo de energia;
- . Maior tempo de vida útil do equipamento;
- . Menor número de fios;
- . Facilidade de Manutenção;
- . Facilidade de Alterações;
- . Mais produção em menos tempo;
- . Facilidade de Implementações;
- . Muito menor custo total.

O PLC NANOLOGIX disponibiliza uma ferramenta de programação em LADDER, capaz de se comunicar e transferir dados através da interface serial RS-232. Sua programação é realizada através do KEYPROGRAM, um aplicativo fácil de usar, desenvolvido para ambiente Windows 98, Me e XP, que pode ser baixado gratuitamente pela Internet.

1.1. Requisitos Mínimos da Ferramenta KeyProgram

Processador: 133 MHz com 32 MB de RAM.

Resolução de Vídeo 800x600 pixels (fontes pequenas)

Sistema Operacional: Windows 95, 98, ME ou XP.

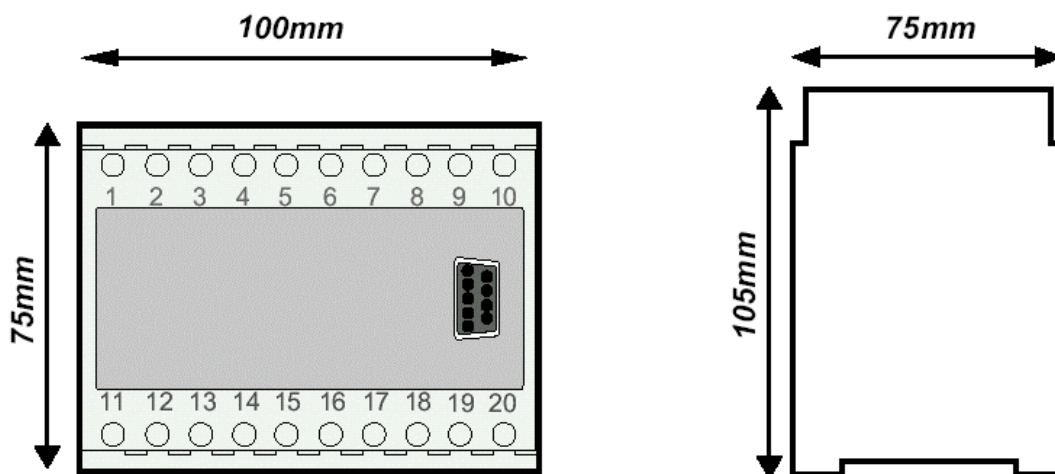
1.2. Configurações Recomendadas

Processador: 400 MHz com 64 MB de RAM ou superior.

Sistema Operacional: Windows 98, ME ou XP.

2. Especificações do Equipamento

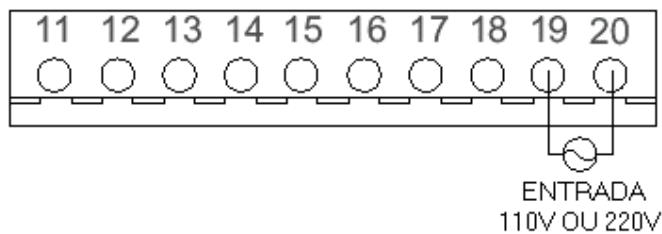
2.1. Dimensões



2.2. Alimentação

Os pinos (19 e 20) devem ser alimentados com 110 VAC ou 220 VAC dependendo do modelo do aparelho.

É apresentada abaixo, a figura dos pinos que devem ser alimentados para a ligação do PLC.

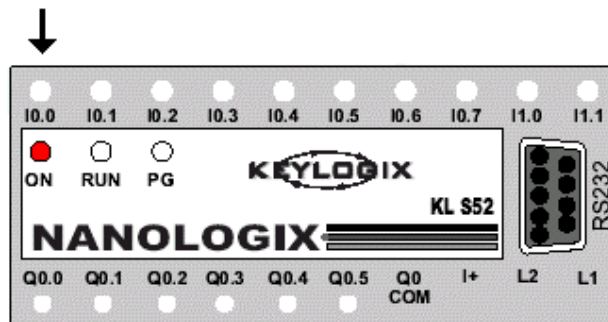


Obs: Verificar a voltagem do seu equipamento antes de ligar.

2.3. LED's de Sinalizações

2.3.1. LED ON – Indicador de equipamento ligado

Quando a entrada do equipamento é alimentada, o LED ON, localizado na parte frontal do PLC, indicará que o PLC está ligado. É mostrada abaixo a localização do LED indicador de equipamento ligado:

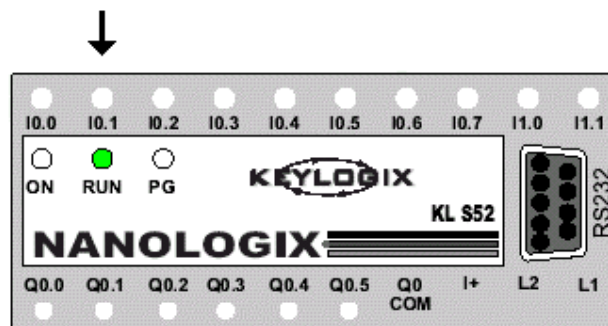


LED indicador de equipamento ligado

2.3.2. LED RUN – Indicador de Lógicas sendo processadas

Quando LED RUN estiver oscilando (ligando e desligando), por aproximadamente um segundo, isto indicará que o PLC está em seu modo de execução, processando uma ou mais lógicas.

O LED RUN, de cor verde, é ilustrado na figura abaixo.



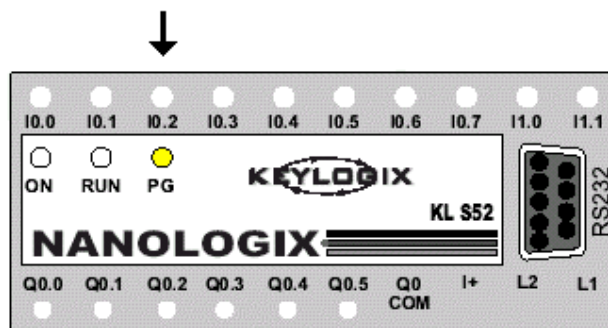
LED indicador de Lógicas sendo processadas

2.3.3. LED PG – Indicador de Programação sendo enviada

Quando LED PG estiver acendendo e apagando por aproximadamente meio segundo, isto indicará que o PLC está em seu modo de programação, fazendo o envio de uma ou mais lógicas que estão no PC para o PLC.

Neste modo o PLC não irá executar lógica alguma, pois o modo de programação apenas interrompe o modo de execução do PLC (modo RUN).

O LED PG, de cor amarela, é ilustrado na figura abaixo.



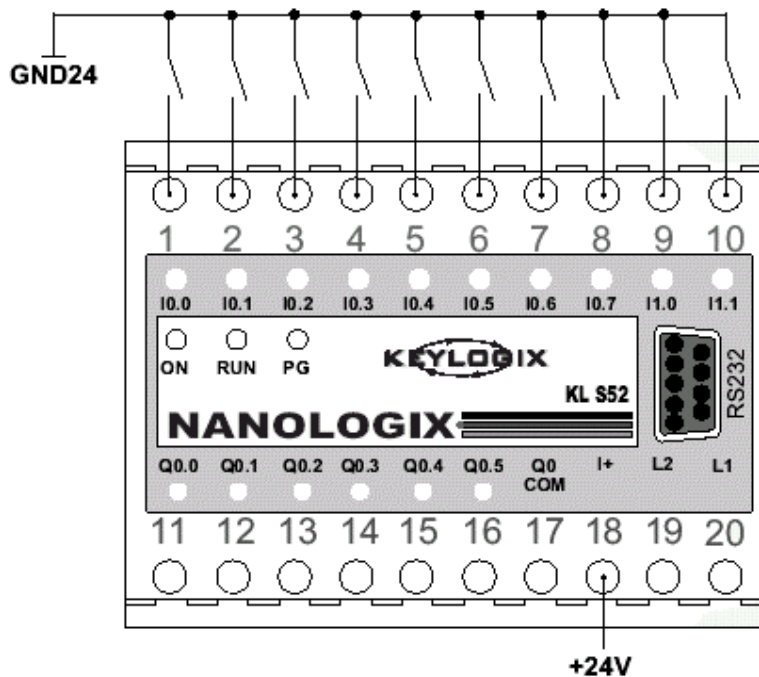
LED indicador de Programação sendo enviada.

2.4. Entradas Digitais

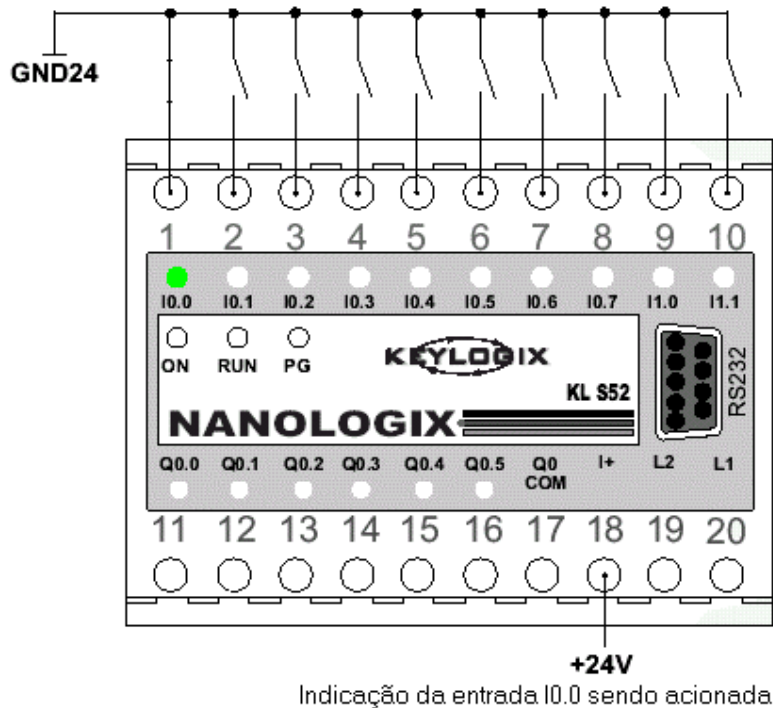
2.4.1. Entrada Tipo N (24Vcc)

A entradas Tipo N são acionadas quando são alimentadas por uma tensão de 24 VDC sendo que o + 24 VDC deve ser conectado ao comum I+ das entradas (pino 18) e os demais pinos (1 ao 10), correspondentes as entradas que se desejam acionar, devem ser ligados no GND dos 24VDC.

É indicado do desenho abaixo:



Quando a entrada é acionada, o LED verde da entrada correspondente ficará aceso, indicando que a entrada está sendo acionada. É ilustrado abaixo o esquema de acionamento de uma entrada.



2.4.2. Características das Entradas

Range de tensão	24Vdc (19,5 à 32Vdc)
Corrente por entrada	10mA em 24Vdc
Isolação	1.500 V
Led Indicador de status	Verde
Quantidade	10

2.4.3. Endereçamento das Entradas

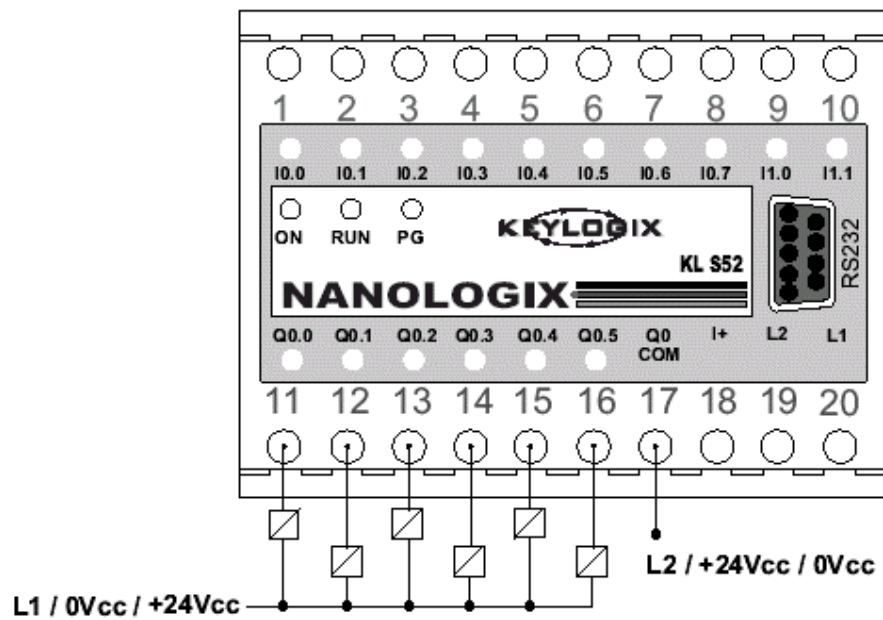
Variável	Endereço	Ordem da Entrada	Descrição
%IO.0	0100h	1ª Entrada	Bits referentes às Entradas Digitais
%IO.1	0101h	2ª Entrada	
%IO.2	0102h	3ª Entrada	
%IO.3	0103h	4ª Entrada	
%IO.4	0104h	5ª Entrada	
%IO.5	0105h	6ª Entrada	
%IO.6	0106h	7ª Entrada	
%IO.7	0107h	8ª Entrada	
%I1.0	0108h	9ª Entrada	
%I1.1	0109h	10ª Entrada	

2.5. Saídas Digitais

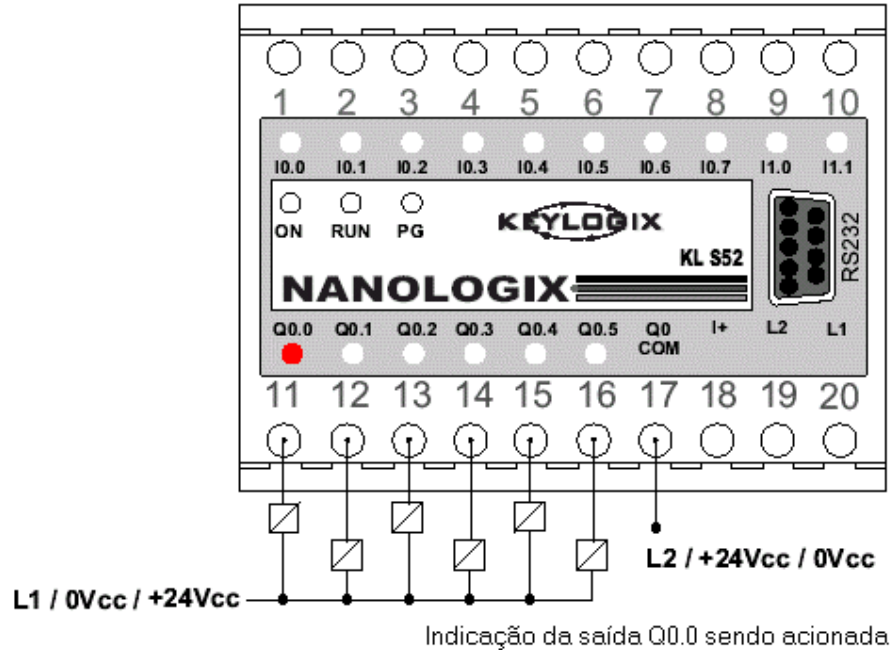
2.5.1. Saídas a Relé

As saídas a relé serão acionadas ou desacionadas de acordo com a lógica que o usuário a programou. As saídas apresentam os contatos comuns, pino 17 (Q0 COM) e pelos contatos Normalmente abertos, pino 11 ao 16 (Q0.0, Q0.1, Q0.2, Q0.3, Q0.4, Q0.5 respectivamente).

Abaixo é representada a forma de ligações das saídas:



Quando a saída é acionada, o LED vermelho das saída correspondente ficará aceso indicando que a saída está sendo acionada. É ilustrado abaixo o esquema de acionamento de uma saída.



Os fechamentos dos contatos deverão ser acionados por algum atuador (dispositivo externo) como contatores, solenóides, relés, lâmpada, indicadores, etc., aceitando valores máximos de tensão e corrente de acordo com a tabela abaixo:

2.5.2. Características das Saídas

Capacidade do Contato		
Corrente de Comutação	2A	3A
Tensão de Comutação	30VCC	250VCA

2.5.3. Endereçamento das Saídas

Variável	Endereço	Ordem da Saída	Descrição
%Q0.0	0110h	1 ^a Saída	Bits referentes às Saídas Digitais
%Q0.1	0111h	2 ^a Saída	
%Q0.2	0112h	3 ^a Saída	
%Q0.3	0113h	4 ^a Saída	
%Q0.4	0114h	5 ^a Saída	
%Q0.5	0115h	6 ^a Saída	

2.6. Comunicação

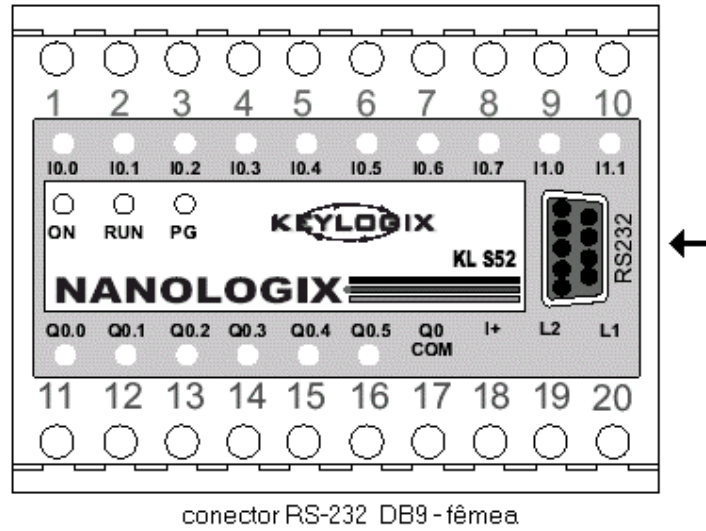
2.6.1. Modo de Comunicação:

A programação e monitoração do CLP são realizadas por meio de um PC, utilizando como meio de comunicação a interface serial RS-232.

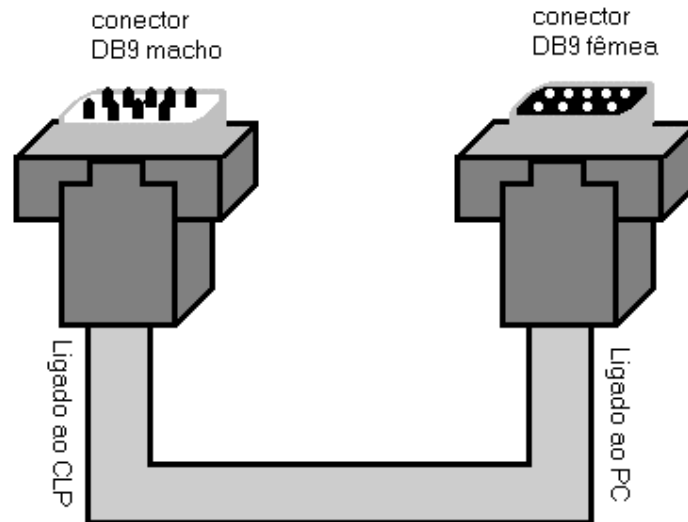
A interface RS-232 utiliza o protocolo de comunicação Keynet para a comunicação entre o CLP e o PC. A plataforma utilizada para programação do CLP é efetuada através do Software Keyprogram.

A taxa de transmissão (Baud Rate) utilizada na comunicação de nosso equipamento, ao PC é de 115200 bps - taxa máxima suportada pelo meio físico.

O equipamento utiliza um conector DB9 (fêmea) para comunicação RS-232 como é mostrado na figura abaixo:

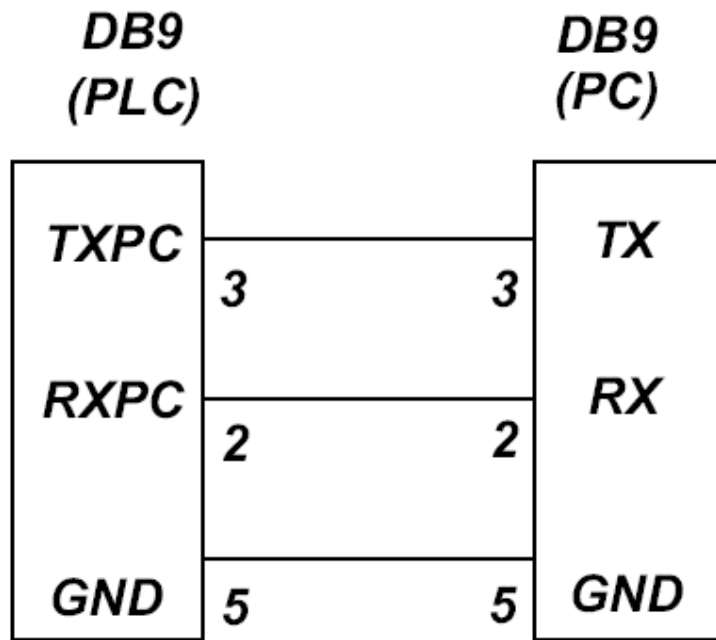


O cabo de comunicação para conectar no equipamento e no PC:



O comprimento máximo na Interface RS-232 do cabo de comunicação do PLC deve ser de no máximo 10 m.

A pinagem do PLC NANOLOGIX obedece à mesma pinagem da porta do PC, em outras palavras:



3. Endereçamento

São apresentados os endereços pertencentes ao NANOLOGIX na tabela a seguir:

Endereços Externos					
Variável	Endereço	Ordem da Entrada	Descrição	Tamanho	Qtde.
%I0.0	0100h	1 ^a Entrada	Bits referentes às Entradas Digitais	1 Bit (Nível 1 ou Nível 0)	10
%I0.1	0101h	2 ^a Entrada			
%I0.2	0102h	3 ^a Entrada			
%I0.3	0103h	4 ^a Entrada			
%I0.4	0104h	5 ^a Entrada			
%I0.5	0105h	6 ^a Entrada			
%I0.6	0106h	7 ^a Entrada			
%I0.7	0107h	8 ^a Entrada			
%I1.0	0108h	9 ^a Entrada			
%I1.1	0109h	10 ^a Entrada			
Variável	Endereço	Ordem da Saída	Descrição	Tamanho	Qtde.
%Q0.0	0110h	1 ^a Saída	Bits referentes às Saídas Digitais	1 Bit (Nível 1 ou Nível 0)	6
%Q0.1	0111h	2 ^a Saída			
%Q0.2	0112h	3 ^a Saída			
%Q0.3	0113h	4 ^a Saída			
%Q0.4	0114h	5 ^a Saída			
%Q0.5	0115h	6 ^a Saída			

Endereços Internos				
Variável	Endereço	Descrição	Tamanho	Qtde.
%T0.Q ~ %T31.Q	0052h 0091h	Saída dos Timers	1 bit	32
%C0.Q ~ %C31.Q	03B8h 03D8h	Saída dos Contadores	1 bit	32
%M0 ~ %M255	0400h 05FFh	Endereço dos bits de memória	1 bit	256
%MB0 ~ %MB255	0600h 07FFh	Endereço dos bytes de memória	1 byte	256
%MR0 ~ %MRB255	0800h 09FFh	Endereço dos bits retentivos	1 bit	16
%MBR0 ~ %MBR255	0A00h 0BFFh	Endereço dos bytes retentivos	1 byte	16
%S0 ~ %S255	0C00h 0CFFh	Endereço dos bits de sistema Ver detalhes na tabela bits de Sistema	1 bit	256
%SB0 ~ %SB255	0D00h 0DFFh	Endereço dos bytes de sistema Ver detalhes na tabela bytes de Sistema	1 byte	256
%MW0 ~ %MW255	0E00h 0EFFh	Endereço de %MW	2 bytes	256
%MWR0 ~ %MW255	1000h 10FFh	Endereço de %MWR	2 bytes	256
%T0.V ~ %T31.V	1200h 123Fh	Contagem Atual dos Timers	2 bytes	32
%T0.P ~ %T31.P	1280h 12BFh	Valor de Preset dos Timers	2 bytes	32
%C0.V ~ %C31.V	01300h 0131Fh	Contagem Atual dos Contadores	2 bytes	32
%C0.P ~ %C31.P	01380H 0139Fh	Valor de Preset dos Contadores	2 bytes	32
%SW0 ~ %SW255	01400h 014FFh	Endereço de Word's de sistema 256 Ver detalhes na tabela do item 7	2 bytes	256

Bits de Sistema	
Bit	Descrição
S0	Sempre Ligado
S1	Sempre Desligado
S2	Ligado Na Primeira Varredura
S3	Desligado Na Primeira Varredura
S4	Clock de 0.5s (0.5 low, 0.5 High)
S5	Clock de 30s (30 low, 30 High)
S6	Chave de Modo (Run/Prog)
S7	Serial (On-line/Off-line)
S8-S255	RESERVA

Bytes de Sistema	
Byte	Descrição
SB0	SEGUNDO
SB1	MINUTO
SB2	HORA
SB3	DIA
SB4	DATA
SB5	MÊS
SB6	ANO
SB7	TELA
SB9	Endereço do PLC (Comunicação serial)
SB10-SB255	RESERVA

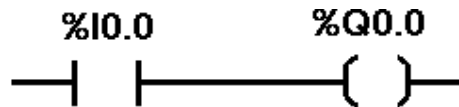
4. Instruções de Programação Ladder

4.1. Contatos

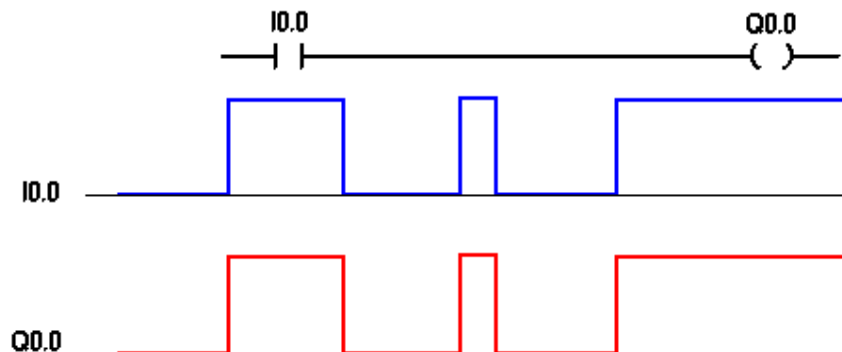
4.1.1. Contato Normalmente Aberto



Descrição: Um contato normalmente aberto é fechado quando o bit associado a ele estiver no estado lógico 1. Caso contrário ele permanece aberto.



No exemplo acima a saída (%Q0.0) vai ser acionada somente quando a entrada (%I0.0) estiver em nível lógico 1.

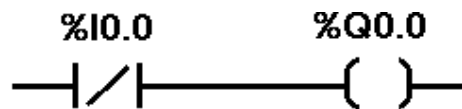


A saída (%Q0.0) ficará acionada quando o contato (%I0.0) permanecer em nível lógico 1. No momento que o contato (%I0.0) estiver em nível lógico 0, a saída (%Q0.0) estará desacionada.

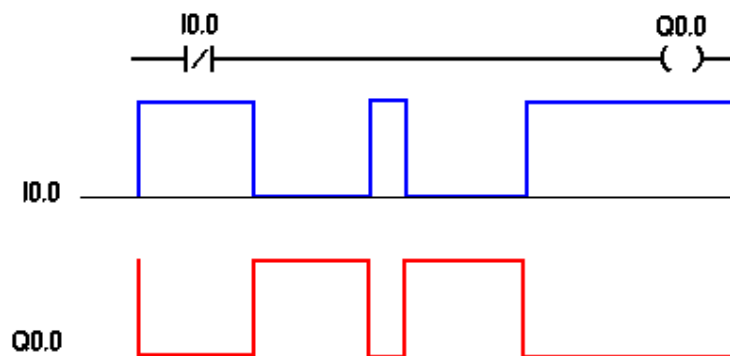
4.1.2. Contato Normalmente Fechado



Descrição: um contato normalmente fechado é aberto quando o bit associado a ele estiver no estado lógico 1. Caso contrário ele permanece fechado (nível lógico 0).



No exemplo acima, a saída (%Q0.0) vai ser acionada somente quando a entrada (%I0.0) estiver com nível lógico 0.

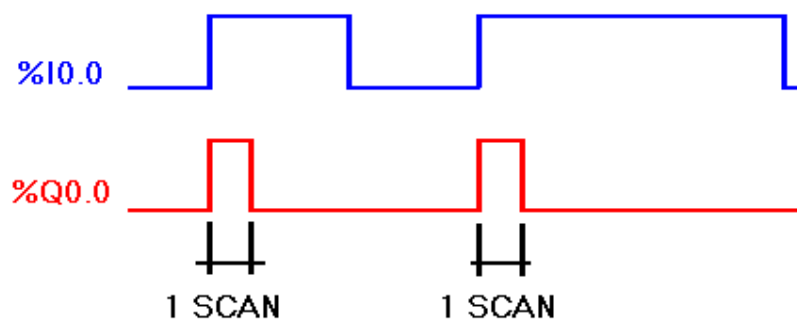
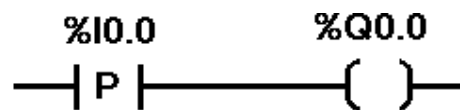


A saída (%Q0.0) permanecerá acionada enquanto a entrada (%I0.0) estiver em nível lógico 0. No momento que o contato (%I0.0) estiver em nível lógico 1, então a saída (%Q0.0) estará desacionada.

4.1.3. Contato por borda positiva



Descrição: um contato por borda positiva gera um pulso em sua saída no período de um scan quando o bit associado a ele passar do estado 0 para estado 1.

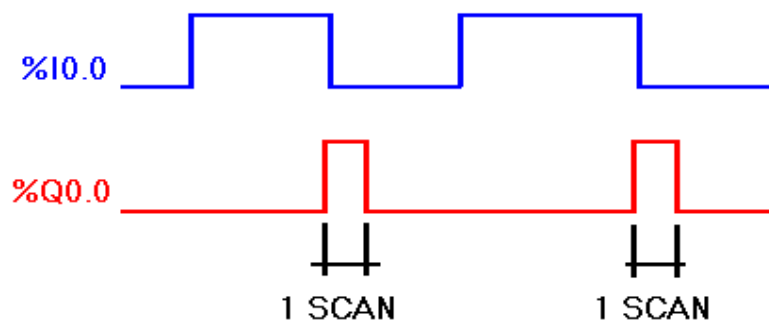
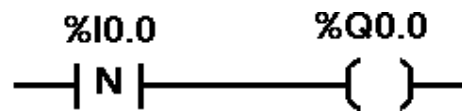


No exemplo do gráfico mostrado acima, a saída (%Q0.0) será acionada quando a entrada (%I0.0) estiver mudando de nível lógico 0 para nível lógico 1, e permanecerá ativada durante o tempo de 1 scan.

4.1.4. Contato por borda negativa



Descrição: um contato por borda negativa gera um pulso em sua saída de período de um scan quando o bit associado a ele passar do estado 1 para estado 0.



No exemplo do gráfico mostrado acima, a saída (%Q0.0) será acionada quando a entrada (%I0.0) mudar do nível lógico 1 para o nível lógico 0 , e permanecerá ativada durante o tempo de 1 scan.

4.1.5. *Parâmetro dos Contatos:*

O programador deve utilizar somente bit's, na utilização dos contatos Normalmente aberto, Normalmente Fechado, por Borda Positiva, por Borda Negativa.

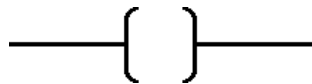
Tipos de Operandos aceitos:

- ❖ %Ixx (Entradas Digitais);
- ❖ %Qxx (Saídas Digitais);
- ❖ %Mxx (Bits de Memória);
- ❖ %MRxx (Bits de Memória Retentiva);
- ❖ %Sxx (Bit de Sistema);
- ❖ %Txx.Q (Saída de um Temporizador);
- ❖ %Cxx.Q (Saída de um Contador);

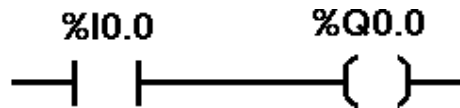
Onde xx é o número do parâmetro ex: I1.0 (9ª entrada)

4.2. Bobinas

4.2.1. *Saída Simples*

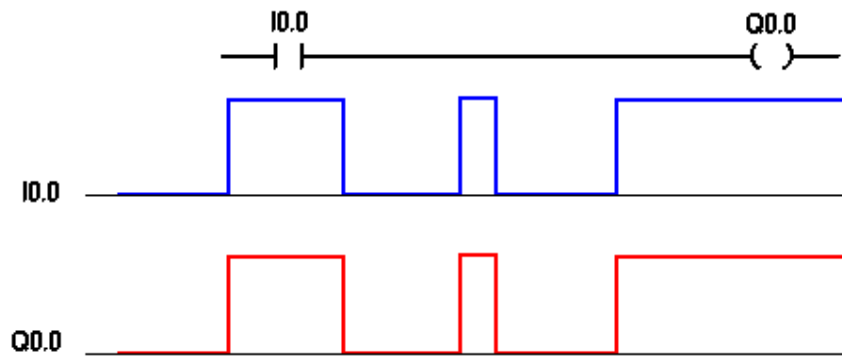


Descrição: uma saída simples transfere para o endereço associado a ela, o valor de sua entrada. Caso a lógica associada à saída estiver em 1 transfere 1 para o endereço, caso contrário transfere 0.



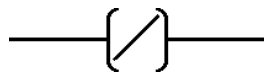
No exemplo acima a saída (%Q0.0) será acionada somente quando a entrada (%I0.0) for acionada.

A saída permanece ativada enquanto a entrada estiver acionada.

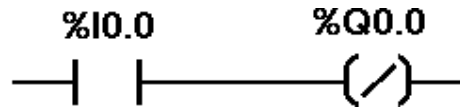


O gráfico acima mostra que a saída (%Q0.0) permanecerá acionado enquanto a entrada (%I0.0) estiver em nível lógico 1. No momento que o contato (%I0.0) estiver em nível lógico 0, então a saída %Q0.0 estará desacionada.

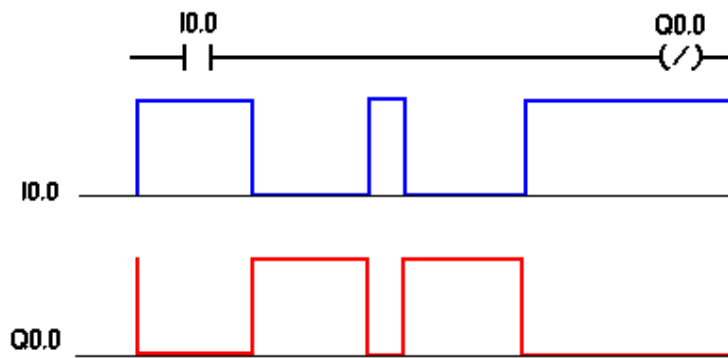
4.2.2. Saída Complementar



Descrição: uma saída complementar transfere para o endereço associado a ela, o inverso do valor de sua entrada. Caso a lógica associada a saída estiver em 0 transfere 1 para o endereço, caso contrário transfere 0.



No exemplo acima a saída (%Q0.0) será acionada somente quando a entrada (%I0.0) estiver desacionada.



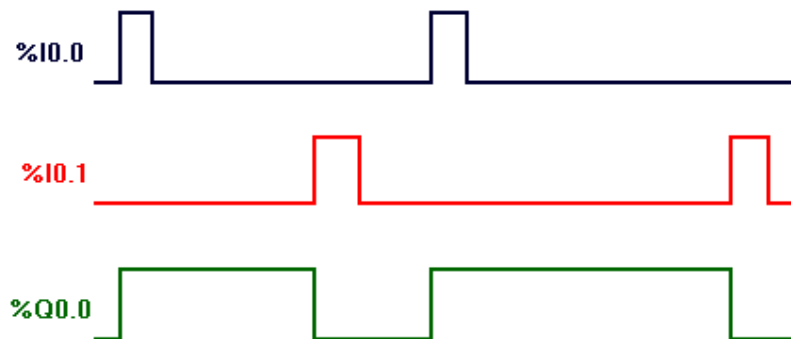
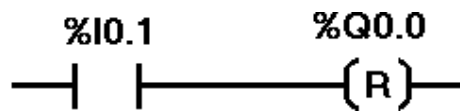
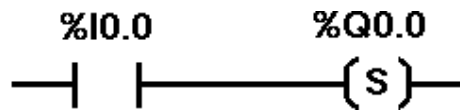
O gráfico acima mostra que a saída (%Q0.0) permanecerá acionado enquanto a entrada (%I0.0) estiver em nível lógico 0. No momento que o contato (%I0.0) estiver em nível lógico 1, então a saída %Q0.0 estará desacionada.

4.2.3. Saída Set e Reset

Saída set ———— (S) ————

Saída reset ———— (R) ————

Descrição: a saída Set sempre é acionada quando há um pulso (transição de nível 0 para nível 1) em sua entrada. A saída permanecerá acionada mesmo que o valor da entrada vá para 0. A saída só irá para 0 através da instrução Reset.



No exemplo acima, a saída (%Q0.0) será acionada e permanecerá acionada quando a entrada (%I0.0) transitar de 0 para 1. Após acionada a saída (%Q0.0) só será desacionada quando a entrada (%I0.1) transitar de nível 0 para 1.

4.2.4. Parâmetro das Saídas:

O programador deve utilizar somente bit's, na utilização da saída simples, complementar, set e reset.

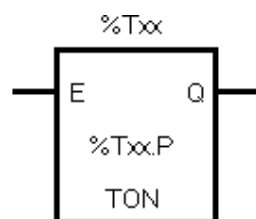
Tipos de Operandos aceitos:

- ❖ %Qxx (Saídas Digitais);
- ❖ %Mxx (Bits de Memória);
- ❖ %MRxx (Bits de Memória Retentiva);
- ❖ %Txx.Q (Saídas dos Temporizadores);
- ❖ %Cxx.Q (Saídas dos Contadores);

Onde xx é o número do parâmetro ex: Q0.4 (5ª saída)

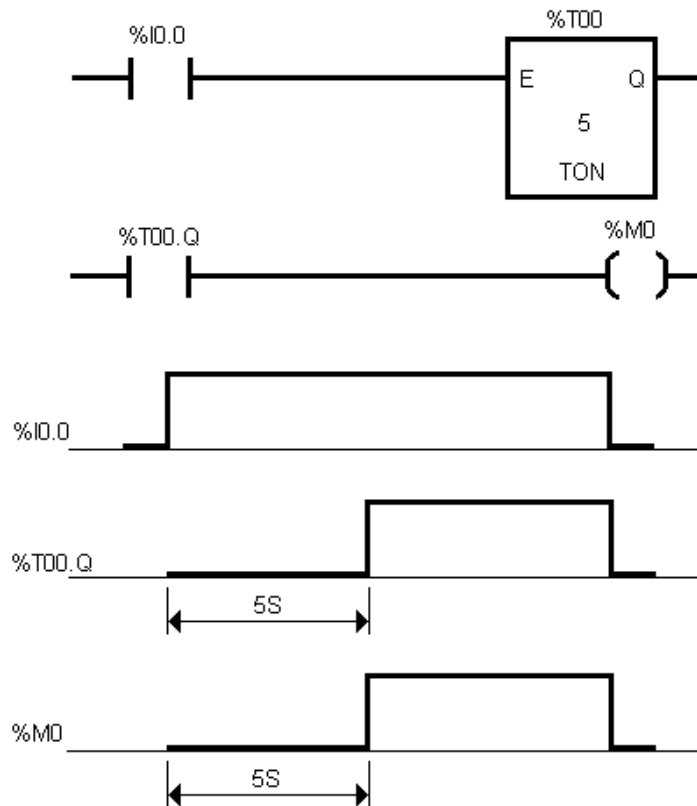
4.3. Timers (Temporizadores)

4.3.1. Timer Tipo TON



Descrição: esta instrução conta um tempo a partir do instante em que a entrada Enable (E) for habilitada. Quando o valor atual do Timer (Txx.V) alcançar o tempo de preset (Txx.P), definido pelo usuário, a saída (Q) do Timer será acionada.

Quando a entrada Enable for desabilitada, o valor atual do Timer será zerado, e a saída Q do Timer será desativada.



No exemplo acima, a saída (Q) do Timer irá acionar enquanto a entrada (%I0.0) permanecer fechada (nível lógico 1) e o tempo atual do timer alcançar o valor do preset, acionando assim, o bit (%M0).

Quando a entrada (%I0.0) estiver aberta (nível lógico 0), imediatamente o valor atual do timer será zerado, irá ser desacionada a saída (Q) do timer, desacionando assim o bit (%M0).

4.3.2. Parâmetro do Timer ON:

O programador ao utilizar os Timer's ON,s deve seguir as seguintes regras:

- ❖ **%Txx:** Número do Timer (T0 a T31), definido pelo usuário.

- ❖ **Timer Base:** Base de Tempo do Timer (1s, 0.1s e 0,01s), definido pelo usuário.

- ❖ **%Txx.P:** Valor do Preset do Timer, definido pelo usuário (Número inteiro de 0 a 65535).

- ❖ **%Txx.V:** Valor Atual do Timer definido por software. (Número inteiro de 0 a 65535).

- ❖ **Q:** Status de saída do Timer definido por software. É ativado quando o valor atual do Timer se igualar ao valor do preset do contador. Bit (0 ou 1).

- ❖ **E:** Enable do Timer de Pulsos, definido pelo usuário. Quando ativado faz a contagem do Timer. Bit (0 ou 1).

Tipos de Operandos aceitos:

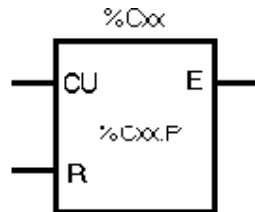
Na saída Q do Timer os parâmetros aceitos são valores que contenham tamanho de 1 bit (nível 0 ou nível 1), ou seja, variáveis que contenham também tamanho de 1 bit. Elas podem ser:

- ❖ **%Qxx (Saídas Digitais);**
- ❖ **%Mxx (Bits de Memória);**
- ❖ **%MRxx (Bits de Memória Retentiva);**
- ❖ **%Txx.Q (Saídas dos Temporizadores);**
- ❖ **%Cxx.Q (Saídas dos Contadores);**

No valor de Preset do Timer os parâmetro aceitos são valores que contenham o tamanho de 1 Word (valor de 0 a 65535).

4.4. Contadores

4.4.1. Contador UP



Descrição: esta instrução faz a contagem de pulsos, com o incremento de uma unidade a cada vez que o contato associado à entrada (CU) muda de estado desligado (nível 0), para estado ligado (nível 1).

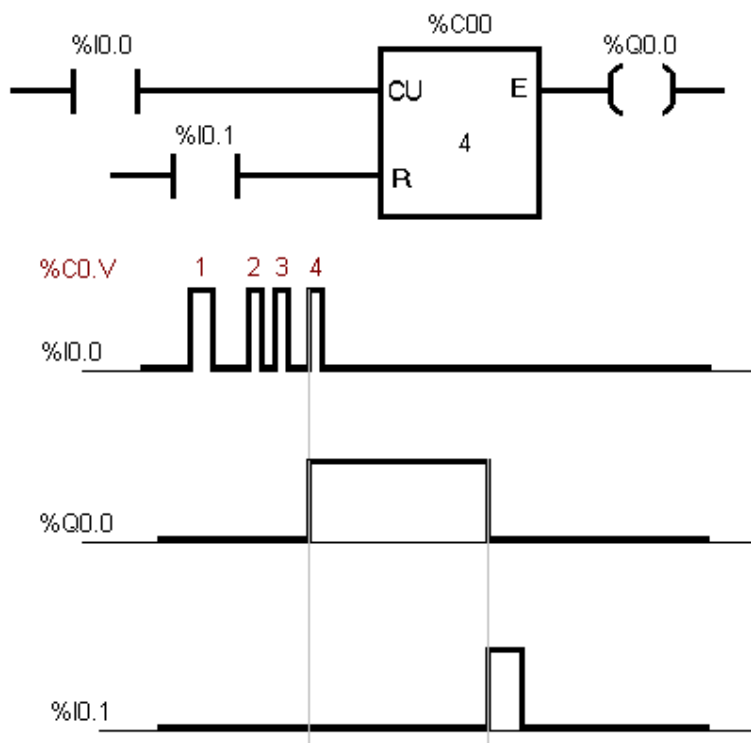
Quando o valor atual do Contador se igualar ao valor de preset então a saída (E) do Contador será acionada.

Quando o Reset (R) do contador for habilitado, o valor do contador atual será igual ao valor zero e a saída (E) do contador será desacionada.

4.4.2. Parâmetro do Contador UP:

- ❖ **%Cxx:** Número do contador (C0 a C31), definido pelo usuário.
- ❖ **%Cxx:P:** Valor do Preset do contador, definido pelo usuário.
(Número inteiro de 0 a 65535).
- ❖ **%Cxx.V:** Valor Atual da contagem definido por software Este valor é incrementado a cada pulso recebido pela entrada (CU) do contador.
(Número inteiro de 0 a 65535).
- ❖ **CU:** Entrada dos pulsos do contador, definido pelo usuário. Bit (0 ou 1).

- ❖ **R:** Reset, definido pelo usuário. Quando ativado, a contagem atual do contador irá apresentar o valor zero e irá desacionar a saída do contador.
Bit (0 ou 1).
- ❖ **E:** Status de saída do contador definido por software. É ativada quando o valor atual de contagem se igualar ao valor do preset do contador.
Bit (0 ou 1).



No exemplo acima, o valor do preset está especificado com o número quatro (4), isto significa que, quando (%I0.0) passar de nível lógico 0 para nível lógico 1 quatro vezes, a saída (E) do contador será acionada. Note que a contagem começa de 0, que é o valor inicial, até o valor do preset do

contador (%Cxx.P), especificado com o número 4. Se a entrada (%I0.1) for acionada, a saída (E) do contador será imediatamente desacionada e o valor atual do contador será zerado .

4.4.3. Tipos de operandos aceitos:

O programador ao utilizar os Contadores UP, deve seguir as seguintes regras:

Na saídas dos Contadores, os parâmetros aceitos são valores que contenham tamanho de 1 bit (nível 0 ou nível 1), ou seja, variáveis que contenham também tamanho de 1 bit. Elas podem ser:

- ❖ **%Qxx (Saídas Digitais);**
- ❖ **%Mxx (Bits de Memória);**
- ❖ **%MRxx (Bits de Memória Retentiva);**
- ❖ **%Txx.Q (Saídas dos Temporizadores);**
- ❖ **%Cxx.Q (Saídas dos Contadores);**
- ❖ **%LTxx (Led da IHM).**

No valor de Preset do Timer os parâmetro aceitos são valores que contenham o tamanho de 1 Word (valores inteiros de 0 a 65535).

4.5. Instruções de Comparação

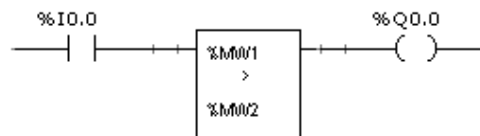
Estas instruções têm como função comparar dois valores, caso estes valores satisfaçam a condição de comparação e a entrada do comparador estiver acionada, então a saída do comparador será acionada, habilitando assim, por exemplo, uma saída.

4.5.1. IGUAL (=)



No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação de igualdade entre o operando 1 e o operando 2. Caso eles sejam iguais, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for diferente ao operando 2, o resultado será nível lógico 0, e conseqüentemente a saída será desacionada.

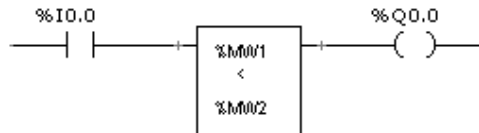
4.5.2. MAIOR QUE (>)



No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação entre o operando 1 e o operando 2. Caso o operando 1 seja maior que o operando 2, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for menor ou igual ao operando 2,

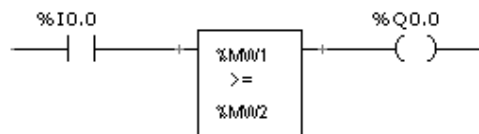
o resultado será nível lógico 0, e conseqüentemente a saída será desacionada.

4.5.3. MENOR QUE (<)



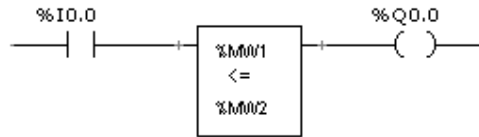
No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação entre o operando 1 e o operando 2. Caso o operando 1 seja menor que o operando 2, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for maior ou igual ao operando 2, o resultado será nível lógico 0, e conseqüentemente a saída será desacionada.

4.5.4. MAIOR OU IGUAL QUE (>=)



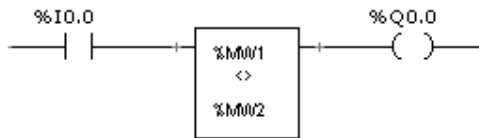
No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação entre o operando 1 e o operando 2. Caso o operando 1 seja maior ou igual ao operando 2, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for menor que operando 2, o resultado será nível lógico 0, e conseqüentemente a saída será desacionada.

4.5.5. MENOR OU IGUAL QUE (<=)



No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação entre o operando 1 e o operando 2. Caso o operando 1 seja menor ou igual ao operando 2, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for maior que o operando 2, o resultado será nível lógico 0, e conseqüentemente a saída será desacionada.

4.5.6. NÃO IGUAL (<>)

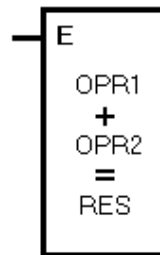


No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação de igualdade entre o operando 1 e o operando 2. Caso eles sejam diferentes, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for igual ao operando 2, o resultado será nível lógico 0, e conseqüentemente a saída será desacionada.

4.6. Operações Matemáticas

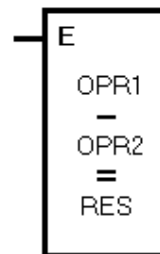
Estas instruções têm como função executar operações aritméticas de dois operandos e colocá-lo em um operando de resposta (RES).

4.6.1. Soma



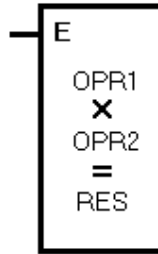
Quando habilitado através da entrada E, executa a soma de OPR1 + OPR2 e coloca o resultado em RES.

4.6.2. Subtração



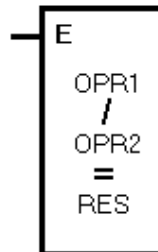
Quando habilitado através da entrada E, executa a Subtração em OPR1 do valor de OPR2 e coloca o resultado em RES.

4.6.3. Multiplicação



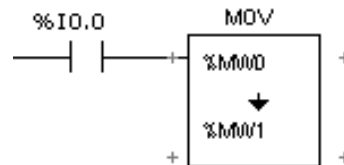
Quando habilitado através da entrada E, executa a multiplicação de OPR1 por OPR2 e coloca o resultado em RES.

4.6.4. Divisão



Quando habilitado através da entrada E, executa a divisão de OPR1 por OPR2 e coloca o resultado em RES.

4.6.5. Move



Quando habilitado, através de um bit de entrada, transfere o valor contido na Variável Origem (representado na figura como MW0), para a variável Destino (representado na figura como MW1).

5. Ferramenta KeyProgram

5.1. Visão Geral do Keyprogram

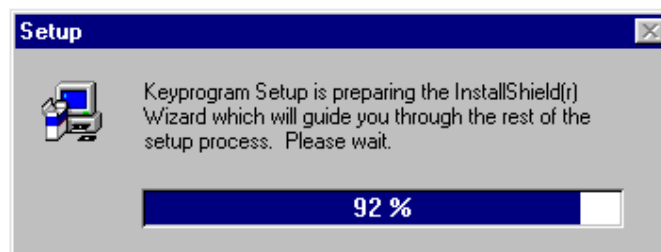
A principal importância da ferramenta Keyprogram é facilitar a programação Ladder em uma plataforma simples, ágil e amigável.

O Keyprogram foi desenvolvido utilizando alguns recursos que o Windows oferece, buscando tornar a sua utilização bem interativa. Portanto, para utilizá-lo não é necessário ser um “expert” em informática, basta ter os conhecimentos básicos para operá-lo através de botões, menus, teclas de atalho e janelas de configuração.

5.2. Instalando o Keyprogram

Para instalar o Keyprogram

1. Certificar-se de possuir o Microsoft Windows 98, ou Windows XP, instalado em seu computador.
2. Obter o software Keyprogram através do endereço eletrônico www.keylogix.com.br
3. Executar o aplicativo keyprogram.
4. Irá aparecer uma tela como é representado abaixo, indicando o progresso de carregamento do software:

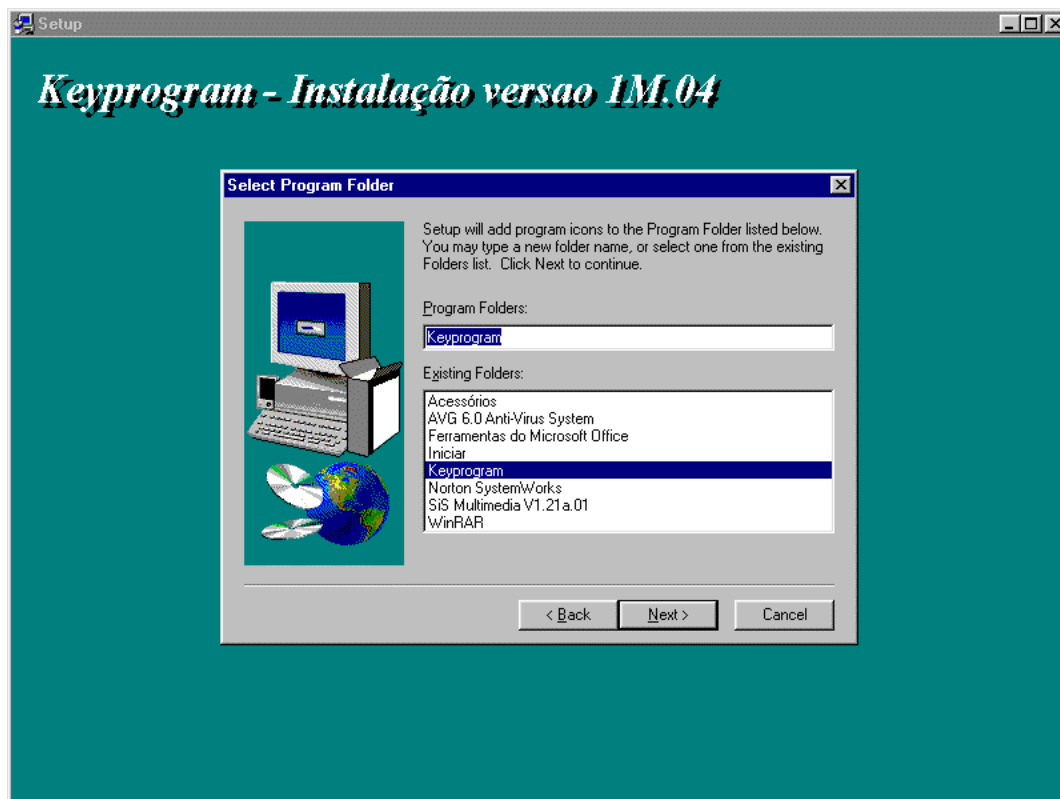


5. Logo em seguida irá aparecer a tela de instalação do Keyprogram como mostra abaixo:

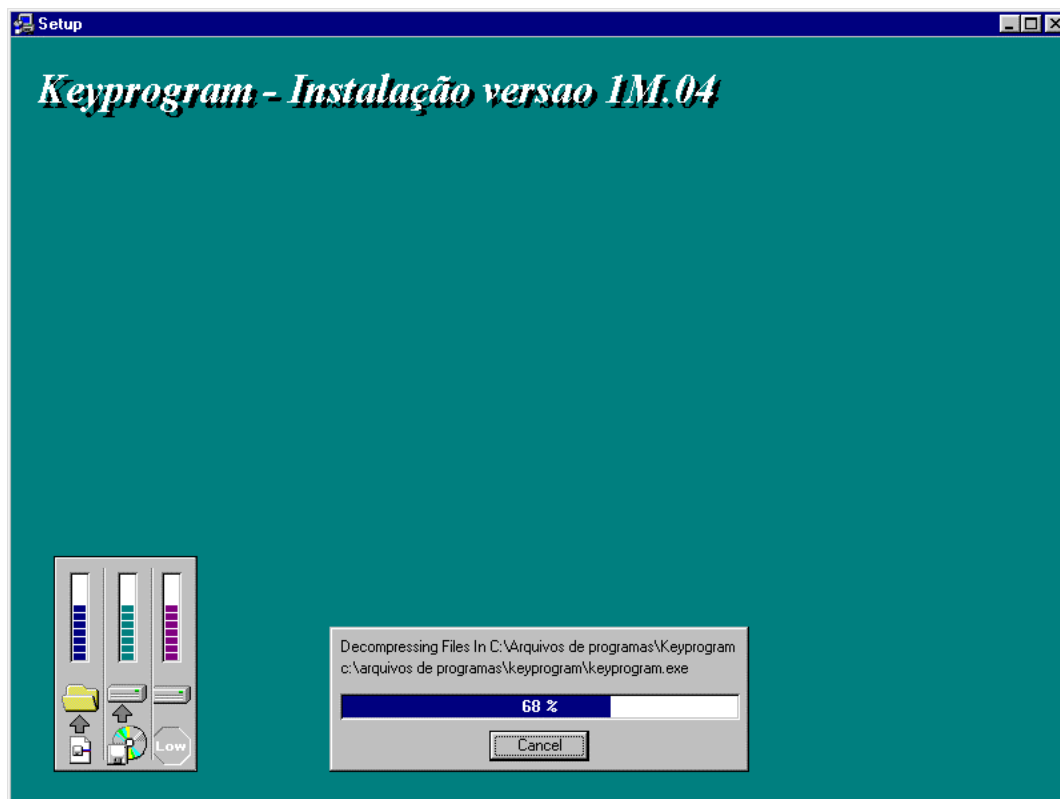


6. Em Destination Directory desta mesma tela (tela acima), é mostrada o diretório padrão onde irá ser instalado o software. Clique em Next se você deseja realmente instalar a pasta Keyprogram no diretório padrão do Windows, ou seja, em C:\Arquivos de programas. Caso queira instalar em um outro diretório, clique em Browse e selecione a pasta de sua escolha.

7. A seguir, clique em Next para ser criada a pasta Keyprogram em sua pasta programas no menu iniciar, como é mostrado abaixo:



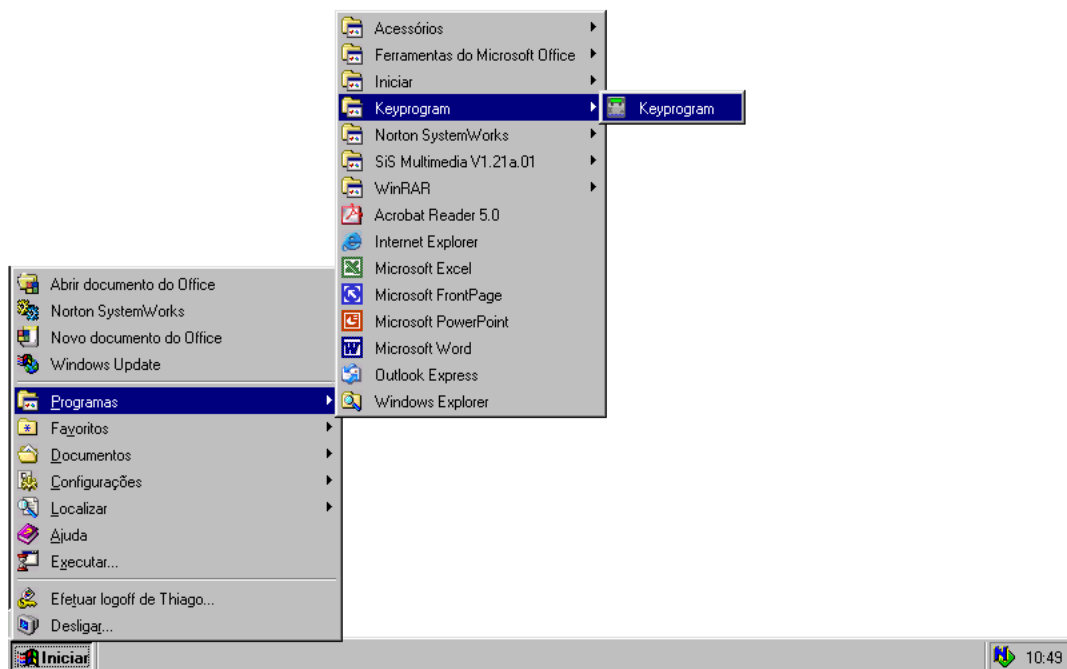
8. Em seguida, o programa irá começar a ser instalado indicando assim uma tela de progresso de instalação, como é apresentada abaixo:



5.3. Iniciando o Keyprogram

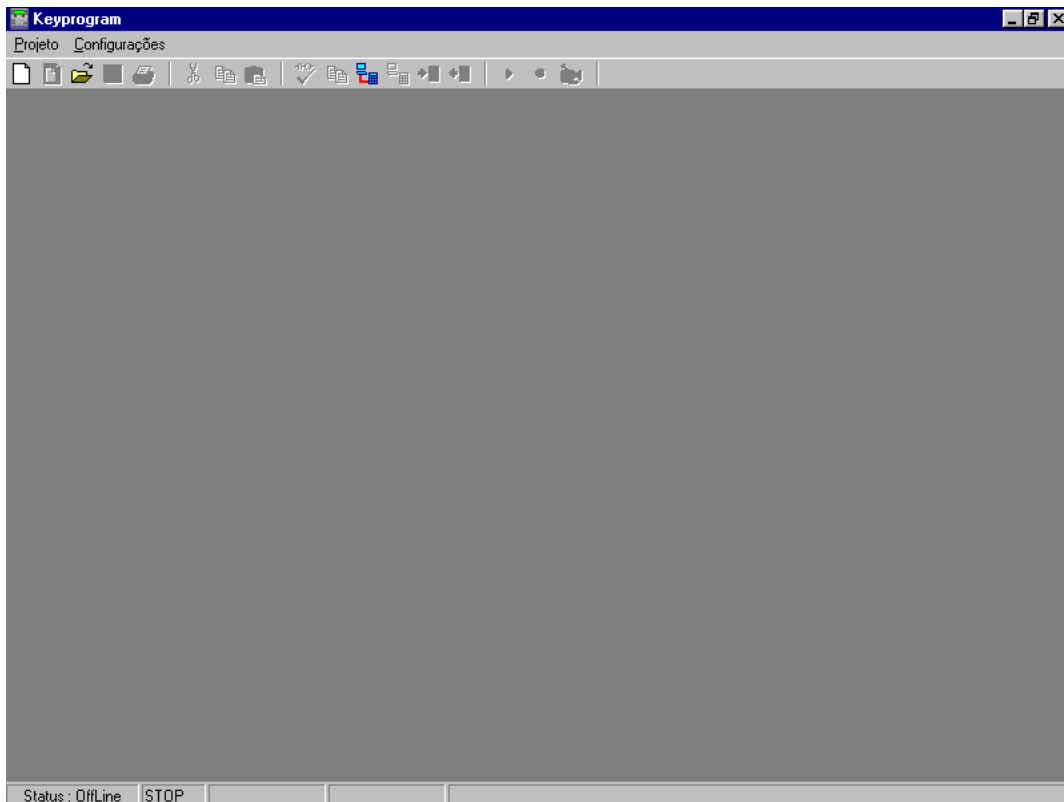
Para entrar no Keyprogram, já instalado corretamente em seu computador, execute os seguintes passos:

1. Clique no Menu Iniciar, programas, Keyprogram e Keyprogram, como pode ser descrito na figura abaixo:



5.4. Tela Inicial

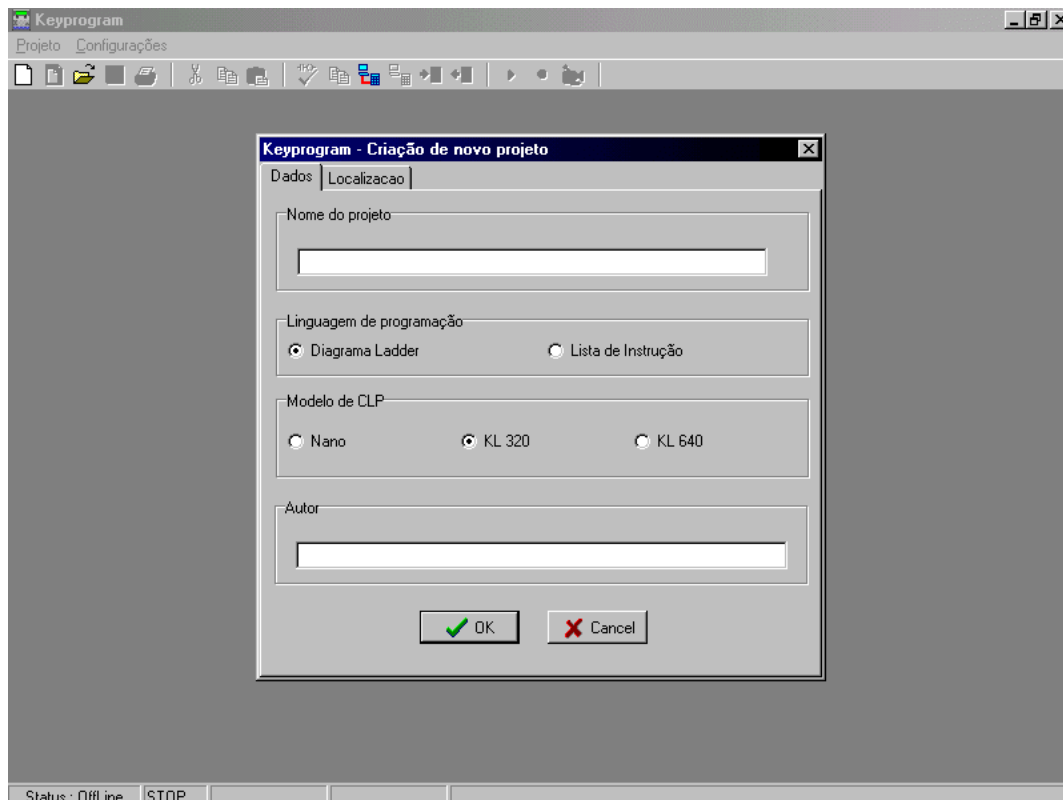
Esta é a tela executada logo na inicialização do Keyprogram:



A tela Inicial é composta por **Barra de Ferramentas**, **Barra de Menus** e **Barra de Tarefas**. Quando é apresentada a **Tela inicial**, os únicos ícones que estarão habilitados para uso, serão os ícones **Novo Projeto**, **Abrir Projeto** e **Conectar ao CLP**, pertencentes à Barra de tarefas. Já os Menus que estarão disponíveis na inicialização são: **Menu Projeto** e **Menu Configurações**, pertencentes ao a barra de Menus.

5.4.1. Janela de Configuração Inicial

A janela de configuração inicial aparece toda vez que se cria um novo projeto e é nela que o usuário define o nome do programa, a localização, o tipo de programação, o modelo do equipamento e o autor do projeto.



A Janela de Configuração Inicial é composta dos seguintes itens:

Nome do Projeto: Indica o nome do projeto.

Linguagem de Programação: é definido se a programação vai ser realizada em Diagrama Ladder ou em Lista de Instrução.

Modelo de CLP: Nesta opção são mostrados todas os modelos disponíveis para essa edição do Keyprogram.

Autor: Representa o autor do projeto.

Localização: é definido em qual diretório será salvo o programa.

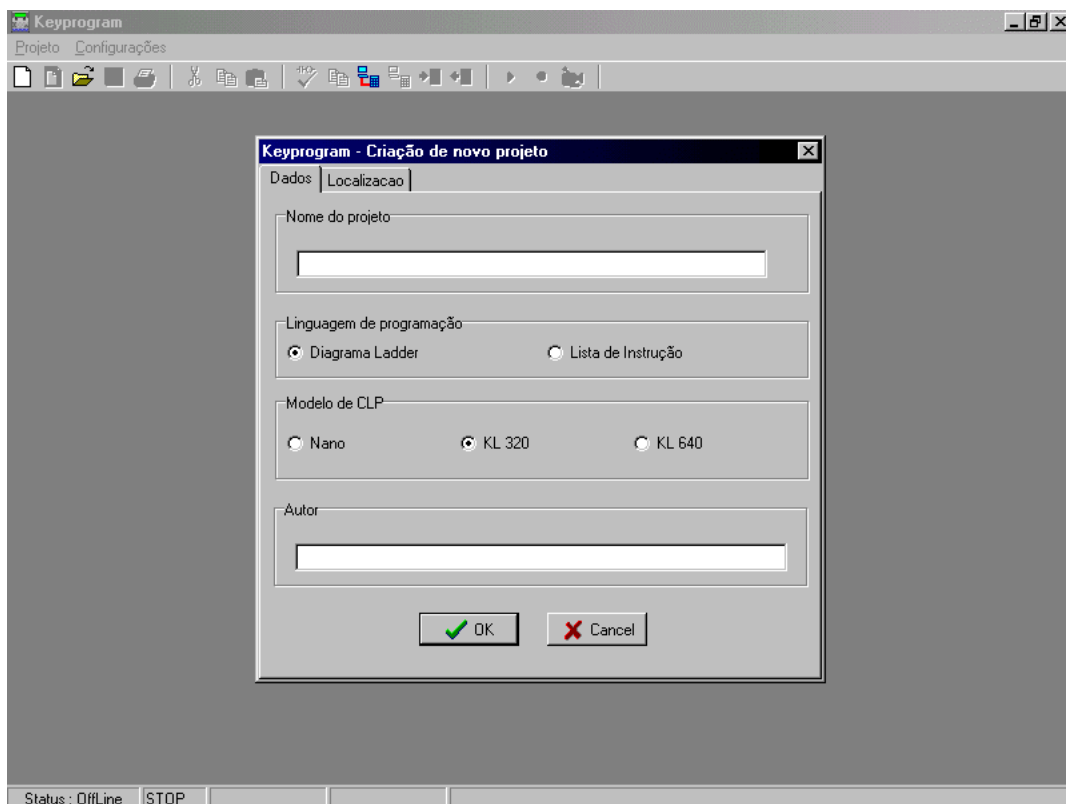
5.4.2. Criando um Projeto

Para criar um novo projeto é preciso definir o nome do projeto, a localização do projeto e o tipo de programação.

Obs: ao se definir o nome do projeto, não utilize: (, . < > = +] [; : “ * ? | / \ { }), pois esses caracteres são utilizados pelo sistema operacional e não são aceitos em nome de projeto.

Estando o software aberto, executam-se as seguintes operações:

1. Clique no ícone  (novo Projeto). Será apresentada a tela abaixo:




2. No item Nome do Projeto, digite o nome do Projeto;
3. No item Linguagem de programação escolha a linguagem em Diagrama Ladder, que é a linguagem que iremos trabalhar;
4. No item Modelo de CLP, escolha o modelo Nano, que é o modelo que estamos utilizando;
5. No item Autor, digite o nome do autor do projeto;
6. Clique em OK;

5.4.3. Abrindo um Projeto

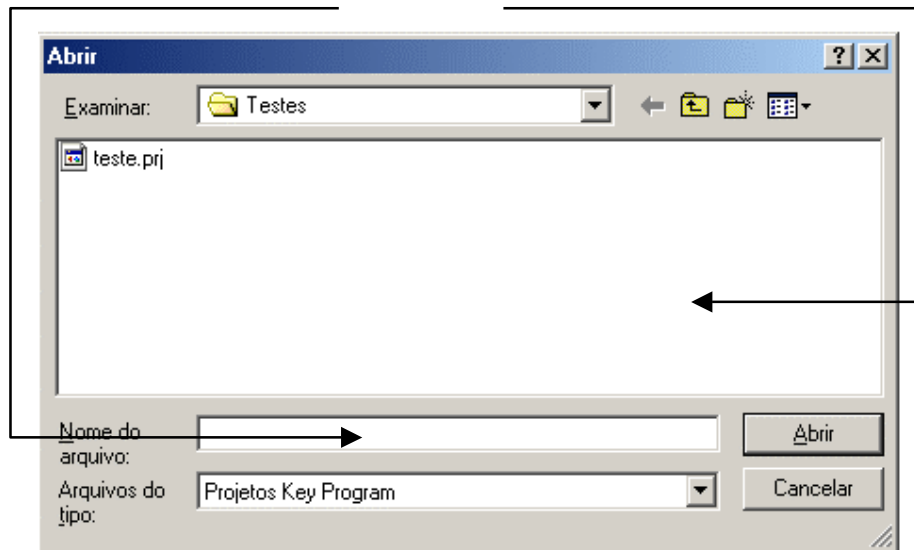
Para se modificar um projeto já existente é necessário que ele seja aberto no Keyprogram.

Para abrir um projeto:

1. Clique no menu **Projeto** (Alt+P) e depois **Abrir Projeto** (Ctrl+A), ou na **Barra de Ferramentas** clique no ícone  botão.
2. Clique em **Examinar** para seleccionar o diretório em que está o projeto.
3. No item **Nome do Arquivo**, digite o nome do arquivo ou selecione um arquivo na área de visualização de arquivos.
4. No item **Arquivo do tipo**, digite ou selecione Projetos Keyprogram.

Digite o nome do arquivo


Área de Visualização de Arquivos



5.4.4. Salvando um Projeto

Depois de feito as alterações no projeto, deve-se salvar o mesmo para que as alterações não sejam perdidas.

✓ **Salvando um projeto:**

Clique no menu **Projeto** (Alt+P) e depois **Salvar Projeto** (Ctrl+B), ou na Barra de Ferramentas clique no botão .

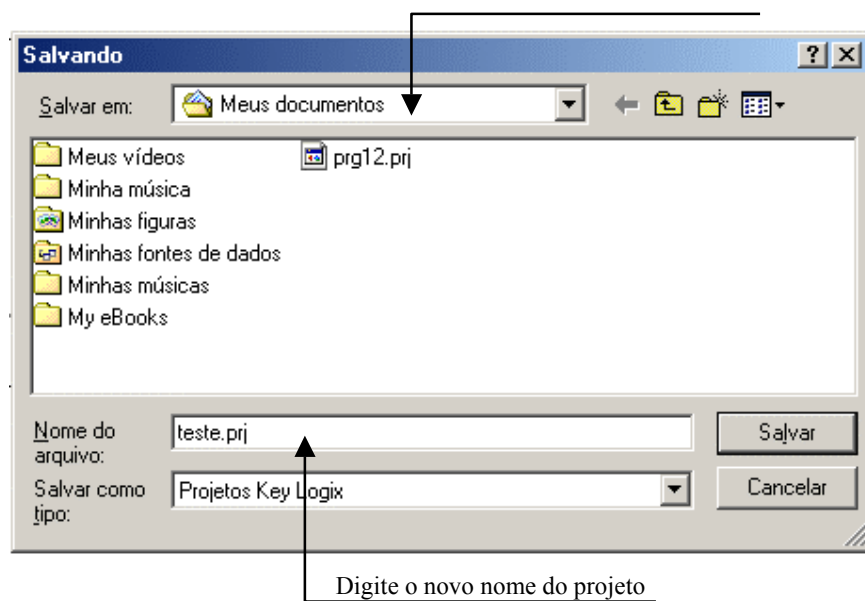
✓ **Salvando um projeto com outro Nome:**

É possível salvar um projeto já salvo com outro nome.

Para salvar o projeto com outro nome:

1. Clique no menu **Projeto** (Alt+P) e depois **Salvar Projeto Como**.
2. No item **Salvar em** escolha o diretório onde será salvo o projeto.
3. No item **Nome do Arquivo**, digite o novo nome do projeto.
4. Depois de digitado o novo nome clique em **Salvar**.

5. Caso já exista um projeto com o mesmo nome o programa irá pedir uma confirmação para substituir o projeto antigo por este que acabou de ser criado.



✓ Fechando um Projeto

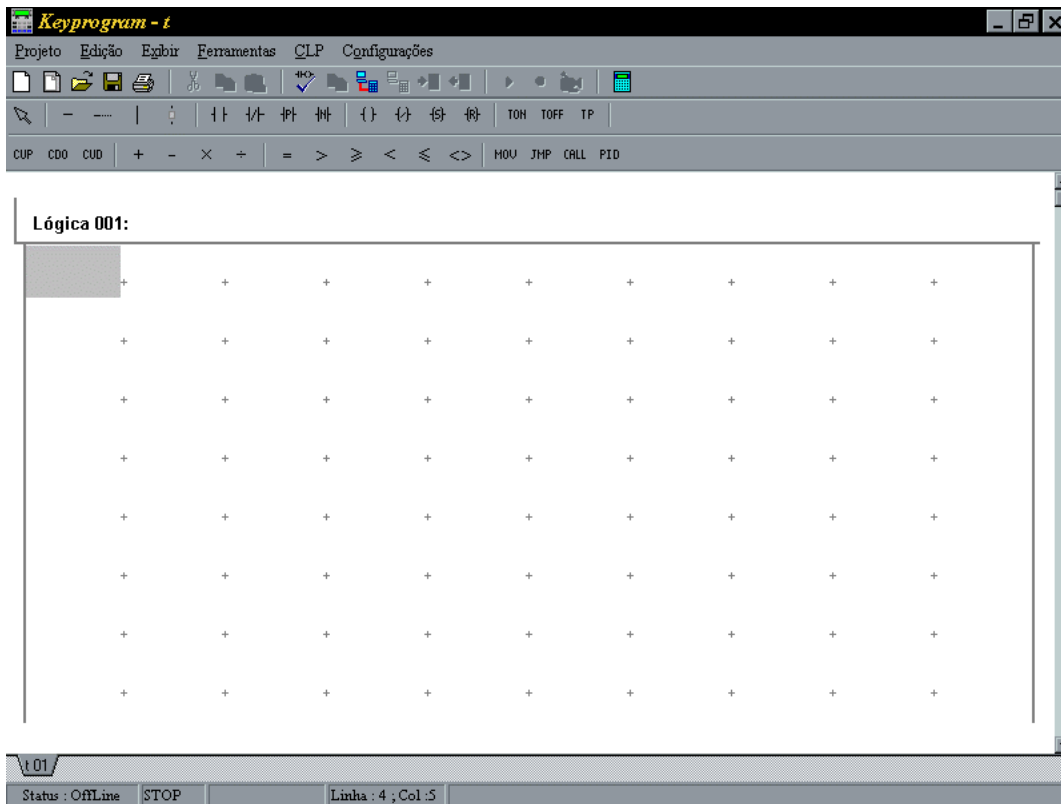
Depois de se modificar um projeto, pode-se fechá-lo para que um novo projeto possa ser criado.

Para fechar um projeto:

1. Clique no menu **Projeto** e depois no item **Fechar Projeto**.

5.5. Área de Trabalho do Keyprogram

A tela abaixo indica como será apresentada a tela para a edição da área lógica.



Note que agora na Tela da **Área de Trabalho**, os guias da **Barra de menus** estão completos e outros ícones da **Barra de tarefas** estão disponíveis.

5.5.1. Barra de Menus



Na **Área de Trabalho** do Keyprogram serão indicados abaixo os significados de todos os itens dos **Menus** da **Barra de menus**:

5.5.1.A. Menu Projeto

Novo Projeto	Ctrl+O
Abrir Projeto	Ctrl+A
Fechar Projeto	
Salvar Projeto	Ctrl+B
Salvar Projeto Como	
Importar Programa IHM	
Exportar Programa IHM	
Nova Sub-rotina	
Importar Sub-rotina	
Excluir Sub-rotina	
Exportar Sub-rotina	
Imprimir	Ctrl+P
C:\gransoft\Keylogix\keyprogram060504\teste\teste.prk	
C:\gransoft\Keylogix\keyprogram060504\testes\testes.prk	
Novo.kpg.prk	
C:\gransoft\ybms\RECALCADOR DE BARRAS\RECALCADOR DE BARRAS.prk	
Sair	

Novo projeto - Cria um novo projeto (programa da lógica do CLP em diagrama ladder e telas de IHM). Antes da criação do projeto o Keyprogram mostrara uma janela com algumas propriedades do novo projeto, que devem ser preenchidas corretamente para a criação eficaz do projeto.

Abrir projeto - Abre um projeto já existente. Deve-se informar ao Keyprogram a localização do projeto. Todos os arquivos do projeto ficam em um único diretório e em um diretório só existem arquivos de um único projeto. A extensão do arquivo de projeto é “.prk”.

Fechar projeto - Fecha o projeto que está aberto atualmente. Ao fechar o projeto o Keyprogram questionará ao usuário se deseja salvar ou não as ultimas alterações feitas, caso elas aconteçam.

Salvar projeto - Salva o projeto que está aberto atualmente.

Salvar Projeto Como: Salva o projeto atual com um nome diferente.

Importar programa IHM: Importa um arquivo com telas de IHM. Esse arquivo deve ter sido exportado de um outro projeto.

Exportar programa IHM: Exporta um arquivo com a programação de telas feita no projeto atual.

Nova Sub-rotina: Cria uma nova sub-rotina dentro do projeto atual.

Importar Sub-rotina: Importa um arquivo exportado de um outro projeto para ser utilizado no projeto aberto atualmente como sub-rotina.

Excluir Sub-rotina: Exclui uma sub-rotina do projeto atual

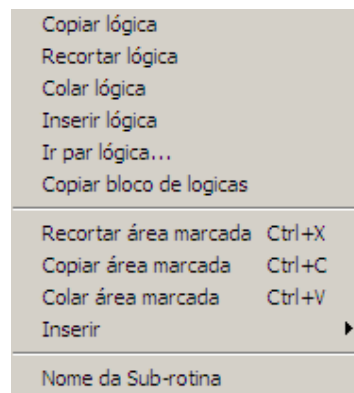
Exportar Sub-rotina: Exporta uma sub-rotina para ser utilizada em um outro projeto.

Imprimir: Abre uma janela onde se pode imprimir: a programação ladder, a lista de instrução, as telas de IHM, a tabela de símbolos.

Arquivos recentes: Exibe os quatro últimos arquivos abertos recentemente.

Sair: Fecha o Keyprogram.

5.5.1.B. Menu Edição



Copiar lógica: Armazena uma determinada lógica para ser colada em um outro local do programa.

Recortar lógica: Apaga uma determinada lógica e armazena para ser colada em um outro local do programa.

Colar lógica: Cola um uma posição determinada do programa a lógica que antes havia sido armazenada.

Inserir lógica: Insere uma nova lógica em uma determinada posição do programa ladder.

Ir para lógica: Mostra uma determinada lógica na tela do Keyprogram.

Copiar bloco de lógicas: Copia um bloco de lógica e cola este bloco em uma outra posição do programa. As posições iniciais e finais do bloco e a posição que o bloco será colado deve ser informado ao usuário.

Recortar Área marcada: Apaga e armazena para ser colada uma área que estiver selecionada na lógica.

Copiar Área marcada: Armazena para ser colada uma área que estiver selecionada na lógica.

Colar Área marcada: Cola uma área que foi selecionada. Esta área é colada na posição atual do cursor.

Inserir Linha: Insere uma nova linha na lógica que estiver aberta na posição em que estiver selecionado.

Inserir Coluna: Insere uma nova coluna na lógica que estiver aberta na posição em que estiver selecionado.

Nome da Sub-rotina: Altera o nome da sub-rotina que estiver selecionada na guia de páginas inferior.

5.5.1.C. Menu Exibir

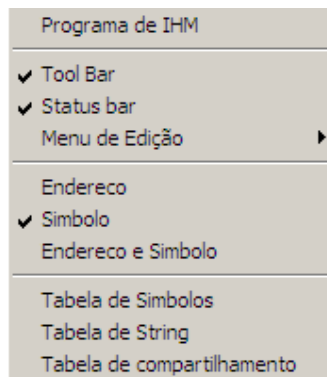


Diagrama Ladder: Este item do menu exibir só aparece quando o Keyprogram estiver na parte de programação da IHM. Ele passa o Keyprogram para a programação em Diagrama Ladder.

Programa de IHM: Este item do menu exibir só aparece quando o Keyprogram estiver na parte de programação do diagrama ladder. Ele passa o Keyprogram para a programação das telas da IHM.

Toolbar: Configura o Keyprogram para exibir ou não o toolbar. O toolbar é uma barra de ferramentas que fica logo abaixo no menu superior. Ela contém os ícones de novo projeto, nova sub-rotina etc. Caso o usuário opte por não visualizar esta barra de ferramentas, todas as funções devem ser escolhidas através do menu superior.

Statusbar: Configura o Keyprogram para a exibição ou não do

Statusbar. O Statusbar é uma barra de informações que fica na parte inferior do Keyprogram, e ele mostra o estado atual do keyprogram quanto à comunicação com o CLP e outras informações de edição do projeto.

Menu de Edição: Configura o Keyprogram quanto à maneira como será exibido o menu de edição. É através do menu de edição que o usuário insere novos componentes no diagrama ladder. Há duas opções de visualização deste menu:

- **Exibir no topo:** Exibe o menu de edição logo abaixo da barra de ferramentas superior.
- **Menu Móvel:** Deixa o menu de edição de uma maneira móvel. A vantagem dessa opção é aumentar a área útil da edição do ladder. A desvantagem é que o menu acaba atrapalhando a edição do diagrama ladder.

Endereço: Os operandos dos componentes do diagrama ladder são mostrados com seu endereço. (Ex.: %I0.0, %Q0.0 e etc).

Símbolo: Os operandos dos componentes do diagrama ladder são mostrados com seu símbolo. Caso um endereço não possua símbolo associado a ele através da tabela de símbolos, é mostrado o seu endereço.

Endereço e Símbolo: Os operandos dos componentes do diagrama ladder são mostrados com seu símbolo e endereço simultaneamente.

Tabela de Símbolo: Exibe um formulário que permitira a edição da tabela de símbolos. Essa tabela associa a um endereço utilizado no projeto um símbolo com 9 caracteres. Esse símbolo é usado edição de cada componente do diagrama ladder e na visualização desse diagrama como um todo.

Tabela de Símbolo: Exibe um formulário que permite a edição da tabela de strings. Essa tabela indexa strings que são enviadas para o CLP por meio do download do projeto. Essas strings são usadas na programação da IHM, nos campos string e seletora.

Tabela de compartilhamento: Exibe um formulário que permite a edição da tabela de compartilhamento. Essa tabela configura quais CLP's e quais endereços o CLP que receberá este projeto estará lendo valores. Essa tabela só se preenche quando o CLP é Mestre na rede.

5.5.1.D. Menu Ferramenta:

Busca Instrução
Próxima Busca Instrução
Busca Operando
Próxima Busca Operando F3
Substituir operando
Monitoramento

Busca instrução: Busca no diagrama ladder a partir da posição atual do cursor uma determinada instrução. Ex: Caso queira buscar a próxima instrução de contato normalmente aberto, digite CNA. Uma tabela completa com as instruções e as respectivas siglas pode ser encontrada no manual do Keyprogram.

Próxima busca instrução: Busca a partir da posição em que foi encontrada a última instrução, um próximo componente no diagrama ladder com a mesma instrução.

Busca Operando: Busca no diagrama ladder a partir da posição atual do cursor um componente que possua um determinado operando. Esse operando pode ser buscado tanto por endereço, como por símbolo.

Próxima busca operando: Busca a partir da posição em que foi encontrado o último operando, um próximo componente no diagrama ladder com o mesmo operando.

Substituir operando: Substitui em um bloco do diagrama ladder, um determinado endereço por outro. Essa substituição pode ser feita através do endereço ou do símbolo associado ao endereço.

Monitoramento: Exibe uma janela para se monitorar os endereços do CLP que estiver conectado ao Keyprogram.

5.5.1.E. Menu CLP:

Conectar	F10
Desconectar	F9
Download	F8
Upload	F7
Run	F6
Stop	F5
Monitorar	F3
Compara	
Apaga Memória	
Memoria Utilizada	
Configura relógio	

Conectar: Abre a porta de comunicação serial e tenta estabelecer uma comunicação com o CLP. Caso essa comunicação ocorra com sucesso, a barra de Status mostrará: Status Online. Caso não houver resposta do CLP será mostrado: Status Offline e numa frequência de 3s o Keyprogram tentará estabelecer a comunicação com o CLP.

Desconectar: Fecha a porta de comunicação serial.

Download: Transfere o projeto (programa do CLP, telas de IHM, tabela de strings, tabela de compartilhamento) aberto atualmente no Keyprogram para o CLP.

Upload: Transfere o projeto (programa do CLP, telas de IHM, tabela de strings, tabela de compartilhamento) que estiver na memória do CLP para o Keyprogram.

Run: Passa o CLP para modo de execução RUN. Neste modo o CLP executa a lógica que estiver em sua memória. Obs.: Só é possível passar para esse modo se o programa que estiver na memória do CLP for igual ao que estiver aberto no Keyprogram.

Stop: Passa o CLP para modo STOP. Neste modo o CLP para de executar o programa que estiver em sua memória.

Monitorar: Mostra no diagrama ladder o valor de cada operando que estiver no mesmo.

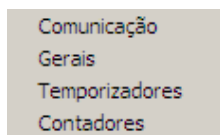
Compara: Compara o programa aberto atualmente no Keyprogram com o programa residente na memória do CLP.

Apaga memória: Apaga a memória do CLP. O projeto que estiver no CLP (lógica, telas, tabela de strings são apagadas).

Memória utilizada: Informa ao usuário qual o tamanho do programa atual no Keyprogram e do programa que estiver no CLP.

Configura relógio: Abre uma janela que permite configurar o relógio do CLP.

5.5.1.F. Menu Configurações:



Comunicação: Abre uma janela para se configurar qual a porta serial será utilizada para se comunicar com o CLP.

Gerais: Abre uma janela para a edição de varias configurações do projeto.

Temporizadores: Dá acesso a uma tabela com as configurações dos temporizadores utilizados atualmente no projeto. Essas configurações podem ser mudadas através dessa tabela.


Contadores: Dá acesso a uma tabela com as configurações dos contadores utilizados atualmente no projeto. Essas configurações podem ser mudadas através dessa tabela.


5.5.2. Barra de Ferramentas


A **Barra de Ferramentas** possui ícones destinados a tarefas tanto da edição de lógicas, quanto para edição de telas de IHM.





Segue a seguir, a descrição de cada ícone:


 **Cria um novo projeto:** Este projeto inclui tanto o diagrama ladder como também as telas da IHM. Caso exista um projeto já aberto este pedirá para salvá-lo e em seguida abrirá um novo;


 **Insere uma nova sub-rotina:** Esta sub-rotina é um programa em ladder ou lista de instrução que poderá ser chamado pelo programa principal ou por outros programas através da função CALL ou JUMP;


 **Abre projeto:** Abre um projeto que tenha sido gravado anteriormente;


 **Salva Projeto:** Salva um projeto, incluindo a programação ladder e a programação das telas de IHM;


 **Imprime:** Através desse ícone pode-se imprimir tanto a programação ladder com todas sub-rotinas, a lista de instrução, telas de IHM e tabela de símbolos. Cada uma dessas impressões pode ser feita separadamente;


 **Recortar:** Apaga e armazena na área de transferência uma parte do diagrama ladder, da lista de instrução ou da tela de IHM que estiver selecionado;


 **Copiar:** Armazena na área de transferência uma parte do diagrama ladder, da lista de instrução ou da tela de IHM que estiver selecionado;

 **Colar:** Copia a área de transferência para o local em que o cursor estiver posicionado no momento;

 **Conectar:** Estabelece a conexão com o CLP que estiver conectado a saída serial do PC;


 **Desconectar:** Desconecta-se do CLP quando este está On-line;


 **Download:** Transfere para o CLP toda a programação ladder e as telas de IHM;


 **Upload:** Transfere do CLP para o PC toda a programação ladder e telas de IHM;

 **Run:** Passa o CLP para modo Run;

 **Stop:** Passa o CLP para modo Stop;

 **Help:** Chama o Help para ajudar o usuário no ponto em que ele estiver programando;

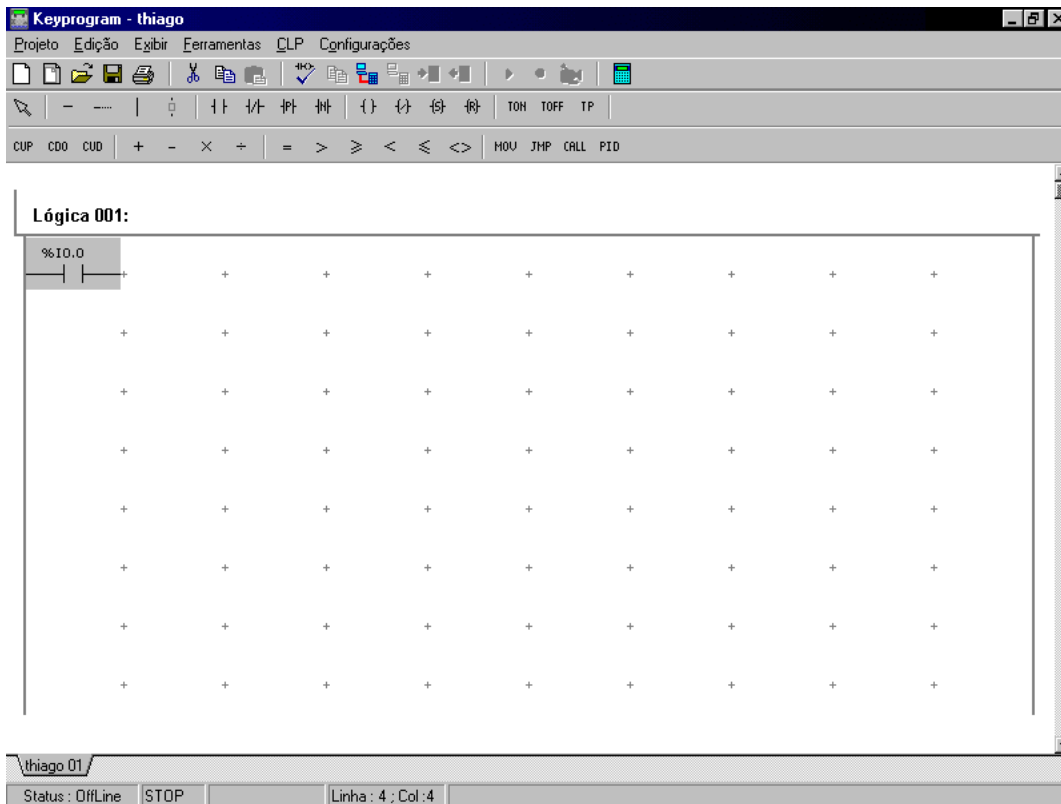
 **Editor Ladder:** Chama o editor de lógica ladder caso o usuário esteja no editor de IHM;

 **Editor IHM:** Chama o editor de telas de IHM caso o usuário esteja no editor de lógica ladder;

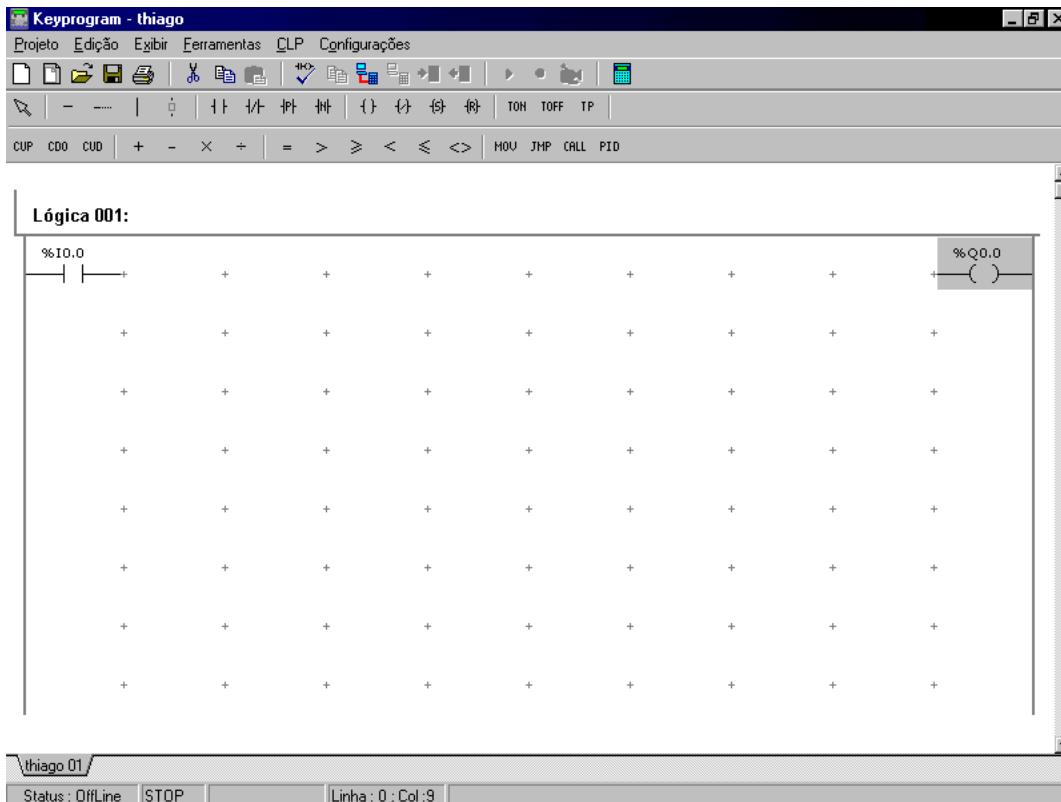
5.6. Criando e executando uma Lógica

Estando aberto um novo projeto ou um projeto já existente execute os seguintes passos para criar uma lógica no programa Keyprogram:

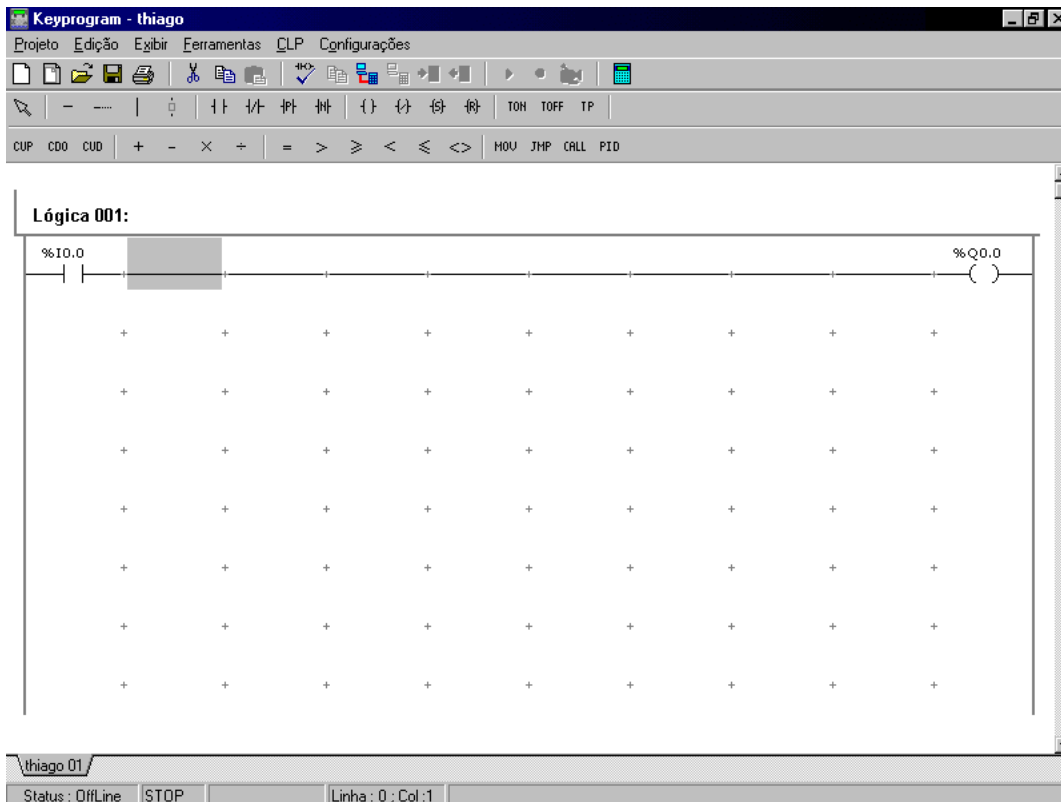
1. Clique no contato Normalmente Aberto (NA) para ser inserido na área de edição. Note que o contato ficará preso no cursor do mouse até que seja clicado novamente em algum ponto da Área de Edição do Keyprogram. Clique na área onde esteja situada a primeira linha e primeira coluna da área de Edição;
2. No contato NA será aberto um campo para que seja especificado o nome do endereço do contato. Digite: %I0.0 e tecle ENTER. A tela ficará assim:



3. A seguir, clique na bobina descrita como: saída simples e clique na primeira linha e última coluna da área de edição;
4. Digite %Q0.0 no campo texto da saída simples e tecle ENTER. Se tudo ocorrer bem ficara assim:



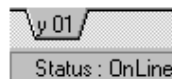
5. Note que agora deve ser fechado o circuito de forma que funcione o programa que estamos propondo. Para isso devemos usar ícone: completa traço horizontal, que fechará o circuito proposto fazendo com que a saída simples responda da seguinte forma: quando o contato NA (%I0.0) estiver aberto, a saída simples (%Q0.0) permanecerá desacionada; já, quando o contato NA (%I0.0) estiver fechado, a saída simples (%Q0.0) será acionada;
6. Nosso programa ficará assim:




7. Em seguida é preciso conectar o PLC ao PC. Para isso, clique no botão:




8. Estando o PLC on-line com o PC, irá mostrar na barra de status, situada no canto inferior esquerdo, a indicação on-line como se pode verificar na figura ao lado:



9. A seguir, é preciso fazer o envio da lógica de programação, criada no Keyprogram, para o PLC, bastando para isso clicar no botão ;

10. Se tudo decorrer bem, então a lógica será enviada ao PLC, e o PLC irá executar tudo aquilo que foi programado na lógica Ladder, criada no Keyprogram;

11. Além disso, para verificar as Lógicas sendo processadas no CLP e monitoradas pelo PC, basta clicar no ícone: , que o software irá indicar quando um contato estiver fechado, uma saída for acionada, etc.


5.7. Áreas de Programação do Keyprogram

Para facilitar a utilização, o Keyprogram reúne em um único software, duas telas de programação:

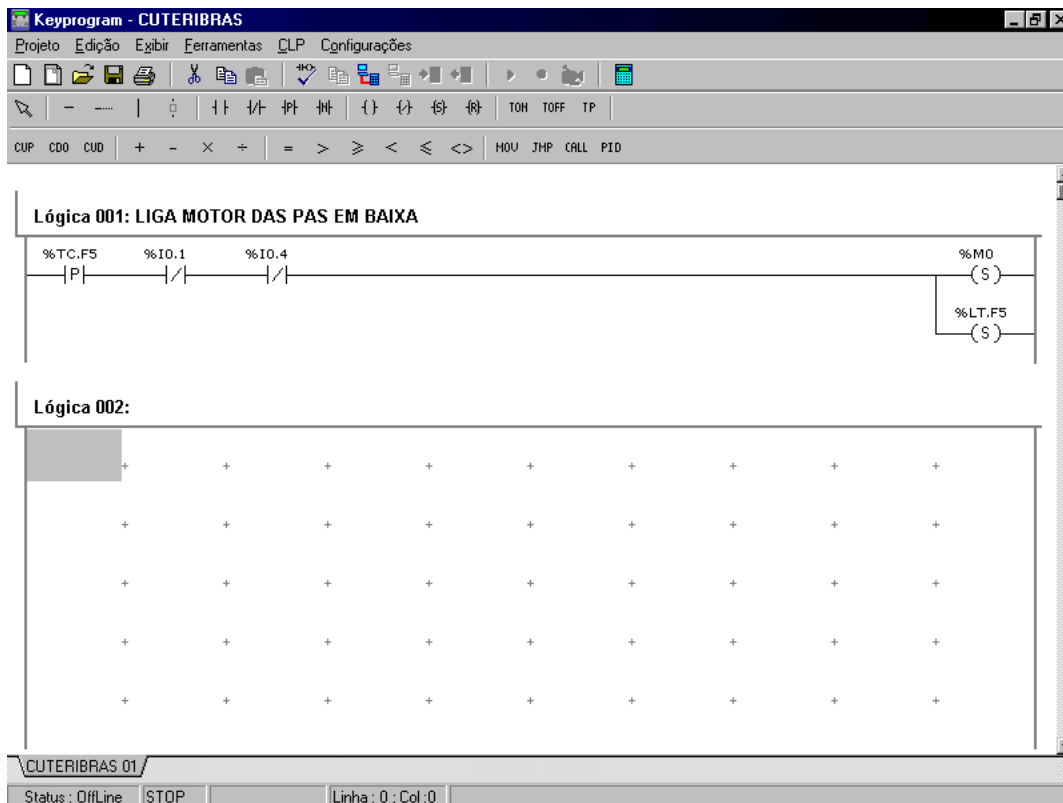
- ✓ **Tela de Programação Ladder:** Utilizado para programar lógicas de contatos executadas pelo PLC.
- ✓ **Tela de Programação IHM:** Usado para programar telas que serão visualizadas no Display da IHM.

Obs: Esta versão de Equipamento NanoLogix não integra a opção de Tela de Programação IHM, pois não possui IHM.

5.7.1. Área de Programação Ladder

Para entrar na área de programação Ladder clique no ícone , localizado na barra de ferramentas. Note que, quando se clica neste ícone, ele se modifica indicando agora a entrada para o modo de Programação IHM.

É mostrada abaixo, a Área de Programação Ladder:








- **Descrição da Barra de Ferramentas (Ladder)**

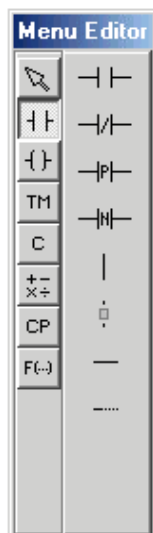
Menu de Edição





✓ **Grupo Geral**



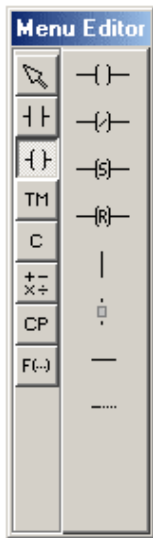
-  **Seta:** deixa o cursor em seu modo default, que é usado para selecionar os elementos da lógica.
-  **Traço vertical:** Insere um traço vertical na lógica entre dois elementos.
-  **Apaga traço vertical:** Ao posicionar o cursor sobre traço vertical e clicar o traço é apagado.
-  **Traço horizontal:** Insere um traço horizontal na célula onde se clicar com o mouse.
-  **Completa ligações na horizontal:** Até se encontrar um outro elemento.





✓ **Grupo de Contatos**



-  Insere **Contato Normalmente Aberto** ao selecionar esse ícone e clicar sobre uma lógica.
-  Insere **Contato Normalmente Fechado** ao selecionar esse ícone e clicar sobre uma lógica.
-  Insere **Contato por Borda Positiva** ao selecionar esse ícone e clicar sobre uma célula.
-  Insere **Contato por Borda Negativa** ao selecionar esse ícone e clicar sobre uma lógica.




✓ **Grupo de Bobinas**



-  **Saída Simples:** Insere uma Saída Simples.
-  **Saída Complementar:** Insere uma Saída Complementar.
-  **Saída Set:** Insere uma Saída Set.
-  **Saída Reset:** Insere uma Saída Off

✓ **Grupo de Temporizadores**



-  **Temporizador de Pulso:** Insere um Temporizador de Pulso
-  **Temporizador On:** Insere um Temporizador On
-  **Temporizador Off:** Insere um Temporizador Off

✓ *Grupo de Operações Matemáticas*



- Bloco Somador:** Insere um Bloco Somador.
- Bloco Subtrator:** Insere um Bloco Subtrator.
- Bloco Divisor:** Insere um Bloco Divisor.
- Bloco Multiplicador:** Insere um Bloco Multiplicador.

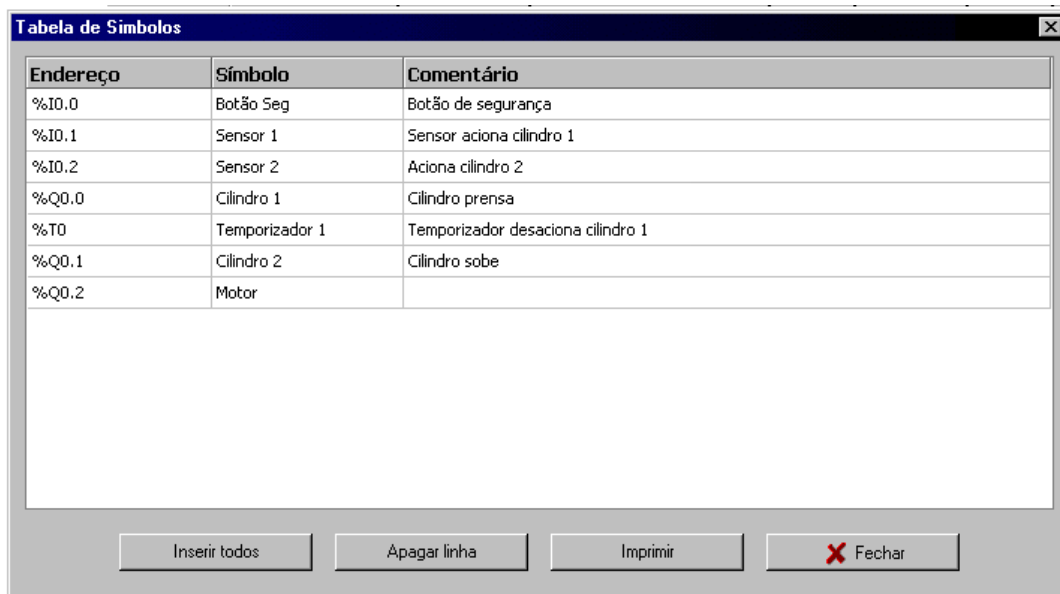
✓ *Grupo de Comparadores*



- Comparador Maior:** Insere um Comparador Maior.
- Comparador Maior Igual:** Insere um Comparador Maior Igual.
- Comparador Igual:** Insere um Comparador Igual.
- Comparador Menor Igual:** Insere um Comparador Menor Igual.
- Comparador Menor:** Insere um Comparador Menor Igual.
- Comparador Diferente:** Insere um Comparador diferente.

5.8. Tabelas de Símbolos

A Tabela de Símbolos é um recurso que ajuda o programador a documentar o programa que está sendo desenvolvido, identificando cada operando do programa.



Endereço	Símbolo	Comentário
%I0.0	Botão Seg	Botão de segurança
%I0.1	Sensor 1	Sensor aciona cilindro 1
%I0.2	Sensor 2	Aciona cilindro 2
%Q0.0	Cilindro 1	Cilindro prensa
%T0	Temporizador 1	Temporizador desaciona cilindro 1
%Q0.1	Cilindro 2	Cilindro sobe
%Q0.2	Motor	

Inserindo uma nova linha: para criar uma nova linha pressione a tecla down do teclado.

Apagando uma linha: selecione a linha que se deseja apagar e pressione o botão Apagar linha.

Inserindo todos os operandos do programa: essa função insere na Tabela de Símbolos todos os operandos do programa que ainda não estiverem nela. Clique no botão Inserir todas.

Ordenando a tabela de símbolos por endereço: dê um duplo clique sobre o título Endereço da tabela. Ela será ordenada da seguinte ordem:


Imprimindo a Tabela de Símbolos: clique no menu Projeto (Alt+P) e no item Imprimir (Ctrl+P) selecione a opção Tabela de Símbolos, Imprimir e OK.

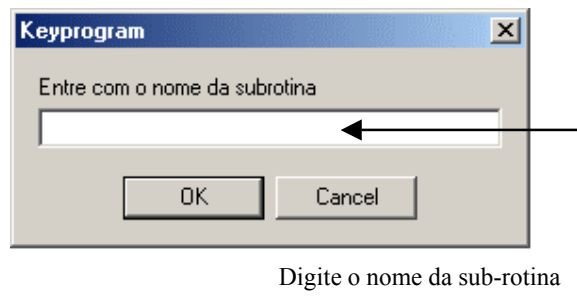
5.9. Sub-rotinas

✓ Criando Nova Sub-rotina

A criação de Sub-rotinas permite que uma função que seja utilizada várias vezes no projeto não precise ser reescrita todas às vezes.

Para criar uma Sub-rotina:

1. Clique no ícone **Nova Sub-rotina**  da **Barra de Ferramentas**.
2. Digite o nome da nova sub-rotina.
3. Clique em **OK**.

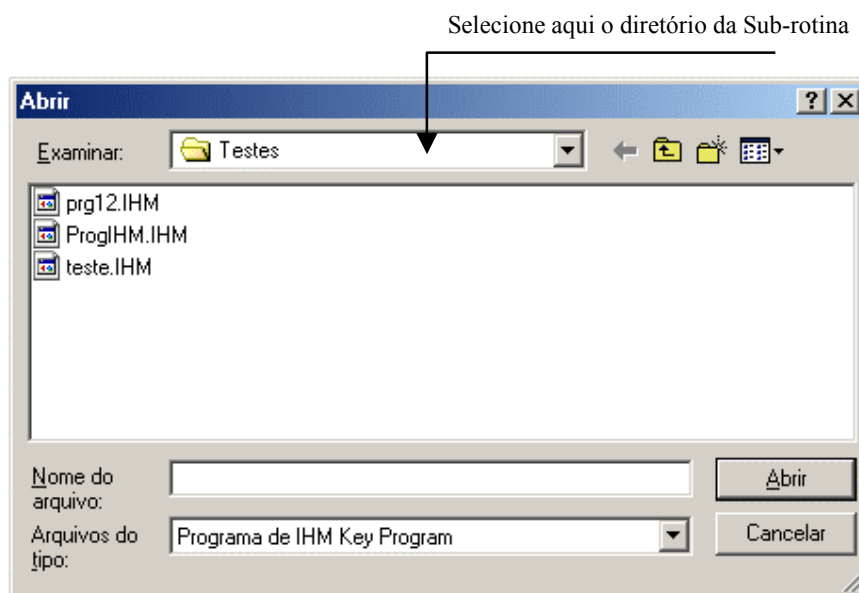


✓ **Importando Sub-rotina**

Se a Sub-rotina que você for criar já estiver feita em outro projeto, use a função Importar Sub-rotina:

Para importar uma Sub-rotina:

1. Clique no menu **Projeto** (Alt+P), no item **Importar Sub-rotina**.
2. No item **Examinar**, indique o diretório em que está a sub-rotina.
3. Clique em **Abrir**.

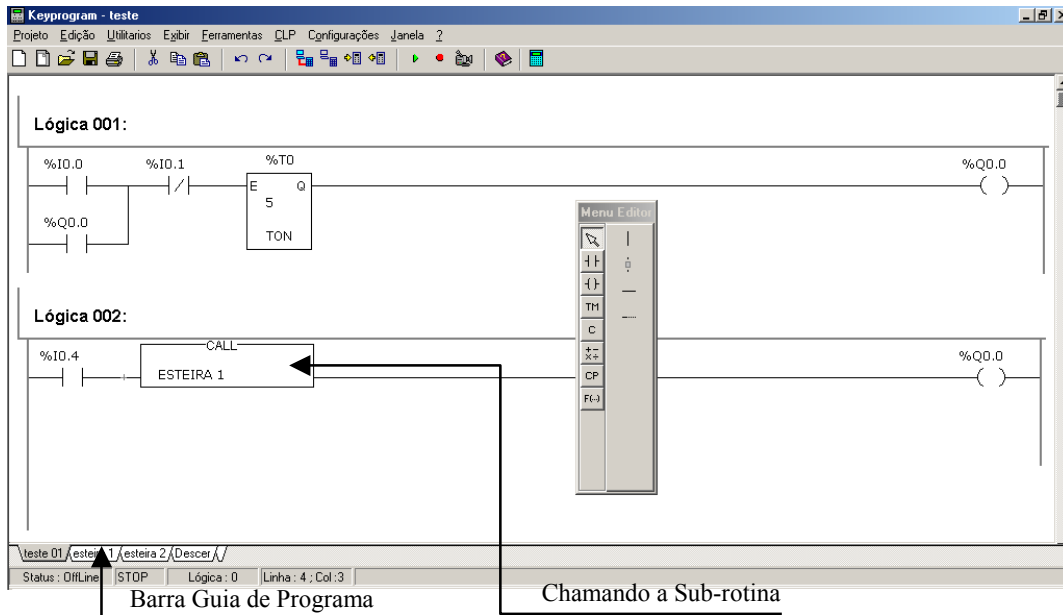


A nova sub-rotina irá aparecer na barra Guia de Programa.

✓ Editando Sub-rotinas

Para editar a Sub-rotina:

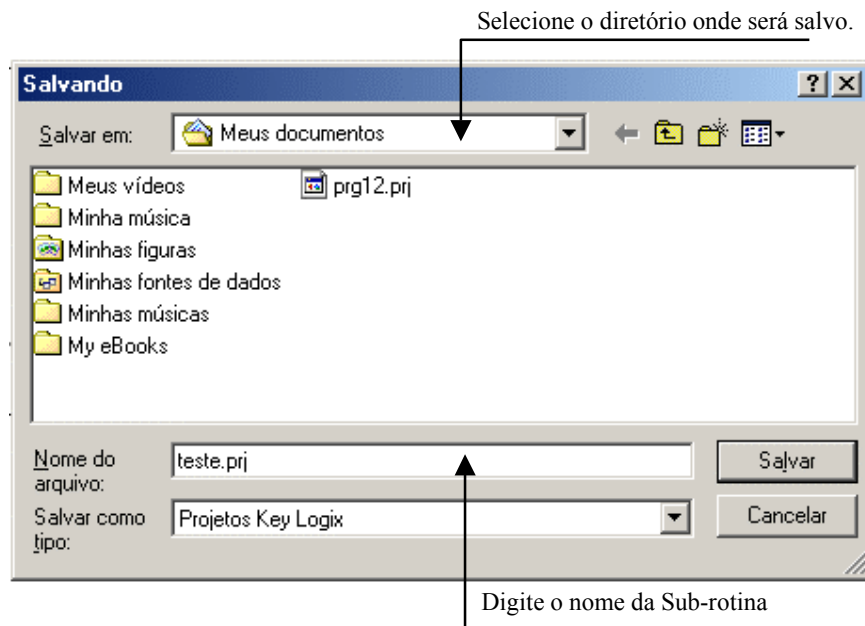
1. Clique sobre a **Sub-rotina** na barra **Guia de Programa**.
2. Para chamar a **Sub-rotina** use um operando do tipo **CALL** do grupo funções especiais.



✓ Exportando Sub-rotina

Para exportar uma Sub-rotina:

1. Clique no menu **Projeto** (Alt+P), no item **Exportar Sub-rotina**.
2. Escolha o local que a Sub-rotina vai ser salva.
3. Digite um nome para a Sub-rotina.
4. Clique em **Salvar**.



5.10. Configuração da Comunicação

A comunicação entre o CLP Keylogix e o Keyprogram é feita pela Porta de Comunicação Serial.

Para configurar a porta serial:

1. Clique no menu Configurações (Alt+O) no item Comunicação.
2. No item Port, selecione a porta de comunicação serial que você irá utilizar.
3. Clique em OK.

Obs: As demais propriedades do item Comunicação são configuradas de acordo com o equipamento utilizado, não altere essas opções, pois fará com que se perca a comunicação serial entre o PLC e o PC.

5.11. Instruções Válidas para o NanoLogix

- ✓ **Contatos**
 - ✓ Contato normalmente aberto
 - ✓ Contato normalmente fechado
 - ✓ Contato por borda positiva
 - ✓ Contato por borda negativa

- ✓ **Bobinas**
 - ✓ Saída simples
 - ✓ Saída complementar
 - ✓ Saída set
 - ✓ Saída reset

- ✓ **Temporizadores**
 - ✓ Temporizador na energização (TON)

- ✓ **Contadores**
 - ✓ Contador UP

- ✓ **Comparadores**
 - ✓ Igual (==)
 - ✓ Maior que (>)
 - ✓ Menor que (<)
 - ✓ Maior ou igual que (>=)
 - ✓ Menor ou igual que (<=)
 - ✓ Diferente (<>)

- ✓ **Especiais**
 - ✓ Move

6. Glossário

- **Baud rate (taxa de Transmissão).** Taxa pela qual os bits de informação são transmitidos através de uma interface serial ou rede de comunicação.
- **Byte.** Unidade de informação composta por oito bits.
- **Controlador Lógico Programável (PLC).** Equipamento que realiza o controle sob o comando de um programa aplicativo escrito em linguagem de relés e blocos. É composto basicamente de Unidade de Processamento, Memória, e dispositivos de entradas e saídas.
- **EEPROM (Electric Erasable Programmable Read Only Memory).** Memória somente de leitura, apagável e programável eletricamente.
- **Flash EPROM.** Memória não volátil apagável eletricamente
- **Instrução.** Operação a ser executada sobre o conjunto de operandos dentro de um programa.
- **Lógica de Programação.** Matriz gráfica onde são inseridas as instruções da linguagem de diagrama de relés que compõem um programa aplicativo. Um conjunto de lógicas ordenadas seqüencialmente constitui um módulo de programa.

- **Menu.** Conjunto de opções disponíveis e exibidas no vídeo por um programa, a serem selecionadas pelo usuário a fim de ativar ou executar uma determinada tarefa.
- **Octeto.** Conjunto de oito bits numerados de 0 a 7.
- **Protocolo.** Regras de procedimentos e formatos convencionais que, mediante sinais de controle, permitem o estabelecimento de uma transmissão de dados e a recuperação de erros entre equipamentos.
- **RAM (Random Access Memory).** Memória de acesso aleatório. Este tipo de memória é volátil, ou seja, seu conteúdo se perde quando é desenergizada.
- **Retentivo.** Indica que o dado armazenado não será apagado quando o equipamento for desenergizado, “retendo” este valor quando o equipamento for ligado novamente.
- **Scan.** Período de uma varredura no programa.
- **Word.** Unidade de informação composta por oito 16 bits ou 2 Bytes.