

CONTROLADOR LÓGICO PROGRAMÁVEL KL640



MANUAL DO USUÁRIO KL640

Série: KL640 – 003

Ref. 0405.003.M04

Ver. 0411.003.001

Todos os direitos reservados. Proibida a reprodução total ou parcial deste manual sem autorização por escrito da Keylogix Automation LTDA.
As informações contidas neste documento apresentam caráter exclusivamente técnico/informativo e a Keylogix, sem qualquer aviso prévio, terá direito de alterar as informações deste documento caso necessite.

Sumário

1. Introdução	9
1.1. Requisitos Mínimos da Ferramenta KeyProgram	10
1.2. Configurações Recomendadas	10
2. Especificações do Equipamento	11
2.1. Instalação	11
2.2. Alimentação	12
2.3. Entradas Digitais	13
2.3.1. Entrada Tipo N (24Vcc).....	13
2.3.2. Entrada Tipo P (24Vcc)	14
2.3.3. Características das Entradas	15
2.3.4. Endereçamento das Entradas	15
2.4. Entrada Rápida.....	16
2.4.1. Características das Entradas Rápidas	17
2.4.2. Endereçamento das Entradas	17
2.5. Saídas Digitais	18
2.5.1. Saídas Tipo N.....	18
2.5.2. Saídas Tipo P	19
2.5.3. Saídas a Relé.....	20
2.5.4. Características das Saídas	21
2.5.5. Endereçamento de Saídas	21
2.6. Entradas Analógicas:	22
2.6.1. Por Corrente.....	22
2.6.2. Por Tensão	23
2.6.3. Características das Entradas Analógicas	24
2.6.4. Endereçamento das Entradas Analógicas.....	24
2.6.5. Descrição das Saídas Analógicas:.....	25
2.6.6. Características das Saídas Analógicas	26
2.6.7. Endereçamento de Saídas	26
2.7. Comunicação.....	27
2.7.1. Modo de Comunicação:	27
3. Endereçamento	30
4. Instruções de Programação Ladder	32
4.1. Contatos	32

4.1.1.	Contato Normalmente Aberto	32
4.1.2.	Contato Normalmente Fechado	33
4.1.3.	Contato por borda positiva.....	34
4.1.4.	Contato por borda negativa.....	35
4.1.5.	Parâmetro dos Contatos:	36
4.2.	Bobinas	36
4.2.1.	Saída Simples.....	36
4.2.2.	Saída Complementar	37
4.2.3.	Saída Set e Reset.....	38
4.2.4.	Parâmetro das Saídas:	40
4.3.	Timers (Temporizadores).....	40
4.3.1.	Timer Tipo TON	40
4.3.2.	Timer Tipo TOFF	42
4.3.3.	Temporizador de Pulso	43
4.3.4.	Parâmetro dos Timers:	45
4.4.	Contadores	47
4.4.1.	Contador UP	47
4.4.2.	Contador DOWN	50
4.4.3.	Contador UP-DOWN.....	52
4.4.4.	Contador Rápido	54
4.4.5.	Tipos de operandos aceitos:	57
4.5.	Instruções de Comparação	58
4.5.1.	IGUAL (=).....	58
4.5.2.	MAIOR QUE (>).....	58
4.5.3.	MENOR QUE (<).....	59
4.5.4.	MAIOR OU IGUAL QUE (>=).....	59
4.5.5.	MENOR OU IGUAL QUE (<=).....	60
4.5.6.	NÃO IGUAL (<>)	60
4.6.	Operações Matemáticas	61
4.6.1.	Soma	61
4.6.2.	Subtração	61
4.6.3.	Multiplicação	61
4.6.4.	Divisão	62
4.7.	Funções Especiais	62
4.7.1.	CALL	62
4.7.2.	JUMP	63
4.7.3.	Move	63
5.	Ferramenta KeyProgram.....	64

5.1.	Visão Geral do Keyprogram	64
5.2.	Instalando o Keyprogram.....	64
5.3.	Iniciando o Keyprogram	68
5.4.	Tela Inicial	69
5.4.1.	Janela de Configuração Inicial.....	70
5.4.2.	Criando um Projeto	71
5.4.3.	Abrindo um Projeto.....	72
5.4.4.	Salvando um Projeto	73
5.5.	Área de Trabalho do Keyprogram	74
5.5.1.	Barra de Menus	75
5.5.1.A.	Menu Projeto	76
5.5.1.B.	Menu Edição	77
5.5.1.C.	Menu Exibir.....	78
5.5.1.D.	Menu Ferramenta:	79
5.5.1.E.	Menu CLP:.....	81
5.5.1.F.	Menu Configurações:.....	82
5.5.2.	Barra de Ferramentas	83
5.6.	Criando e executando uma Lógica.....	85
5.7.	Áreas de Programação do Keyprogram	89
5.7.1.	Área de Programação Ladder.....	89
•	Descrição da Barra de Ferramentas (Ladder).....	90
	Menu de Edição	90
✓	Grupo Geral	90
✓	Grupo de Contatos	90
✓	Grupo de Bobinas	91
✓	Grupo de Temporizadores.....	91
✓	Grupo de Operações Matemáticas	92
✓	Grupo de Comparadores	92
✓	Grupo de Funções Especiais	93
5.7.2.	Área de Trabalho da IHM	93
•	Descrição da Barra de Ferramentas (IHM)	95
5.8.	Tabelas de Símbolos	96
5.9.	Lista de String.....	97
5.10.	Configuração dos Temporizadores	98
5.11.	Janela de Configurações das Páginas.....	99
5.12.	Campos	100
5.12.1.	Campo de Edição	100

5.12.2. Campo de Visualização.....	101
5.12.3. Campo String	102
5.12.4. Campo Seletor.....	102
5.13. Sub-rotinas	103
5.14. Configuração da Comunicação	106
5.15. Instruções Válidas para o KL640.....	107
6. Glossário	109

Informações Gerais

Este capítulo apresenta informações sobre o conteúdo deste manual e suas funcionalidades, descreve as principais características do PLC KL640.

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 KL640.

- 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 KL640 é um dos modelos mais evoluídos da nossa linha de PLC's.

Engloba display de cristal líquido (LCD) de 4x16 (4 linhas por 16 dígitos) com Backlight, teclado numérico, teclas de funções especiais, teclas de navegação, led's indicadores independentes, módulos de entradas digitais (tipo N ou tipo P), módulos de saídas digitais (tipo N ou tipo P), módulos de entradas rápidas, módulos de entradas analógicas, módulos de saídas analógicas, tudo isso contido em um único equipamento.

Possui também memória RAM de 32KB, EEPROM de 30KB e FLASH de 64KB, Circuito de Watch-Dog, relógio de tempo real, dois canais de comunicação serial, sendo um canal RS232 e um canal RS485.

Construído com o objetivo de ser inserido no ambiente industrial de Processos de Produção para aumentar a velocidade e controlar suas produções o PLC KL640 adaptou o dinamismo, controle e robustez em um equipamento pequeno, ideal para aplicações de pequeno porte. Nosso equipamento veio para substituir os sistemas a Relés (painéis eletromecânicos) trazendo as vantagens de:

- . Menor espaço na instalação;
- . Maior economia 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 KL640 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

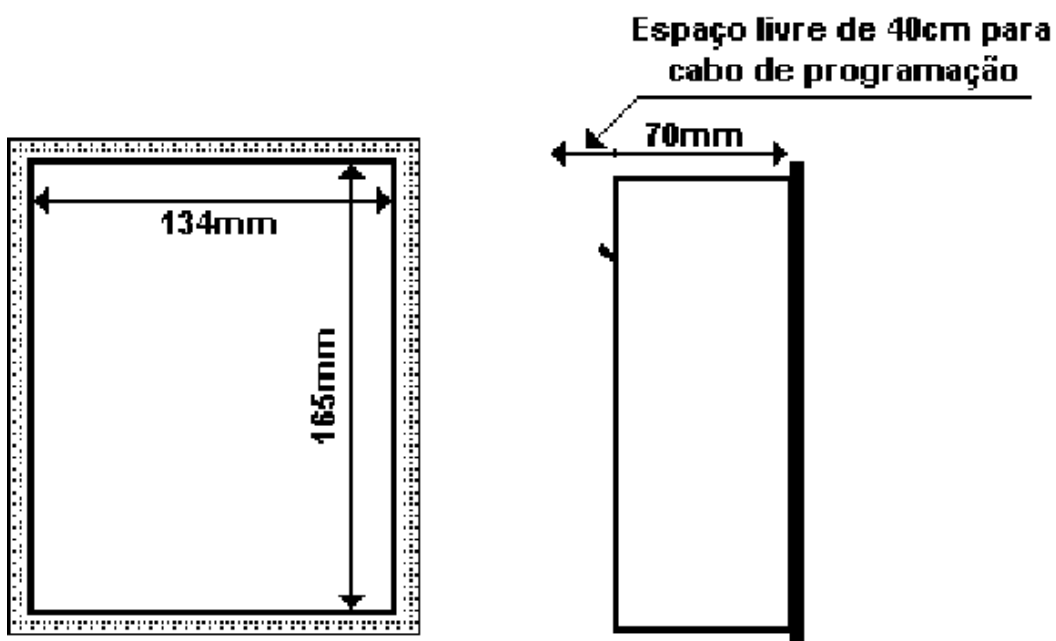
Processador: 400 MHz com 64 MB de RAM ou superior.

Sistema Operacional: Windows 98, ME ou XP.

2. Especificações do Equipamento

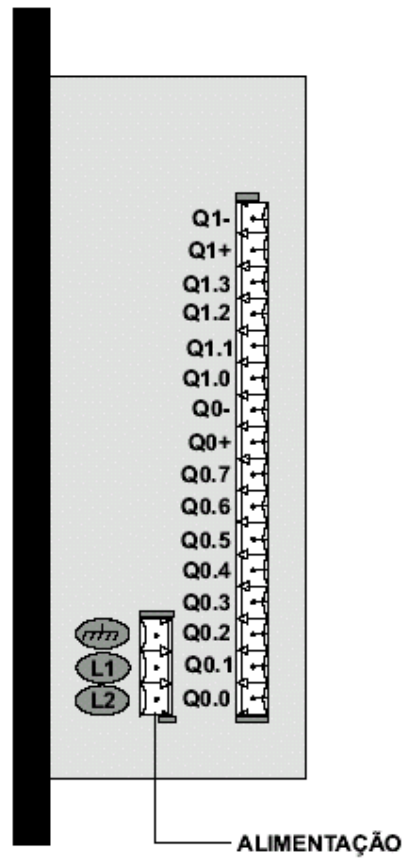
2.1. Instalação

Dimensões para o rasgo de instalação



2.2. Alimentação

Os pinos (L1 e L2) devem ser alimentados com 110 VAC ou 220 VAC dependendo do modelo do equipamento. É apresentada a seguir, a figura dos pinos que devem ser alimentados para a ligação do PLC.

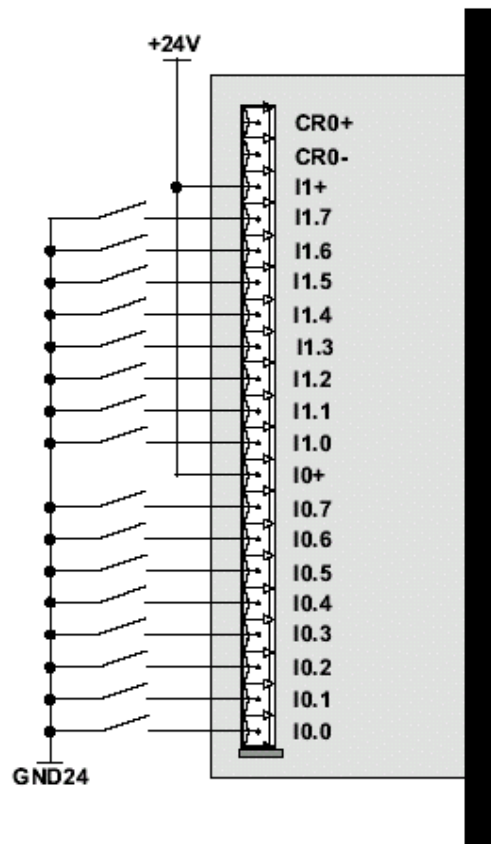


2.3. Entradas Digitais

2.3.1. Entrada Tipo N (24Vcc)

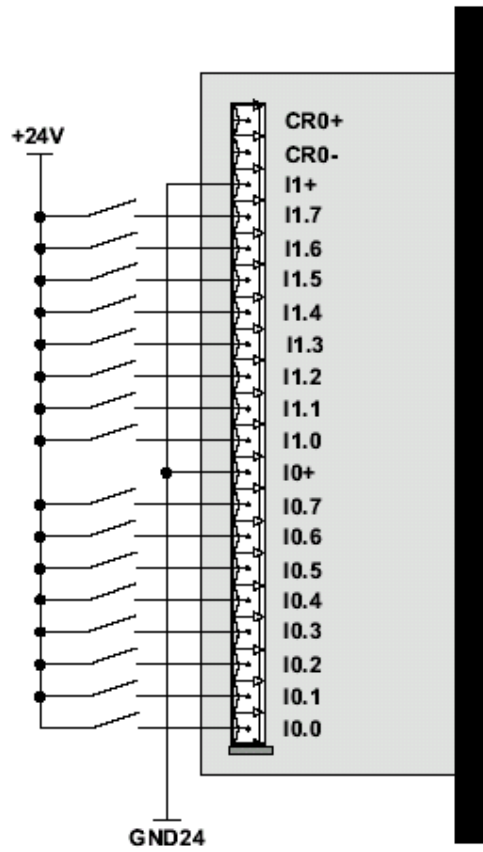
A entrada do Tipo N é acionada com 0 Vcc, como mostra abaixo:

O PLC KL640 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.



2.3.2. Entrada Tipo P (24Vcc)

A entrada do Tipo P é acionada com +24 Vcc, como mostra abaixo:



2.3.3. 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	32

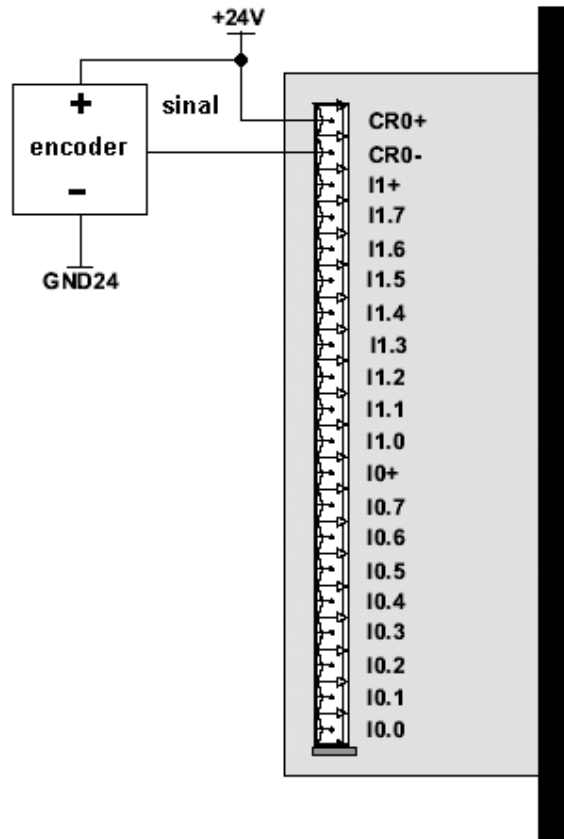
2.3.4. Endereçamento das Entradas

Quantidade de Entradas	Endereços	Descrição
8	%I0.0 a %I0.7	Bits referentes às Entradas Digitais
16	%I0.0 a %I1.7	
24	%I0.0 a %I2.7	
32	%I0.0 a %I3.7	

2.4. Entrada Rápida

A entrada Rápida é acionada com +24 Vcc em CR0+ e Gnd em CR0-, como pode ser representada abaixo:

Possui frequência máxima de 5 KHz e sua contagem é unidirecional.



Exemplo de ligação da entrada rápida utilizando encoder

2.4.1. Características das Entradas Rápidas

Entrada de Contagem Rápida	
Range de Tensão Quantidade	24Vdc (19,5 à 32Vdc)
Contagem por	Borda de Descida
Frequência Máxima	5KHz (quadrada, 50%)
Largura de Pulso Mínima Quantidade	100 s 1

2.4.2. Endereçamento das Entradas

A entrada rápida propriamente dita não possui endereço físico, mas é representada pelo do contador rápido através dos seus blocos. É representada abaixo a tabela de endereçamento desses blocos:

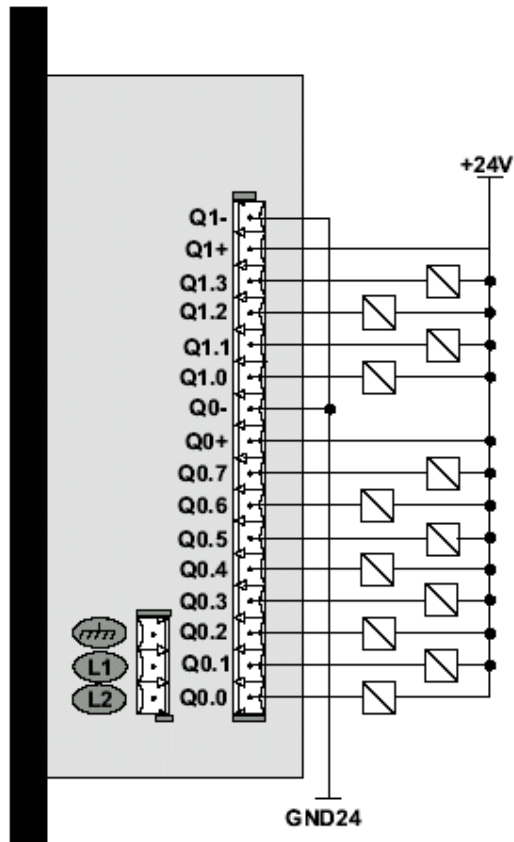
Quantidade máxima dos Blocos	Endereços	Descrição
8	%CR0 a %CR7	Bits referentes aos Blocos

2.5. Saídas Digitais

2.5.1. Saídas Tipo N

As saídas Tipo N serão acionadas ou desacionadas de acordo com a lógica que o usuário a programou. As saídas apresentam os contatos comuns, descrito por (Q0- a Q3-) e pelos contatos Normalmente aberto - NA (Q0.0 a Q3.3).

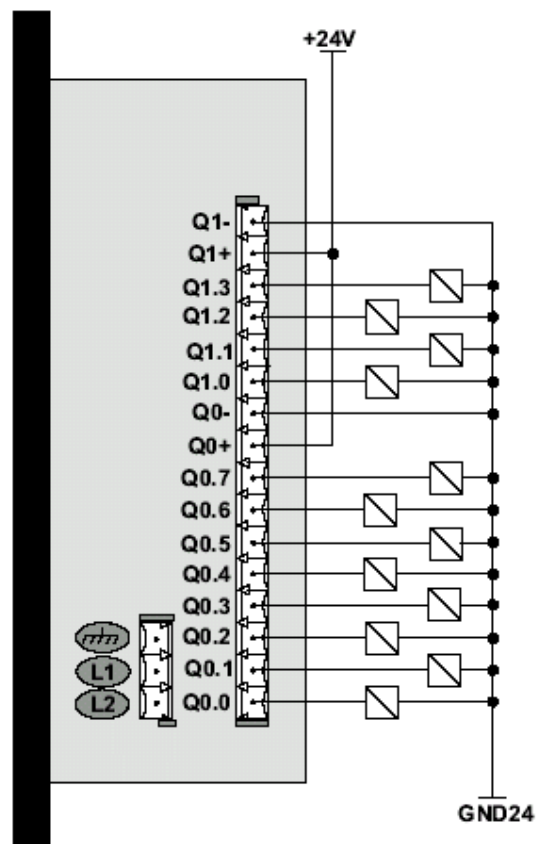
Abaixo é representada a forma de ligações das saídas:



2.5.2. Saídas Tipo P

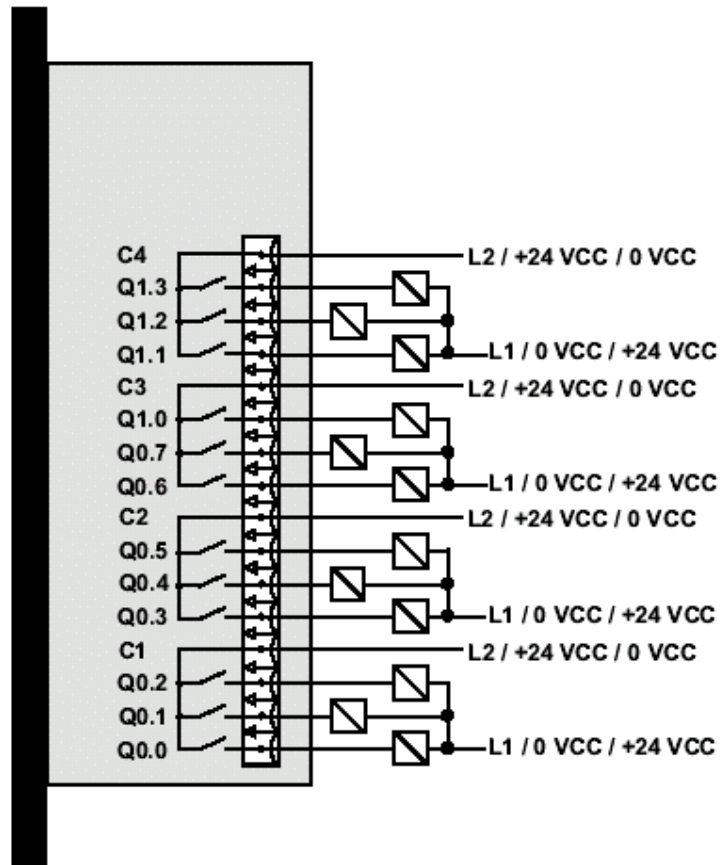
As saídas Tipo P serão acionadas ou desacionadas de acordo com a lógica que o usuário a programou. As saídas apresentam os contatos comuns, descrito por (Q0+ a Q3+) e pelos contatos Normalmente aberto - NA (Q0.0 a Q3.3).

Abaixo é representada a forma de ligações das saídas:



2.5.3. 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, descrito por (%Q0+ e %Q3+) e pelos contatos Normalmente aberto - NA (%Q0.0 a %Q3.3).



2.5.4. Características das Saídas

Range de tensão	24Vdc (14 à 32Vdc)
Corrente máxima	2A
Isolação	1.500 V
Led Indicador de status	Vermelho
Quantidade	28

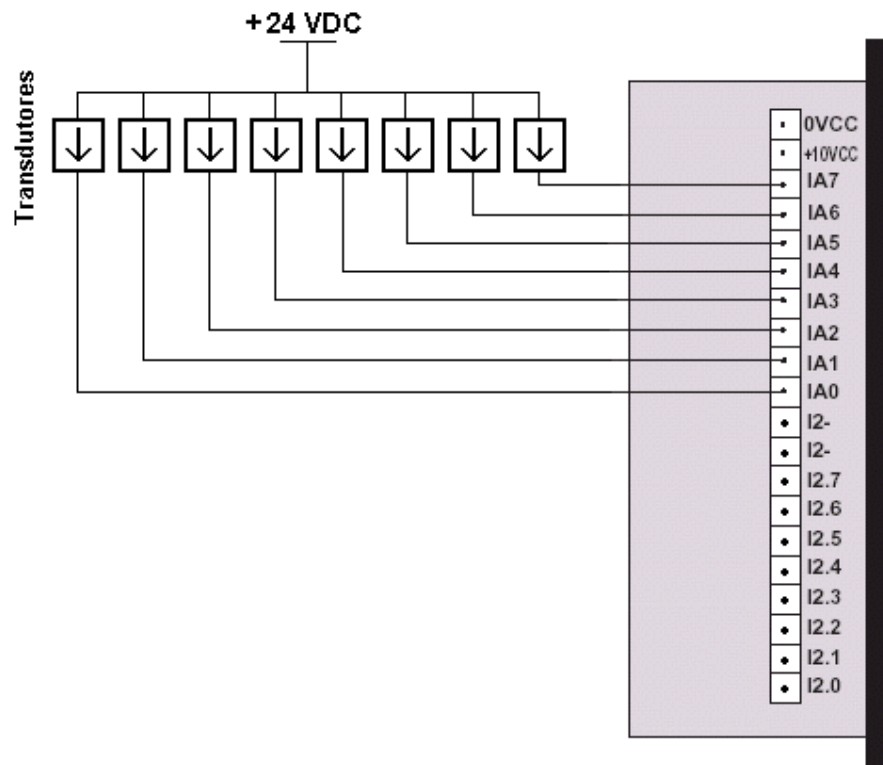
2.5.5. Endereçamento de Saídas

Quantidade de Saídas	Endereços	Descrição
8	%Q0.0 a %Q0.7	Bits referentes às Saídas Digitais
12	%Q0.0 a %Q1.3	
20	%Q0.0 a %Q2.3	
28	%Q0.0 a %Q3.3	

2.6. Entradas Analógicas:

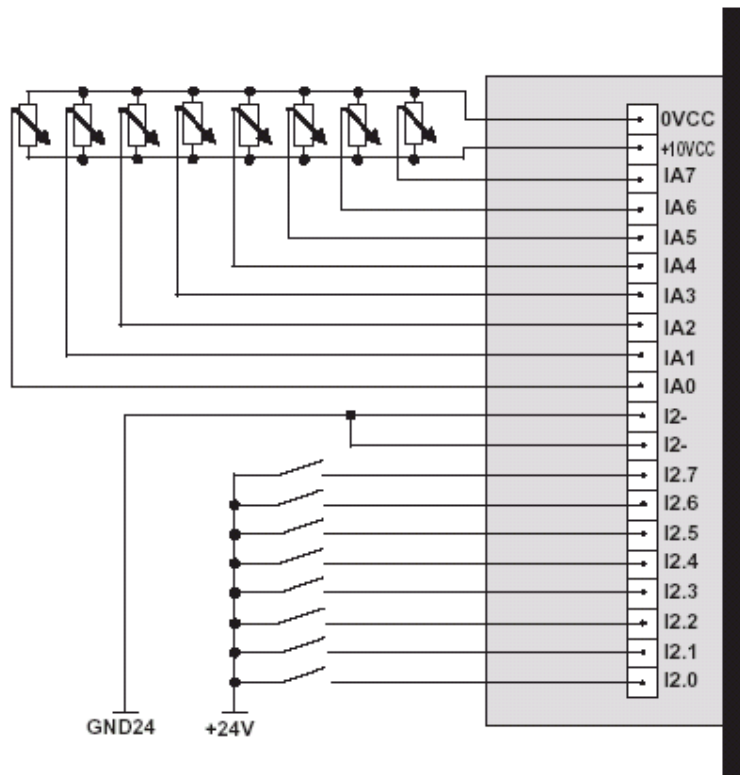
2.6.1. Por Corrente

É representado abaixo, o esquema de utilização das Entradas Analógicas por tensão:



2.6.2. Por Tensão

É representado abaixo, o esquema de utilização das Entradas Analógicas por tensão:



2.6.3. Características das Entradas Analógicas

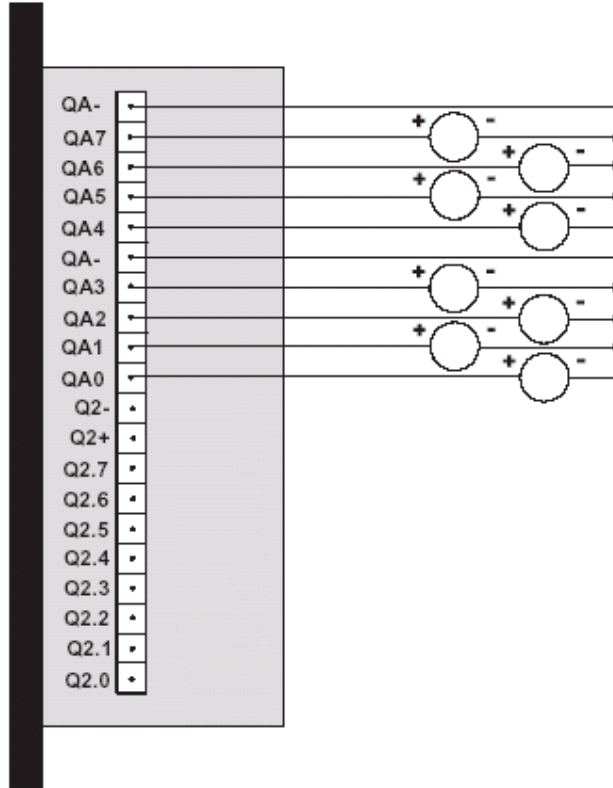
Entradas Analógicas	
IA0	0 a 10Vcc ou 0 a 20mA
IA1	0 a 10Vcc ou 0 a 20mA
IA2	0 a 10Vcc
IA3	0 a 10Vcc
IA4	0 a 10Vcc
IA5	0 a 10Vcc
IA6	0 a 10Vcc
IA7	0 a 10Vcc
Resolução em Tensão	12 bits (2,44mV)
Impedância em Tensão	10K Ω
Resolução em Corrente	12 bits (4,8 μ A)
Impedância em Corrente	500 Ω
Quantidade	8

2.6.4. Endereçamento das Entradas Analógicas

Quantidade de Entradas	Endereço	Descrição
2	%IA0 a %IA1	12 bits de resolução referentes às Entradas Analógicas
4	%IA0 a %IA3	
8	%IA0 a %IA7	

2.6.5. Descrição das Saídas Analógicas:

É representada abaixo, a forma de ligação das Saídas Analógicas:



2.6.6. Características das Saídas Analógicas

Entradas Analógicas	
QA0	0 a 10Vcc
QA1	0 a 10Vcc
QA2	0 a 10Vcc
QA3	0 a 10Vcc
QA4	0 a 10Vcc
QA5	0 a 10Vcc
QA6	0 a 10Vcc
QA7	0 a 10Vcc
Resolução	8 bits (39mV)
Impedância	200Ω
Quantidade	8

2.6.7. Endereçamento de Saídas

Quantidade de Saídas	Endereço	Descrição
2	%QA0 a %QA1	8 bits de resolução referentes às Saídas Analógicas
4	%QA0 a %QA3	
8	%QA0 a %QA7	

2.7. Comunicação

2.7.1. Modo de Comunicação:

Possui dois canais de comunicação, possibilitando a comunicação pelo canal RS-485, ou pelo canal RS-232.

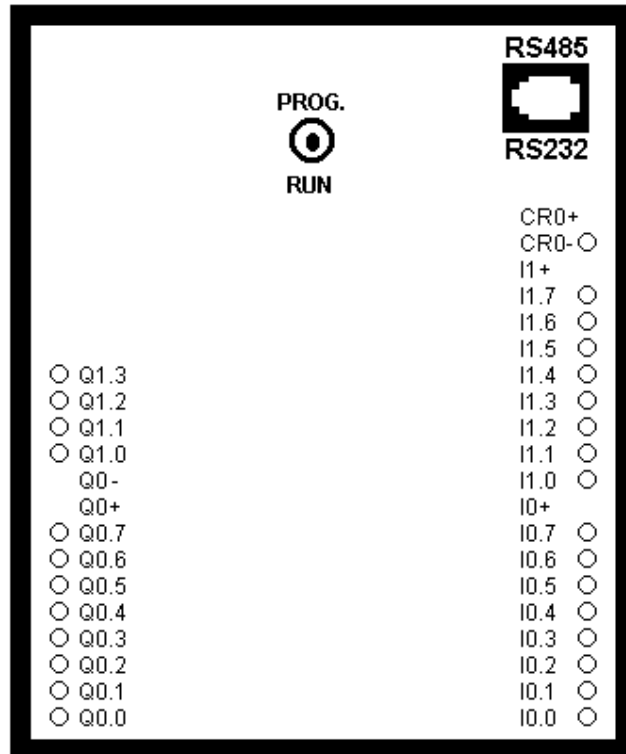
A programação e monitoração do CLP são realizadas por meio de um PC, utilizando como meio de comunicação a interface serial RS-232.

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.

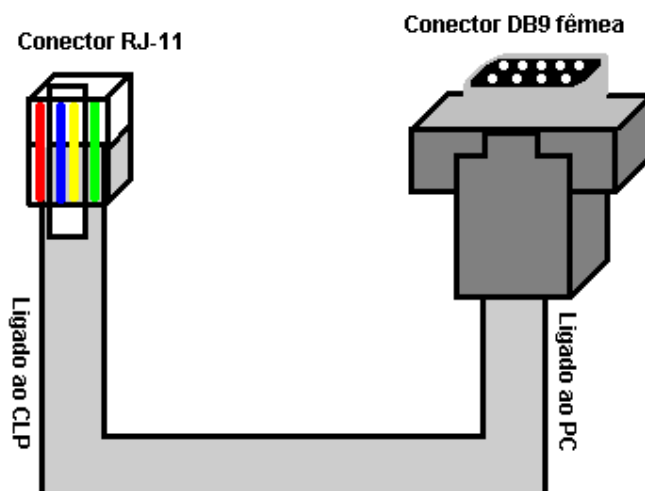
A taxa de transmissão (Baud rate) utilizada na comunicação de nosso equipamento, ao PC é de 115200 bps - taxa máxima suportada pelo meio físico.

O equipamento utiliza um conector RJ-11 de 6 pinos para conexão da interface RS-232 na parte traseira do equipamento, como é mostrado na figura abaixo:

Comunicação Serial



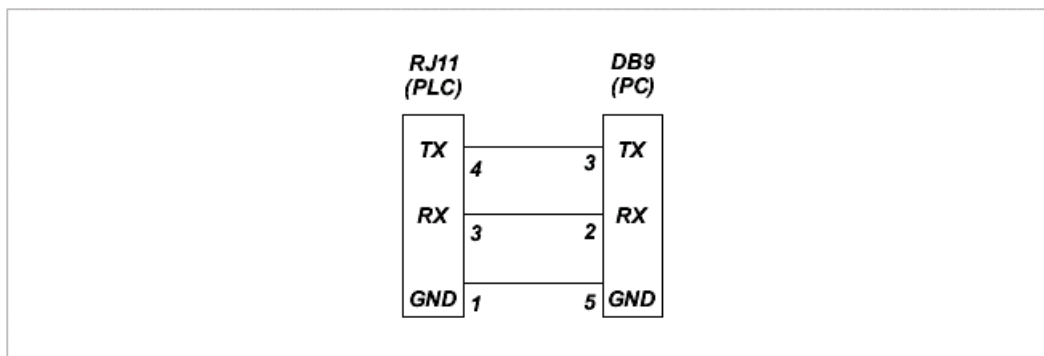
Cabo de comunicação utilizado para conectar o PLC ao PC:



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

Já para a Interface RS-485 o comprimento máximo permitido é de aproximadamente 1Km.

Cabo de Programação



3. Endereçamento

Endereços Externos					
Descrição	Endereços				Tamanho
Entradas Digitais	8	16	24	32	1 Bit (Nível 1 ou Nível 0)
	%I0.0 a %I0.7	%I0.0 a %I1.7	%I0.0 a %I2.7	%I0.0 a %I3.7	
Saídas Digitais	8	12	20	28	1 Bit (Nível 1 ou Nível 0)
	%Q0.0 a %Q0.7	%Q0.0 a %Q1.3	%Q0.0 a %Q2.3	%Q0.0 a %Q3.3	
Entrada Rápida	1 entrada / 8 blocos: %CR0 a %CR7				1 Bit (Nível 1 ou Nível 0)
Entradas Analógicas		2	4	8	12 Bits (De 0 a 4095)
	-	%IA0 a %IA1	%IA0 a %IA3	%IA0 a %IA7	
Saídas Analógicas	0	2	4	8	8 Bits (De 0 255)
	-	%QA0 a %QA1	%QA0 a %QA3	%QA0 a %QA7	
Teclas de IHM	%TC.OP1, %TC.OP2, %TC.UP, %TC.DOWN, %TC.MANUT, %TC.ALT, %TC.ENTER, %TC0 ~ %TC9, %TC.F1 ~ %TC.F8				1 Bit (Nível 1 ou Nível 0)
Leds de Indicação	%LT.F1 a %LT.F8				1 Bit (Nível 1 ou Nível 0)

Endereços Internos		
Descrição	Endereços	Tamanho
Bits de Memória	%M0 ~ %M255	1 Bit (Nível 1 ou Nível 0)
Bits Retentivos	%MR0 ~ %MR255	1 Bit (Nível 1 ou Nível 0)
Bytes de Memória	%MB0 ~ %MB255	1 Byte (De 0 a 255)
Bytes Retentivos	%MBR0 ~ %MBR255	1 Byte (De 0 a 255)
Word de Memória	%MW0 ~ %MW255	2 Bytes (De 0 a 65535)
Word Retentivos	%MWR0 ~ %MWR255	2 Bytes (De 0 a 65535)
Sempre Ligado	%S0	1 Bit (Nível 1 ou Nível 0)
Sempre Desligado	%S1	1 Bit (Nível 1 ou Nível 0)
Ligado Na Primeira Varredura	%S2	1 Bit (Nível 1 ou Nível 0)
Desligado Na Primeira Varredura	%S3	1 Bit (Nível 1 ou Nível 0)
Clock de 0.5s (0.5 low, 0.5 High)	%S4	1 Bit (Nível 1 ou Nível 0)
Clock de 30s (30 low, 30 High)	%S5	1 Bit (Nível 1 ou Nível 0)
Chave de Modo (Run/Prog)	%S6	1 Bit (Nível 1 ou Nível 0)
Serial (On-line/Off-line)	%S7	1 Bit (Nível 1 ou Nível 0)
RESERVADO PARA USO FUTURO	%S8-%S255	1 Bit (Nível 1 ou Nível 0)

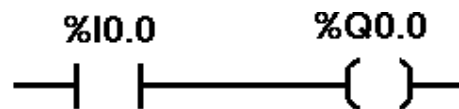
4. Instruções de Programação Ladder

4.1. Contatos

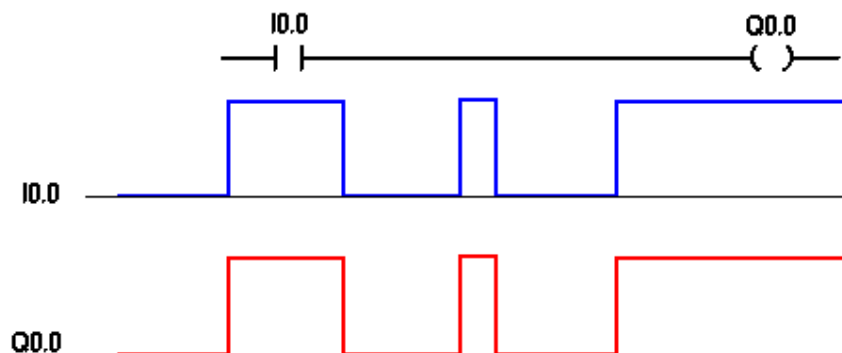
4.1.1. Contato Normalmente Aberto



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.

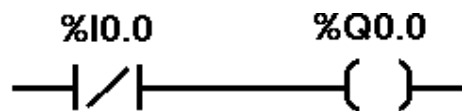


A saída (%Q0.0) ficará acionada quando o contato (%I0.0) permanecer em nível lógico 1. No momento que o contato (%I0.0) estiver em nível lógico 0, a saída (%Q0.0) estará desacionada.

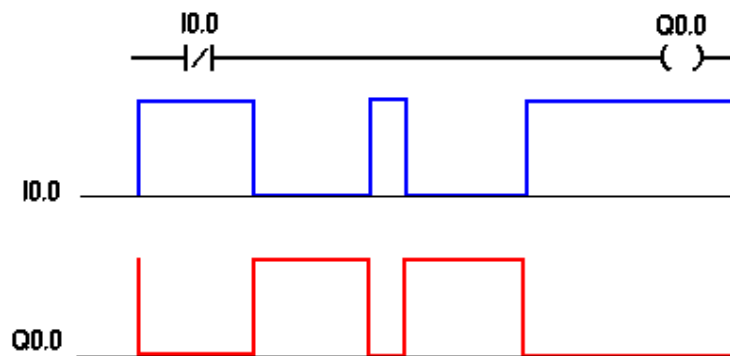
4.1.2. Contato Normalmente Fechado



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



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

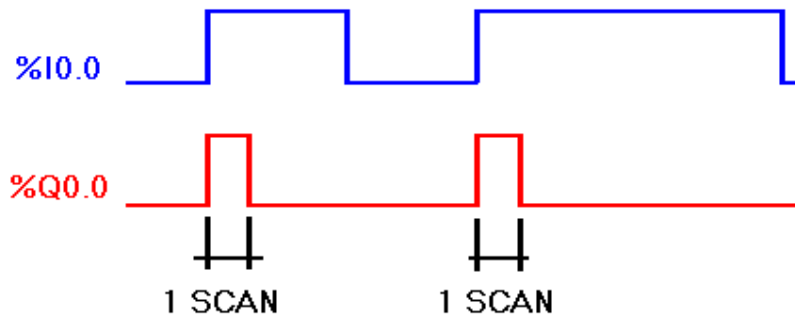
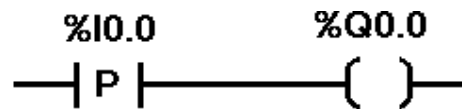


A saída (%Q0.0) permanecerá acionado enquanto a entrada (%I0.0) estiver em nível lógico 0. No momento que o contato (%I0.0) estiver em nível lógico 1, então a saída (%Q0.0) estará desacionada.

4.1.3. 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.

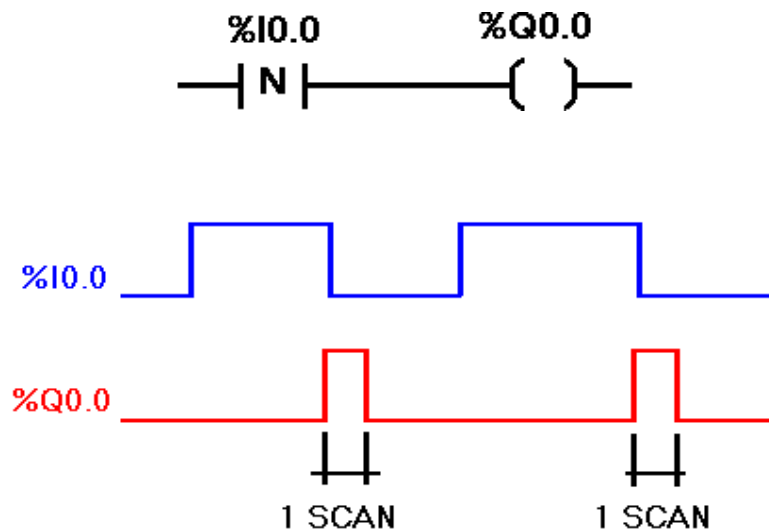


No exemplo do gráfico mostrado acima, a saída (%Q0.0) será acionada quando a entrada (%I0.0) estiver mudando de nível lógico 0 para nível lógico 1, e permanecerá ativada durante o tempo de 1 scan.

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);
- ❖ %TCxx(Tecla do Teclado);
- ❖ %LTxx (Led da IHM);
- ❖ %Sxx (Bit de Sistema);
- ❖ %Txx.Q (Saída de um Temporizador);
- ❖ %Cxx.E, %Cxx.D, %Cxx.F (Saídas de um Contador).

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

4.2. Bobinas

4.2.1. Saída Simples

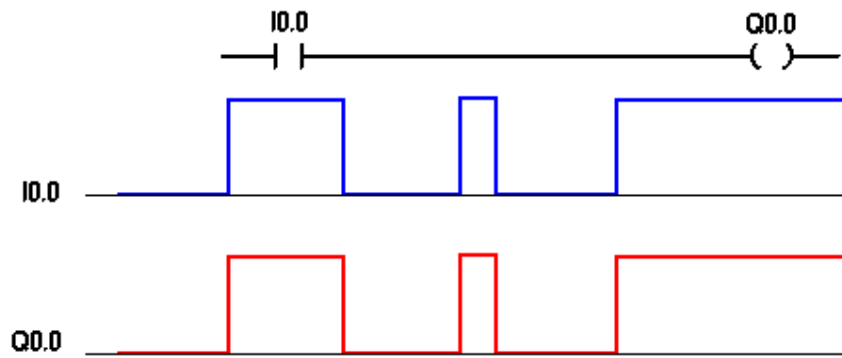


Descrição: uma saída simples transfere para o endereço associado a ela, o valor de sua entrada. Caso a lógica associada à saída estiver em 1 transfere 1 para o endereço, caso contrário transfere 0.



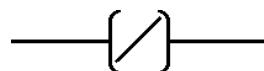
No exemplo acima a saída (%Q0.0) será acionada somente quando a entrada (%I0.0) for acionada.

A saída permanece ativada enquanto a entrada estiver acionada.

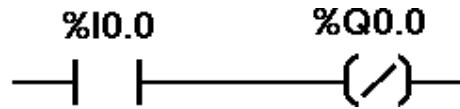


O gráfico acima mostra que a saída (%Q0.0) permanecerá acionado enquanto a entrada (%I0.0) estiver em nível lógico 1. No momento que o contato (%I0.0) estiver em nível lógico 0, então a saída %Q0.0 estará desacionada.

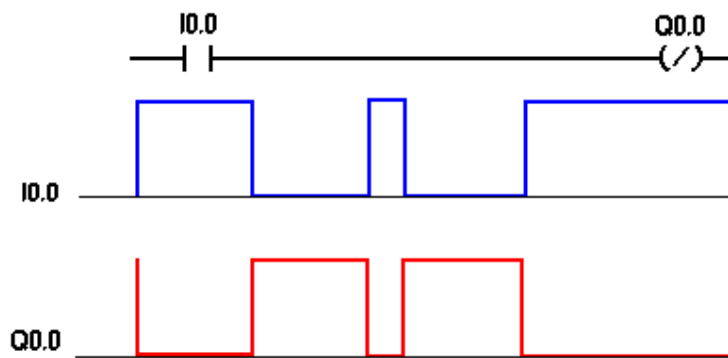
4.2.2. Saída Complementar



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.

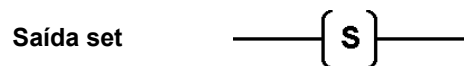


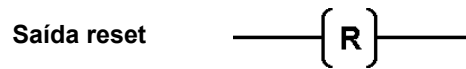
No exemplo acima a saída (%Q0.0) será acionada somente quando a entrada (%I0.0) estiver desacionada.



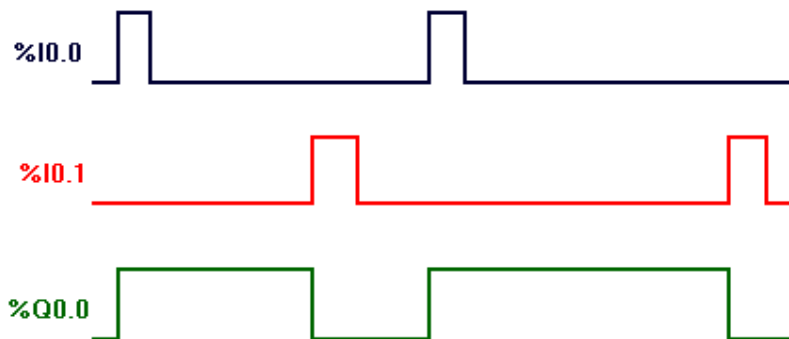
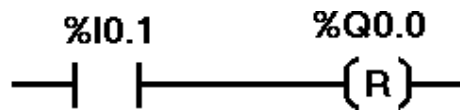
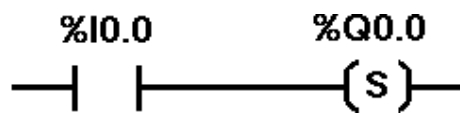
O gráfico acima mostra que a saída (%Q0.0) permanecerá acionado enquanto a entrada (%I0.0) estiver em nível lógico 0. No momento que o contato (%I0.0) estiver em nível lógico 1, então a saída %Q0.0 estará desacionada.

4.2.3. Saída Set e Reset





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) for 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.

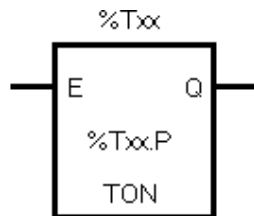
Tipos de Operandos aceitos:

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

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

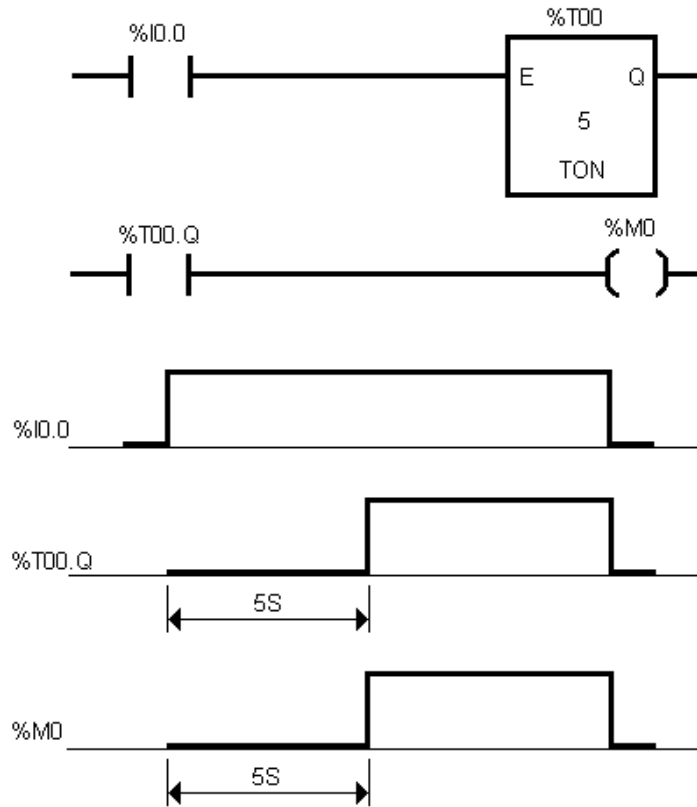
4.3. *Timers (Temporizadores)*

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.

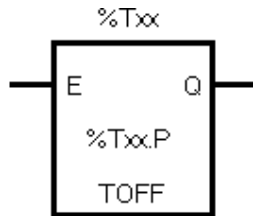
Quando a entrada Enable for desabilitada, o valor atual do Timer será zerado, e a saída Q do Timer será desativada.



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. Timer Tipo TOFF



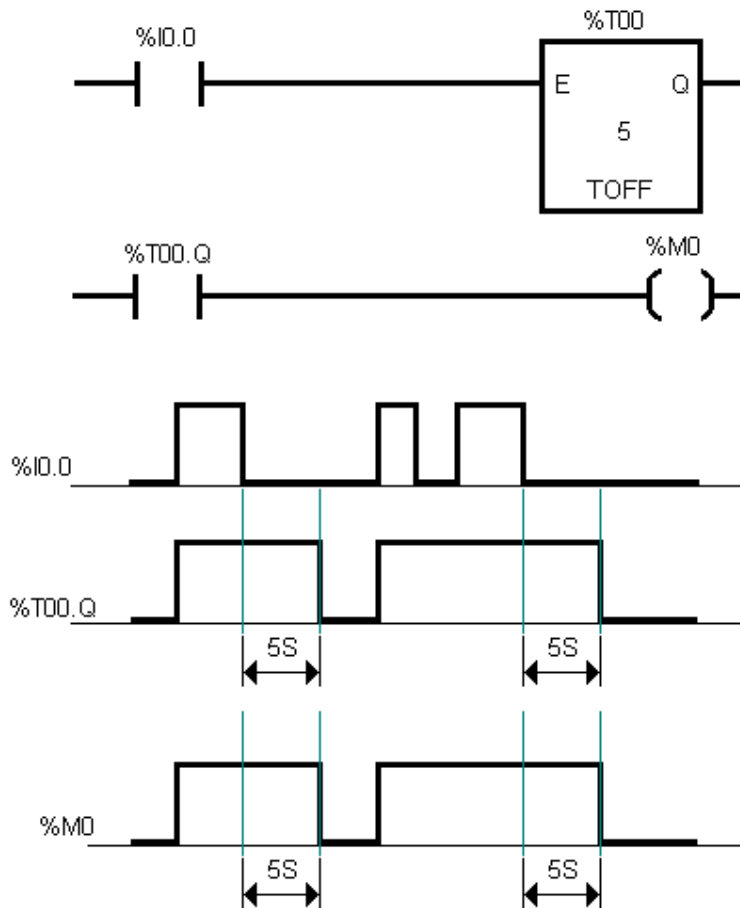
Descrição: esta instrução conta um tempo a partir do instante em que a entrada Enable (E) do Timer estiver desabilitada.

No momento que Entrada (E) for acionada a saída instantaneamente irá ser ativada.

Quando a Entrada do Timer mudar de estado (ir para nível 0), o valor atual do Timer (Txx.V) irá contar até chegar no valor de preset (Txx.P).

Quando o valor atual do Timer (Txx.V) alcançar o tempo de preset (Txx.P), definido pelo usuário, a saída (Q) do Timer será desacionada.

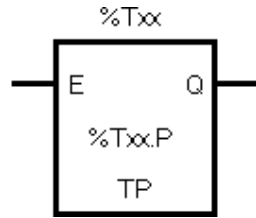
Quando a entrada Enable (E) for habilitada, o valor atual do Timer (Txx.V) será zerado e saída (Q) do Timer irá ser acionada.



No exemplo acima, a saída (Q) do timer é acionada quando a entrada (%I0.0) estiver fechada (nível lógico 1). Enquanto a entrada (%I0.0) estiver aberta (nível lógico 0), o valor atual do timer irá contar até que ele se iguale ao valor do preset, quando se igualar, imediatamente a saída (Q) do timer irá ser desacionada, desacionando assim, o bit (%M0).

Quando a entrada (%I0.0) estiver aberta (nível lógico 0) , o valor de contagem atual do Timer (%Txx.V) terá o valor zero.

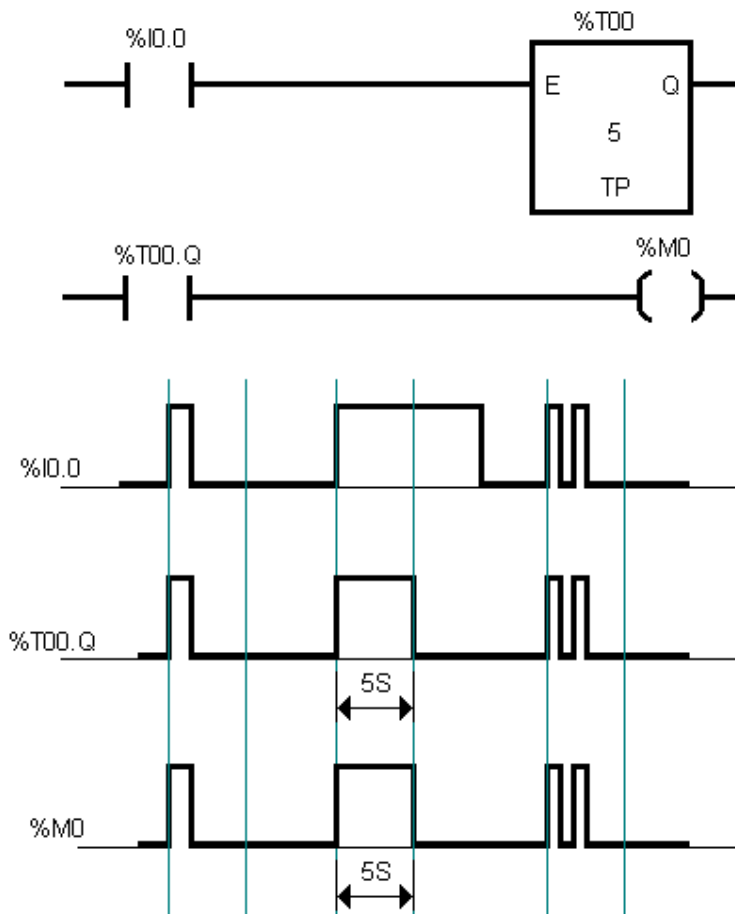
4.3.3. Temporizador de Pulso



Descrição: esta instrução conta um tempo a partir do instante em que a entrada Enable (E) do Timer estiver habilitada. Neste momento a saída (Q) do Timer é acionada e o valor atual do Timer (Txx.V) começará a contar enquanto a entrada (E) permanecer habilitada .

Quando o valor atual (Txx.V) se igualar ao valor do Tempo de Preset (Txx.P), a saída (Q) do Timer será desativada.

Quando a entrada (E) estiver em desabilitada, o valor atual do Timer irá ser zerado e a saída (Q) será desativada.



No exemplo acima, a saída (Q) do timer será acionada quando a entrada (%I0.0) estiver habilitada. Nesse mesmo momento, o timer começa a contar até chegar no tempo determinado pelo preset. Quando o valor atual do Timer for igual ao valor especificado pelo preset, a saída do timer (Q) será desacionada. Se a entrada (%I0.0) for habilitada e em seguida desabilitada em um tempo menor que o valor especificado pelo preset, então a saída (Q) do timer será acionada e desacionada no mesmo intervalo de tempo da entrada (%I0.0).

4.3.4. Parâmetro dos Timers:

O programador ao utilizar os Timer ON's, OFF,s e de Pulsos 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:

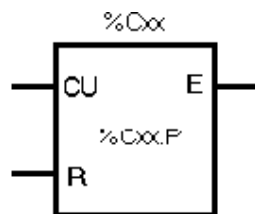
Na saída Q dos Timers os parâmetros aceitos são valores que contenham tamanho de 1 bit (nível 0 ou nível 1), ou seja, variáveis que contenham também tamanho de 1 bit. Elas podem ser:

- ❖ %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 (valor de 0 a 65535).

4.4. Contadores

4.4.1. Contador UP



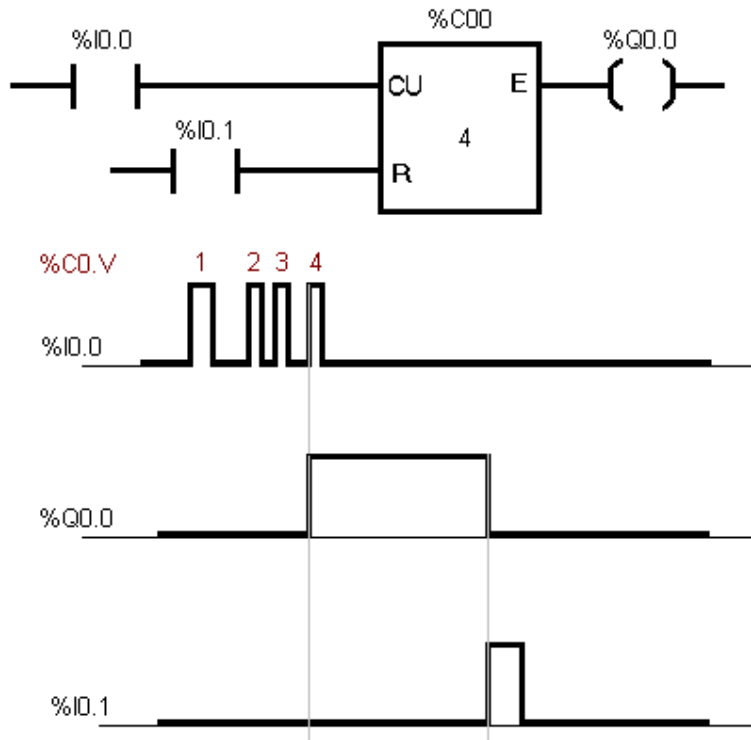
Descrição: esta instrução faz a contagem de pulsos, com o incremento de uma unidade a cada vez que o contato associado à entrada (CU) muda de estado desligado (nível 0), para estado ligado (nível 1).

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.

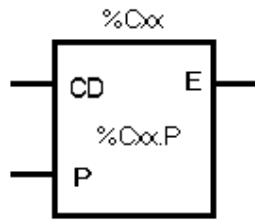
Parâmetros do Contador UP:

- ❖ **%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.
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 .

4.4.2. Contador DOWN



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

O valor atual do Contador (Cxx.V) é inicialmente carregado com o valor do preset (Cxx.P). Quando o valor atual do Contador se igualar a zero, a saída do Contador (E) será acionada.

Quando o Preset (P) do contador for habilitado, o valor do contador atual será igual ao valor do preset.

Parâmetros do Contador DOWN

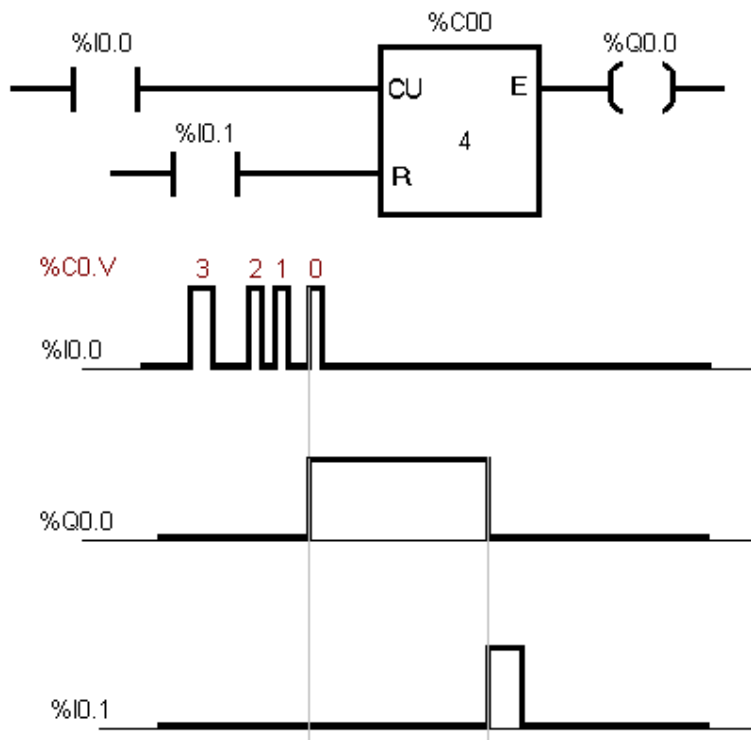
- ❖ **%Cxx:** Número do contador (C0 a C31), definido pelo usuário.

- ❖ **%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 decrementado a cada pulso recebido pela entrada (CD) do contador.
(Número inteiro de 0 a 65535).

- ❖ **CD:** Entrada dos pulsos do contador, definido pelo usuário. Bit (0 ou 1).

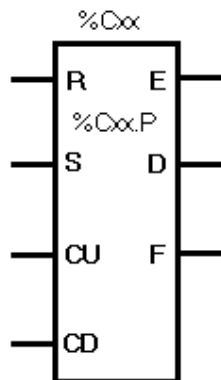
- ❖ **P:** Preset, definido pelo usuário. Quando ativado, a contagem atual do contador irá apresentar o valor do preset. Bit (0 ou 1).
- ❖ **E:** Status de saída do contador definido por software. É ativada quando o valor atual de contagem se igualar a zero. Bit (0 ou 1).



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 com o valor do preset, especificado com o número 4, até

o número zero. Se a entrada (%I0.1) for acionada, a saída (E) do contador será imediatamente desacionada.

4.4.3. Contador UP-DOWN



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) ou com o decremento de uma unidade a cada vez que o contato associado à entrada (CD) muda de estado desligado (nível 0), para estado (nível 1).

Quando o valor atual do contador for igual ou maior ao valor do preset, a saída (E) do contador será acionada. Já, quando o valor atual do contador for menor que o valor do preset, a saída será desacionada.

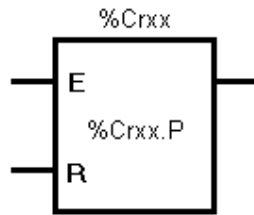
Quando o Reset (R) do contador estiver habilitado, o valor do contador atual será zerado.

Quando o Set (S) do contador estiver habilitado, o valor do contador atual será igual ao valor do preset e a saída (E) imediatamente ficará acionada.

Parâmetros do Contador UP-DOWN:

- ❖ **%Cxx:** Número do contador (C0 a C31), definido pelo usuário.
(Número inteiro de 0 a 65535).
- ❖ **%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).
- ❖ **CD:** Entrada dos pulsos do contador Down, definido pelo usuário. Bit (0 ou 1).
- ❖ **CU:** Entrada dos pulsos do contador UP, definido pelo usuário. Bit (0 ou 1).
- ❖ **S:** Set definido pelo usuário. Quando ativado, a contagem atual do contador irá apresentar o valor do preset.
Bit (0 ou 1).
- ❖ **R:** Reset definido pelo usuário. Quando ativado, a contagem atual do contador irá apresentar o valor zero.
Bit (0 ou 1).
- ❖ **E:** Status de saída do contador Up definido por software. É ativada quando o valor atual de contagem se igualar ao valor do preset do contador.
Bit (0 ou 1).
- ❖ **D:** Status de saída do contador Down definido por software. É ativada quando o valor atual de contagem se igualar a zero.
Bit (0 ou 1).

4.4.4. Contador Rápido



Descrição: Esta instrução é destinada na contagem de pulsos recebidos pela entrada rápida (pinos CR0+ e CR0-). Possibilitando uma frequência máxima de 5 KHz.

A cada pulso recebido na entrada rápida, é incrementado o valor atual de contagem (Crx.V) no bloco do contador rápido (Cr0. a Cr7).

Estando a entrada Enable (E) habilitada é feito uma comparação do valor atual (Crx.V) com o valor do Preset (Crx.P) do bloco do contador rápido. Caso o valor atual for igual ou maior que o valor do preset o bit relacionado ao bloco (Crx.Q) será acionado.

Caso a entrada Enable (E) não estiver habilitada, a comparação não será efetuada, no entanto, o valor atual da visualização continua sendo incrementado a cada pulso recebido da entrada rápida.

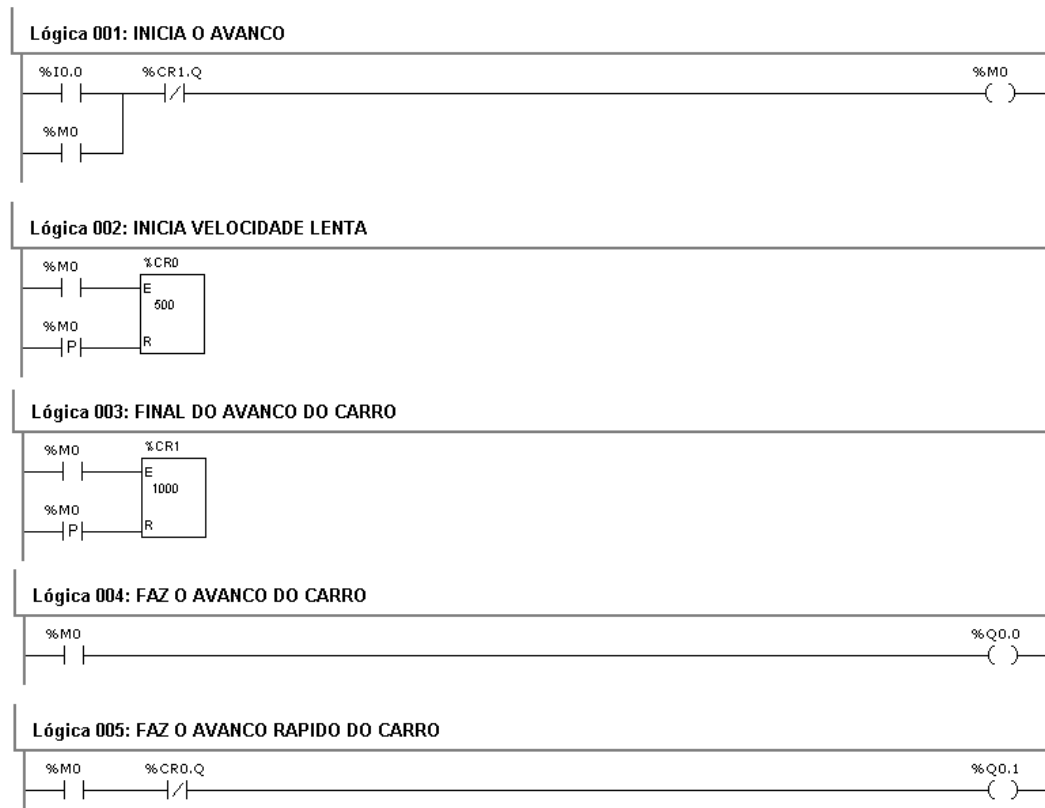
O valor atual do bloco do contador rápido pode ser zerado a qualquer instante através do acionamento do Reset (R) no bloco do contador rápido.

Parâmetros do Contador rápido:

- ❖ **%CRx:** Número do bloco do contador rápido (Cr0 a Cr7), definido pelo usuário. Podendo ter até 8 blocos em apenas um programa;
- ❖ **%CRx.P:** Valor do Preset do contador definido pelo usuário. (Número inteiro de 0 a 65535).

- ❖ **%CRx.V:** Valor Atual da contagem definido por software Este valor é incrementado a cada pulso recebido pela entrada (E) do contador.
(Número inteiro de 0 a 65535).
- ❖ **R:** Reset do contador definido pelo usuário . Bit (0 ou 1);
- ❖ **E:** Entrada do contador definido pelo usuário. Bit (0 ou 1).
- ❖ **%CRx.Q:** Status de saída do contador rápido. Bit (0 ou 1).
Indica que o valor atual do contador rápido (Crx.V) já chegou ao valor de preset (Crx.P) do contador rápido, acionando, portanto, esta saída.

Exemplo de Lógica utilizando Contador Rápido



Quando a entrada (I0.0) for acionada, o bit de memória (M0) é acionado, permanecendo acionado enquanto o bit (Cr1.Q) estiver desacionado.

O bit (M0) habilita os dois blocos do contador rápido, e ao mesmo tempo reseta os blocos através do contato de pulso (acionado durante uma varredura do programa). Neste momento, os valores das visualizações dos dois blocos estarão com 0.

Na lógica 4, o bit (M0) aciona a saída (Q0.0), avançando assim o carro.

Na lógica 5, o bit (M0) também aciona a saída (Q0.1), que faz o avanço rápido, A saída (Q0.0) permanecerá acionada enquanto o bit (Cr0.Q) estiver desacionado.

Quando a visualização do contador rápido (Cr0.V) for igual ou maior que o valor de preset (Cr0.P), o bit (Cr0.Q) será acionado, desligando a saída (Q0.1), fazendo assim, com que o carro diminua a velocidade.

O carro permanecerá avançando, enquanto o bit (Cr1.Q) não for acionado, sendo que este bit, só será acionado quando o valor da visualização (Cr1.V) for igual ou maior que o valor do preset (Cr1.P) do bloco (Cr1). Ao acionar o bit de saída do contador rápido (Cr1.Q), o bit (M0) será desacionado na lógica 1, desligando assim, a saída do avanço do carro (Q0.0).

Exemplo de Aplicação

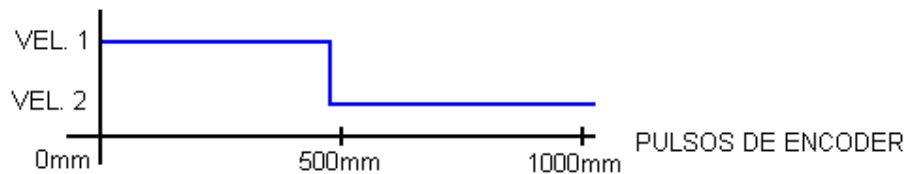
Exemplo de aplicação utilizando bloco do contador rápido em uma máquina curvadora de tubos, onde o avanço do carro que transporta o tubo a ser encurvado avança em duas velocidades, sendo:

1º Avanço rápido (de 0m a 500m)

2º Avanço lento (de 501m a 1000m)

O carro avança até a distância de 500m com uma velocidade rápida, e depois avança até a distância 1000m com velocidade lenta.

Obs: Para tornar fácil o entendimento do exemplo, considera-se que a cada pulso recebido do encoder equivale a 1 milímetro.



4.4.5. Tipos de operandos aceitos:

O programador ao utilizar os Contadores UP, DOWN, UP-DOWN e Rápido, devem 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 dos contadores os parâmetro aceitos são valores que contenham o tamanho de 1 Word (valores inteiros de 0 a 65535).

4.5. Instruções de Comparação

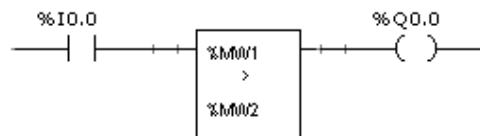
Estas instruções têm como função comparar dois valores, caso estes valores satisfaçam a condição de comparação e a entrada do comparador estiver acionada, então a saída do comparador será acionada, habilitando assim, por exemplo, uma saída.

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.

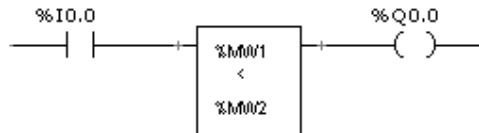
4.5.2. MAIOR QUE (>)



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,

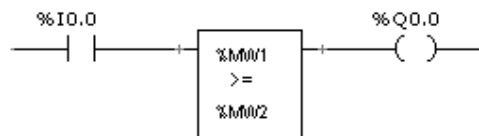
o resultado será nível lógico 0, e conseqüentemente a saída será desacionada.

4.5.3. MENOR QUE (<)



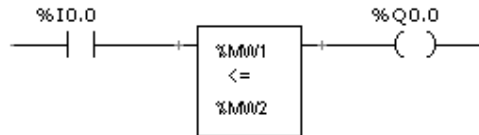
No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação entre o operando 1 e o operando 2. Caso o operando 1 seja menor que o operando 2, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for maior ou igual ao operando 2, o resultado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

4.5.4. MAIOR OU IGUAL QUE (>=)



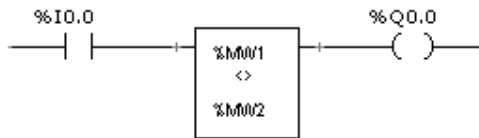
No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação entre o operando 1 e o operando 2. Caso o operando 1 seja maior ou igual ao operando 2, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for menor que operando 2, o resultado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

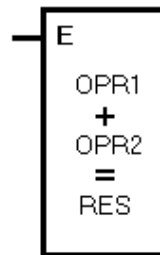


No exemplo acima, quando a entrada %I0.0 estiver habilitada inicia-se a comparação de igualdade entre o operando 1 e o operando 2. Caso eles sejam diferentes, o resultado será nível lógico 1, e conseqüentemente a saída será acionada. Já, se o operando 1 for igual ao operando 2, o resultado será nível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

4.6. Operações Matemáticas

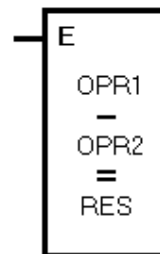
Estas instruções têm como função executar operações aritméticas de dois operandos e colocá-lo em um operando de resposta (RES).

4.6.1. Soma



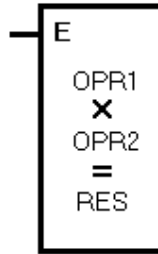
Quando habilitado através da entrada E, executa a soma de OPR1 + OPR2 e coloca o resultado em RES.

4.6.2. Subtração



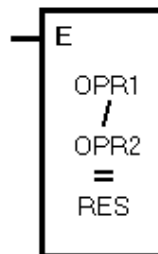
Quando habilitado através da entrada E, executa a Subtração em OPR1 do valor de OPR2 e coloca o resultado em RES.

4.6.3. Multiplicação



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

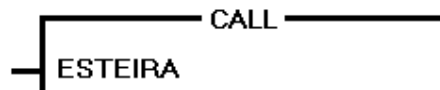
4.6.4. Divisão



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

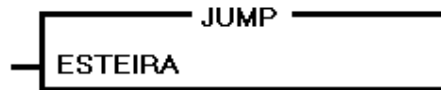
4.7. Funções Especiais

4.7.1. CALL



Quando a entrada da instrução CALL é habilitada, ao passar pela instrução o programa executa a sub-rotina indicada em CALL e após a execução volta para o mesmo ponto do programa que chamou a sub-rotina.

4.7.2. **JUMP**



Quando a entrada da instrução JUMP é habilitada, ao passar pela instrução o programa pula para a sub-rotina indicada em JUMP. Esse pulo não tem retorno, ele pula para a sub-rotina e não retorna para a rotina que chamou a sub-rotina.

4.7.3. **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

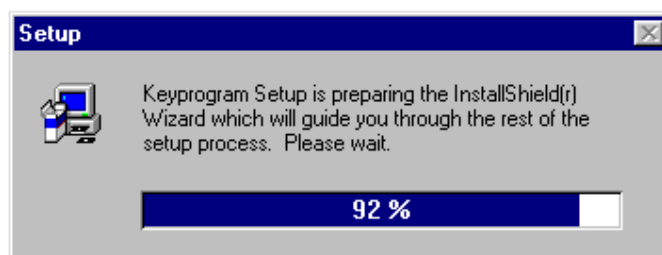
A principal importância da ferramenta Keyprogram é facilitar a programação Ladder em uma plataforma simples, ágil e amigável.

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:

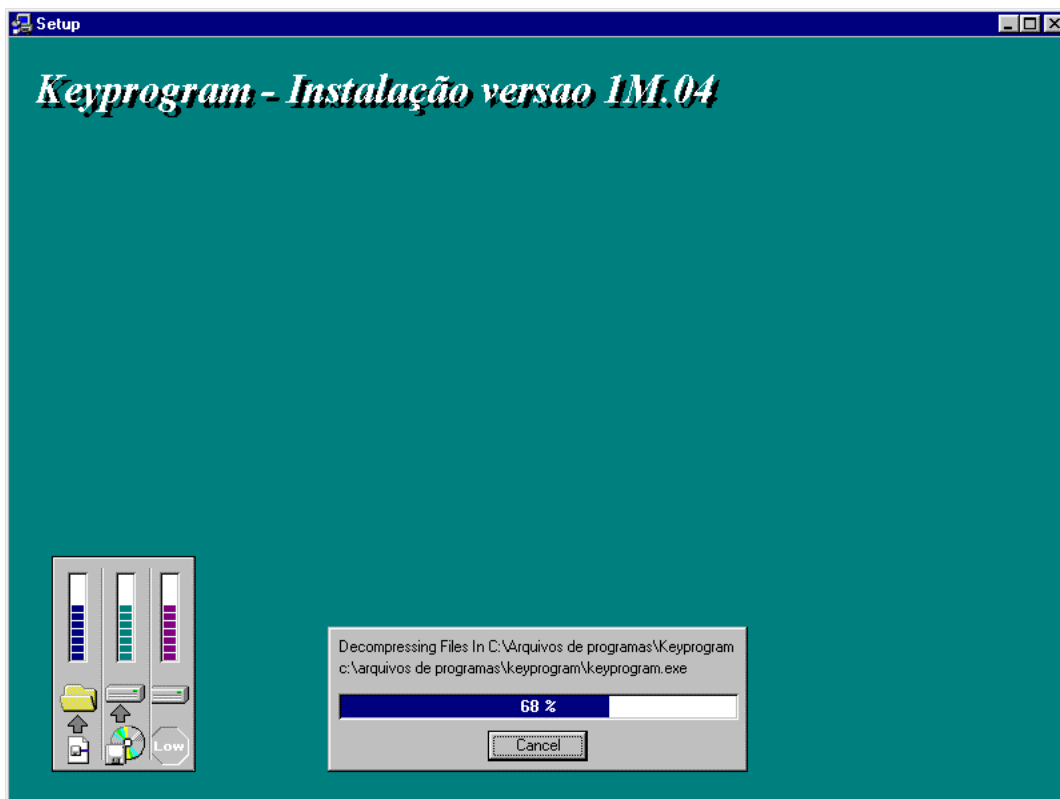


6. Em Destination Directory desta mesma tela (tela acima), é mostrada o diretório padrão onde irá ser instalado o software. Clique em Next se você deseja realmente instalar a pasta Keyprogram no diretório padrão do Windows, ou seja, em C:\Arquivos de programas. Caso queira instalar em um outro diretório,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



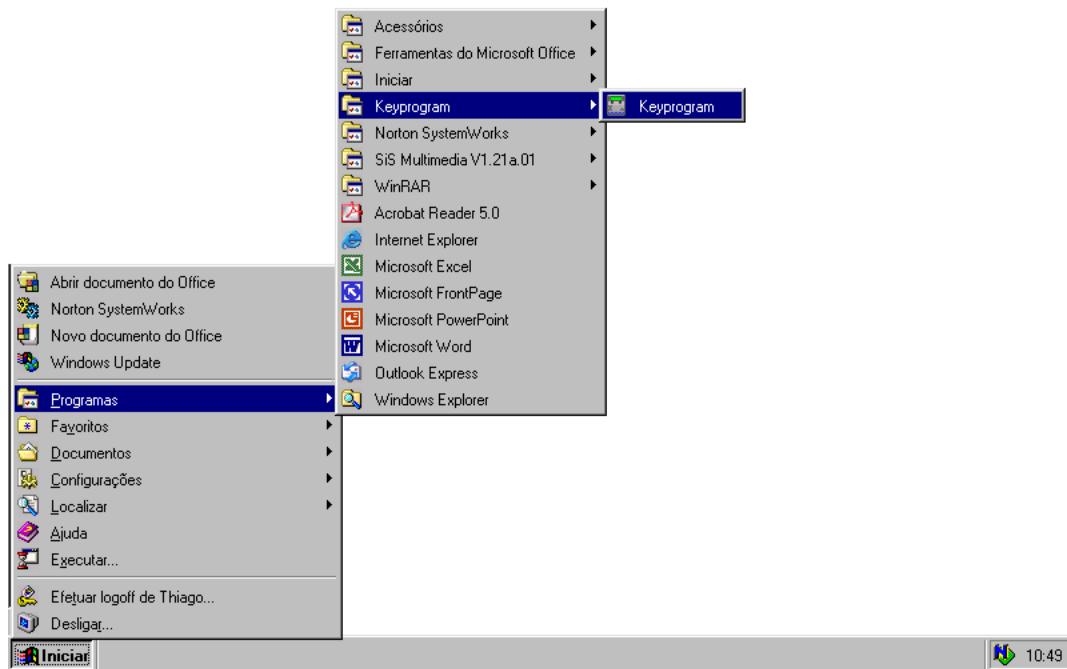
8. Em seguida, o programa irá começar a ser instalado indicando assim uma tela de progresso de instalação, como é apresentada abaixo:



5.3. Iniciando o Keyprogram

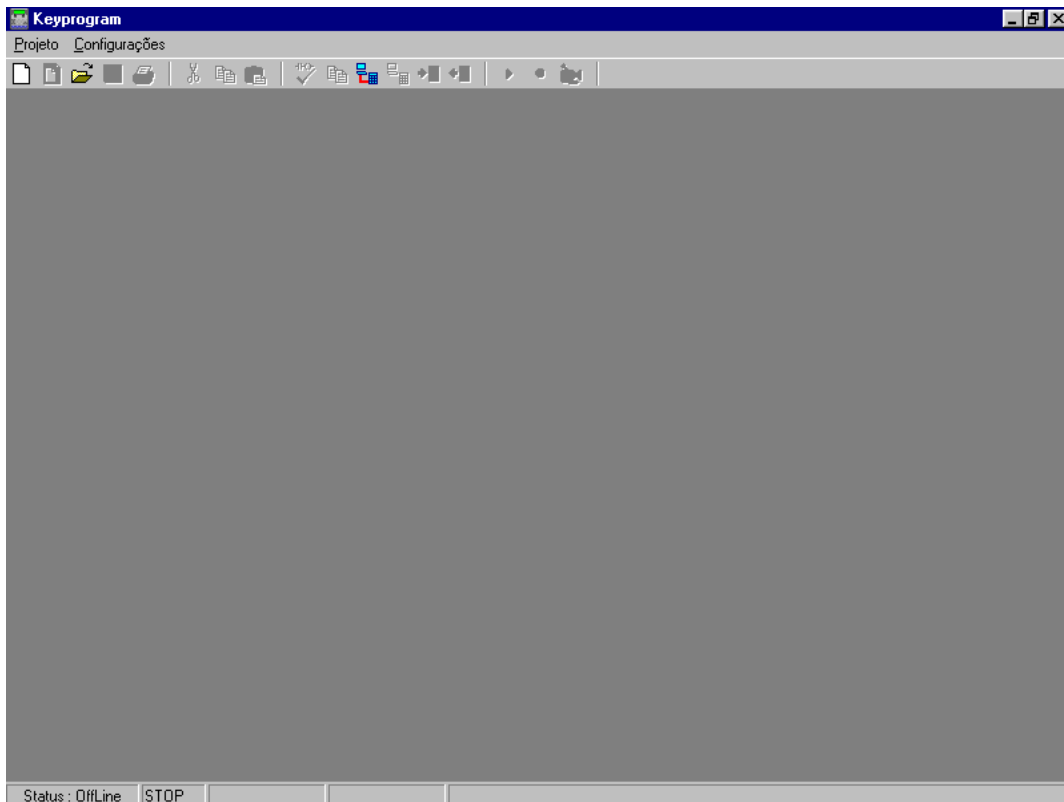
Para entrar no Keyprogram, já instalado corretamente em seu computador, execute os seguintes passos:

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



5.4. Tela Inicial

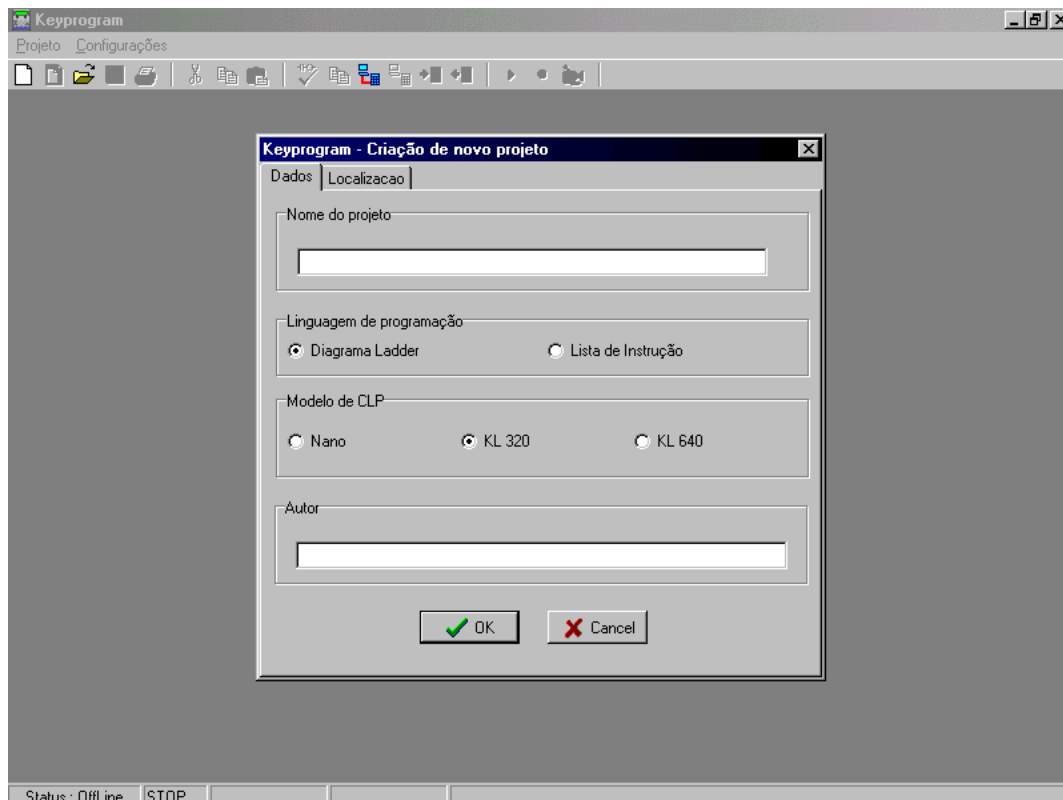
Esta é a tela executada logo na inicialização do Keyprogram:



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.

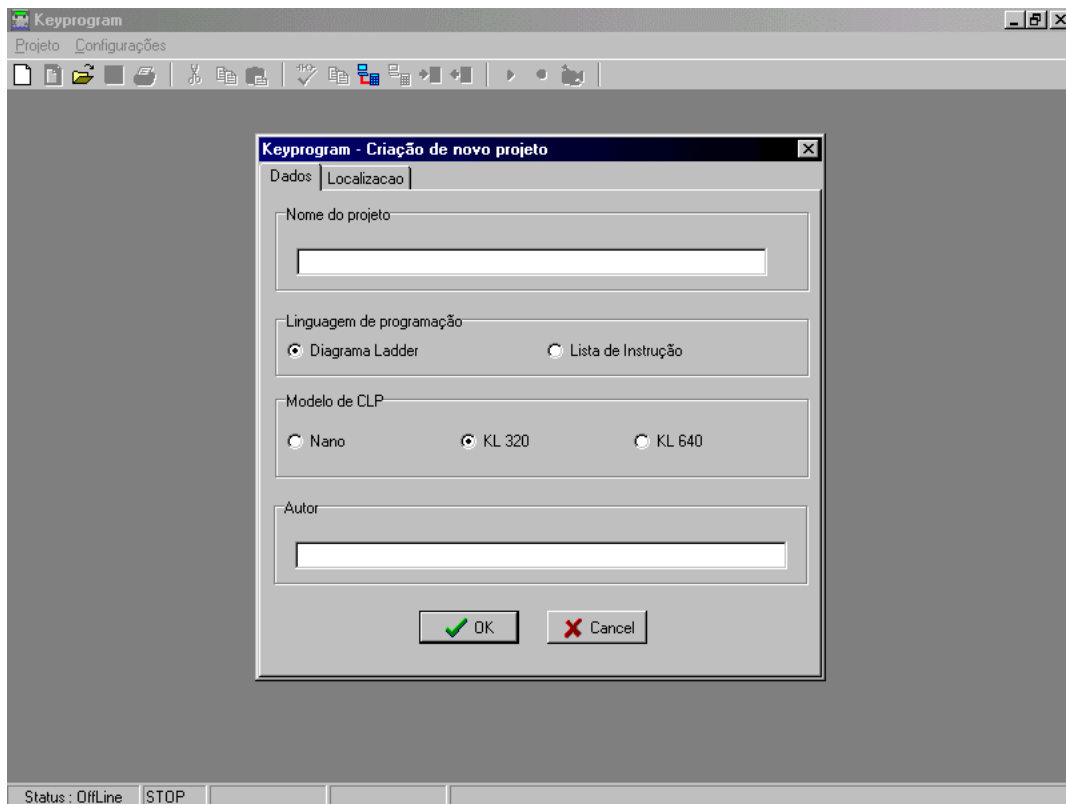
5.4.2. Criando um Projeto

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 KL640, 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

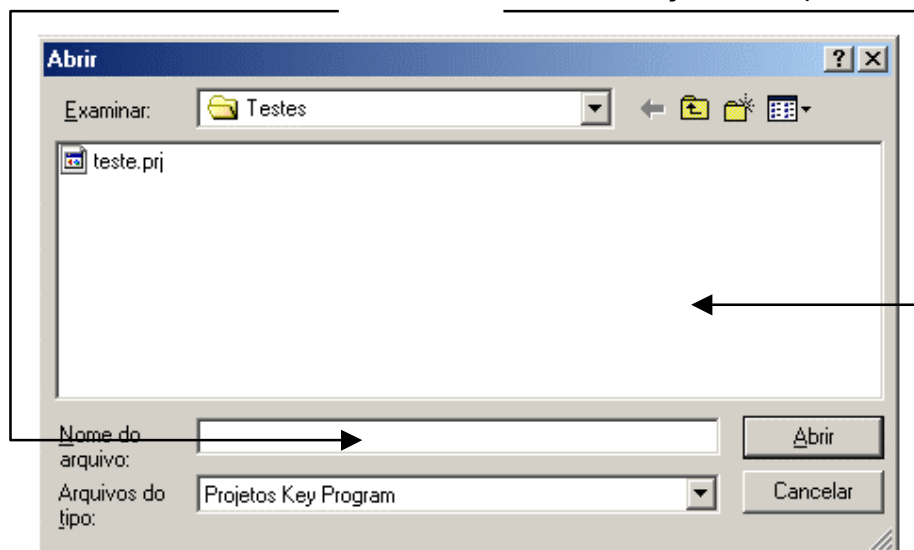
Para se modificar um projeto já existente é necessário que ele seja aberto no Keyprogram.

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 selecionar 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 .

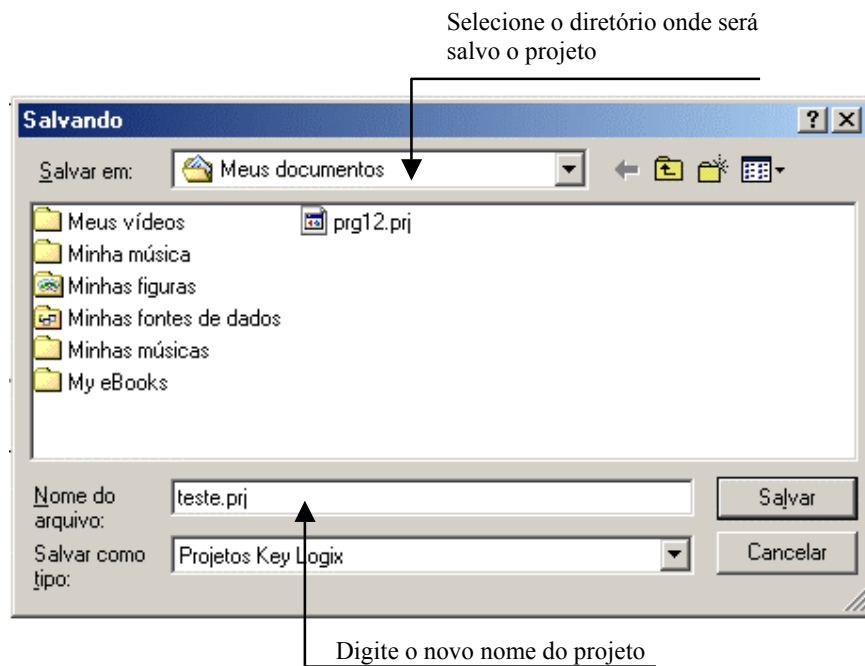
✓ **Salvando um projeto com outro Nome:**

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.

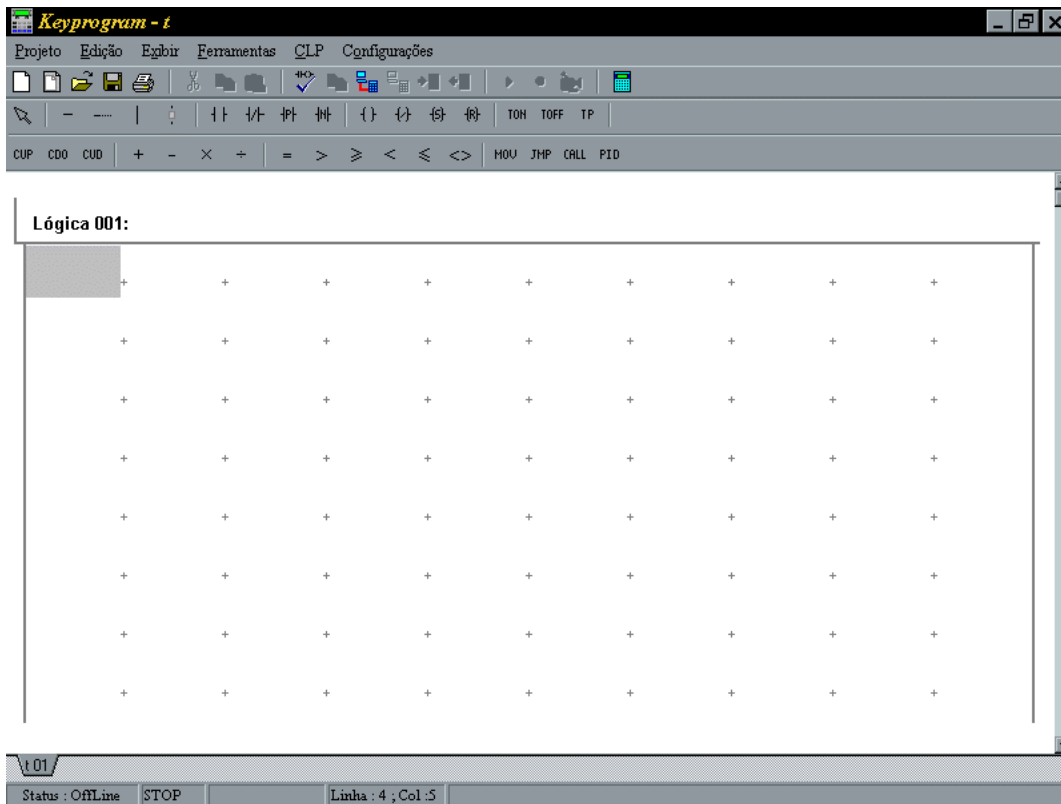
Para fechar um projeto:

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.

Esta tela é disposta de 8 linhas e 8 colunas por lógica, onde o programador irá inserir seus contatos, bobinas, etc. nesta área, **respeitando a quantidade do número de linhas e colunas por 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

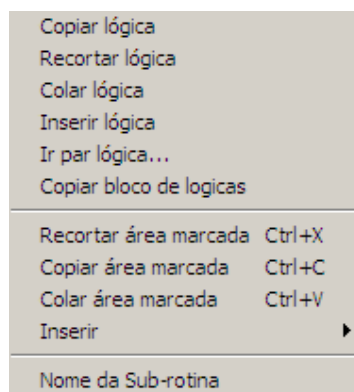
Exportar Sub-rotina: Exporta uma sub-rotina para ser utilizada em um outro projeto.

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 em 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

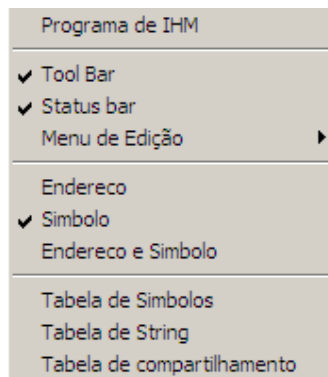


Diagrama Ladder: Este item do menu exibir só aparece quando o Keyprogram estiver na parte de programação da IHM. Ele passa o Keyprogram para a programação em Diagrama Ladder.

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.

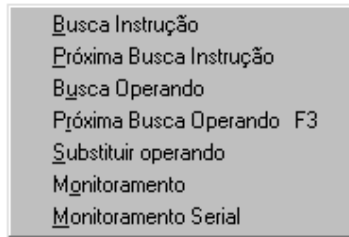
Endereço e Símbolo: Os operandos dos componentes do diagrama ladder são mostrados com seu símbolo e endereço simultaneamente.

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: 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.

Monitoramento Serial: Exibe uma janela para se monitorar todos os endereços e valores que estão sendo enviados e recebidos através da interface serial.

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	
Modelo do CLP	

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.

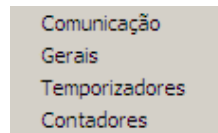
Apaga memória: Apaga a memória do CLP. O projeto que estiver no CLP (lógica, telas, tabela de strings são apagadas).

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.

Modelo do CLP: Exibe uma caixa de texto contendo o modelo do equipamento que está sendo utilizado.

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;


 **Monitorar:** Entra na tela de monitoramento;


 **Tabela de Strings:** Exibe a janela da Lista de Strings;

 **Tabela de Símbolos:** Exibe a janela da Tabela de Símbolos;

 **Tabela de Compartilhamento:** Exibe a janela de Compartilhamento;

 **Receitas:** Exibe a janela de Receitas;

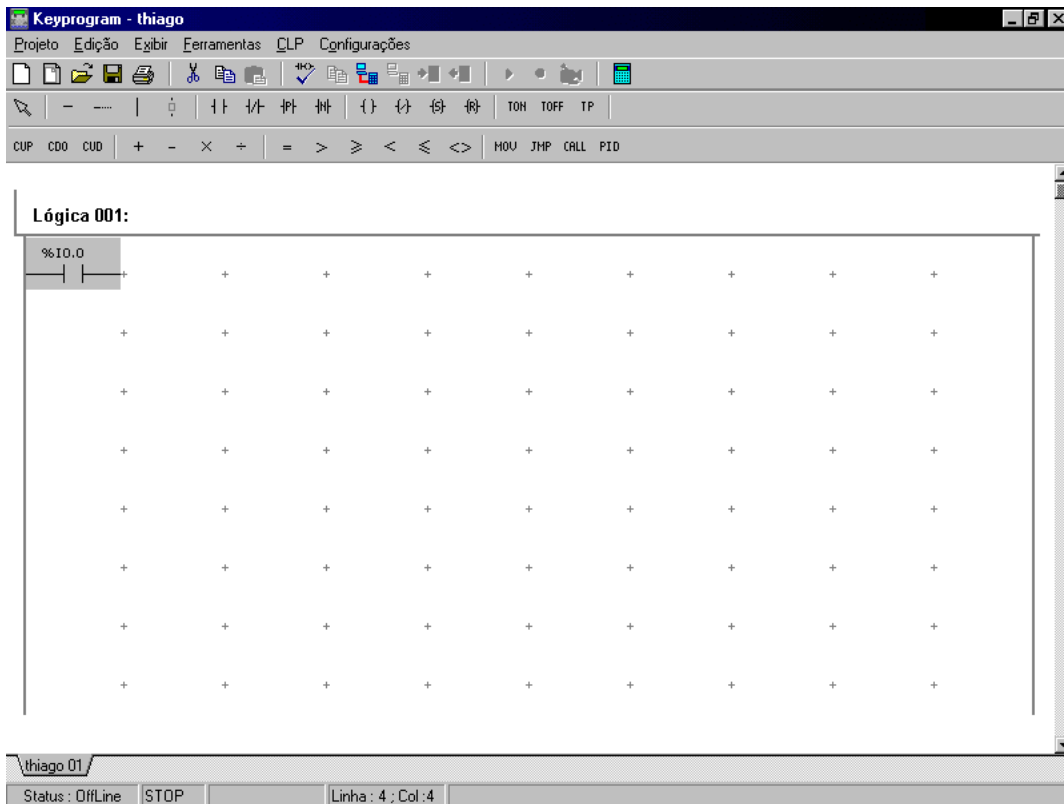
 **Editor Ladder:** Chama o editor de lógica ladder caso o usuário esteja no editor de IHM;

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

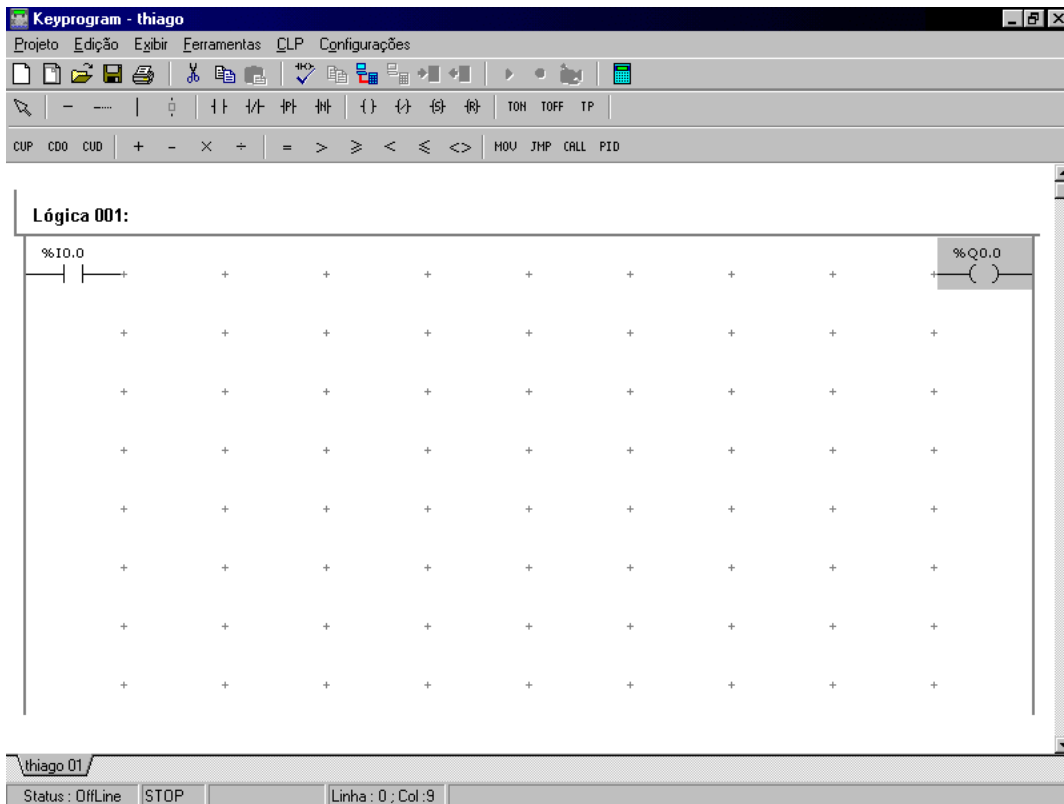
5.6. Criando e executando uma Lógica

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

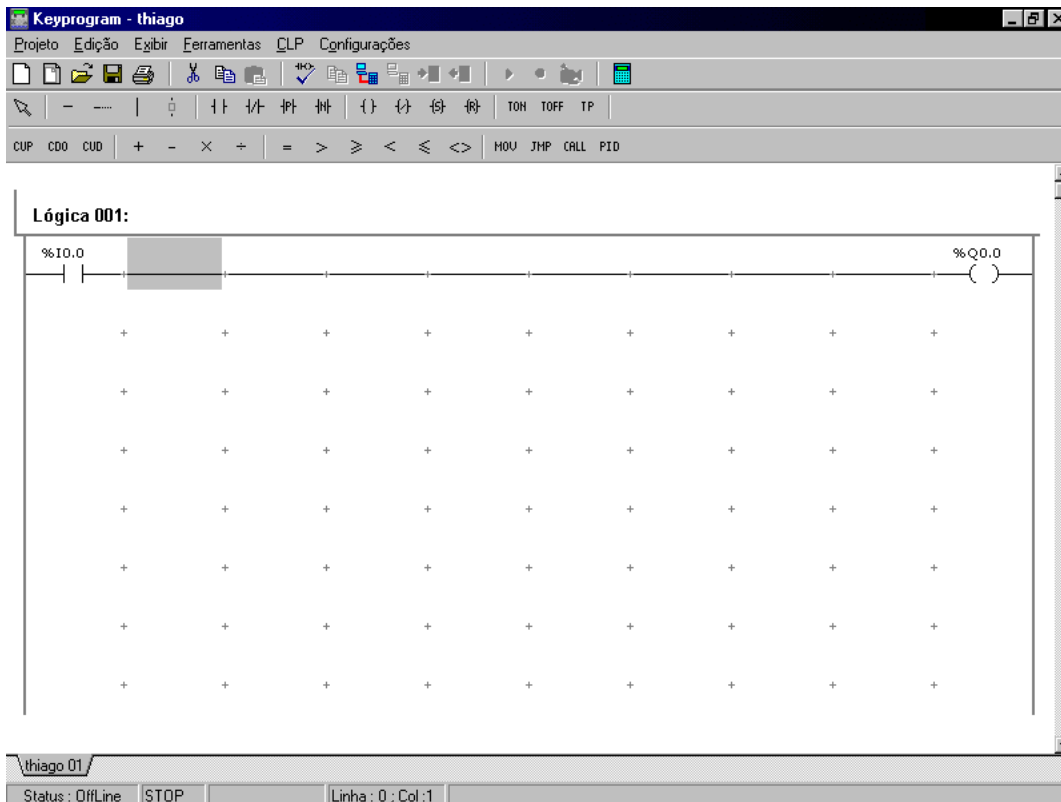
1. Clique no contato Normalmente Aberto (NA) para ser inserido na área de edição. Note que o contato ficará preso no cursor do mouse até que seja clicado novamente em algum ponto da Área de Edição do Keyprogram. Clique na área onde esteja situada a primeira linha e primeira coluna da área de Edição;
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:



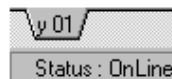
5. Note que agora deve ser fechado o circuito de forma que funcione o programa que estamos propondo. Para isso devemos usar ícone: completa traço horizontal, que fechará o circuito proposto fazendo com que a saída simples responda da seguinte forma: quando o contato NA (%I0.0) estiver aberto, a saída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

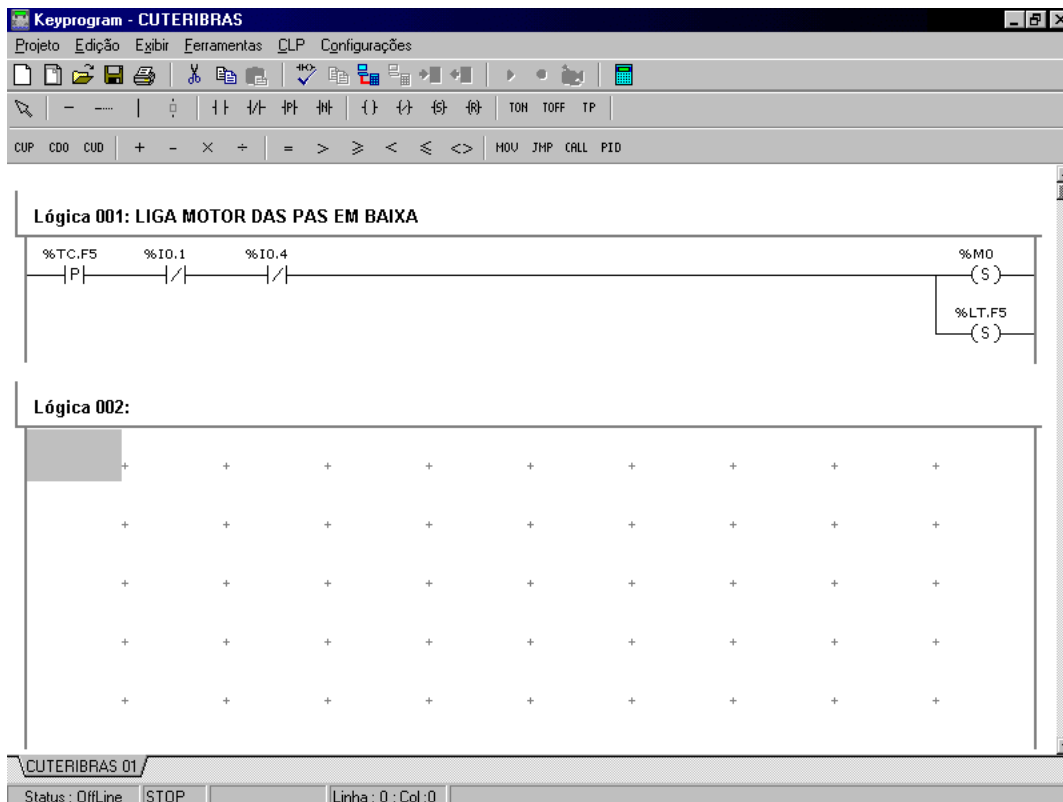
Para facilitar a utilização, o Keyprogram reúne em um único software, duas telas de programação:

- ✓ **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.

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)**

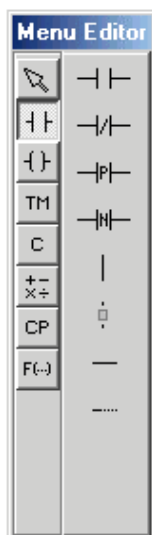
Menu de Edição





✓ **Grupo Geral**



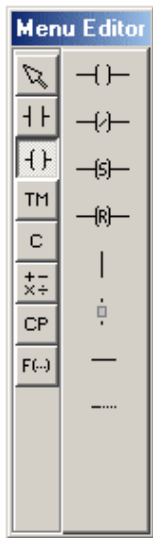
-  **Seta:** deixa o cursor em seu modo default, que é usado para selecionar os elementos da lógica.
-  **Traço vertical:** Insere um traço vertical na lógica entre dois elementos.
-  **Apaga traço vertical:** Ao posicionar o cursor sobre traço vertical e clicar o traço é apagado.
-  **Traço horizontal:**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



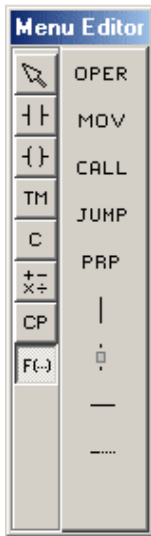
-  **Bloco Somador:** Insere um Bloco Somador.
-  **Bloco Subtrator:** Insere um Bloco Subtrator.
-  **Bloco Divisor:** Insere um Bloco Divisor.
-  **Bloco Multiplicador:** Insere um Bloco Multiplicador.






✓ *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.

✓ **Grupo de Funções Especiais**

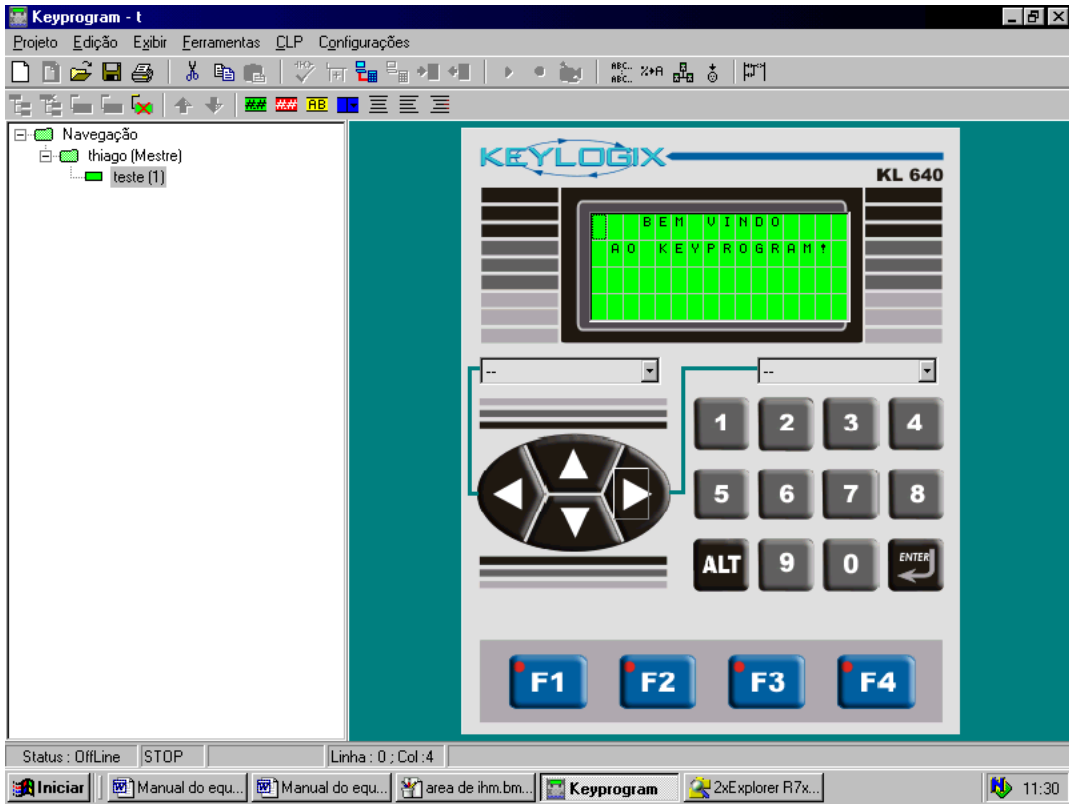


-  Operate: Insere um bloco Operate.
-  Movimentador: Insere um bloco Movimentador.
-  Call: Insere um bloco Call.
-  Jump: Salta para alguma parte do programa.
-  PRP: Insere um bloco Proporcional.

5.7.2. Área de Trabalho da IHM

Para entrar na área de programação da IHM 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 Ladder.

É mostrada abaixo, a Área de Programação Ladder:



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

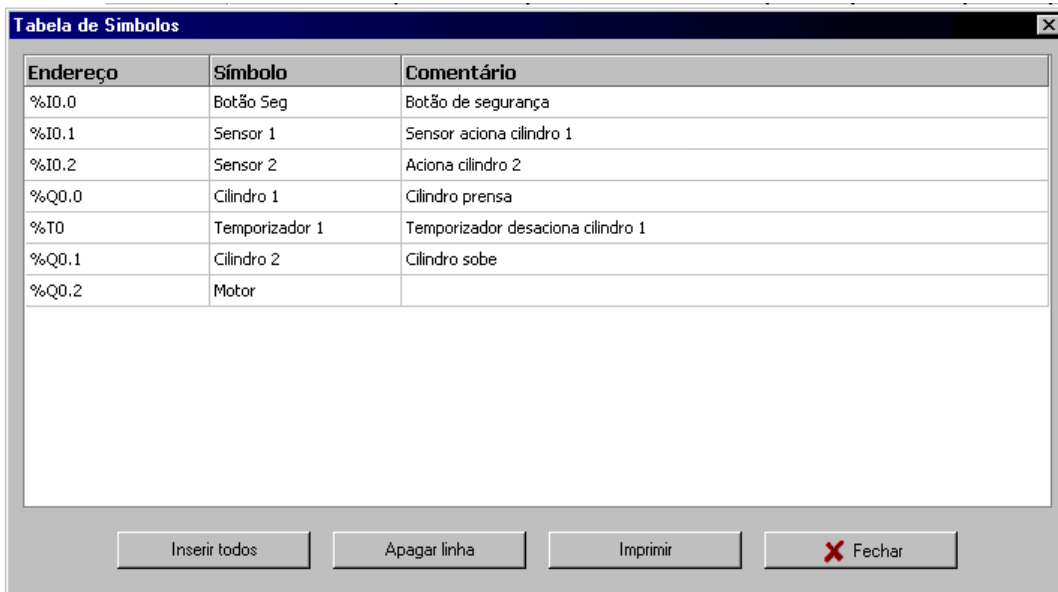
Na barra de tarefas da IHM estão as principais funções que você irá utilizar na programação da IHM.



Ícone	Função	Descrição
	Novo grupo de páginas	Cria um novo grupo de páginas.
	Elimina grupo de páginas	Esta função elimina o grupo de página selecionado.
	Nova página	Esta função cria uma nova página dentro do grupo selecionado.
	Nova página de Bargraph	Esta função cria uma nova página de Bargraph dentro do grupo selecionado. A tela de Bargraph mostra em um gráfico o valor de uma determinada variável.
	Elimina página	Esta função exclui a pagina selecionada.
	Sobe grupo/página	Esta função inverte a ordem do grupo/pagina, subindo o grupo/página selecionado.
	Desce grupo/página	Esta função inverte a ordem do grupo/pagina, descendo o grupo/página selecionado.
	Proteção da página	Coloca uma senha na página.
	Campo de Edição	Permite alterar o valor de uma determinada variável através do teclado da IHM.
	Campo de visualização	Esta função mostra na tela da IHM o valor de um operando do programa.
	Campo de String	Esta função mostra um String na tela da IHM.
	Campo Seletor de String	Esta função permite ao usuário escolher uma string, entre as pré-selecionadas pelo programador.
	Centralizar Texto	Esta função centraliza o texto da IHM
	Alinhar Texto à esquerda	Esta função alinha à esquerda o texto da IHM.
	Alinhar Texto à direita	Esta função alinha à direita o texto da IHM.

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.

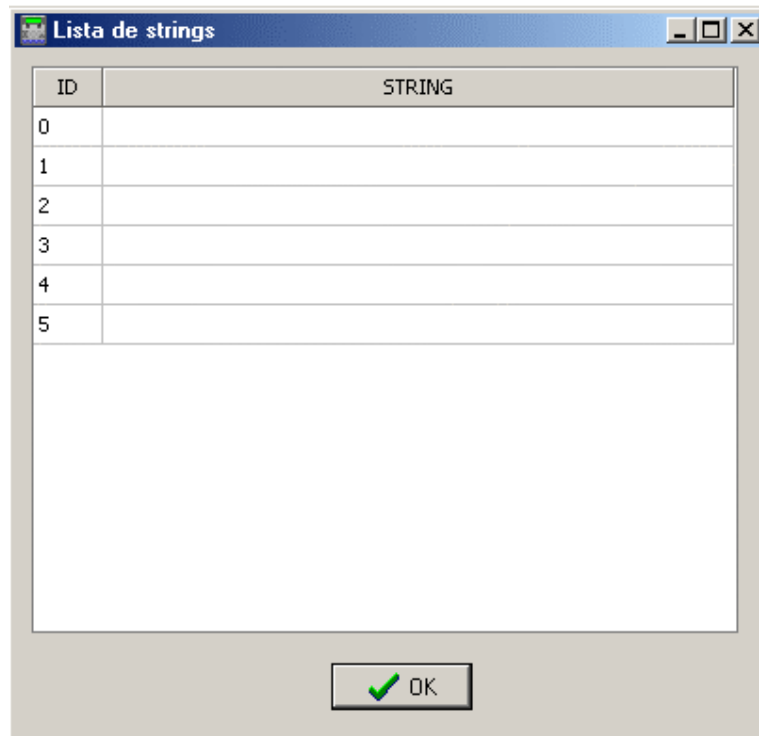
Inserindo todos os operandos do programa: essa função insere na Tabela de Símbolos todos os operandos do programa que ainda não estiverem nela. Clique no botão Inserir todas.

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. Lista de String

A tabela Lista de String é onde o usuário lista as Strings que irá usar no projeto.



ID	STRING
0	
1	
2	
3	
4	
5	

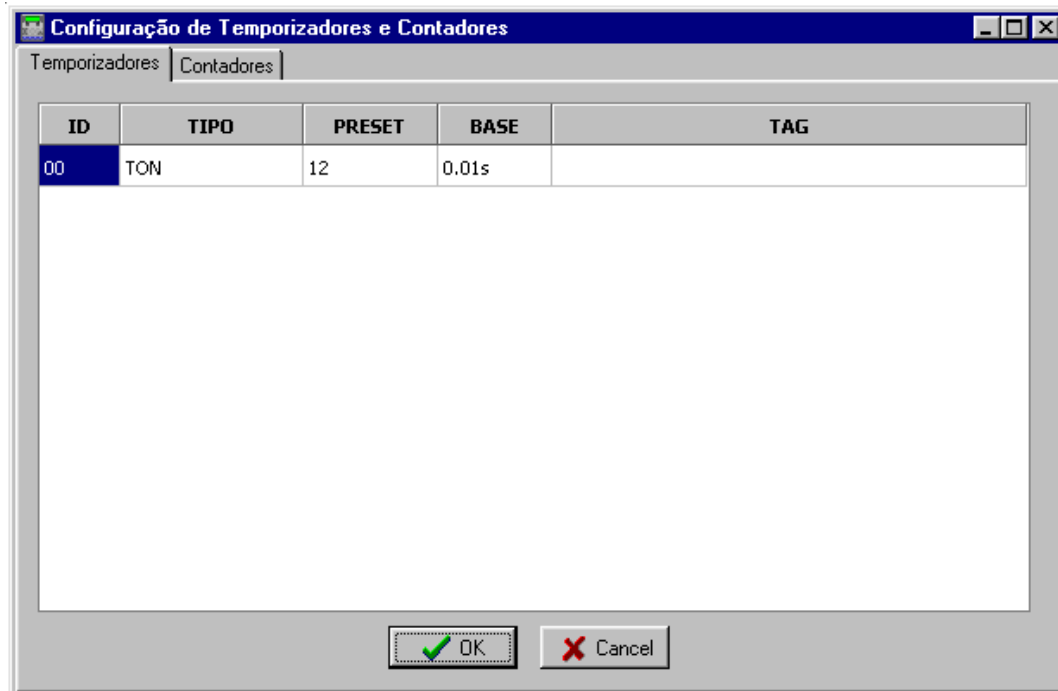
OK

ID: Número de identificação da String.

String: Strings que serão editadas pelo usuário.

5.10. Configuração dos Temporizadores

É nessa janela que são configurados os parâmetros dos Temporizadores.



ID: Número identificador do contador / temporizador.

Tipo: Tipo do contador / temporizador.

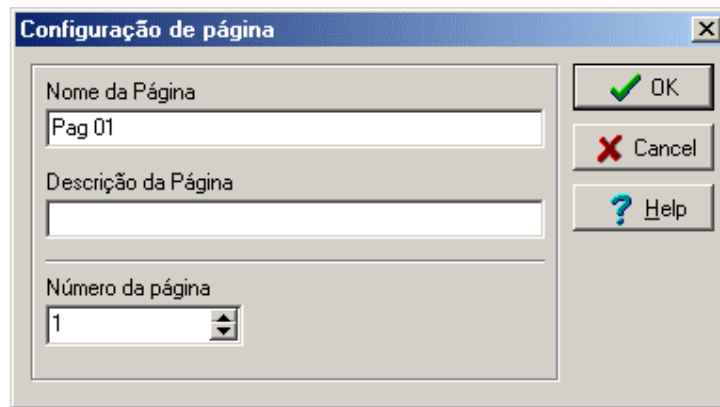
Preset: Valor de preset do contador / temporizador.

Base: Base de tempo de contador / temporizador.

Tag: Descrição do contador / temporizador.

5.11. Janela de Configurações das Páginas

Esta é a janela de configuração das páginas da IHM.

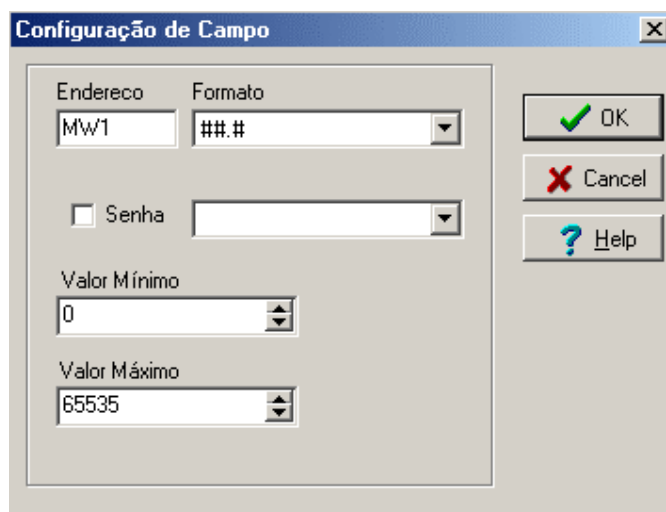


The image shows a dialog box titled "Configuração de página" with a close button (X) in the top right corner. The dialog contains three input fields and three buttons. The first field is labeled "Nome da Página" and contains the text "Pag 01". The second field is labeled "Descrição da Página" and is empty. The third field is labeled "Número da página" and contains the number "1". To the right of the input fields are three buttons: "OK" with a green checkmark icon, "Cancel" with a red X icon, and "Help" with a blue question mark icon.

5.12. Campos

5.12.1. Campo de Edição

Esta é a janela de configuração do campo edição.



Endereço: Endereço da variável (byte ou Word).

Formato da Variável: Formato da variável a ser impresso na tela.

Senha: Indica se a o campo será ou não protegido por senha.

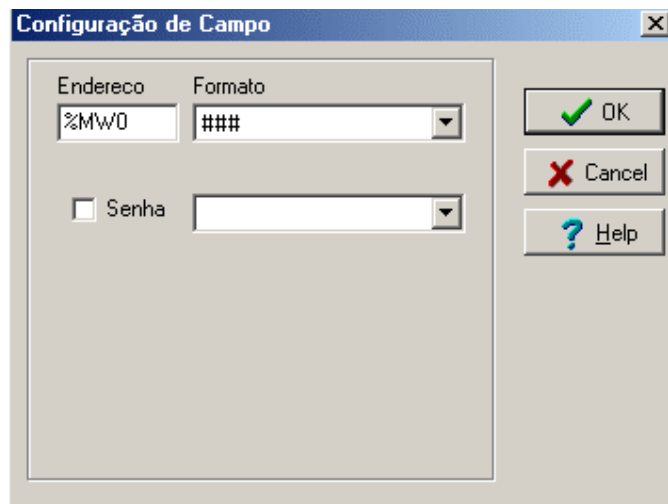
Nível da Senha: Define o nível da senha que será protegido.

Valor Mínimo: Valor mínimo de edição da variável.

Valor Máximo: Valor máximo de edição da variável.

5.12.2. Campo de Visualização

Esta é a janela de configuração do campo visualização



Endereço: Endereço da variável (byte ou Word).

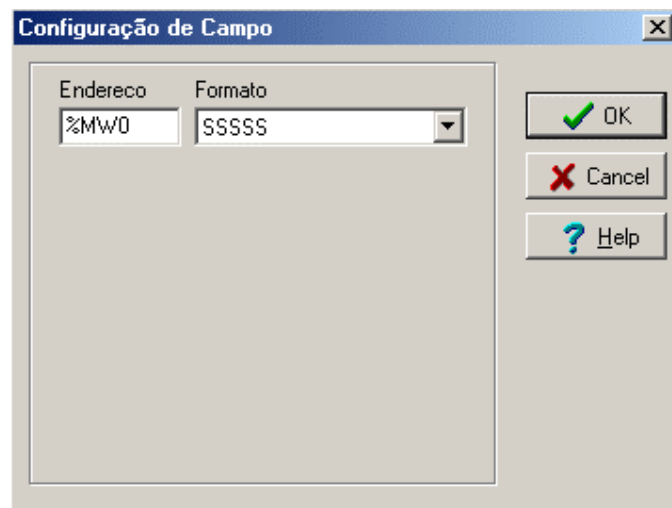
Formato da Variável: Formato da variável a ser impresso na tela.

Senha: Indica se a o campo será ou não protegido por senha.

Nível da Senha: Define o nível da senha que será protegido.

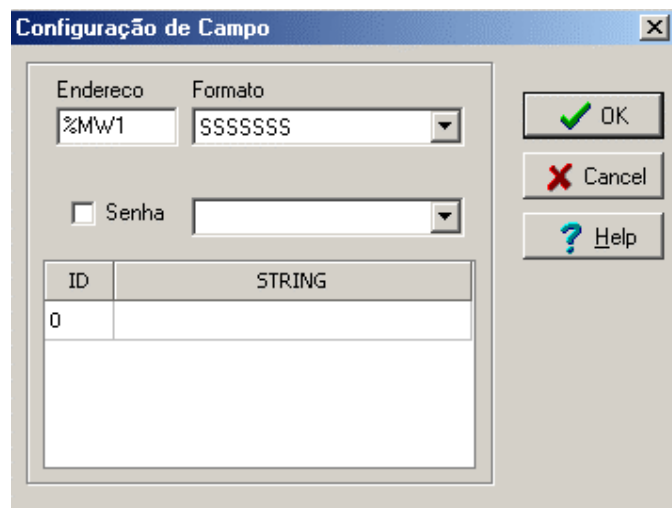
5.12.3. Campo String

Esta é a janela de configuração do campo string.



5.12.4. Campo Seletor

Esta é a janela de configuração do campo Seletor.




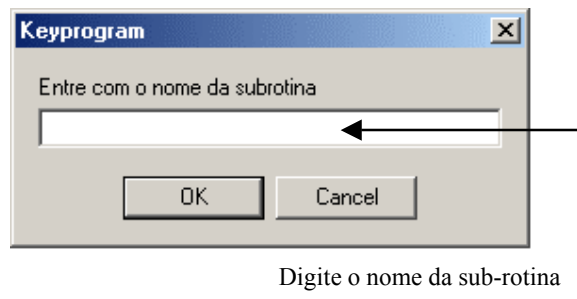
5.13. Sub-rotinas

✓ 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**.

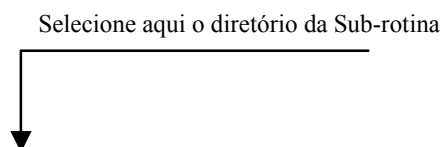


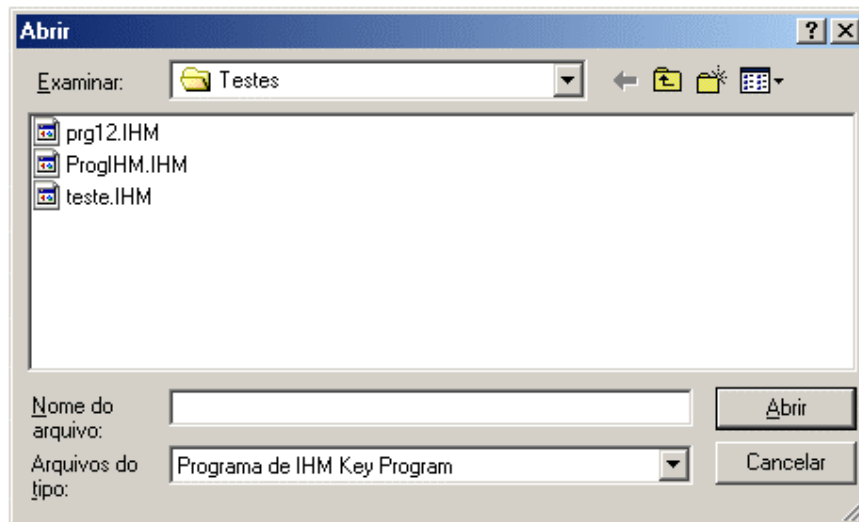
✓ Importando Sub-rotina

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**.



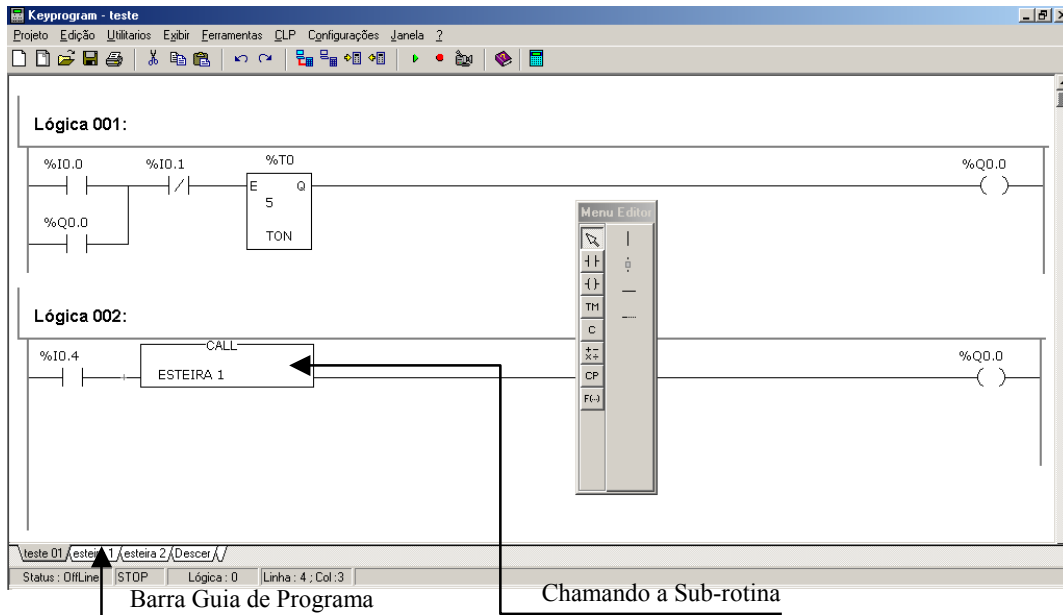


A nova sub-rotina irá aparecer na barra Guia de Programa.

✓ Editando Sub-rotinas

Para editar a Sub-rotina:

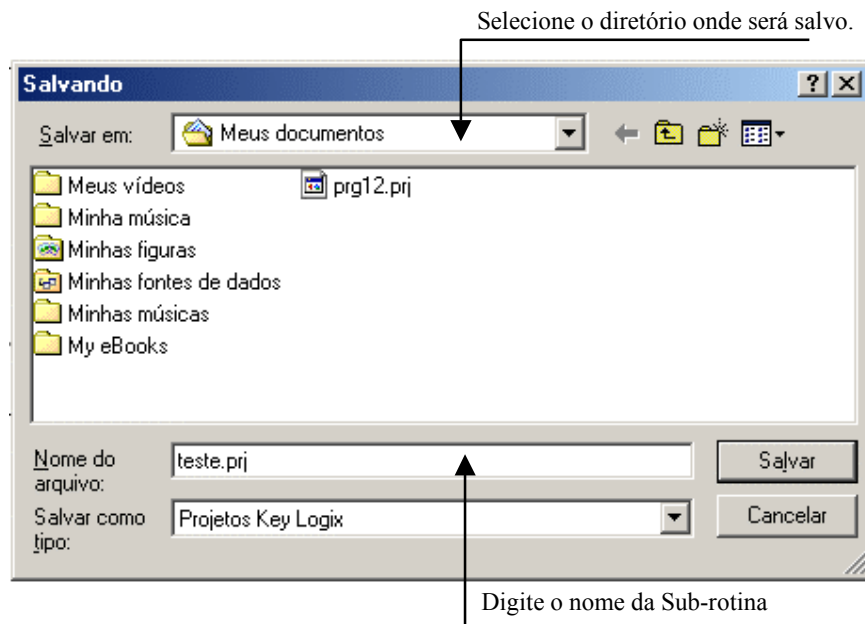
1. Clique sobre a **Sub-rotina** na barra **Guia de Programa**.
2. Para chamar a **Sub-rotina** use um operando do tipo **CALL** do grupo funções especiais.



✓ 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**.



5.14. 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.

5.15. Instruções Válidas para o KL640

✓ 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
- ✓ Temporizador na desenergização
- ✓ Temporizador de pulso

✓ Contadores

- ✓ Contador UP
- ✓ Contador Down
- ✓ Contador Up-Down
- ✓ Contador Rápido

✓ Aritméticos

- ✓ Somador
- ✓ Subtrator
- ✓ Multiplicador
- ✓ Divisor

✓ Comparadores

- ✓ Igual
- ✓ Maior
- ✓ Maior Igual
- ✓ Menor
- ✓ Menor Igual
- ✓ Diferente

✓ Especiais

- ✓ Move
- ✓ Jump
- ✓ Call

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.