

AL-1000/AL-500
Manual de Programação

Ref. 6399 - 000 Rev. C 07/92



Condições Gerais de Fornecimento

Todos os Direitos Reservados

Nenhuma parte deste documento pode ser copiado ou reproduzido de alguma forma sem o consentimento prévio e por escrito da ALTUS Sistemas de Informática Ltda., que reserva-se o direito de efetuar alterações sem prévio comunicado.

Segurança de Pessoas e Instalações

Conforme legislação vigente, do **Código de Defesa do Consumidor**, informamos os seguintes aspectos relacionados com a segurança de pessoas e instalações do cliente:

Os equipamentos de Automação Industrial, fabricados pela ALTUS, são robustos e confiáveis devido ao rígido controle de qualidade a que são submetidos. No entanto, equipamentos eletrônicos de controle industrial (Controladores Programáveis, Comandos Numéricos, etc) podem causar danos, às máquinas ou processos por eles controlados, no caso de defeito em suas partes e peças, erros de programação ou instalação, podendo inclusive colocar em risco vidas humanas.

O usuário deve analisar as possíveis consequências destes defeitos e providenciar instalações adicionais externas de segurança que, em caso de necessidade, atuem no sentido de preservar a segurança do sistema, principalmente nos casos de testes e da instalação inicial.

É imprescindível a leitura completa dos Manuais e/ou Características Técnicas do produto, antes da instalação ou utilização do mesmo.

Garantia

A ALTUS garante os seus equipamentos contra defeitos reais de fabricação pelo prazo de doze meses a partir da data da emissão da nota fiscal.

Esta garantia é dada em termos de manutenção de fábrica, ou seja, o transporte de envio e retorno do equipamento até a fábrica da ALTUS, em Porto Alegre, correrá por conta do cliente.

A garantia será automaticamente suspensa caso sejam introduzidas modificações nos equipamentos por pessoal não autorizado pela ALTUS. A ALTUS exime-se de quaisquer ônus referentes a reparos ou substituições, em virtude de falhas provocadas por agentes externos aos equipamentos, pelo uso indevido dos mesmos, bem como resultantes de caso fortuito ou força maior.

A ALTUS garante que os seus equipamentos funcionam de acordo com as descrições contidas explicitamente em seus Manuais e/ou Características Técnicas, não garantindo a satisfação de algum tipo particular de aplicação dos equipamentos.

A ALTUS desconsiderará qualquer outra garantia, direta ou implícita, principalmente quando se tratar de fornecimento de terceiros.

Pedidos de informações adicionais sobre o fornecimento e/ou características dos equipamentos e serviços ALTUS devem ser feitos por escrito. A ALTUS não se responsabiliza por informações fornecidas sobre seus equipamentos sem registro formal.

SUMÁRIO

1- Introdução à Programação	1.1
Introdução	1.1
Linguagem de Programação	1.3
Operandos	1.3
Relés (R)	1.3
Auxiliares (A)	1.4
Memórias (M)	1.4
Constante (K)	1.5
Lógica	1.5
Ciclo de Varredura	1.6
Configuração	1.9
2- Instruções Básicas	2.1
Introdução	2.1
Bobina	2.2
Simbologia no Diagrama de Relés	2.2
Descrição	2.2
Contato Normalmente Aberto	2.3
Simbologia no Diagrama de Relés	2.3
Descrição	2.3
Contato Normalmente Fechado	2.4
Simbologia no Diagrama de Relés	2.4
Descrição	2.4
Ligação Horizontal e Vertical	2.5
Simbologia no Diagrama de Relés	2.5
Descrição	2.5
Bobina Liga/Bobina Desliga	2.6
Simbologia no Diagrama de Relés	2.6
Descrição	2.6
Relé Mestre/Fim de Relé Mestre	2.7
Simbologia no Diagrama de Relés	2.7
Descrição	2.7
Bobina de Salto	2.8
Simbologia no Diagrama de Relés	2.8
Descrição	2.8
Relé de Pulso	2.9

Simbologia no Diagrama de Relés	2.9
Descrição	2.9
Temporizador	2.10
Simbologia no Diagrama de Relés	2.10
Descrição	2.10
Contador	2.12
Simbologia no Diagrama de Relés	2.12
Descrição	2.12
Soma	2.14
Simbologia no Diagrama de Relés	2.14
Descrição	2.14
Subtração	2.15
Simbologia no Diagrama de Relés	2.15
Descrição	2.15
Multiplicação	2.16
Simbologia no Diagrama de Relés	2.16
Descrição	2.16
Divisão	2.17
Simbologia no Diagrama de Relés	2.17
Descrição	2.17
Conversão D/B e B/D	2.18
Simbologia no Diagrama de Relés	2.18
Conversão Decimal para Binário	2.18
Conversão Binário para Decimal	2.18
Descrição	2.18
Movimentação	2.21
Simbologia no Diagrama de Relés	2.21
Descrição	2.21
Movimentador de Imagem	2.23
Simbologia no Diagrama de Relés	2.23
Descrição	2.23

3- Instruções Estendidas 3.1

Introdução	3.1
Instruções para Tratamento de Tabelas	3.2
Carrega Tabela	3.2
Simbologia no Diagrama de Relés	3.2
Descrição	3.2
Carrega Tabela 00	3.4
Carrega Tabela 01 a 63	3.4
Movimentador de Tabela	3.4

Simbologia no Diagrama de Relés	3.4
Descrição	3.5
Leitura de Conteúdo	3.5
Escrita de Conteúdo	3.5
Sequenciador	3.7
Simbologia no Diagrama de Relés	3.7
Descrição	3.7
Contador Bidirecional	3.10
Simbologia no Diagrama de Relés	3.10
Descrição	3.10
Conversão Analógico / Digital	3.11
Simbologia no Diagrama de Relés	3.11
Descrição	3.11
Conversão Digital / Analógica	3.13
Simbologia no Diagrama de Relés	3.13
Descrição	3.13
Impressão	3.14
Impressão de Textos	3.14
Simbologia no Diagrama de Relés	3.14
Impressão dos Conteúdos das Memórias	3.15
Simbologia no Diagrama de Relés	3.15
Descrição	3.16
4- Instruções Especiais	4.1
Introdução	4.1
Instrução Especial 00 - Interrupção de Tempo	4.2
Simbologia no Diagrama de Relés	4.2
Descrição	4.2
Restrições ao Uso da Instrução Especial 00	4.5
Instrução Especial 01 - Norma ISA4A	4.6
Simbologia no Diagrama de Relés	4.6
Descrição	4.6
Instrução Especial 02 -Movimentação Indireta(*)	4.8
Simbologia no Diagrama de Relés	4.8
Descrição	4.8
Tempo de Execução	4.9
Instrução Especial 03 (*)	4.10
Simbologia no Diagrama de Relés	4.10
Teste ou Liga/Desliga Relé ou Auxiliar Indexado	4.10
Descrição	4.10
Teste de Relé ou Auxiliar Indexado	4.10

Liga / Desliga Relé ou Auxiliar Indexado	4.11
Instrução Especial 04 - Recebe Caracter	4.12
Simbologia no Diagrama de Relés	4.12
Descrição	4.12
Instrução Especial 05 - Instrução Pt100	4.14
Simbologia no Diagrama de Relés	4.14
Descrição	4.14
Instrução Especial 06 - Retorno da Interrupção de Tempo	4.16
Simbologia no Diagrama de Relés	4.16
Descrição	4.16
Instrução Especial 07 - Instrução Termopar	4.17
Simbologia no Diagrama de Relés	4.17
Descrição	4.17
Instrução Especial 08 - Leitura de Pontos Analógico Isolados	4.22
Simbologia no Diagramas de Relés	4.22
Descrição	4.22
Instrução Especial 09 - Move Tabela Indexado	4.25
Simbologia no Diagrama de Relés	4.25
Descrição	4.25
Instrução Especial 10 - Movimentação de Blocos (*)	4.27
Simbologia no Diagrama de Relés	4.27
Descrição	4.27
Instrução Especial 11 - Retorno de Subrotina(*)	4.29
Simbologia no Diagrama de Relés	4.29
Descrição	4.29
Instrução Especial 12 - P.I.D.	4.31
Simbologia no Diagrama de Relés	4.31
Descrição	4.31
Instrução Especial 13 - Horário e Calendário	4.38
Simbologia no Diagrama de Relés	4.38
Descrição	4.38
Tratamento de Variáveis BCD	4.40
Generalidades	4.40
Instrução Especial 14 - Transferência/Deslocamento BCD	4.41
Simbologia no Diagrama de Relés	4.41
Descrição	4.41
Instrução Especial 15 - Soma/Subtração/Comparação em 7 Dígitos BCD	4.44
Simbologia no Diagrama de Relés	4.44
Descrição	4.44
Instrução Especial 16 - Conversão BCD-BIN/BIN-BCD	4.47

Simbologia no Diagrama de Relés	4.47
Descrição	4.47
Tratamento de Arquivos	4.48
Generalidades	4.48
Instrução Especial 17 - Move Arquivo (*)	4.50
Simbologia no Diagrama de Relés	4.50
Descrição	4.50
Contador Rápido	4.51
Descrição do Módulo	4.51
Instrução Especial 18 - Contador Rápido	4.53
Simbologia no Diagrama de Relés	4.53
Descrição	4.53
Instrução Especial 19 - Interface de LEDs	4.56
Simbologia no Diagrama de Relés	4.56
Descrição	4.56
5- Informações Adicionais	5.1
Área de Memória Ocupada pelo Programa do Usuário	5.1
Tempo de Execução das Instruções	5.5

Lista de Figuras

Figura 1 - Exemplo de Diagrama Elétrico	1.2
Figura 2 - Exemplo de Programa de CP	1.2
Figura 3 - Lógica de Programação	1.5
Figura 4 - Fluxo de Energia	1.6
Figura 5 - Ciclo de Varredura	1.7
Figura 6 - Varredura em uma Lógica	1.7
Figura 7 - Ciclo de Execução de um Programa	1.8
Figura 8 - Tela de Configuração do TP AL-3800	1.10
Figura 9 - Lógica com Bobina e Contatos NA	2.2
Figura 10 - Lógica com Contatos NF	2.4
Figura 11 - Funcionamento das Instruções de Conversão BCD	2.19
Figura 12 - Listagem da Instrução CT	3.3
Figura 13 - Funcionamento da Instrução Seqüenciador	3.8
Figura 14 - Sequência de Alarme ISA4A	4.6
Figura 15 - Utilização da Instrução ESP 02	4.9
Figura 16 - Diagrama de Lógica da Instrução PID	4.32
Figura 17 - Movimentação de Dígitos	4.41
Figura 18 - Deslocamento para a Esquerda	4.42
Figura 19 - Deslocamento para a Direita	4.42
Figura 20 - Deslocamento p/ Esquerda Movendo Dígito	4.43
Figura 21 - Deslocamento p/ Direita Movendo Dígito	4.43
Figura 22 - Soma BCD com 7 Dígitos e Sinal	4.45
Figura 23 - Comparação BCD	4.46
Figura 24 - Declaração de Arquivos	4.49
Figura 25 - Diagrama de Blocos do Módulo Contador Rápido	4.52

Lista de Tabelas

Tabela 1 - Comportamento da Instrução Temporizador	2.10
Tabela 2 - Comportamento da Instrução Contador	2.13
Tabela 3 - Exemplos de Aplicação da Instrução MOV	2.22
Tabela 4 - Exemplos de Aplicação da Instrução MOI	2.24
Tabela 5 - Caracteres de Controle	3.15
Tabela 6 - Freqüências de Interrupção	4.3
Tabela 7 - Baud Rates	4.12
Tabela 8 - Linearização da Instrução PT100	4.15
Tabela 9 - Refer. Entrada Analógica/Valores Instr. Termopar	4.20

Tabela 10 - Rel. Dados de Engenharia/Normalizados dos Termop. . . .	4.20
Tabela 11 - Parâmetros da Instrução PID	4.33
Tabela 12 - Parametrização da Instrução PID	4.37
Tabela 13 - Organização da Tabela da Instrução Relógio	4.38
Tabela 14 - Instrução de Contatos	5.2
Tabela 15 - Instruções Básicas de Blocos	5.3
Tabela 16 - Instruções Estendidas e Especiais	5.3
Tabela 17 - Tempos de Execução das Instruções Estendidas	5.5
Tabela 18 - Tempos de Execução das Instruções Básicas	5.6
Tabela 19 - Tempos de Execução das Instruções Especiais	5.7

Prefácio

Introdução

Este manual apresenta uma descrição geral das instruções dos Controladores Programáveis e substitui o capítulo referente à programação, que era parte integrante dos manuais dos Terminais de Programação ALTUS.

Organização Geral do Manual

O capítulo 1 apresenta a linguagem de programação dos controladores programáveis AL-1000 e AL-500. Este, inclui formato de programação, dados de operandos, ciclos de varreduras e configuração.

O Capítulo 2 descreve o conjunto básico de instruções presente em todos os Controladores Programáveis. Este, inclui instruções tais como: contatos, bobinas, operações aritméticas, temporizadores, contadores, movimentadores, entre outras.

O Capítulo 3 é dedicado a descrição de instruções para tratamento de tabelas, seqüenciamentos, conversão A/D e D/A e impressão de textos e conteúdos de memórias, que constituem o conjunto estendido de instruções.

No Capítulo 4 são descritas, as instruções para interfaceamento com os módulos especiais ALTUS, como Contador Rápido e Multiplexador de LEDs, bem como

aquelas que envolvem movimentação de blocos de operandos, tratamento de arquivos e utilização de sub-rotinas.

O Capítulo 5 apresenta os tempos de execução e a área de memória ocupada pelas instruções e pelo programa.

Capítulo 1

Introdução à Programação

1.1. Introdução

A linguagem de programação dos Controladores Programáveis ALTUS (Linguagem de Relés) se caracteriza pela facilidade no aprendizado por indivíduos que não tenham conhecimento em programação, mas tenham conhecimento em interpretação de diagramas elétricos.

As figuras 1 e 2 dão uma idéia de semelhança entre um esquema elétrico e um programa de CP.

As séries AL-500 e AL-1000 são totalmente compatíveis a nível de software de usuário. O fato de existir esta compatibilidade entre os dois modelos permite que o usuário seja treinado apenas uma vez e sejam usados os mesmos terminais de programação.

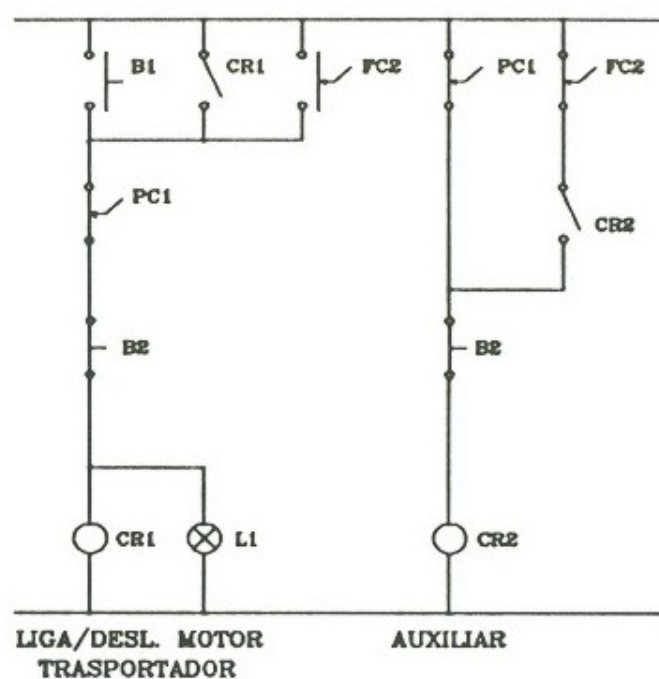


Figura 1 - Exemplo de Diagrama Elétrico

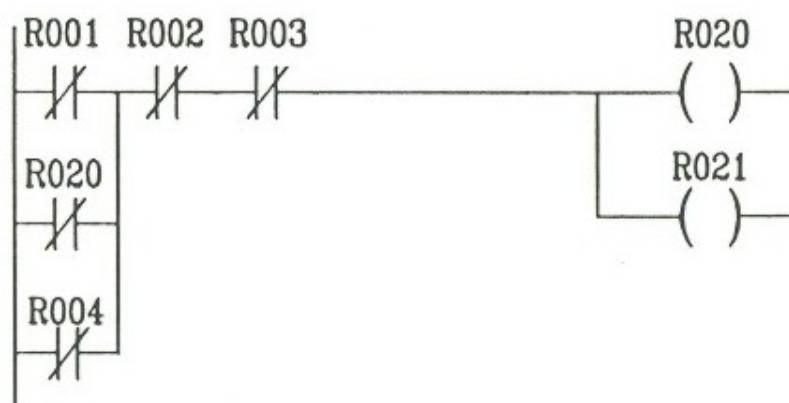


Figura 2 - Exemplo de Programa de CP

1.2. Linguagem de Programação

A estrutura da linguagem de programação, bem como o tipo de operandos usados serão descritos neste manual, porém o usuário necessita ter em mãos o manual de Utilização do Terminal de Programação.

Existem três Terminais de Programação que podem ser usados nas séries AL-500 e AL-1000:

- AL-1800;
- AL-2800;
- AL-3800.

As informações necessárias para o uso dos mesmos encontram-se nos seus respectivos manuais de utilização.

1.2.1. Operandos

Existem cinco tipos de operandos disponíveis nesta linguagem de programação:

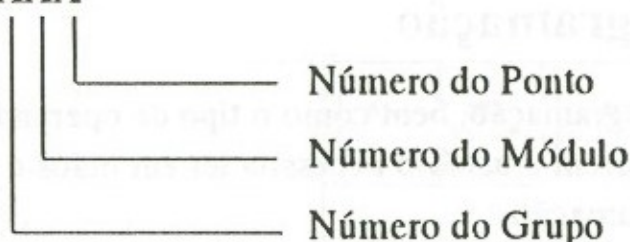
- Relés;
- Auxiliares;
- Memórias;
- Tabelas;
- Constantes.

1.2.1.1. Relés (R)

O operando tipo relé referência no programa do usuário um ponto físico de entrada ou saída.

A forma de endereçamento físico dos módulos de entradas ou saídas está descrita no Manual de Utilização AL-1000/AL-500 cód. 6299-000.4.

RXXXX

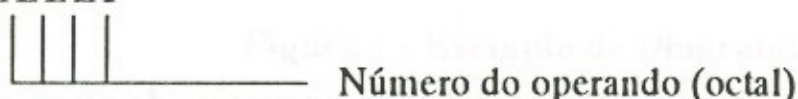


O endereçamento dos módulos de entradas ou saídas é feito em base octal, portanto o operando real (R) também é em base octal. Pode-se utilizar o número máximo de operandos Reais dependendo da configuração e do modelo de UCP usado.

1.2.1.2. Auxiliares (A)

Os operandos Auxiliares têm as mesmas características que os operandos Reais, representando estados intermediários da lógica do usuário.

AXXXX



O número máximo de operandos Auxiliares depende do modelo de UCP utilizado.

1.2.1.3. Memórias (M)

O operando tipo memória é usado no programa para armazenar valores que vão desde 0000 até 9999 (digitais).

MXXXX



O número máximo de operandos Memória depende do modelo usado.

1.2.1.4. Constante (K)

O operando constante é usado no programa sempre que for necessário especificar um valor constante.

KXXXX

 Valor constante (decimal)

1.2.2. Lógica

Um programa de usuário é formado por um conjunto de lógicas.

A lógica de programação é formada por 32 células posicionadas numa matriz de 8 colunas por 4 linhas (Figura 3), na qual serão editadas as instruções. Uma instrução pode ocupar uma ou mais células.

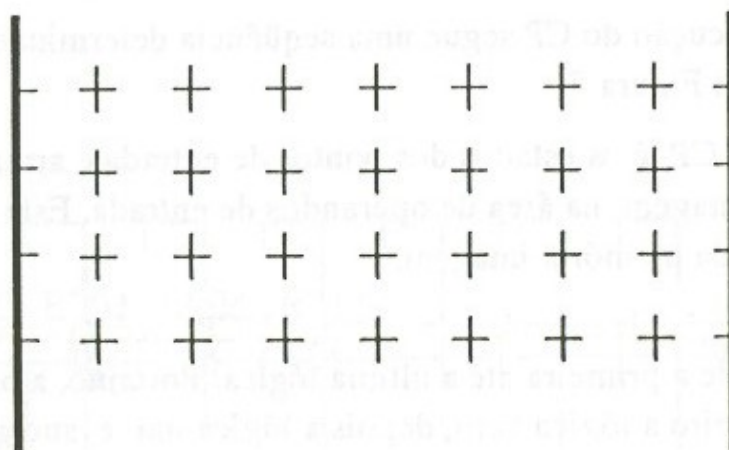


Figura 3 - Lógica de Programação

A barra esquerda da lógica é chamada de barra de energia, a qual está sempre energizada (nível lógico "1"). A energia flui na lógica sempre da esquerda para a direita, de cima para baixo e de baixo para cima (Figura 4), não sendo possível o fluxo de energia da direita para a esquerda.

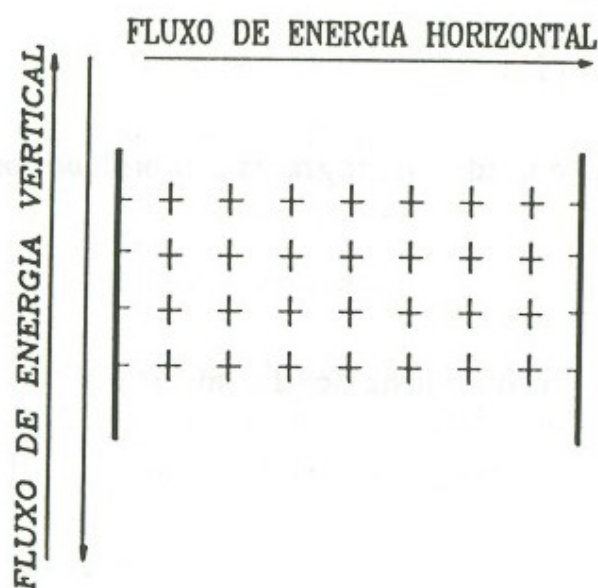


Figura 4 - Fluxo de Energia

1.2.3. Ciclo de Varredura

O ciclo de execução do CP segue uma seqüência determinada, conforme ilustram a Figura 5 e a Figura 7.

Inicialmente o CP lê os estados dos pontos de entrada e armazena esta informação na memória imagem, na área de operandos de entrada. Esta operação é chamada de atualização da memória imagem.

Após a atualização da memória imagem, a UCP executa o programa de maneira seqüencial, desde a primeira até a última lógica. Portanto, a ordem da execução das lógicas é primeiro a lógica zero, depois a lógica um e sucessivamente até a última lógica.

No fim da execução do programa a memória imagem é atualizada, novamente, e os valores dos operandos de saída são colocados nos módulos de saída.

A varredura na lógica é executada coluna por coluna, de cima para baixo (Figura 6) e da esquerda para a direita, ou seja, primeiro são executadas as instruções que estão na primeira coluna, depois a segunda coluna até a oitava coluna. Na coluna, as instruções são executadas de cima para baixo, ou seja, primeiro é executada a instrução da primeira célula, depois a segunda célula até a quarta célula.

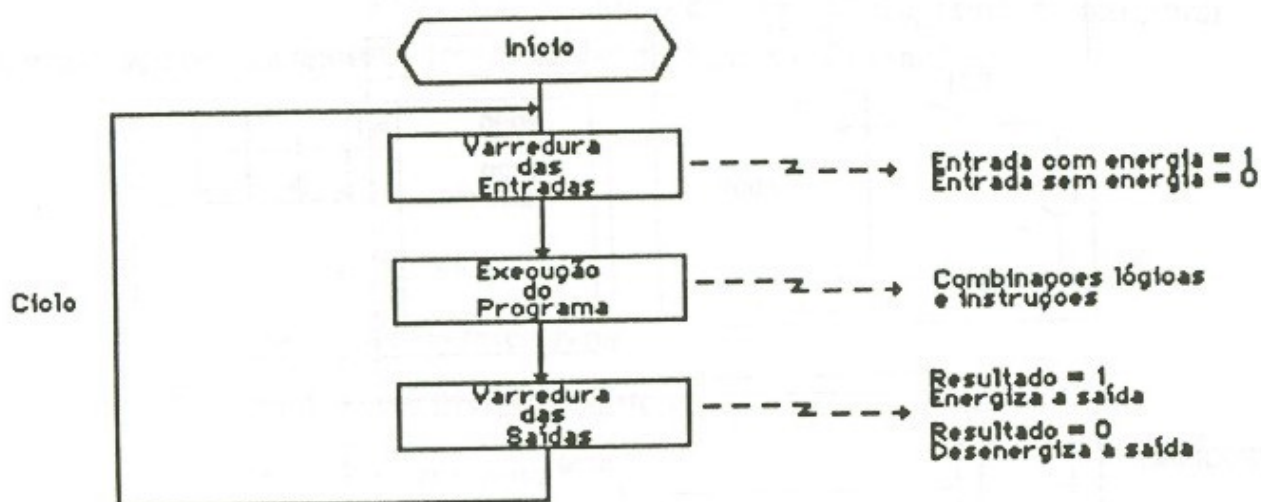


Figura 5 - Ciclo de Varredura

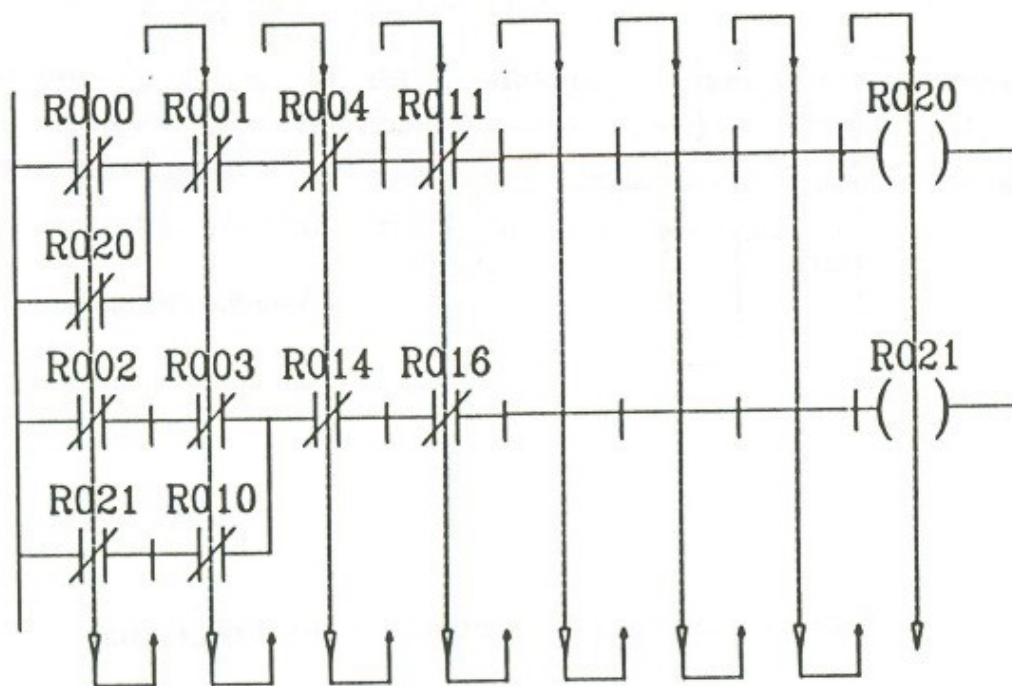


Figura 6 - Varredura em uma Lógica

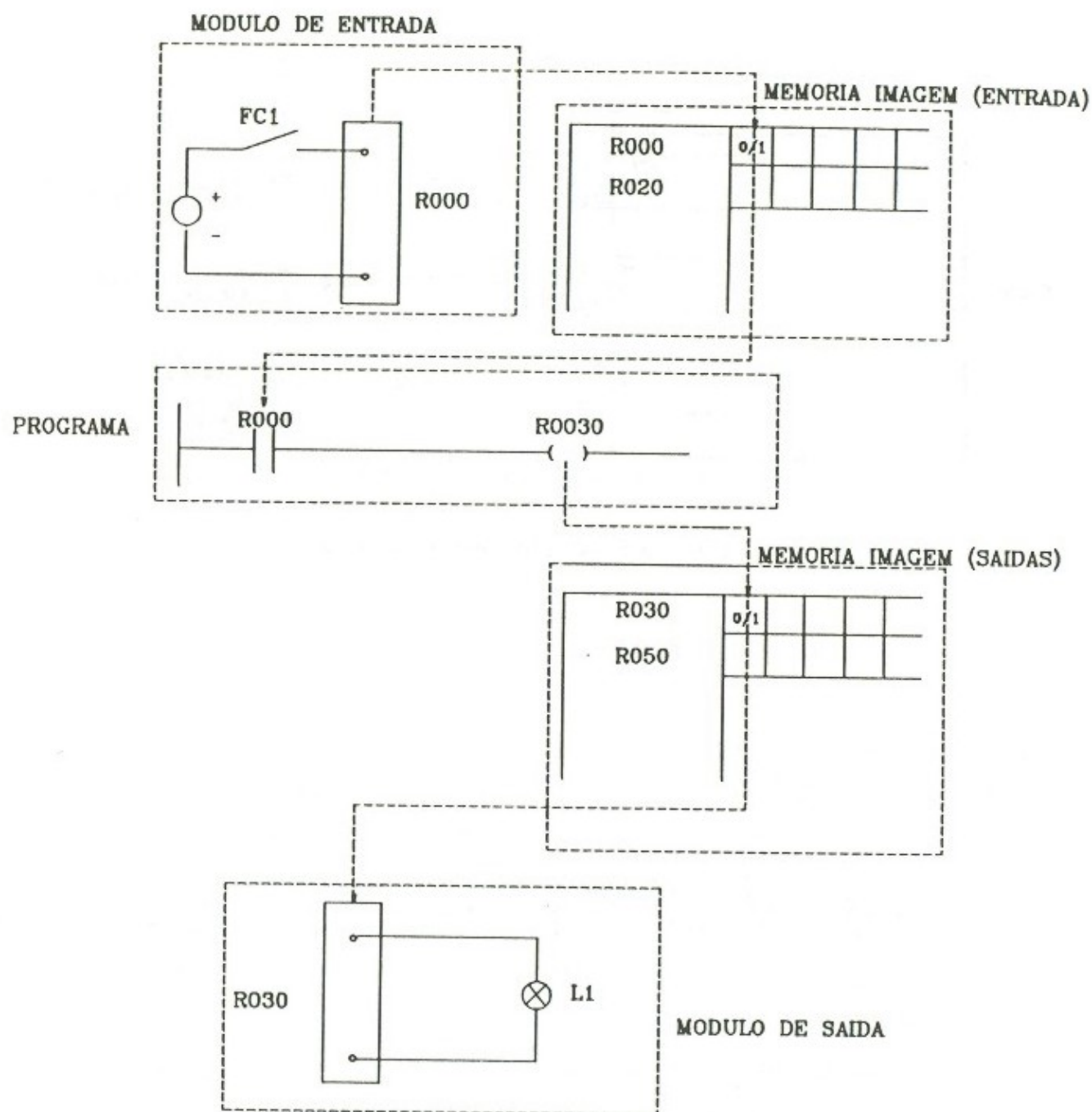


Figura 7 - Ciclo de Execução de um Programa

1.3. Configuração

A primeira atitude do usuário na hora de fazer um programa, é configurar o sistema. Para isso, o mesmo necessita das seguintes informações:

- Tipo de UCP;
- Número de octetos de entrada;
- Número de octetos de saída;
- Número de pontos Reais retentivos;
- Número de pontos Auxiliares retentivos;
- Número de Memórias retentivas;
- Número de grupos (apenas no AL-1000/512);
- Número de entradas especiais (apenas no AL-500);
- Número de saídas especiais (apenas no AL-500);
- Número de módulos duplos de entrada;
- Número de módulos duplos de saída;
- Programação "on line" (apenas no AL-1000/512).

A configuração é feita através do Terminal de Programação pelo comando Parâmetros, no AL-3800; CNFIG e PARAM, no AL-2800 e CMD 0 e CMD 29 no AL-1800. A maneira de usar os comandos de edição de configuração está descrita no manual de utilização do respectivo Terminal de Programação.

Configuracao do Controlador:		Parametros genericos:	
Tipo de controlador	128	Parametro 00	000
Numero de entradas	04	Parametro 01	000
Numero de saidas	06	Parametro 02	004
Reles retentivos	04	Parametro 03	004
Auxiliar retentivos	00	Parametro 04	000
Memorias retentivas	0040	Parametro 05	000
Memoria disponivel	16	Parametro 06	000
Numero do programa	010	Parametro 07	000
Velocidade de imp	9600	Parametro 08	000
		Parametro 09	000

SETAS-Movem=HOME-Inicio=END-Fim=PGUP-Topo=PGDN-Ult=ENTER-Conf=ESC-Aband

Visualizacao de Logicas: CTRL HOME/END: Logica inicial/final
 PGUP/PGDN: Log ant/post +/-: Lig vert/horiz INS:desce para logica
 HOME/END: Log inicial/final F7- movimentacao de colunas na edicao
 F1-Auxilio=F2-Salva prog=F3-Cont busca=F4-Base(Dec,Bin,Oct,Hex)=F5-Ciclo

Figura 8 - Tela de Configuração do TP AL-3800

Capítulo 2

Instruções Básicas

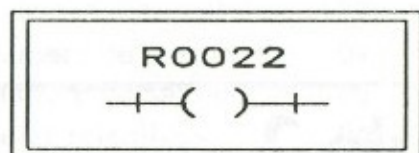
2.1. Introdução

A linguagem utilizada pelos Controladores Programáveis ALTUS AL-500 e AL-1000 é a Linguagem de Relés e Blocos. Esta linguagem é gráfica e apresenta a vantagem de ser similar a diagramas de relés convencionais. Os Terminais de Programação AL-1800, AL-2800 e AL-3800 programam os CPs com o conjunto básico de instruções apresentado neste capítulo. O programa escrito com estas instruções é compilado pelo sistema, podendo ser enviado sob forma de código objeto para o Controlador Programável ou ser gravado em módulo de memória EPROM para ser executado.

Observe que para a explicação de muitas instruções utilizam-se analogias com o funcionamento elétrico do dispositivo equivalente àquela instrução.

2.2. Bobina

2.2.1. Simbologia no Diagrama de Relés



2.2.2. Descrição

A instrução BOBINA modifica a memória imagem do Controlador Programável, com a posição modificada sendo determinada pelo operando da instrução. O valor colocado depende da lógica que especifica a função desejada pelo usuário.

Todos os contatos do programa com operando idêntico ao de instrução BOBINA comportam-se sob o comando desta. Se na Figura 9 a "corrente lógica" energiza a bobina R0020, o CONTATO NORMALMENTE ABERTO R0020 fechará, pois na memória imagem estará o valor "1".

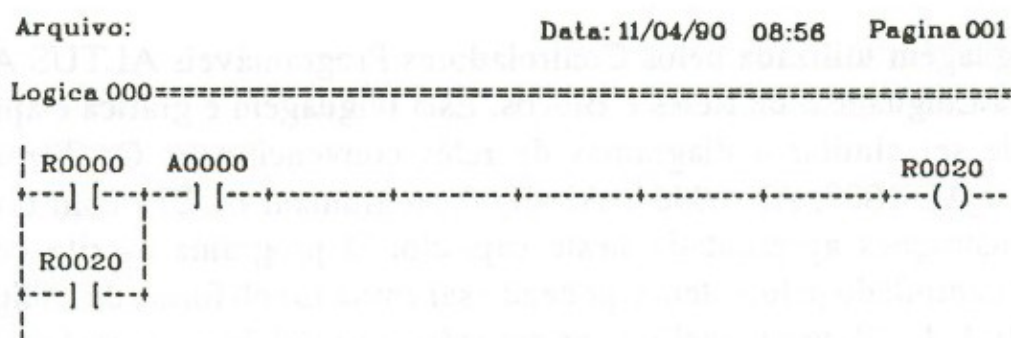


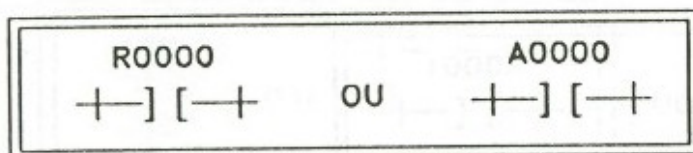
Figura 9 - Lógica com Bobina e Contatos NA

É importante salientar que podem existir no máximo quatro instruções BOBINA em uma mesma lógica de programa, estas devem localizar-se sempre na coluna sete de cada lógica.

O preenchimento de ligações horizontais de uma lógica da coluna de valor mais alto que possua instruções até a coluna 7 é feito automaticamente pelo sistema ao ser encerrada a edição da lógica.

2.3. Contato Normalmente Aberto

2.3.1. Simbologia no Diagrama de Relés



2.3.2. Descrição

Esta instrução reflete, logicamente, no programa, o comportamento real de um contato elétrico de um relé.

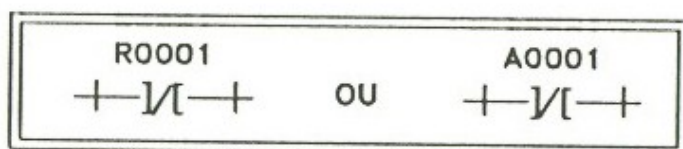
Este contato fecha-se quando circula "corrente lógica" na sua bobina de controle. Conforme a condição do bit correspondente ao seu operando na memória esteja em "1" ou "0", o CONTATO NORMALMENTE ABERTO estará fechado ou aberto, respectivamente.

Observe, na Figura 9, que para os contatos R0000, A0000 e R0020 permitirem a passagem da "corrente lógica" é necessário que os mesmos estejam fechados. Para tanto, o valor dos operandos R0000, A0000 e R0020 na memória imagem deverá ser "1".

O CONTATO NA de um mesmo operando pode ser repetido "n" vezes no programa do usuário.

2.4. Contato Normalmente Fechado

2.4.1. Simbologia no Diagrama de Relés



2.4.2. Descrição

As características desta instrução são as mesmas daquela denominada CONTATO NA, exceto pelo fato de o estado da instrução em questão corresponder à condição inversa do bit de seu operando na memória imagem, ou seja, conforme o bit esteja em "0" ou "1" o contato estará fechado ou aberto.

Podemos observar na Figura 10, que os contatos R0001, A0001 e R0021 permitem a passagem da "corrente lógica". Isto persiste enquanto os valores dos operandos correspondentes na memória imagem forem "0".

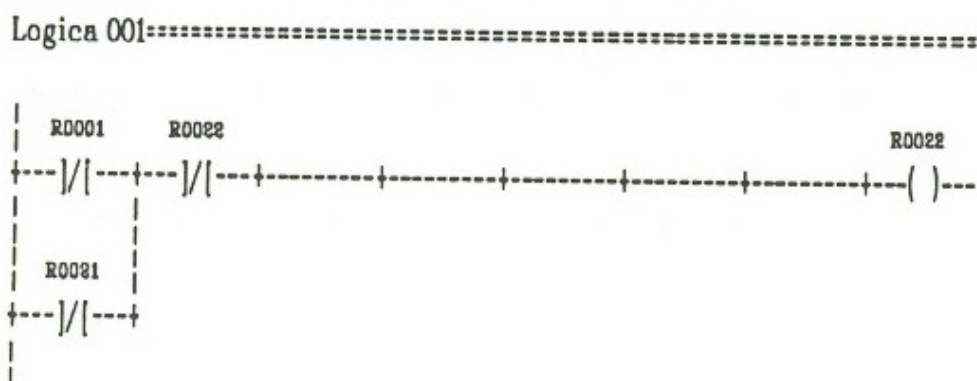
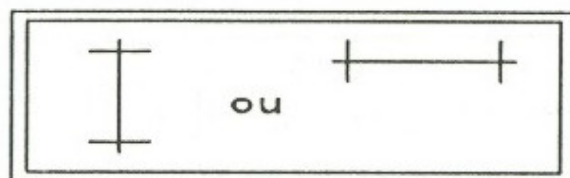


Figura 10 - Lógica com Contatos NF

O CONTATO NF de um mesmo operando pode ser repetido "n" vezes no programa do usuário.

2.5. Ligação Horizontal e Vertical

2.5.1. Simbologia no Diagrama de Relés



2.5.2. Descrição

Estas instruções não possuem operandos e são utilizadas para desenhar, de forma completa o diagrama de relés em uma lógica.

A LIGAÇÃO HORIZONTAL permite que a "corrente lógica" obtenha um caminho de uma célula para outra à sua direita.

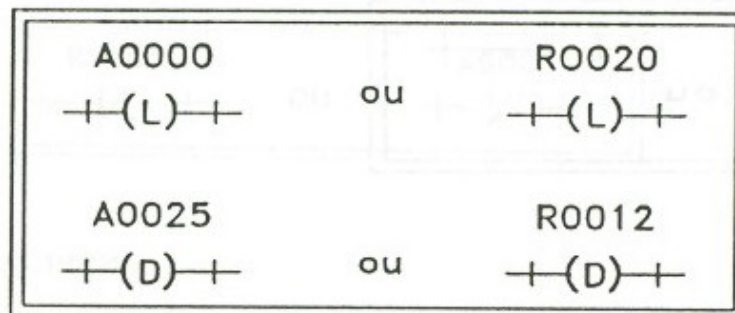
A LIGAÇÃO VERTICAL permite que a "corrente lógica" obtenha caminho de uma célula para outras duas, determinando a construção de ramos de contatos paralelos.

A ligação vertical estabelece um ramo paralelo entre a linha da célula onde é declarada e a linha seguinte, a partir da coluna seguinte àquela onde a instrução é declarada. Se a instrução LIGAÇÃO VERTICAL for declarada na linha 0 e coluna 0 (veja a Figura 10), dois ramos iniciando na coluna 1 são especificados, um na coluna 0 e outro na linha 1.

No Programador AL-3800, as teclas `e` possuem característica de "toggle", ou seja, permitem a retirada ou inclusão da ligação correspondente, conforme já exista ou não. No AL-1800 e AL-2800 existem teclas separadas para inclusão e retirada de cada tipo de ligação (ver capítulo sobre teclado do terminal no Manual de Utilização).

2.6. Bobina Liga/Bobina Desliga

2.6.1. Simbologia no Diagrama de Relés



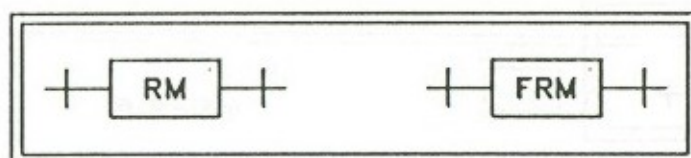
2.6.2. Descrição

As instruções BOBINA LIGA e BOBINA DESLIGA, quando energizadas, determinam que os bits dos operandos correspondentes na memória imagem assumam os valores "1" e "0", respectivamente.

Ambas as instruções, ao serem desenergizadas, não provocam qualquer alteração no valor dos operandos imagens correspondentes.

2.7. Relé Mestre/Fim de Relé Mestre

2.7.1. Simbologia no Diagrama de Relés



2.7.2. Descrição

As instruções RELÉ MESTRE e FIM DE RELÉ MESTRE podem ser programadas apenas uma vez em cada lógica e sempre ocupam a coluna 7 de uma das linhas da mesma.

As instruções em questão atuam em conjunto, definindo uma região que abrange desde a lógica seguinte àquela onde está declarada a instrução RELÉ MESTRE (RM) até a lógica onde existir a instrução FIM DE RELÉ MESTRE (FRM).

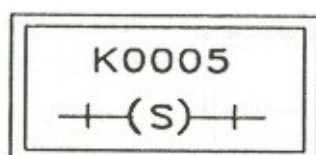
A instrução RM, quando tem sua entrada desenergizada, provoca a desenergização de todos os ramos conectados à direita na barra lógica do diagrama de relés até a lógica onde está presente a instrução FRM. Caso esteja energizada, a instrução atua como se estivesse ausente do programa.

Estas instruções atuam sempre na lógica seguinte à que estão declaradas.

Instruções que não necessitam de energização para serem executadas são independentes do estado das regiões limitadas por RM e FRM. Instruções com saídas energizáveis independentes da habilitação do barramento e bobinas eventualmente ligadas a estas serão normalmente atualizadas.

2.8. Bobina de Salto

2.8.1. Simbologia no Diagrama de Relés



2.8.2. Descrição

A instrução BOBINA DE SALTO serve para controlar a seqüência de um programa.

Seu operando é uma constante que determina o número de lógicas a serem saltadas a partir de sua energização.

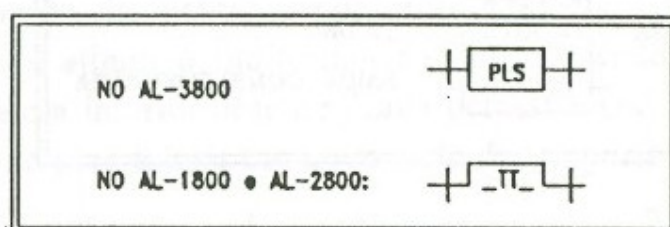
As instruções pertencentes a estas lógicas não serão executadas, permanecendo o estado de seus operandos inalterado. Quando desenergizada a BOBINA DE SALTO, a lógica seguinte àquela em que esta instrução está declarada será executada.

O salto será efetivado a partir do ponto onde a instrução estiver. Assim, se houver bobinas nas linhas subseqüentes da lógica àquela onde a mesma se encontra, estas serão saltadas.

Se a constante KXXXX for maior que o número de lógicas subseqüentes àquela onde a BOBINA DE SALTO está declarada, a primeira lógica do programa do usuário será a próxima a ser executada.

2.9. Relé de Pulso

2.9.1. Simbologia no Diagrama de Relés



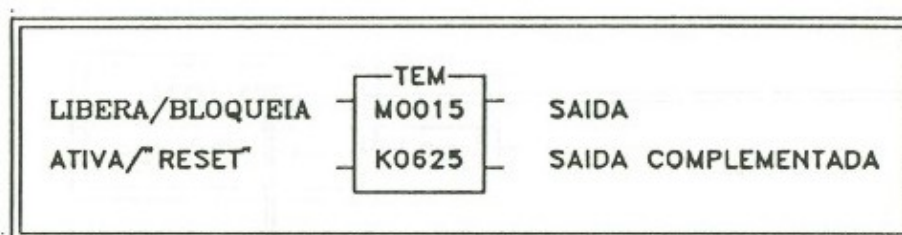
2.9.2. Descrição

A instrução RELÉ DE PULSO, quando tem o estado de sua entrada alterado de desenergizado para energizado, gera um pulso na saída com a duração de uma varredura de programa, ou seja, no início da varredura seguinte, sua saída é desenergizada.

Esta instrução pode ser utilizada apenas oito vezes em um único programa e não pode ser usada dentro da zona de atuação de uma bobina de salto, a menos que todas as instruções de pulso sejam saltadas simultaneamente.

2.10. Temporizador

2.10.1. Simbologia no Diagrama de Relés



2.10.2. Descrição

Esta instrução possui dois operandos.

O da célula superior, sempre do tipo M, especifica a memória acumuladora de tempo (acumula valores correspondentes a 0,1 s).

O da célula inferior determina o tempo limite para energização da saída da célula superior e pode ser do tipo K ou M.

A utilização de operando do tipo M determina uma memória, cujo conteúdo especifica o valor limite, e permite a implementação de tempos variáveis com uma única instrução.

Convenção : 1 = Energizada

0 = Desenergizada

MA = Memória acumuladora de tempo

ENTRADA DA CELULA SUPERIOR	ENTRADA DA CELULA INFERIOR	SAIDA DA CELULA SUPERIOR	SAIDA DA CELULA INFERIOR	ESTADO DO TEMPORIZADOR
0	0	0	1	RESETADO MA = 0
1	0	0	1	RESETADO MA = 0
1	1	0 / 1	1 / 0	TEMPO TRANS-CORRIDO TEMPORIZANDO
0	1	0 / 1	1 / 0	TEMPO TRANS-CORRIDO BLOQUEADO

Tabela 1 - Comportamento da Instrução Temporizador

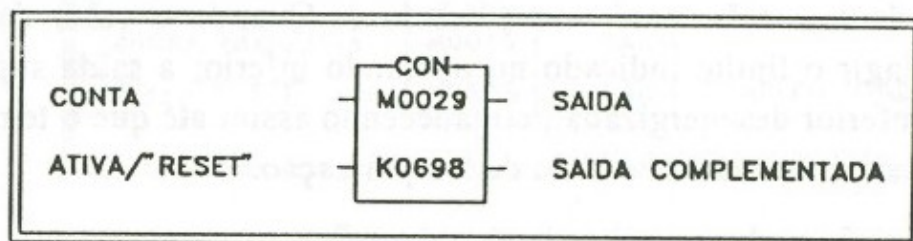
O valor máximo de contagem de uma instrução TEMPORIZADOR é de 999,9 segundos. Este valor corresponde a 9999 na memória acumuladora. Para obter temporizações maiores utiliza-se temporizadores em cascata ou um temporizador como base de tempo e vários contadores.

O estado das entradas Libera/Bloqueia (L/B) e Ativa /Reset (A/R) controlam o funcionamento da instrução conforme a tabela 1. Quando o valor da memória acumuladora atingir o limite indicado no operando inferior a saída superior será energizada e a inferior desenergizada permanecendo assim até que o temporizador seja resetado para iniciar um novo ciclo de temporização.

Esta instrução não pode ser utilizada na coluna 7.

2.11. Contador

2.11.1. Simbologia no Diagrama de Relés



2.11.2. Descrição

Esta instrução possui dois operandos.

O da célula superior, sempre do tipo M, especifica a memória que contabiliza os eventos.

O da célula inferior estabelece o valor limite de contagem para energização da saída da célula superior e pode ser do tipo K ou M.

A utilização de operando do tipo M define uma memória, cujo conteúdo especifica o valor limite, e permite a implementação de contagens com valor limite variável utilizando uma única instrução.

O estado das entradas Conta(C) e Ativa/"Reset"(A/R) controlam o funcionamento da instrução conforme a tabela 2. Quando o valor da memória acumuladora (operando superior) atingir o limite indicado no operando inferior a saída superior será energizada permanecendo assim até que o contador seja resetado. Nesta situação novos eventos (entrada "C" passando de 0 para 1) são desconsiderados.

Esta instrução não pode ser utilizada na coluna 7.

Convenção : 1 = Energizada

0 = Desenergizada

MA = Memória acumuladora de eventos

ENTRADA DA CELULA SUPERIOR	ENTRADA DA CELULA INFERIOR	SAIDA DA CELULA SUPERIOR	SAIDA DA CELULA INFERIOR	ESTADO DO CONTADOR
0	0	0	1	RESETADO MA → 0
1	0	0	1	RESETADO MA → 0
0 → 1	1	0 / 1	1 / 0	* **
1 → 0	1	0 / 1	1 / 0	*** ****

* MA → MA + 1 ; Conta evento

MA ← Limite

** MA MA + 1 ; Conta evento

MA = Limite

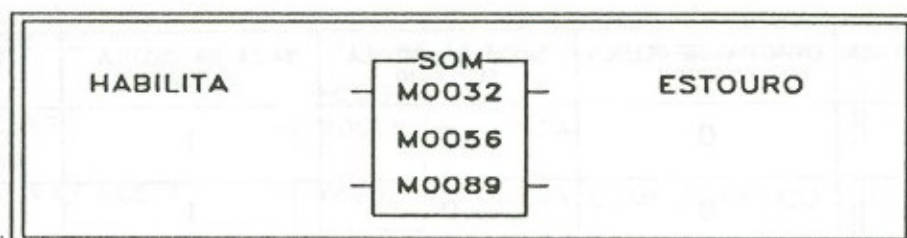
*** MA # Limite

**** MA = Limite

Tabela 2 - Comportamento da Instrução Contador

2.12. Soma

2.12.1. Simbologia no Diagrama de Relés



2.12.2. Descrição

Esta instrução ocupa três células de uma coluna de uma lógica de programação.

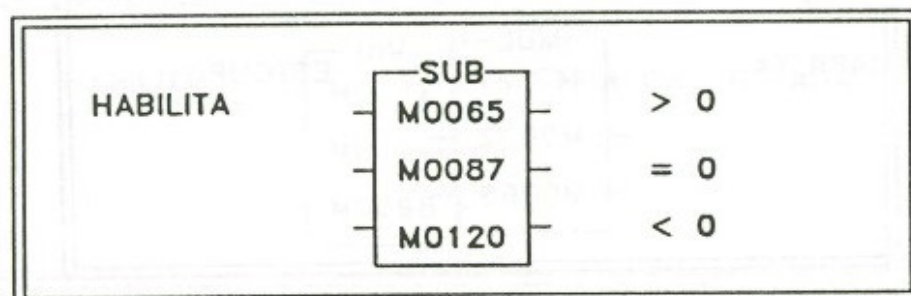
Quando a entrada do bloco que representa a instrução no diagrama de relés é energizada, os operandos especificados nas duas primeiras células são somados e o resultado armazenado no operando da terceira.

Embora os operandos parcela possam ser do tipo constante ou memória, o operando da terceira célula deve necessariamente ser do tipo memória.

Se a soma exceder o valor máximo armazenável numa memória (9999), o resultado final será 9999 e a saída ESTOURO será energizada, permanecendo neste estado até que outra operação seja executada.

2.13. Subtração

2.13.1. Simbologia no Diagrama de Relés



2.13.2. Descrição

Esta instrução é de três células.

Quando a entrada HABILITA é energizada, o operando da segunda célula é subtraído do operando da primeira e o resultado é armazenado na memória especificada na terceira célula.

Embora os operandos da primeira e segunda células possam ser do tipo constante ou memória, o operando da terceira célula deve, necessariamente, ser do tipo memória.

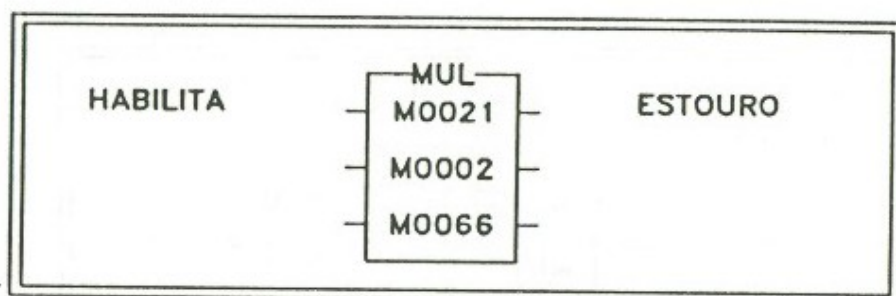
As linhas de saída "> 0", "= 0" e "< 0" podem ser usadas para efetuar comparações e são acionadas de acordo com o resultado da operação de subtração. Quando o resultado é negativo, a saída "< 0" é acionada e a memória do resultado recebe o valor absoluto da subtração.

Se a entrada HABILITA não está energizada, todas as saídas são desenergizadas.

Esta instrução não pode ser utilizada na coluna 7.

2.14. Multiplicação

2.14.1. Simbologia no Diagrama de Relés



2.14.2. Descrição

Esta instrução é de três células.

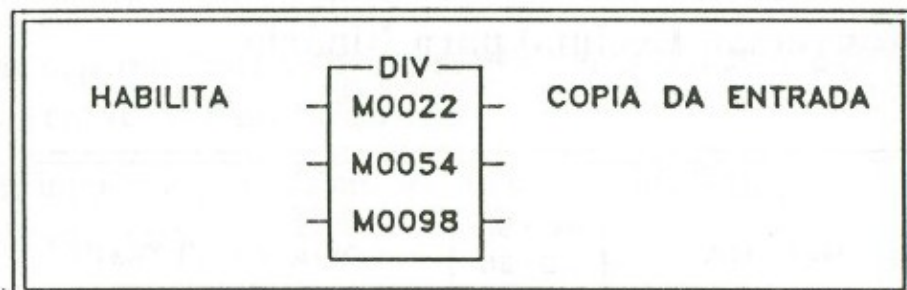
Quando a entrada HABILITA é energizada, ocorre a multiplicação do conteúdo da memória ou constante especificado na primeira célula pela memória ou constante especificada na segunda. O resultado é armazenado na memória especificada na terceira célula.

Embora os operandos da primeira e segunda célula possam ser do tipo constante ou memória, o operando da terceira célula deve ser necessariamente uma memória. Caso o resultado exceda o valor máximo armazenável em uma memória (9999), o resultado final será 9999 e a saída ESTOURO será energizada, permanecendo assim até que seja efetuada uma nova operação.

Esta instrução não pode ser utilizada na coluna 7.

2.15. Divisão

2.15.1. Simbologia no Diagrama de Relés



2.15.2. Descrição

Esta instrução é de três células.

Quando a entrada HABILITA é energizada, ocorre a divisão do operando da primeira célula pelo da segunda, sendo o resultado armazenado na memória especificada na terceira célula.

Embora os operandos da primeira e segunda células podem ser do tipo memória ou constante, o da terceira célula deve ser, necessariamente, uma memória.

O primeiro operando pode assumir qualquer valor entre 0 e 9999, mas o segundo somente entre 0 e 255. O resultado terá valores entre 0 e 9999, sendo o resto armazenado na memória M0000.

Quando é feita uma divisão por 0, o valor 9999 é armazenado na memória destino e a memória M0000 não é alterada.

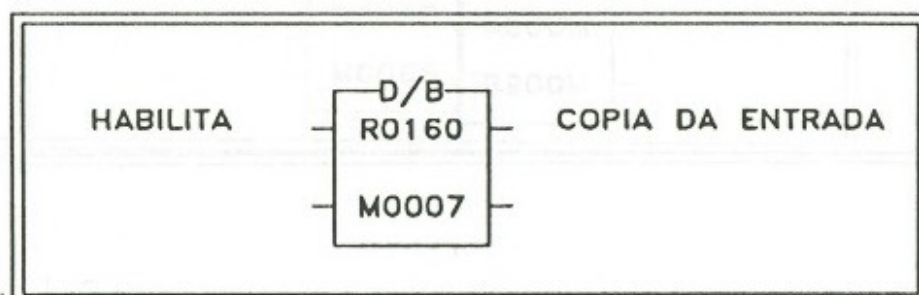
A saída é cópia da entrada HABILITA.

Esta instrução não pode ser utilizada na coluna 7.

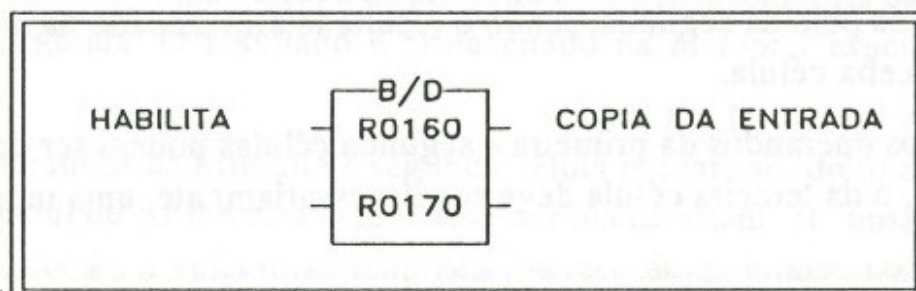
2.16. Conversão D/B e B/D

2.16.1. Simbologia no Diagrama de Relés

2.16.1.1. Conversão Decimal para Binário



2.16.1.2. Conversão Binário para Decimal



2.16.2. Descrição

As informações de conversão são instruções de entrada e saída de dados na forma de numeração BCD ("Binary Coded Decimal").

A instrução de conversão decimal para binário permite a aquisição, pelos CPs ALTUS, de dados com valores entre 0 e 9999 expressos sob a forma BCD. Internamente, estes dados serão armazenados em binário em um operando memória (M).

A Figura 11 mostra, esquematicamente, o funcionamento da instrução de conversão decimal para binário (D/B) utilizando os módulos de interface para chaves rotativas do tipo "thumbwheel". Quando a entrada habilita é energizada, o valor 8842, por exemplo, é convertido da chave rotativa, onde está composto pelos bits 1000100001000010, para binário com a composição de bits 0010001010001010.

A célula superior desta instrução recebe um operando relé (R), que especifica o endereço da chave "thumbwheel".

A célula inferior especifica um operando memória (M), que recebe o valor binário resultado da conversão.

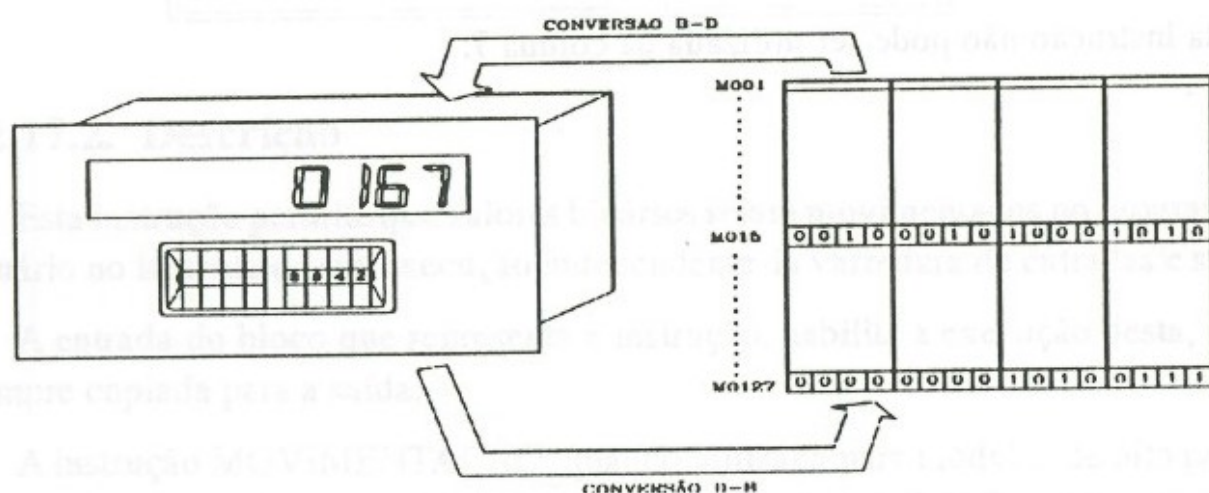


Figura 11 - Funcionamento das Instruções de Conversão BCD

A instrução de conversão de binário para decimal (B/D) converte um valor interno, residente em um operando do tipo memória (M) ou constante (K), para um módulo da saída, que expressará este valor em BCD.

A Figura 11 mostra, esquematicamente, o funcionamento da instrução de conversão binário para BCD, utilizando os módulos de interface para displays de 4 dígitos de sete segmentos. Quando a entrada HABILITA é energizada o valor 167, por exemplo, expresso em binário (0000001010011) em uma das memórias M(operando superior), é convertido para o valor correspondente (0000000101100111) em BCD, que é apresentado no display.

A célula superior desta instrução recebe uma memória que contém o valor binário que será convertido.

Na célula inferior é especificado um operando do tipo R, que define o endereço do módulo de saída que vai receber a informação em BCD.

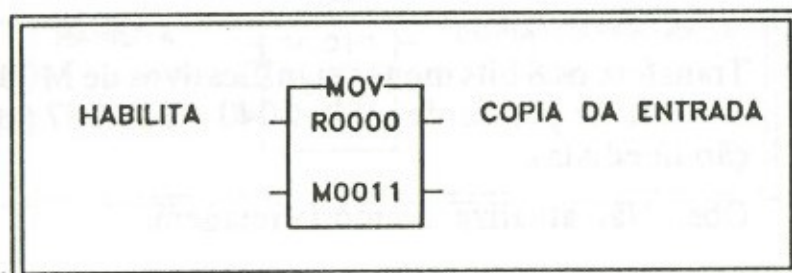
No caso da instrução de B/D a memória imagem do operando destino não é atualizada. A transferência é feita diretamente para o módulo de saída especificado pelo operando R da 2ª célula no momento da execução da instrução.

As instruções B/D e D/B podem ser utilizadas utilizando módulos digitais como origem ou destino da movimentação, neste caso, apenas os 8 bits menos significativos da memória origem são convertidos em saídas no módulo digital destino (instrução B/D). Se a origem for um módulo digital de entrada (instrução D/B) os seus 8 pontos codificados em 2 dígitos BCD serão convertidos em valor binário na memória destino. A saída é cópia do estado da entrada.

Esta instrução não pode ser utilizada na coluna 7.

2.17. Movimentação

2.17.1. Simbologia no Diagrama de Relés



2.17.2. Descrição

Esta instrução permite que valores binários sejam movimentados no programa do usuário no instante de sua execução independente da varredura de entradas e saídas.

A entrada do bloco que representa a instrução, habilita a execução desta, sendo sempre copiada para a saída.

A instrução MOVIMENTAÇÃO, quando utilizada para módulos de oito pontos, transfere apenas valores entre 0 e 255. A imagem dos pontos de E/S nunca é atualizada pela mesma.

Esta instrução não pode ser utilizada na coluna 7.

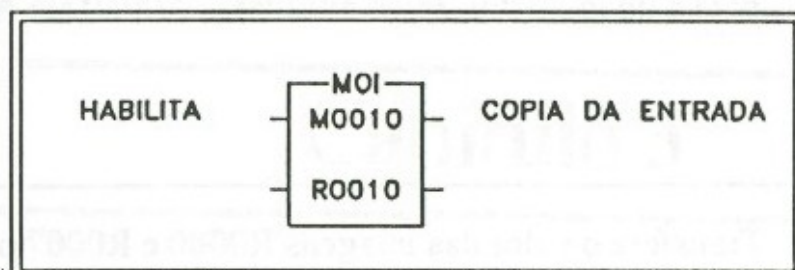
A Tabela 3 apresenta exemplos de atuação para as diferentes modalidades desta instrução.

	<p>Transfere o valor do octeto de entradas reais R0020 para os 8 bits menos significativos de M0015 (leitura imediata).</p>
	<p>Transfere os 8 bits menos significativos de M0032 para o octeto de pontos de saída R040 até R0047 (atualização imediata). Obs.: Não atualiza memória imagem.</p>
	<p>Transfere o conteúdo de M0012 para M0005.</p>
	<p>Transfere a constante 128 (decimal) para M0012.</p>
	<p>Transfere o equipamento binário de 125 (01111101) para o octeto de saída R0050 até R0057 (atualização imediata). Obs.: Não atualiza memória imagem.</p>
	<p>Transfere o octeto de pontos R0010 a R0017 para o octeto R0100 a R0107 (transferência imediata). Obs.: Não atualiza memória imagem.</p>

Tabela 3 - Exemplos de Aplicações da Instrução MOV

2.18. Movimentador de Imagem

2.18.1. Simbologia no Diagrama de Relés



2.18.2. Descrição

Esta instrução permite a transferência de octetos (8 bits) de operandos entre as memórias imagem de relés (R), auxiliares (A) e memórias (M).

No caso de operandos M, somente seus 8 bits menos significativos são movimentados, enquanto os 8 mais significativos não são considerados pela instrução.

A instrução é executada quando a entrada HABILITA é energizada, sendo sempre copiada para a saída.

Esta instrução não pode ser usada na coluna 7.

A Tabela 4 apresenta exemplos de atuação desta instrução.

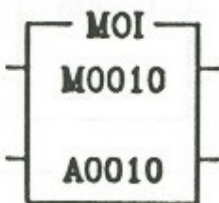
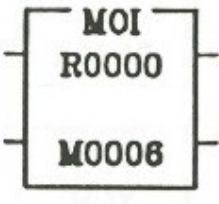
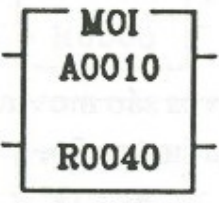
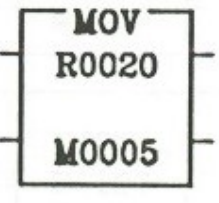
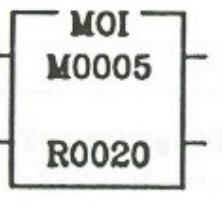
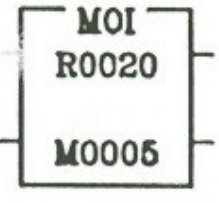
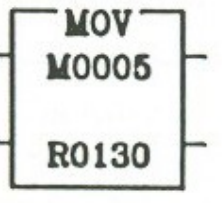
	Transfere os 8 bits menos significativos de M0010 para a imagem dos relés auxiliares A0010 até A0017.	
	Transfere o valor das imagens R0000 e R0007 para os bits menos significativos de M0006.	
	Transfere as imagens dos relés auxiliares A0010 a A0017 para a memória imagem de R0040 a R0047.	
		Faz uma atualização dos pontos de imagem de entrada durante a execução do programa. A instrução MOV lê os pontos reais do módulo de entrada e a instrução MOI carrega-os na imagem.
		Faz uma atualização dos pontos de saída durante a execução do programa. A instrução MOI lê a imagem de saída para a memória M0005 e a instrução MOV carrega-os nos pontos de saída reais.

Tabela 4 - Exemplos de Aplicações da Instrução MOI

Capítulo 3

Instruções Estendidas

3.1. Introdução

As Instruções Estendidas introduzem a possibilidade de manipulação de tabelas e arquivos, utilização de instruções seqüenciador, contagem bidirecional, conversão A/D e D/A, impressão de textos (ou conteúdos de memória) e incluem as Instruções Especiais.

3.2. Instruções para Tratamento de Tabelas

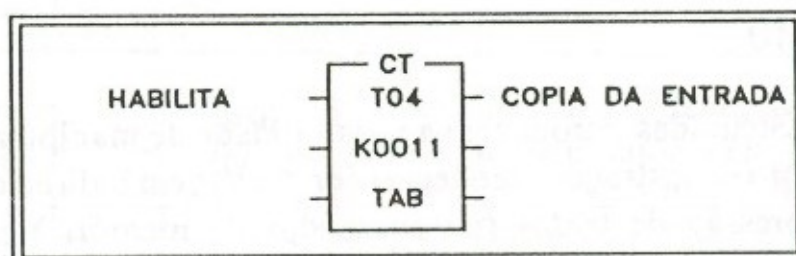
As tabelas são operandos que podem armazenar grupos numéricos, podendo ser utilizadas para realizar seqüenciamentos, armazenamento de valores, decodificação numérica, etc.

Cada uma das 64 tabelas (numeradas de 0 a 63) disponíveis em um programa pode ter 64 posições (numeradas de 0 a 63) que armazenam valores numéricos ou octetos binários. O número total de posições de todas as tabelas utilizadas em um programa não pode ultrapassar 640, exceto no CP AL-1000/512, onde o número total pode chegar até 3007 posições.

O número de posições de cada tabela e o número de tabelas utilizado deve ser declarado no início do programa, levando-se em consideração os limites já mencionados.

3.2.1. Carrega Tabela

3.2.1.1. Simbologia no Diagrama de Relés



3.2.1.2. Descrição

Quando a entrada HABILITA é energizada a instrução transfere os valores programados para as posições da tabela especificada.

Os operandos correspondentes às três células que a compõem são os seguintes:

- 1ª célula TXX Especifica o número da tabela a ser carregada.
- 2ª célula KXXXX Especifica o número de posições da tabela.
- 3ª célula TABELA É o conteúdo a ser carregado na tabela.

O preenchimento do conteúdo na 3ª célula é realizado, em cada posição, com o auxílio das teclas de deslocamento de cursor.

Após o preenchimento da 1ª e 2ª células, ao posicionarmos o cursor na 3ª célula, a tabela em questão é visualizada inteira no formato XX : 0000. Os dois primeiros dígitos definem o número da posição. No AL-1800 as posições são visualizadas uma por vez, seguindo as mesmas regras dos outros terminais.

Utilizando as teclas de deslocamento, incrementamos o índice de posição da tabela, obtendo acesso ao conteúdo desta posição de tabela. Para introduzir novos conteúdos basta pressionar as teclas numéricas correspondentes.

As bases numéricas possíveis nos Programadores AL-3800 e AL-2800 são a decimal, a octal, a binária e a hexadecimal, sendo distinguíveis pela apresentação do mnemônico correspondente na tela. No Programador AL-1800 estão disponíveis as bases decimal e octal.

Para informações sobre o procedimento de mudança de base, o usuário deve consultar o conteúdo "Teclas Funcionais" do Manual de Utilização do Terminal de Programação.

A instrução CARREGA TABELA (CT), deve ser localizada sempre na coluna 7 (veja Figura 12).

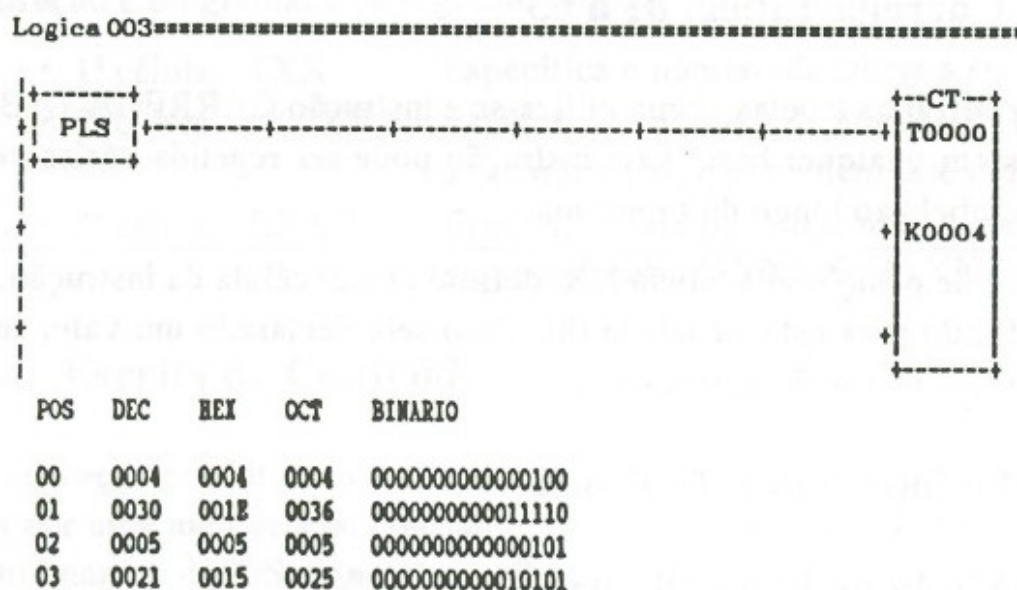


Figura 12 - Listagem da Instrução CT

Esta instrução está funcionalmente dividida em dois tipos: CARREGA TABELA 00 e CARREGA TABELA 01 A 63.

3.2.1.3. Carrega Tabela 00

Na tabela referenciada define-se quantas tabelas serão utilizadas e de quantas posições cada uma se compõe.

Os operandos que a constituem são os seguintes:

- 1ª célula T00 Especifica a tabela 00.
- 2ª célula K00XX É o número de tabelas utilizadas no programa, incluindo-se a T00.
- 3ª célula TAB Especifica o número de posições de cada tabela a partir da T00 (aconselha-se usar a base decimal).

A Figura 12 apresenta um exemplo de instrução carrega tabela 0 que declara três tabelas com 30, 5 e 25 posições cada uma, além da própria tabela 0.

A instrução CARREGA TABELA 00 deve ser sempre executada no início do programa, antes de qualquer outra instrução que manipula tabelas e, preferencialmente, apenas na primeira varredura, utilizando-se da instrução RELÉ DE PULSO.

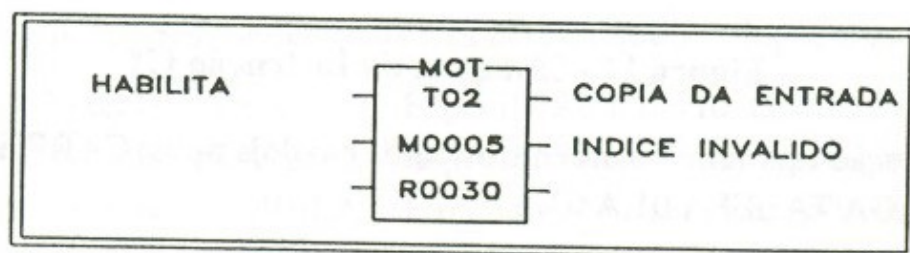
3.2.1.4. Carrega Tabela 01 a 63

Para inicializar as tabelas acima utiliza-se a instrução CARREGA TABELA XX com valores em qualquer base. Esta instrução pode ser repetida várias vezes para uma mesma tabela ao longo do programa.

O número de posições da tabela XX, definido na 2ª célula da instrução, deve ser igual ao definido para esta na tabela 00. Caso seja declarado um valor inferior as últimas posições não serão carregadas.

3.2.2. Movimentador de Tabela

3.2.2.1. Simbologia no Diagrama de Relés



3.2.2.2. Descrição

Esta instrução possibilita a transferência entre posições de tabelas e operandos memória, relés e auxiliares. A segunda célula da instrução contém um operando do tipo constante ou memória que determina (aponta) a posição da tabela que será lida ou escrita.

Uma única instrução MOT pode ser utilizada para movimentar qualquer posição da tabela através da alteração do conteúdo da memória apontadora. Caso seja utilizada uma constante, a transferência é feita sempre de/para a mesma posição.

A instrução é executada quando a entrada habilita é energizada.

3.2.2.3. Leitura de Conteúdo

Esta instrução permite ler o conteúdo de uma posição da tabela XX apontada por uma memória ou constante e carregá-lo em uma memória, octeto imagem de E/S ou octeto auxiliar.

A instrução é programada da seguinte forma:

- 1ª célula TXX Especifica o número da tabela a ser lida.
- 2ª célula K00XX Especifica o número da posição (K) a ser lida ou a memória (M) que contém este número.
- 3ª célula RXXX Especifica para onde o conteúdo da posição deve ser transferido: MXXXX, RXXX0 ou AXXX0.

3.2.2.4. Escrita de Conteúdo

Esta instrução permite escrever um conteúdo em uma posição da tabela XX apontada por uma memória ou constante. Este conteúdo pode estar em uma memória, octeto imagem de E/S ou octeto auxiliar.

Neste caso, a instrução é programada da seguinte forma:

- 1ª célula MXXX Especifica de onde o valor deve ser transferido.
- 2ª célula K00XX Especifica o número da posição (K) a ser alterada na tabela ou a memória (M) que contém este número.
- 3ª célula TXXX Especifica o número da tabela para onde será transferido o conteúdo.

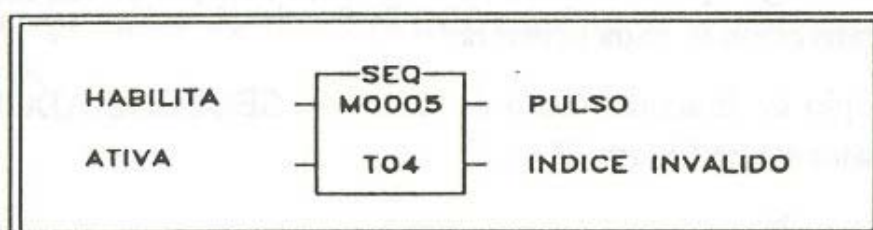
Se o valor da segunda célula for maior que a maior posição da tabela, a transferência não será efetuada e a saída **ÍNDICE INVÁLIDO** é acionada.

A saída **ÍNDICE INVÁLIDO** é desenergizada quando ocorre uma operação válida ou quando a instrução é desabilitada.

Esta instrução não pode ser utilizada na coluna 7.

3.3. Seqüenciador

3.3.1. Simbologia no Diagrama de Relés



3.3.2. Descrição

Esta instrução permite o controle de um seqüenciador com base em um conjunto de condições de evolução especificadas passo a passo contidas em uma tabela. As condições de evolução de um passo para o seguinte são operandos do tipo AXXXX (auxiliares) o que permite a utilização de condições completamente genéricas (tempos, eventos, etc.) conforme o acionamento dos auxiliares no programa.

Compõe-se de duas células, com as seguintes funções:

- 1ª célula MXXXX Especifica a memória que indica o passo atual do seqüenciamento servindo como índice da tabela de condições de evolução.
- 2ª célula TXX Especifica a tabela que contém as condições de evolução determinadas pelos operandos do tipo A endereçados nesta tabela (valores octais de endereço).

Quando as entradas HABILITA e ATIVA estão energizadas, a instrução acessa o conteúdo da tabela TXX apontado por MXXXX e examina o operando do tipo Relé Auxiliar indicado nesta posição da tabela. Se este auxiliar estiver energizado, MXXXX é incrementada e na saída é dado um pulso de desenergização com duração de um ciclo de varredura de programa, indicando que houve a evolução de um passo.

A cada evolução no seqüenciamento o próximo elemento da tabela é examinado e o procedimento é repetido até a última posição da tabela quando ocorre o retorno à primeira posição (MXXXX=0) caso a condição de evolução seja satisfeita.

Caso o elemento da tabela corresponda a um operando do tipo Relé Auxiliar que está desenergizado, nada ocorrerá, permanecendo o valor de MXXXX inalterado.

Assim, se MXXXX contém o valor 3, a posição 3 da tabela XX será examinada. Supondo que o conteúdo desta posição seja 155 (octal), a instrução examinará o valor do auxiliar A0155 na memória imagem. Caso o auxiliar esteja em estado 1, o valor do operando MXXXX será incrementado de uma unidade (recebe o valor 4) e um pulso de desenergização será dado na saída com duração de um ciclo de varredura. Se A0155 estiver em 0, nada ocorrerá.

O princípio de funcionamento da instrução SEQÜENCIADOR é apresentado esquematicamente na Figura 13.

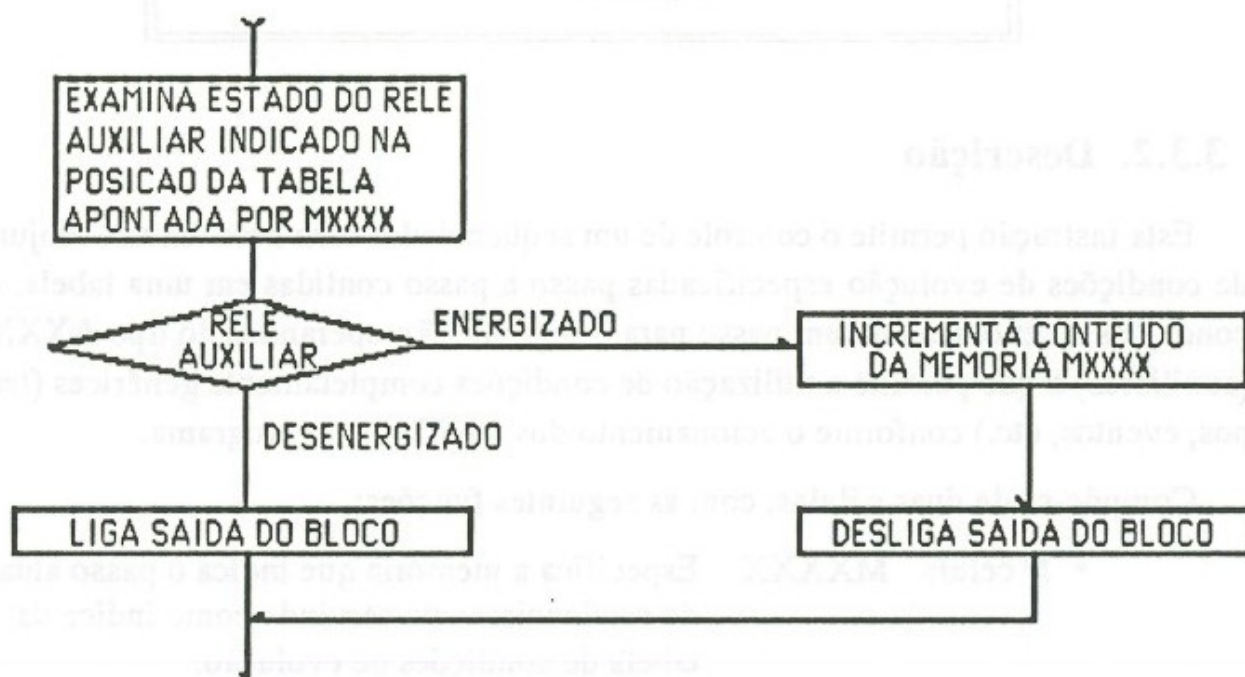


Figura 13 - Funcionamento da Instrução Seqüenciador

A entrada ATIVA, quando desenergizada, zera a memória indicadora de posição.

A entrada HABILITA, quando desenergizada, bloqueia o incremento de MXXXX, independentemente das condições de evolução da tabela.

Em ambos os casos, a saída ÍNDICE INVÁLIDO é desenergizada.

Tendo chegado ao fim de uma tabela de condições de evolução, a memória indicadora de posição recebe o valor zero e a seqüência é reiniciada.

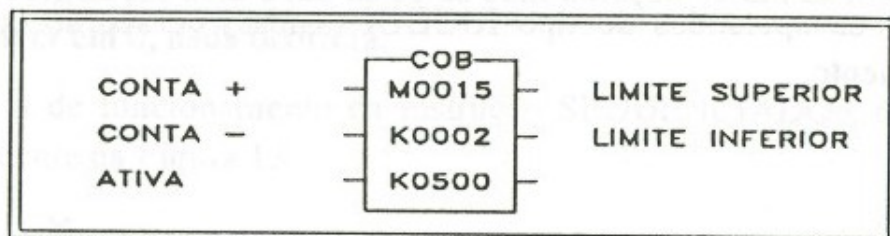
Se a memória indicar uma posição inexistente na tabela de condições de evolução ou se for referenciada uma tabela não definida, a saída ÍNDICE INVÁLIDO será ativada.

A saída PULSO é desabilitada somente no instante em que o pulso é emitido.

A memória MXXX pode ser utilizada como índice de outras tabelas que contenham valores binários contendo seqüências de acionamentos de pontos de saída. Neste caso, deve ser utilizado uma instrução MOT que movimenta os valores da tabela para os operandos do tipo RXXXX (saídas) de acordo com o passo do seqüenciamento.

3.4. Contador Bidirecional

3.4.1. Simbologia no Diagrama de Relés



3.4.2. Descrição

Esta instrução permite contagens em ambos sentidos, isto é, incrementa ou decrementa o conteúdo de um operando do tipo memória, sendo programada da seguinte maneira:

- 1ª célula MXXXX Memória acumuladora.
- 2ª célula K0XXX Especifica o valor do incremento/decremento.
- 3ª célula KXXXX Especifica o valor limite de contagem, podendo ser indicado por constante (KXXXX) ou memória (MXXXX).

A contagem ocorre sempre que a entrada ATIVA estiver energizada e uma das entradas CONTA passar do estado 0 para 1.

Quando a memória acumuladora atingir um valor igual ou superior ao indicado no limite, a memória acumuladora recebe o valor do limite e a saída LIMITE SUPERIOR é ligada.

Quando a memória acumuladora atingir o valor 0" (zero), a saída LIMITE INFERIOR é ligada.

Se ambas as saídas (LIMITE SUPERIOR e LIMITE INFERIOR) não estão ligadas, a saída da terceira célula é ligada.

Se, em um mesmo ciclo, ambas as entradas CONTA mudarem do estado "0" para "1", tudo se passará como se não houvesse contagem.

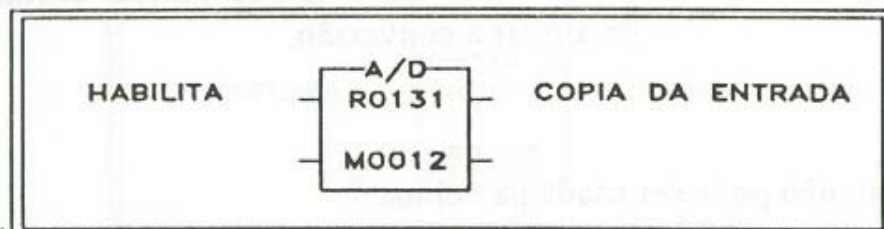
Quando a entrada ATIVA é desenergizada, a memória acumuladora é zerada e a saída LIMITE INFERIOR energizada.

O valor máximo de incremento/decremento é 255.

Esta instrução não pode ser usada na coluna 7.

3.5. Conversão Analógico / Digital

3.5.1. Simbologia no Diagrama de Relés



3.5.2. Descrição

Esta instrução permite a leitura do valor de um dos pontos de um módulo de entrada analógico e o armazenamento deste valor digitalizado em uma memória MXXXX.

É permitida a leitura de oito pontos por módulo, de forma individual.

A instrução é programada da seguinte forma:

- 1ª célula RXXXX Especifica o endereço do ponto de entrada analógico.
- 2ª célula MXXXX Especifica a memória destino.

A conversão ocorre independente do estado de entrada, sendo a saída cópia da entrada.

A única maneira de desabilitar esta instrução é saltando-a.

O valor analógico (tensão ou corrente) presente na entrada é convertido interativamente para um valor digital entre 0 e 1000 correspondente ao início e fim da escala analógica usada (0 a 10 V, 4 a 20 mA, etc), podendo necessitar até 14 ciclos de varredura para ser completamente convertido.

Se a instrução for saltada, o processo será interrompido, ocasionando conversão incompleta e errônea.

Equivalência:

- 0V ou 4mA correspondem ao valor digital 0000;
- 5V, 10V ou 20mA correspondem ao valor digital 1000;
- Os demais valores correspondem a valores digitais proporcionais nesta faixa.



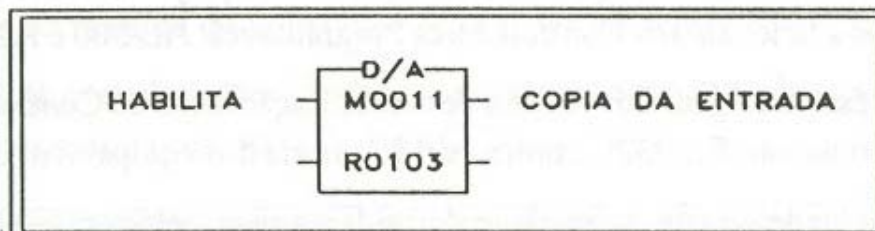
Esta instrução, ao ser saltada na execução do programa do Controlador Programável, pode ocasionar erro de conversão.

Não se deve escrever na memória destino, sob risco de alterar a conversão.

Esta instrução não pode ser usada na coluna 7.

3.6. Conversão Digital / Analógica

3.6.1. Simbologia no Diagrama de Relés



3.6.2. Descrição

Esta instrução converte um valor digital de 0 a 1000 contido na memória indicada na primeira célula em um valor analógico proporcional no ponto de saída indicado na segunda célula. O valor analógico convertido depende da escala utilizada no módulo de saída analógico destino (0 a 10 V, 4 a 20 mA, etc).

A instrução é programada da seguinte maneira:

- 1ª célula MXXXX Especifica a memória fonte.
- 2ª célula RXXXX Especifica o endereço do ponto de saída analógica.

A conversão é imediata e ocorre, independente do estado da entrada, sempre que a instrução é declarada no Controlador Programável.

Não se recomenda saltar esta instrução na execução do programa. Isto faria com que o ponto analógico perdesse o seu valor gradativamente, pois este necessita ser reescrito a cada ciclo de varredura.

Esta instrução não pode ser utilizada na coluna 7.

3.7. Impressão

A instrução IMPRESSÃO é uma ferramenta de programação extremamente poderosa para gerar relatórios, mensagens, formatação de "display" ou vídeo sem a necessidade de módulos adicionais aos Controladores Programáveis AL-500 e AL-1000.

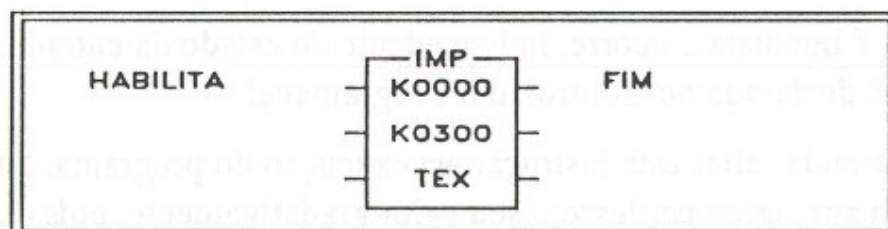
A impressora é ligada ao conector de comunicação serial do Controlador Programável utilizando o padrão RS-232C compatível à maioria dos equipamentos periféricos.

A velocidade de comunicação é definida na parametrização inicial do programa através do comando PARÂMETRO do AL-3800 ou pelo comando CMD 29 do AL-1800 ou ainda pela tecla PARM do AL-2800.

A instrução IMPRESSÃO permite o envio pelo Controlador Programável, via canal serial, de qualquer código ASCII, possibilitando tanto a impressão das mensagens como também o envio de caracteres de controle de dispositivos de saída. A instrução também pode ser utilizada para impressão de conteúdos de operandos do tipo memória.

3.7.1. Impressão de Textos

3.7.1.1. Simbologia no Diagrama de Relés



Para a impressão de textos, o preenchimento da instrução ocorre da seguinte forma:

- 1ª Célula K0000 Obrigatoriamente deve conter K0000, indicando que trata-se de impressão de mensagens contendo caracteres ASCII.
- 2ª Célula KXXXX Especifica, através de um operando tipo constante, quantos caracteres contém a mensagem controlada por esta instrução. O limite é 128 caracteres.
- 3ª Célula TEXTO Nesta célula poderão ser digitados ou observado os símbolos ou caracteres que serão enviados na mensagem.

A entrada de mensagens para impressão é feita utilizando-se as teclas que contêm ou que auxiliam a geração dos caracteres ASCII.

Nos Terminais de Programação AL-1800 e AL-2800, existem teclas adequadas especiais para a entrada destes caracteres. Para identificá-las verificar o capítulo referente a teclas do Manual de Utilização destes equipamentos.

Para o AL-3800, a tabela a seguir fornece os caracteres de controle e os respectivos valores para serem usados com <ALT> ou <CTRL>.

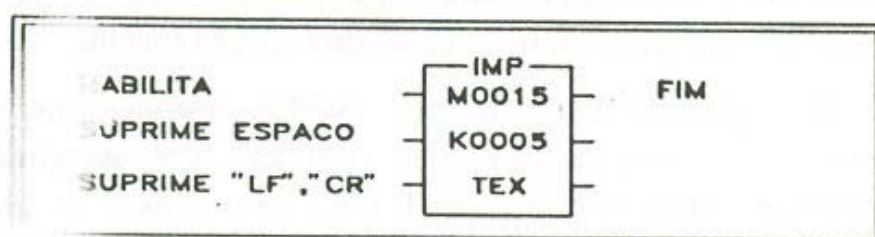
Carac	<ALT>	<CTRL>
NUL	0	␣
SOH	1	A
STX	2	B
ETX	3	C
EOT	4	D
ENQ	5	E
ACK	6	F
BEL	7	G
BS	8	H
HT	9	I
LF	10	J
VT	11	K
FF	12	L
CR	13	M
SO	14	N
SI	15	O

Carac	<ALT>	<CTRL>
DLE	16	P
DC1	17	Q
DC2	18	R
DC3	19	S
DC4	20	T
NAK	21	U
SYN	22	V
ETB	23	W
CAN	24	X
EM	25	Y
SUB	26	Z
ESC	27	[
FS	28	\
GS	29]
RS	30	^
US	31	-

Tabela 5 - Caracteres de Controle

3.7.2. Impressão dos Conteúdos das Memórias

3.7.2.1. Simbologia no Diagrama de Relés



3.7.2.2. Descrição

Para impressão de conteúdos de memória o preenchimento da instrução deve ser feito da seguinte forma:

- 1ª célula MXXXX Especifica o primeiro operando do tipo memória a ser impresso.
- 2ª célula K0XXX Especifica, através de operando tipo constante, o número de operandos memória a serem impressos. O número máximo é 128 memórias por instrução.
- 3ª célula Não utilizada.

Para imprimir o conteúdo de um número de memórias maior que 128 no CP AL-1000/512, deverão ser usadas tantas instruções IMPRESSÃO quantas forem necessárias.

O formato da impressão do quadro memória é o seguinte: VVVV bbbb VVVV bbbb VVVV bbbb, onde VVVV é o valor do conteúdo do operando do tipo memória e bbbb são quatro espaços em branco entre um valor e outro.

Ao término da impressão dos valores, um "line feed" e "carriage return" são efetuados automaticamente, fazendo com que a impressora vá para a linha seguinte retornando o carro para o início da mesma.

O usuário deve observar que a quantidade de memórias a serem impressas por um mesmo comando não ultrapasse as dimensões da linha do formulário de impressão. Os espaços em branco e o "CR" e "LF" podem ser suprimidos, através da energização da segunda e terceira entrada da instrução respectivamente.

A impressão é um processo paralelo à execução do programa do usuário. A instrução dispara este processo apenas na varredura em que a entrada HABILITA estiver energizada.

Não é possível executar mais de uma instrução de impressão simultaneamente, isto é, após a ativação de uma instrução de impressão, o canal serial é reservado para esta e todas as outras instruções são bloqueadas (não importando se a entrada HABILITA esteja energizada). Quando a instrução ativada chega ao fim da impressão o canal serial é desbloqueado e a próxima instrução IMP (com a entrada HABILITA energizada) encontrada na sequência do programa será executada.

A saída do bloco que representa a instrução IMPRESSÃO será energizada somente quando o periférico terminar de receber todo o conteúdo da mensagem ou dos valores de memória, permanecendo a saída energizada por uma varredura completa.

Se a entrada de uma instrução IMPRESSÃO continuar habilitada após o término da sua execução, esta só atuará novamente caso não exista outra habilitada ou se todas as outras habilitadas já tiverem atuado.

Basta um pulso com duração de uma varredura para disparar uma instrução IMPRESSÃO.

Se uma instrução IMPRESSÃO está ativa, esta ocupa o canal serial do Controlador Programável, bloqueando assim, seu uso para a instrução de recepção (ESP 04), se esta estiver ativada.

Quando um Programador estiver interligado ao Controlador Programável AL-500 ou AL-1000 via canal serial, as instruções IMPRESSÃO são automaticamente bloqueadas. Os procedimentos de comunicação do Programador AL-3800 com o CP ALTUS AL-1000/512 e de impressão deste Controlador Programável podem ser concomitantes em virtude do mesmo possuir duas interfaces seriais padrão RS-232C.

Esta instrução não pode ser utilizada na coluna 7.

Capítulo 4

Instruções Especiais

4.1. Introdução

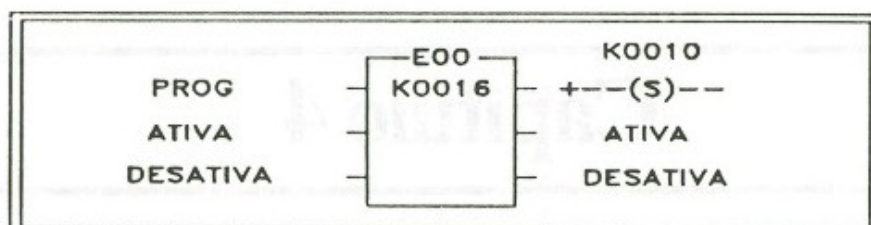
A disponibilidade destas instruções depende da versão de software do Controlador Programável. As Instruções Especiais disponíveis no CP AL-500 são assinaladas com um asterisco.

Estas instruções são sempre declaradas em três células, cabendo ao usuário verificar se os operandos de cada célula estão corretos, pois caso não estejam, o Programador não acusará erro. Sobre as Instruções Especiais não é efetuada nenhuma consistência durante sua edição.

A edição das mesmas é sempre efetuada selecionando o comando ESPECIAIS do Terminal de Programação ou programador e o número correspondente da instrução.

4.2. Instrução Especial 00 - Interrupção de Tempo

4.2.1. Simbologia no Diagrama de Relés



4.2.2. Descrição

Esta instrução tem por objetivo programar e controlar a execução de um trecho de programa que deve ser executado a intervalos de tempo selecionados.

A instrução está disponível apenas nas UCPs AL-1000/128 e 256, nos modelos "ICON" e "P2".

- 1ª Célula KXXXX ou MXXXX Indica a frequência com que o temporizador interno irá gerar as interrupções. O valor da constante K ou o conteúdo da memória M fornece um índice para uma "tabela de frequências", onde estão as frequências que podem ser programadas.
- 2ª Célula Vazia
- 3ª Célula Vazia

Quando a entrada PROG é energizada, a instrução verifica a frequência das interrupções que deve ser usada conforme a tabela 6, e o número de lógicas que vai ser "saltada" para atingir a 1ª lógica da rotina que será executada a cada interrupção. A entrada PROG deve ser energizada ao menos uma vez para inicialização da instrução. Após a inicialização as interrupções são executadas independente do estado da entrada PROG.

A segunda entrada (ATIVA), habilita a execução do trecho de programa selecionado, quando ocorrer a interrupção gerada pelo temporizador interno.

A terceira entrada (DESATIVA), desabilita a execução do trecho de programa selecionado quando o temporizador interno gerar interrupções, isto é, o temporizador

interno continua a gerar interrupções na frequência programada, no entanto, nenhum trecho de programa é executado.



Todas estas entradas são "memorizadas" pela instrução. Por isso, o usuário ao programá-la, deve fornecer os sinais de entrada na forma de um pulso com duração de uma varredura.

A tabela a seguir mostra os valores possíveis para a primeira célula e suas respectivas frequências programadas.

VALOR DE K OU CONTEÚDO DE M	FREQ. (Hz) INTERRUPÇÃO	PERÍODO (1 int. a cada) INTERRUPÇÃO (ms)
0	15	66,7
1	15	66,7
2	16	62,5
3	20	50
4	25	40
5	32	31,25
6	40	25
7	50	20
8	64	15,625
9	80	12,5
10	100	10
11	124,9966	8,00021
12	160	6,25
13	200	5,0
14	250,0271	3,99956
15	320	3,125
16	400	2,5
17	500,0543	1,99978
18	640	1,5625
19	800	1,25
20	999,5662	1,00043
21	1250,4749	0,79969
22	1600	0,625
23	1999,1323	0,50021

Tabela 6 -Frequências de Interrupção

1 - Caso as entradas ATIVA ou DESATV estejam energizadas, o conteúdo da primeira célula é ignorado, podendo a a mesma estar inclusive vazia.

2 - Quando a entrada PROG estiver energizada e o operando da primeira célula não uma é constante, ele é considerado como sendo um operando do tipo memória, independentemente do tipo de operando programado.

3 - Caso o valor do operando da primeira célula seja igual ou maior que o número de opções existentes na tabelas de frequências, a maior frequência disponível é programada.

4 - A instrução "INTERRUPÇÃO DE TEMPO" deve obrigatoriamente:

- estar na linha 0 da coluna 6;
- ser a única instrução desta coluna;
- possuir na saída da primeira célula uma instrução "BOBINA DE SALTO" para a primeira lógica do trecho de programa que será executado a cada interrupção de tempo.

5 - As lógicas que compõem uma rotina de interrupção de tempo podem estar em qualquer ponto do programa, no entanto o usuário deve providenciar para que elas não sejam executadas no ciclo normal de execução do programa. Assim, recomenda-se que elas sejam editadas como as últimas lógicas de um programa, colocando-se antes desta área uma instrução BOBINA DE SALTO SEMPRE HABILITADA e com número de lógicas a saltar superior ao existente no resto do programa.

6 - As seguintes instruções não devem existir no trecho de programa executado nas interrupções de tempo:

- Relé de Pulso;
- . Relé Mestre e Fim de Relé Mestre;
- Impressão;
- Temporizador;
- Bobina de Salto;
- Interrupção de Tempo;
- Termopar.

4.2.3. Restrições ao Uso da Instrução Especial 00

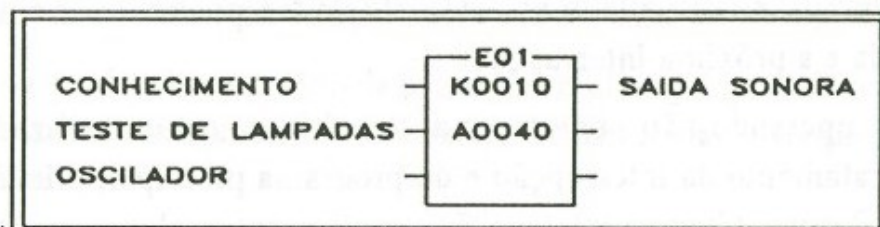
O tempo de programação da interrupção na instrução especial 00 deve ser sempre bem superior ao tempo necessário para executar a rotina associada a interrupção, pois o programa principal só executado no intervalo de tempo em que a rotina não está sendo executada e a próxima interrupção.

Um mesmo operando não pode ter sua memória imagem atualizada dentro da sub-rotina de tratamento da interrupção e do programa principal, principalmente se a instrução for Bobina. Como a interrupção acontece em qualquer ponto do ciclo de varredura, se o programa for interrompido durante a execução desta instrução ou durante a atualização da memória, poderá ser perdido o seu valor.

Os contatos podem ser utilizados, pois não alteram nenhum operando.

4.3. Instrução Especial 01 - Norma ISA4A

4.3.1. Simbologia no Diagrama de Relés



4.3.2. Descrição

Esta instrução realiza a função de alarme com uma única instrução no programa, acionando um conjunto de saídas conforme a norma ISA4A (veja Figura 14).

		NORMAL	ANORMAL	CONHECIMENTO ANTES DO RETORNO AO NORMAL	RETORNO AO NORMAL DEPOIS DO CONHECIMENTO	RETORNO AO NORMAL ANTES DO CONHECIMENTO	CONHECIMENTO DEPOIS DO RETORNO AO NORMAL
VISUAL	1ª						
	OUTROS						
SONORO							

LÂMPADA APAGADA
 LÂMPADA ACESA
 LÂMPADA PISCANDO

SONORO LIGADO
 SONORO DESLIGADO

Figura 14 - Sequência de Alarme ISA4A

A instrução especial 01 é programada da seguinte forma:

- 1ª Célula Especifica, através da constante KXXXX, o número de pontos de alarme.
- 2ª Célula Especifica, através de AXXX0, o endereço do primeiro relé auxiliar que será utilizado como condição de alarme.
- 3ª Célula Especifica, através de RXXX0 o endereço do primeiro relé de saída que será acionado em caso de alarme.

Observe que estes endereços devem possuir o dígito menos significativo igual a 0 (zero).

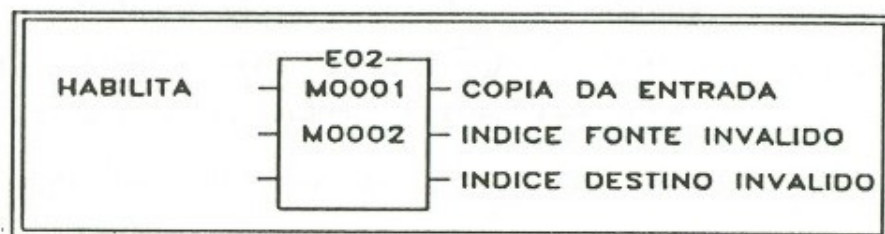
A primeira entrada é o conhecimento do alarme; a segunda é o teste de lâmpadas acendendo todos os relés de saída do alarme e a terceira é o sincronismo. A frequência com que pisca o primeiro alarme que disparou é determinada pelo oscilador conectado a esta entrada.

Quando, numa mesma varredura, dois ou mais alarmes estiverem acionados durante a execução desta instrução, aquele de menor endereço na imagem de auxiliares será o primeiro a ser acionado, desde que não haja nenhum disparado.

É permitida somente uma instrução ISA4A por programa.

4.4. Instrução Especial 02 -Movimentação Indireta(*)

4.4.1. Simbologia no Diagrama de Relés



4.4.2. Descrição

Esta instrução realiza a movimentação indireta de um valor contido em uma memória para outra memória, ambas indicadas por índices diferentes do tipo M ou K.

As células da instrução são programadas da seguinte maneira:

- 1ª Célula Especifica o índice (K ou M) da memória que contém o valor fonte a ser movimentado. Se o operando for uma memória, o valor contido nesta será tomado como índice.
- 2ª Célula Especifica o índice do tipo K ou M da memória para a qual será transferido o valor fonte. Se o operando for uma memória, o valor contido nesta será tomado como índice.
- 3ª Célula Seu conteúdo não é examinado pela instrução.

Quando a entrada HABILITA é energizada, a instrução acessa o conteúdo da memória cujo endereço é apontado pelo índice fonte e move-o para outra memória, cujo endereço é apontado pelo índice destino.

A saída da primeira célula é igual à entrada. A saída da segunda e terceira células, quando ligadas, indicam, respectivamente, que os endereços índice fonte e destino são superiores ao número máximo de memórias disponíveis (127 nos CPs AL-500 e AL-1000/128/256 e 1023 no CP AL-1000/512).

A Figura 15 mostra um trecho de programa que permite visualizar os valores de todas as memórias do Controlador Programável em um módulo visor (R0170). O endereço da memória é dado em chaves decimais (R0160). Observe que isto só ocorre com a entrada R0000 energizada.

4.4.3. Tempo de Execução

AL-500, AL-1000/128, AL-1000/256 - Ativada: 230 μ s
Desativada: 115 μ s

AL-1000/512 - Ativada: 161 μ s
Desativada: 80 μ s

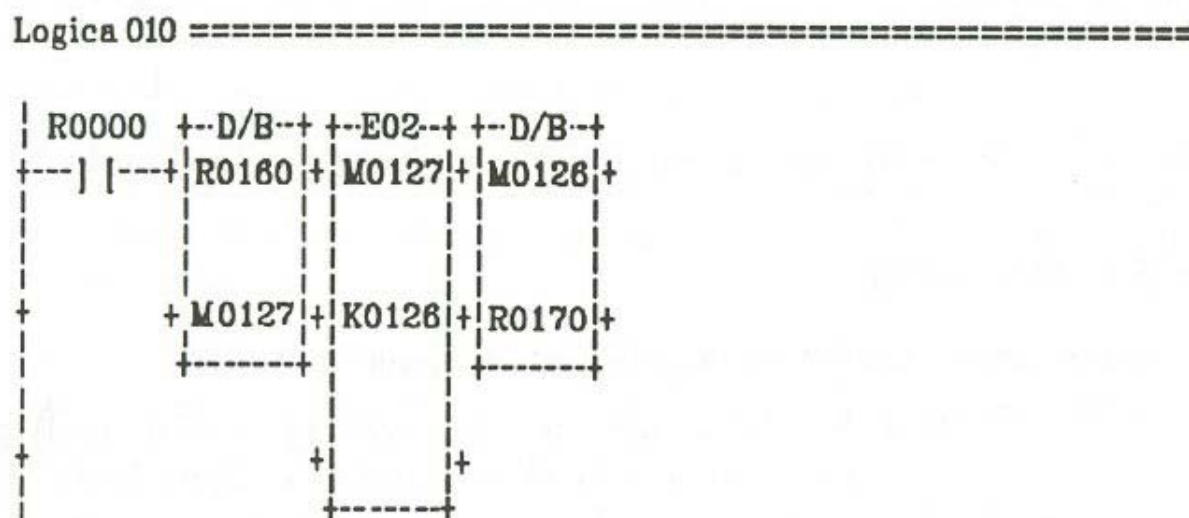
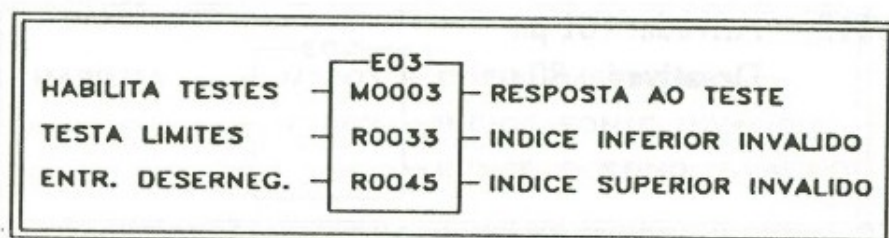


Figura 15 - Utilização da Instrução ESP 02

4.5. Instrução Especial 03 (*)

4.5.1. Simbologia no Diagrama de Relés



4.5.1.1. Teste ou Liga/Desliga Relé ou Auxiliar Indexado

4.5.2. Descrição

As células da instrução são programadas da seguinte maneira:

- 1ª Célula Especifica a memória cujo conteúdo (0 a 255 decimal) aponta para o auxiliar ou relé a ser testado ou ligado/desligado.
- 2ª Célula Especifica o endereço do primeiro relé ou auxiliar válido na instrução.
- 3ª Célula Especifica o endereço do último relé ou auxiliar válido na instrução.

Observe que os operandos das 2ª e 3ª células especificam um trecho de atuação de instrução na memória imagem. No caso destes operandos corresponderem a R0033 e R0045, respectivamente, esta instrução só valerá para os elementos de R0033 a R0037 e de R0040 a R0045. Esta instrução apresenta duas funções distintas dependendo da entrada da 3ª célula.

4.5.2.1. Teste de Relé ou Auxiliar Indexado

Neste caso, se as entradas "HABILITA TESTE" e da 3ª célula estiverem, respectivamente, energizada e desenergizada, o estado do relé ou auxiliar apontado pela memória índice é examinado. Conforme o estado esteja em "1" ou "0", a saída "RESPOSTA AO TESTE" será ligada ou não. Se o relé ou auxiliar apontado pela memória índice estiver fora dos limites definidos pelos parâmetros da 2ª e 3ª células, a saída ÍNDICE INFERIOR INVÁLIDO ou ÍNDICE SUPERIOR INVÁLIDO é

ligada e a saída da 1ª célula desligada. Sempre que a entrada "TESTA LIMITES" é ligada, ocorre o teste de um índice que se reflete nas respectivas saídas.

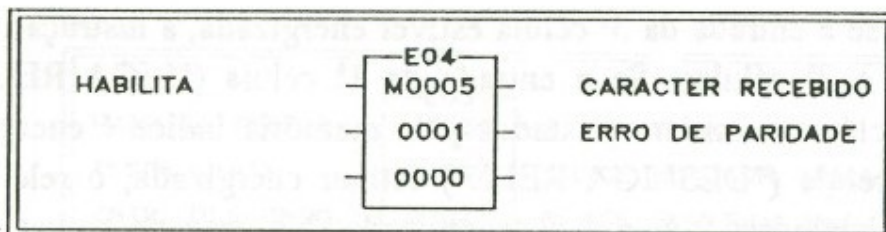
4.5.2.2. Liga / Desliga Relé ou Auxiliar Indexado

Neste caso, se a entrada da 3ª célula estiver energizada, a instrução examina as entradas da 1ª e 2ª células. Se a entrada da 1ª célula ("LIGA RELÉ") estiver energizada, o relé ou auxiliar apontados pela memória índice é energizado. Se a entrada da 2ª célula ("DESLIGA RELÉ") estiver energizada, o relé ou auxiliar apontados pela memória índice é desenergizado. No caso de ambas as entradas estarem energizadas, a entrada DESLIGA RELÉ tem prioridade. Se o relé ou auxiliar apontado pela memória índice estiver fora dos limites definidos pelos parâmetros da 2ª e 3ª células, a saída ÍNDICE SUPERIOR ou ÍNDICE INFERIOR INVÁLIDO é ligada. Esta verificação é feita somente no momento em que uma das entradas LIGA ou DESLIGA RELÉ é habilitada. A saída da 1ª célula é o "OU" lógico das entradas da 1ª e 2ª células.

Observe que os valores de índice contidos na memória estão em base decimal, devendo o cálculo do seu correspondente em base octal ser feito pelo usuário. Por exemplo: A0377 (octal) = A0255 (decimal).

4.6. Instrução Especial 04 - Recebe Caracter

4.6.1. Simbologia no Diagrama de Relés



4.6.2. Descrição

Esta instrução permite a recepção de caracteres via canal serial, sendo programada com os seguintes parâmetros:

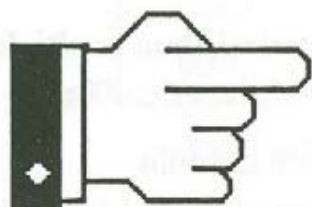
- 1ª célula Especifica a memória destino onde será armazenado o caracter recebido.
- 2ª célula Indicador de Baud-Rate : seleciona a velocidade de recepção de caracteres, segundo a tabela abaixo:
- 3ª célula Não é interpretada.

CONSTANTE	VALOR BPS
K0000	75
K0001	110
K0002	300
K0003	600
K0004	1200
K0005	2400
K0006	4800
K0007	9600

Tabela 7 -Baud Rates

Para valores de K maiores que 7 são considerados apenas os 3 bits menos significativos da representação binária do mesmo.

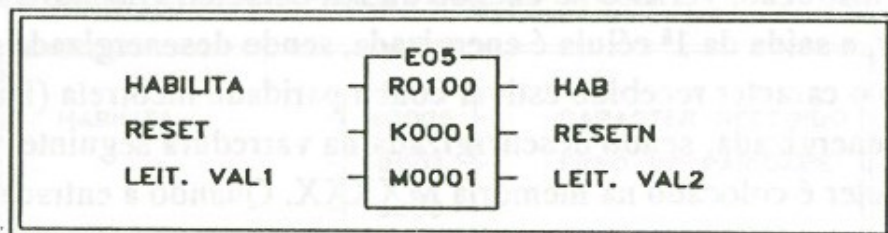
Quando a entrada HABILITA estiver energizada, não existindo nenhuma rotina de impressão em curso e estando o cabo apropriado (AL-1376) conectado, a instrução altera o baud-rate, desabilita o canal serial para outras instruções e, a cada varredura, ao executar a instrução, verifica se chegou algum caracter. Quando da chegada de algum caracter, a saída da 1ª célula é energizada, sendo desenergizada na varredura seguinte. Caso o caracter recebido estiver com a paridade incorreta (ímpar), a saída da 2ª célula é energizada, sendo desenergizada na varredura seguinte. Caso estiver correto, o caracter é colocado na memória MXXXX. Quando a entrada HABILITA não estiver energizada, o baud-rate volta ao valor definido no comando PARÂMETROS para a velocidade da impressora e o canal serial é liberado para outras instruções.



Pode existir somente uma Instrução Especial 04 por programa. O formato do caracter recebido deve ser de 10 bits com 1 bit de "start", 7 bits de dados, 1 bit de paridade par e 1 bit de "stop".

4.7. Instrução Especial 05 - Instrução Pt100

4.7.1. Simbologia no Diagrama de Relés



4.7.2. Descrição

O módulo AL-1117 lê os valores de temperatura de forma multiplexada. As leituras de cada canal são separadas por um intervalo mínimo de 400ms.

- 1ª célula RXXX0 Indica o endereço do módulo.
- 2ª célula KXXXX Indica a faixa de temperatura escolhida.
- 3ª célula Especifica o número da memória inicial ou o número da tabela onde serão colocados os resultados do processamento. Para memórias deve-se alocar 5 posições consecutivas para alocação dos resultados, a partir da memória indicada nesta célula. Para tabelas o tamanho mínimo deverá ser de 5 posições.

1ª posição: uso exclusivo do executivo para apontador de canais e contador de tempos;

2ª posição: resultado do canal 0;

3ª posição: resultado do canal 1;

4ª posição: resultado do canal 2;

5ª posição: resultado do canal 3.

A entrada HABILITA determina a execução da instrução (ativo em "1").

Quando a entrada RESETN é desenergizada o apontador de canal do módulo, o contador de tempo e as memórias ou tabela onde serão colocados os resultados são zerados bem como a saída LEIT.VAL2.

LEIT.VAL1 é uma entrada auxiliar que faz parte da lógica para o sinal de saída.

A saída HAB é cópia da entrada HABILITA.

A saída RESETN é cópia da entrada RESETN.

A saída LEIT.VAL2 é energizada quando o dado da última conversão é válido e as entradas RESETN e LEIT.VAL1 estão energizadas. Quando LEIT.VAL2 está desenergizada, indica que houve um erro na conversão do canal (ruptura ou curto circuito), neste caso o valor 9999 é armazenado na memória ou posição de tabela correspondente. Os valores exibidos para as diversas faixas são:

K	Faixa de temp.	Dado linearizado
0000	-4095 a +4095	Não linearizado
0001	-30 a +50 °C	0000 a 8000
0002	-30 a +370 °C	0000 a 4000
0003	-30 a +770 °C	0000 a 8000

Tabela 8 -Linearização da Instrução PT100

O dado linearizado dividido pelo fator de escala variável para cada faixa de temperatura (K=0001: 100; K=0002: 10; K=0003: 10) e subtraído de 30 corresponde ao valor medido em graus Celsius.

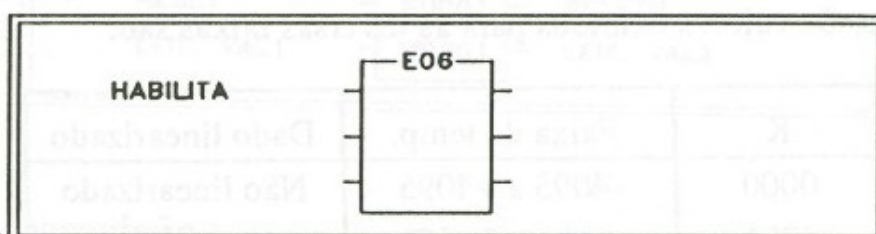


Não pode ser chamada uma ESP05 para um mesmo módulo mais de uma vez durante um ciclo, já que o tempo mínimo de conversão por canal é de 400ms.

O resultado colocado em memória ou tabela está em binário.

4.8. Instrução Especial 06 - Retorno da Interrupção de Tempo

4.8.1. Simbologia no Diagrama de Relés



4.8.2. Descrição

Esta instrução tem por objetivo indicar ao processador o fim de um trecho de programa que é executado a cada interrupção de tempo, retornando a execução para o programa principal.

A instrução está disponível apenas nas UCPs AL-1000/128 e 256 nos modelos "ICON" e "PROC2".

A entrada HABILITA habilita o retorno do trecho de programa executado a cada interrupção de tempo, transferindo o controle do processador para o programa principal.

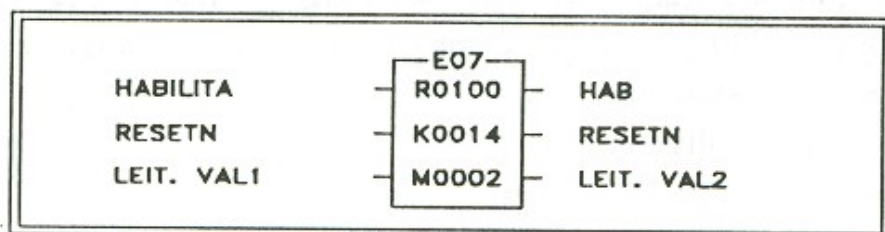
- 1ª. Célula vazia
- 2ª. Célula vazia
- 3ª. Célula vazia



Caso a entrada HABIL não esteja energizada, as saídas da segunda e terceira células serão cópias da entrada.

4.9. Instrução Especial 07 - Instrução TERMOPAR

4.9.1. Simbologia no Diagrama de Relés



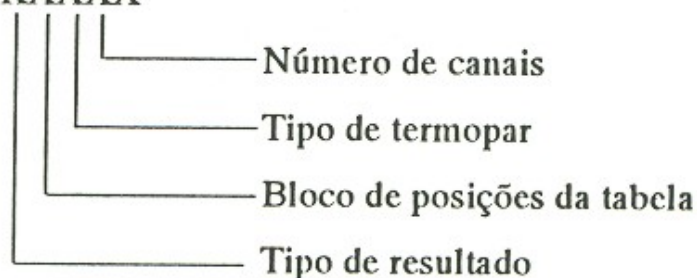
4.9.2. Descrição

Esta instrução tem por objetivo ler o valor de temperatura fornecido pelo módulo termopar e linearizá-lo, colocando o resultado em memória ou tabela.

A instrução está disponível apenas nas UCPs AL-1000/128 e 256 nos modelos "ICON", "PROC2", "PROC3" e, na AL-1000/512, a partir da versão IA.

- 1ª Célula RXXX0 Indica o endereço do módulo.
- 2ª Célula KXXXX Cada dígito desta constante representa um significado conforme indicado abaixo.
- 3ª Célula MXXXX ou TXXXX Especifica o número da memória inicial ou número da tabela onde serão armazenados os valores da conversão.

Significado de KXXXX



- Número de canais: Indica o número de canais a serem convertidos, por exemplo, se o número de canais for 2 será feita a conversão dos canais 0 e 1.

- Tipo de termopar: 1: Tipo j; 2: tipo R; 3: Tipos; 4: Tipo K; 5: Tipo B.

- Bloco de posições da tabela: Caso os valores das conversões sejam armazenados em uma tabela é possível armazenar os resultados de várias instruções termopar em uma única tabela. Neste caso este dígito indica qual o bloco de 4 posições na tabela deve ser utilizada para armazenar as conversões de cada instrução.

- Tipo de Resultado: 0: Resultado em graus centígrados; 1: Resultado em escala de 0 a 1000.

A utilização das 5 memórias ou posições de tabelas indicadas na terceira célula:

- 1ª posição: uso exclusivo do executivo para apontador de canais e contador de tempos (400 ms - tempo do módulo para realizar a conversão);
- 2ª posição: resultado do canal 0;
- 3ª posição: resultado do canal 1;
- 4ª posição: resultado do canal 2;
- 5ª posição: resultado do canal 3;



O número colocado em MXXXX será o da primeira posição e, conforme o número de canais escolhidos, as memórias seguintes serão alocadas para colocação dos resultados. O número de memórias alocadas é igual ao nº de canais + 1. Neste caso, o segundo dígito de KXXXX não surtirá qualquer efeito.

As posições reservadas pelo executivo não devem ser acessadas ou alteradas, sob pena de obtenção de resultados errados.

No caso da terceira célula ser tabela, esta deve conter o número da tabela na qual serão colocados os resultados. As posições da tabela em que serão colocados os valores é dada pelo número de ordem (definido no segundo dígito de KXXX) multiplicado por 4.

O tamanho da tabela deve ser o número de módulos termopar utilizados no sistema multiplicado por 5. As últimas posições da tabela são alocadas pelo programa executivo.

O número de canais a serem convertidos é sempre 4.

A entrada HAB habilita a execução da instrução, ativa em nível lógico "1".

A entrada RESETN, que é ativo em nível lógico "0", tem a seguinte função:

- zerar apontador de canal do módulo;
- zerar contador de tempo;
- zerar memórias ou tabela (onde for colocado o resultado);
- zerar saída LEIT.VAL2.

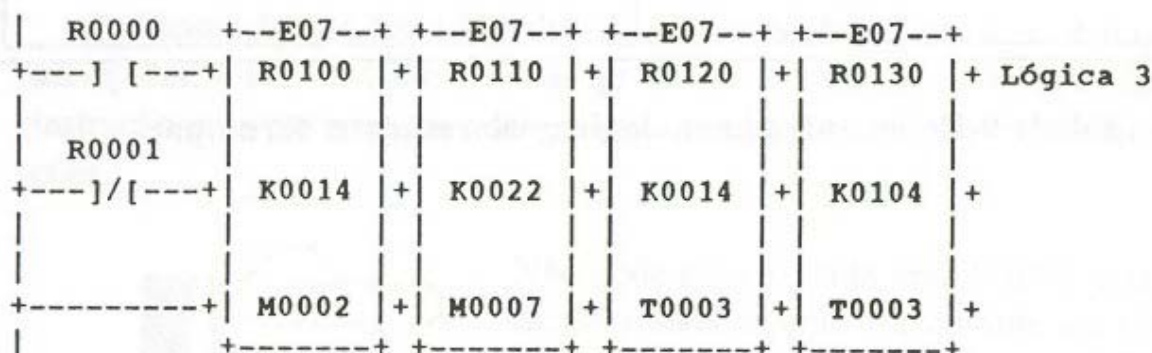
A entrada LEIT.VAL1 faz parte da lógica para o sinal de saída LEIT.VAL2.

A saída HAB é cópia da entrada HAB.

A saída RESETN é cópia da entrada RESETN.

Estes sinais foram implementados desta maneira pensando em habilitar e resetar módulos termopares simultaneamente, permitindo assim a programação com instruções sucessivas em uma mesma lógica.

Ex.:



Acionando a entrada R0000, são habilitadas todas as instruções E07 da lógica 3.

Acionando a entrada R0001, são ressetadas todas as E07 da lógica 3.



Para que seja entendido o RESETN, é necessário que a instrução esteja habilitada.

Neste exemplo, a primeira E07 faz a linearização de um módulo colocado no endereço R0100; o número de canais a serem lidos é 4, o módulo termopar é do tipo J e os resultados serão colocados a partir da M0002.

A saída LEIT.VAL2 é energizada quando o dado da última conversão é válido e as entradas RESETN e LEIT.VAL1 estão energizadas. Quando a saída LEIT.VAL2 está desenergizada significa que houve um erro na conversão (por exemplo termopar em curto, invertido ou interrompido).

VALOR ANALOGICO NA ENTRADA DO CONV. A/D	VALOR FORNECIDO PELA INSTRUCAO
0 A 999 mV	dado ok
-99 A -1 mV	8099 a 8001
MENOR QUE -99 mV	8999
MAIOR QUE 999 mV	9999

Tabela 9-Refer. entrada analógica/valores instr. Termopar

TIPO DE TERMOPAR	DADO EM VARIÁVEL DE ENGENHARIA (graus)	DADO NORMALIZADO
J	000 a 1000	000 a 1000
R	000 a 1500	000 a 1000
S	000 a 1500	000 a 1000
K	000 a 1250	000 a 1000
B	800 a 1800	000 a 1000

Tab.10-Rel. dados de engenharia/normalizados dos termop.

O dado normalizado corresponde ao dado em graus dividido pelo fator de escala variável para cada tipo de termopar (tipo J: 1.00, tipo K: 1.25, tipos R e S: 1.50) e subtraído de 800 para o termopar tipo B.

Podemos observar que para termopares do tipo J os valores normalizado e em variável de engenharia são iguais.

Colocamos à disposição do usuário estas duas opções para que se possa optar entre utilizar o valor normalizado para entrada em uma instrução PID ou se colocar num "display" a temperatura em graus, por exemplo.

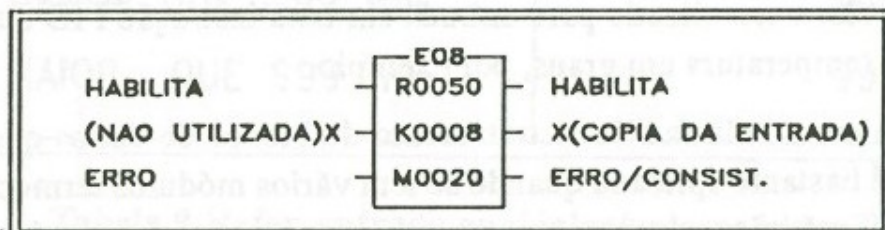
A disposição em tabela facilita o tratamento de blocos de dados que se queira manipular; esta é bastante aplicada quando se tem vários módulos termopares. Cada instrução trata um módulo, carregando numa tabela comum a vários módulos/instruções.



Não pode ser chamada uma ESP07 para um mesmo módulo mais de uma vez durante um ciclo, já que o tempo mínimo de conversão por canal é de 400 ms. O resultado colocado em memória ou tabela está em binário.

4.10. Instrução Especial 08 - Leitura de Pontos Analógicos Isolados

4.10.1. Simbologia no Diagramas de Relés



4.10.2. Descrição

Para efetuar a leitura dos pontos de um cartão de entradas analógicas isoladas AL-1116, utiliza-se no programa do usuário a instrução especial 08.

Os resultados das conversões são colocados em tabelas ou memórias, conforme opção do usuário.

A instrução está disponível apenas nas UCPs AL-1000/128 e 256 modelo "PROC 2" e "PROC 3" no AL-1000/512 a partir da versão 1A.

- 1ª Célula Endereço do ponto de entrada; no caso acima, o primeiro ponto a ser convertido será o de endereço R0050.
- 2ª Célula Especifica a constante K, que indica quantos pontos serão convertidos, contando-se a partir do endereço definido na primeira célula.
- 3ª Célula Especifica o operando destino inicial do tipo M ou T, a partir do qual serão colocados os resultados das conversões.

No exemplo, com memória inicial M0020 serão atualizadas as memórias M0020, M0021, M0022, M0023, M0024, M0025, M0026 e M0027.

A primeira entrada habilita a execução da instrução.

A entrada ERRO habilita a saída ERRO/CONSIST. Não inibe a execução da instrução.

A saída HABILITA é cópia do sinal da entrada HABILITA.

A saída X é cópia da entrada X .

A saída de ERRO/CONSIST pode ser ativada de duas formas:

- a) Entrada de erro ativada: ativa a saída de erro sem inibir a execução da instrução.
- b) Entrada de erro desativada e detecção de erro de consistência de parâmetros: significa que alguma das células possui valor inválido. Inibe a execução da instrução.

Os típicos erros de consistências são:

- Saturação do cartão: Como os pontos a serem convertidos devem pertencer ao mesmo cartão, é considerado erro de consistência a escolha, por exemplo, de um relé inicial 027 com número de entradas a serem convertidas igual a 6.

O último dígito do número do relé somado ao número de pontos a serem convertidos deve ser igual ou inferior a 8.

- Tabela não definida: Ocorre caso o usuário coloque o número de uma tabela não definida. A instrução não será executada e será ativada a saída de erro de consistência.

- Saturação de Tabela: Ocasionalmente pela escolha de um certo número de pontos que não cabem na tabela definida, ou seja, a tabela possui tamanho insuficiente.

- Saturação de memória: Ocasionalmente a escolha de um número inicial de memória que não permite a colocação do número de conversões escolhido.

Exemplo de Propagação do Erro de Consistência:

		+--E08--+		+--E08--+		+--E08--+	
HABILITA	1	R0010	+---1---	R0024	+---1---	R0030	+ 1
	X	K0008	+---X---	K0006	+---X---	K0007	+
	X						
ERRO	0	M0020	+---0---	M0030	+---1---	T0034	+ 1
		Instr.1		Instr.2		Instr.3	

- Os números colocados sobre as linhas correspondem ao nível lógico existente, onde: "0" corresponde a desenergizado; "1" corresponde a energizado e X a um nível lógico qualquer ("0" ou "1").

- A tabela 4 (T04) tem tamanho igual ou superior a 7.

Analisando o programa acima, verifica-se que, energizando a entrada HAB da primeira instrução, habilitamos todas as instruções da lógica.

A linha central tem seu nível lógico X propagando-se sem qualquer alteração.

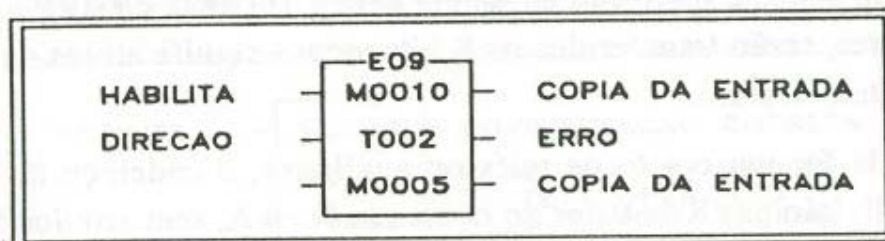
No início do trecho tem-se nível "0" na entrada de erro da primeira instrução. A execução desta instrução dá-se da seguinte forma: são lidas 8 entradas analógicas a partir de R0010 (R0010 a R017), que, após serem convertidas para valores digitais, são colocados sequencialmente em 8 posições de memória, começando por M020. A saída de erro não é energizada.

A segunda instrução apresenta erro de consistência, que é causado por saturação do cartão. Isto acontece porque é selecionada uma leitura (e conversão) de 6 pontos a partir de R024. A instrução não é executada e portanto é ativada a saída de erro.

A terceira instrução da lógica apresentada tem sua entrada de erro energizada, o que leva sua saída de erro automaticamente para nível 1. Como não há erro de consistência na instrução, esta será executada efetuando conversão de 7 valores a partir do endereço R0030 (R0030 a R0036) e colocando os respectivos resultados na tabela 4.

4.11. Instrução Especial 09 - Move Tabela Indexado

4.11.1. Simbologia no Diagrama de Relés



4.11.2. Descrição

Esta instrução permite ao usuário que realize a transferência de dados entre um operando e alguma posição de tabela, esta podendo ser indicada pelo valor de uma constante ou memória, possibilitando assim a indexação de tabelas. As células da instrução são programadas da seguinte maneira:

- 1ª Célula Especifica o operando R, A ou M de/para onde será movido o dado, conforme a entrada DIREÇÃO esteja desenergizada/energizada.
- 2ª Célula Especifica a tabela de/para onde será movido o dado, conforme a entrada DIREÇÃO esteja desligada/ligada.
- 3ª Célula Especifica a posição da tabela, através de operando K ou M de/para onde será movido o dado.

Quando a entrada HABILITA é energizada a instrução é executada.

A entrada DIREÇÃO determina o sentido da movimentação. Se estiver energizada, a instrução moverá o dado indicado na primeira célula para a posição indicada na terceira célula da tabela (indicada na segunda célula). Observe que a tabela, neste caso, pode ser indicada através de uma memória, constante ou tabela e a posição através de uma memória ou constante. Se estiver desenergizada, a instrução moverá da tabela apontada na 2ª célula o dado existente na posição indicada pela 3ª célula para o operando especificado na 1ª célula.

As saídas da 1ª e 3ª células são cópias das entradas. A saída da 2ª célula indica ocorrência de algum dentre os tipos de erros indicados a seguir:

- a) tentativa de acesso a tabela inexistente;

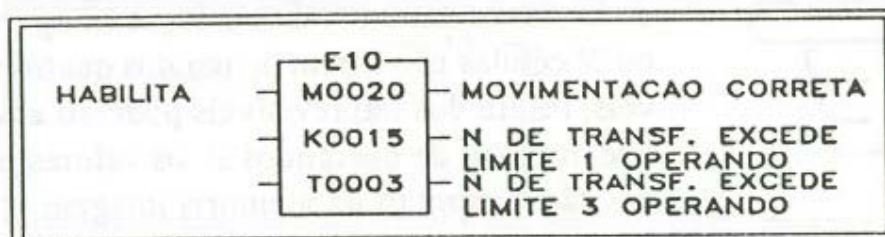
- b) posição de tabela inválida;
- c) tentativa de movimentação para constante quando a 2ª entrada está desenergizada;
- d) operandos inválidos para a(s) células(s).

Se o fluxo de dados é realizado da tabela para a 1ª célula e esta é um octeto de relés ou auxiliares, serão transferidos os 8 bits menos significativos da posição da tabela para o octeto destino.

Se a 1ª célula for um octeto de relés ou auxiliares, o endereço do octeto será calculado pela divisão por 8 do valor do operando R ou A, sem arredondamento.

4.12. Instrução Especial 10 - Movimentação de Blocos (*)

4.12.1. Simbologia no Diagrama de Relés



4.12.2. Descrição

Esta instrução realiza a movimentação entre blocos de operandos com uma única instrução de programa. As células da instrução são programadas da seguinte maneira:

- 1ª Célula Especifica o operando fonte inicial do tipo M, R, A ou T a partir do qual será feita a movimentação.
- 2ª Célula Especifica a constante K que indica quantos operandos devem ser movimentados.
- 3ª Célula Especifica o operando destino inicial do tipo M, R, A ou T, a partir do qual será feita a transferência.

Quando a entrada HABILITA é energizada, a instrução move, a partir do operando fonte inicial, tantos operandos quantos indicados pela constante K para os operandos destino, a partir do operando destino inicial.

Desta forma, é possível mover memórias, octetos de relés ou auxiliares ou posições de tabelas para qualquer operando do sistema.

Caso a constante seja igual a 0, nenhuma movimentação é feita, sendo todas as saídas da instrução desenergizadas. Do mesmo modo, se o número de movimentações exceder o limite do operando fonte (tabela não definida, índice inválido, memória maior que M0127, octeto maior que 0370, etc.) a instrução não é executada e a saída da 2ª célula energizada. Se, eventualmente, o número de movimentações exceder o limite do operando destino, a movimentação não é executada e a saída da 3ª célula energizada. Contudo, se o número de movimentações estiver dentro do limite dos operandos, então a instrução é executada e a saída da 1ª célula energizada. A movimentação de octetos (que ocupam 1 byte) para memórias ou tabelas (que

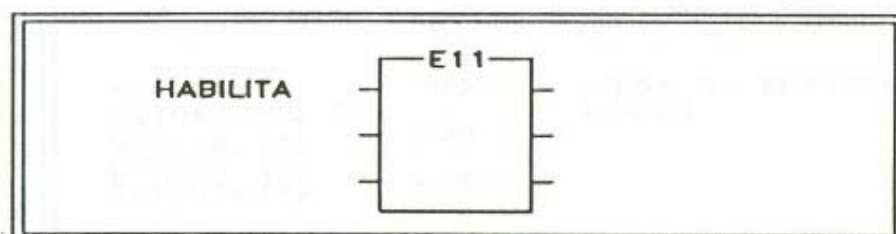
ocupam 2 bytes), é feita para o byte menos significativo destes operandos, sendo o byte mais significativo igualado a zero. Na movimentação de memórias e tabelas para octetos, apenas o byte menos significativo destes operandos é movido.



Não é feita consistência quanto ao tipo de operandos. Assim, o operando da 2ª célula é sempre encarado como uma constante. Contudo, se os operandos da 1ª ou 3ª células não forem de um dos quatro tipos possíveis, resultados imprevisíveis poderão advir. Note-se que, no caso de operandos R, os valores movimentados são sempre os da memória imagem.

4.13. Instrução Especial 11 - Retorno de Subrotina(*)

4.13.1. Simbologia no Diagrama de Relés



4.13.2. Descrição

A instrução BOBINA DE SALTO, ao ser executada, armazena o endereço da próxima lógica que seria executada, caso ela estivesse desabilitada.

O trecho de programa para o qual foi desviado o processamento, como resultado desta instrução, se for uma subrotina, deverá ter como última instrução uma instrução RETORNO DE SUBROTINA e não poderá conter dentro da subrotina instruções BOBINA DE SALTO.

As lógicas que compõem uma subrotina só podem ser executadas como resultado da atuação de uma instrução BOBINA DE SALTO. Assim, recomenda-se que as subrotinas sejam colocadas nas últimas lógicas de um programa, formando uma área específica de subrotinas.

A fim de evitar que sejam executadas no ciclo normal do programa, deve-se colocar antes desta área uma instrução BOBINA DE SALTO SEMPRE HABILITADA e com número de lógicas a saltar superior ao presente no resto do programa.

A instrução BOBINA DE SALTO para subrotina deve ser a última de uma lógica.

Não é possível chamar-se uma subrotina de dentro de outra subrotina, que está sendo executada.

A Instrução Especial 11 - Retorno de Subrotina permite que se retorne de uma subrotina para o programa principal. As células da instrução não necessitam de qualquer operando.

Quando a entrada HABILITA é energizada, a instrução retorna a execução do programa para a lógica seguinte à da instrução BOBINA DE SALTO que originou o

desvio para aquela subrotina. Caso a entrada não esteja habilitada, os sinais dos pontos de entrada da instrução são transferidos para a saída e a subrotina continua a ser executada. Isto permite que se tenha diversos pontos de retorno possíveis dentro da subrotina.



4.1.1. Descrição

A instrução BOBINA DE SALTO, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada.

O endereço de programa para o qual foi desviado o processamento, como resultado desta instrução, se for uma subrotina, desviará no ponto de retorno de subrotina. RETORNO DE SUBROTINA e não apenas como dentro da subrotina. Instruções BOBINA DE SALTO.

As linhas que compõem uma subrotina podem ser executadas com o resultado de uma instrução de salto. A instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada.

A instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada. A instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada.

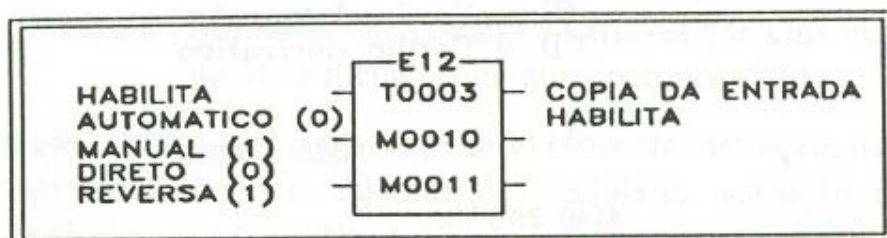
A instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada. A instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada.

A instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada. A instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada.

Quando a instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada. A instrução de salto, ao ser executada, armazena o endereço da próxima linha que está executada, caso ela estiver habilitada.

4.14. Instrução Especial 12 - P.I.D.

4.14.1. Simbologia no Diagrama de Relés



4.14.2. Descrição

A Instrução Especial 12 implementa o algoritmo de controle P.I.D. com uma única instrução no programa de usuário. A partir de uma variável medida (VM) e do ponto de ajuste desejado (PA) a instrução calcula qual deve ser o valor da variável atuada (VA). Este valor é calculado periodicamente, levando em consideração os fatores proporcionais, integrais e derivativos programados. O bloco funcional equivalente a esta instrução e seu diagrama de lógica é mostrado na Figura 16.

As características mais importantes apresentadas pelo laço de controle são:

- Desaturação da integral (anti-"reset windup").
- Acompanhamento da saída no modo manual e comutação manual/automática balanceada ("output tracking and bumpless transfer").
- Ação direta ou reversa.
- Limites de saída máximo e mínimo ajustáveis.
- Ação derivativa calculada sobre várias amostragens.
- Capacidade de realizar integral lógica.
- Deslocamento com sinal.
- Tempo de execução no pior caso de 3ms.
- Resolução de saída de 1:1000.

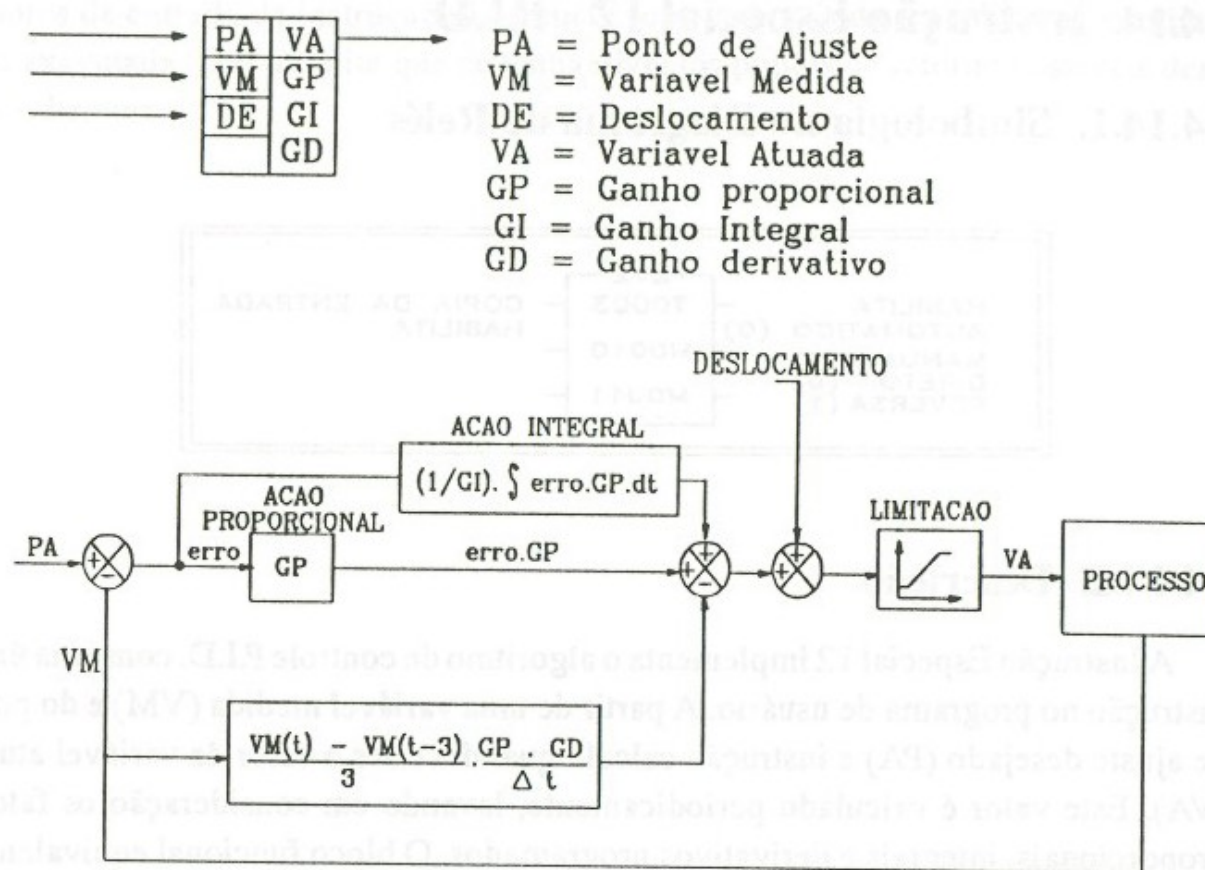


Figura 16 - Diagrama de Lógica da Instrução PID

O uso da instrução P.I.D., juntamente com o programa de usuário, passa a permitir uma série de facilidades que são mais facilmente integradas ao sistema, sem o uso de controladores externos, como por exemplo:

- Função auto/manual.
- Inibição do fator integral/derivativo.
- "Loops" cascadeados.
- Geração de ponto de ajuste automática (curvas).
- Manipulação pelo programa dos parâmetros de controle (controle adaptativo).
- Modificação da política de controle em função do estado do processo.

A instrução deve ser programada da seguinte forma:

- 1ª célula Tabela que contém os parâmetros utilizados pelo algoritmo de controle.
- 2ª célula Memória que contém a variável medida do processo (geralmente a memória de uma instrução A/D).
- 3ª célula Memória que contém a variável que atua no processo (geralmente a memória de uma instrução D/A).

Além dos parâmetros indicados no próprio bloco da instrução, outros parâmetros devem ser carregados na tabela indicada na 1ª célula da instrução. Esta tabela deve possuir 16 posições, sendo utilizada para definir os parâmetros utilizados pelo algoritmo de controle e armazenar resultados intermediários.

A tabela 11 apresenta os parâmetros que devem ser carregados em cada posição da tabela, bem como seus valores mínimos e máximos.

POSICAO DE TABELA	PARAMETRO ARMAZENADO	FORMULA	VARIACAO MAXIMA PERMITIDA	VARIACAO NA TABELA
0	GANHO PROPORCIONAL x 10	$GP \times 10$ $\Delta T / CI$ $GD / 3\Delta T$	GP: 0,1 A 100,0 CI: 1 A 1000 seg/rep ΔT : 0,1 A 10 seg GD: 1 A 1000 seg ΔT : 0,1 A 10 seg	1 A 1000 0000,0001 A 0010,0000 0000,0333 A 3333,3333
1	FATOR INTEGRAL - PARTE FRAC.			
2	FATOR INTEGRAL - PARTE INTEIRA			
3	FATOR DERIVATIVO - PARTE FRAC.			
4	FATOR DERIVATIVO - PARTE INTEIRA			
5	DESLOCAMENTO	DES	0 A 1000	0 A 1000
6	VALOR MINIMO DE SAIDA	MIN	0 A 1000	0 A 1000
7	VALOR MAXIMO DE SAIDA	MAX	0 A 1000	0 A 1000
8	PONTO DE AJUSTE	PA	0 A 1000	0 A 1000
9	VAR. MEDIDA N - 1			0 A 1000
10	VAR. MEDIDA N - 2			0 A 1000
11	VAR. MEDIDA N - 3			0 A 1000
12	ERRO			0 A 1000
13	ACAO PROPORCIONAL x 10			0 A 65535
14	ACAO INTEGRAL, PARTE FRAC. x 10			0 A 65535
15	ACAO INTEGRAL, PARTE INTEIRA x 10			0 A 65535

Tabela 11 - Parâmetros da instrução PID

Na tabela 11, VM, VA, DES, PA, MAX e MIN têm como variação a faixa de 0 a 1000, o que corresponde a uma variação de 0 a 100% nas variáveis do processo. Ganho integral igual a 1 segundo por repetição significa máximo efeito integral. Ganho derivativo igual a 1000 segundos significa máximo efeito derivativo.

As posições de 9 a 15 são usadas pela instrução para cálculos internos, não devendo ser alteradas pelo usuário. A posição 12 (Erro) pode ser consultada pelo programa de usuário.

As posições 14 e 15 acumulam o fator integral. Podem ser zeradas externamente, se necessário. Recomenda-se que, na especificação inicial da tabela, estas posições sejam zeradas para evitar que o valor aleatório fique armazenado.

Para possibilitar uma maior velocidade de execução, alguns parâmetros para os blocos de ação proporcional, integral e derivativa devem ser carregados na tabela de parâmetros já pré-calculados. Sendo valores relativamente fixos, evita-se que sejam calculados pela instrução a cada amostragem.

Os parâmetros que devem ser pré-calculados são:

a) Ganho Proporcional x 10 - Posição 0

É calculado multiplicando-se o ganho proporcional por 10.

b) Fator Multiplicativo Integral - Posições 1 e 2

É calculado dividindo-se o intervalo de amostragem (t) pelo ganho integral (GI). A unidade de t é segundos, sendo seu valor mínimo de 0,1 segundos e máximo de 10,0 segundos, devendo ser igual ao intervalo de tempo em que a rotina é executada. A unidade de GI é segundos/repetição.

c) Fator Multiplicativo Derivativo - Posições 3 e 4

É calculado dividindo-se o ganho derivativo (GD) pelo intervalo de amostragem (t) e pelo valor 3. A unidade de GD é segundos, podendo variar de 1 até 1000 segundos. Recomenda-se que, quanto maior o valor de GD (maior seu efeito), maior deve ser o intervalo de amostragem. Mesmo para valores de $D = 1$ segundo, o intervalo de amostragem deve ser maior que 0,2 segundos. Caso não seja tomado este cuidado, o termo derivativo produzirá apenas "ruído" e a ação de controle será muito brusca.

d) Deslocamento - Posição 5

Permite que se introduza um deslocamento ("BIAS") na variável atuada, evitando que erros negativos causem saturação no valor mínimo de saída. Geralmente este valor é ajustado para 50% (500) ou é igual ao PA, se o ganho proporcional é pequeno.

e) Valor Mínimo de Saída e Valor Máximo de Saída - Posições 6 e 7

São valores opcionais que limitam a excursão da variável atuada, podendo ser variados dinamicamente em função das condições operacionais. Se $MAX = 1000$ e $MIN = 0$, não é feita nenhuma limitação.

f) Ponto de Ajuste - Posição 8

É o valor desejado ("set point") para a variável atuada. Seu valor pode ser variado em função da política de controle desejada.

As demais posições de tabela são tratadas exclusivamente pela instrução P.I.D., não devendo ser manipuladas pelo programa de usuário.

Além da tabela de parâmetros, outras informações devem ser passadas para a instrução através de relés auxiliares (A0374 até A0377). Como estes relés são fixos para todos os "loops", eles devem ter seus estados atualizados antes da chamada de cada instrução P.I.D. Suas funções são as seguintes:

Sinal de Termo Integral: A0374

É atualizado pela instrução P.I.D., indicando, quando desenergizado, que o termo integral é positivo; caso contrário é negativo. Pode ser lido pelo programa, se necessário, devendo ser atualizado antes da execução da instrução.

Sinal do Deslocamento: A0375

Indica para a instrução qual o sinal do deslocamento:

- Desenergizado: positivo
- Energizado: negativo

Normalmente o sinal será positivo, devendo ser atualizado antes da execução da rotina, a menos que todos os loops tenham o mesmo sinal.

Inibe Ação Derivativa : A0376

Deve ser atualizado antes da instrução, a menos que todos os "loops" trabalhem no mesmo modo.

- Energizado: inibe ação derivativa
- Desenergizado: não inibe.
- Inibe ação integral: A0377
- Energizado: ação integral não é calculada, permanecendo com o último valor calculado antes da inibição, a menos que os valores limites sejam excedidos.

- Desenergizado: ação integral é calculada.

Algumas características da instrução são descritas abaixo:

A desaturação da ação integral (anti-"reset windup") é feita de modo a evitar que o termo integral continue a acumular erro, quando um distúrbio no processo causa a saturação da saída do controlador em alguns dos limites. No momento em que a variável de saída viola algum dos limites (máximo ou mínimo), o termo integral é mantido em seu último valor calculado.

Isto assegura que haverá uma resposta do controlador tão logo desapareça o distúrbio que o levou a saturar a saída.

O algoritmo pode ser colocado no modo manual (energizando a 2ª entrada). Neste modo, a rotina não afeta a variável de saída, mas a acompanha, isto é, em função do valor da variável de saída determinada pelo usuário e da variável medida do processo, os termos proporcionais e derivativo são calculados e o termo integral é forçado para um valor adequado, de modo que, quando ocorrer a transição do manual para automático, o controlador possa reassumir o controle com o valor inicial da saída igual ao último valor da saída no modo manual. A isto chamamos de comutação manual/automática balanceada ("bumpless").

A forma de controle pode ser direta ou reversa. Esta seleção é feita desenergizando ou energizando a 3ª entrada do bloco da instrução. A ação reversa deve ser selecionada caso a variável medida diminua quando a variável atuada aumenta.

O intervalo entre amostragens de um "loop" P.I.D. pode variar de 0,1s até 10,0s. No entanto, é responsabilidade do usuário programar um "disparador" de rotinas, ou seja, um trecho de programa que somente habilite a rotina P.I.D. nos intervalos de tempo desejados. Note que o valor do intervalo de amostragem usado para cálculo dos fatores multiplicativos integral e derivativo, deve coincidir com o intervalo de amostragem fornecido pelo "disparador". Como cada "loop" leva até 3ms, é aconselhável que cada "loop" de controle diferente seja disparado em diferentes varreduras do programa.

Como exemplo de utilização, digamos que os valores de ajuste desejados para um "loop" de controle sejam os seguintes:

- PA = 62%;
- GP = 5x (GP = 100/banda proporcional em %);
- GI = 100 SEGUNDOS/REPETIÇÃO (1 = máximo efeito integral);

- GD = 5 segundos (1000 = máximo efeito derivativo);
- t = 1 segundo;
- DES = 50%;
- MAX = 80%;
- MIN = 0%.

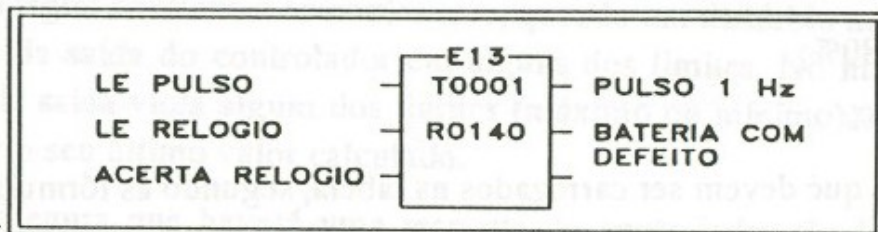
Os valores que devem ser carregados na tabela, segundo as fórmulas indicadas passam a ser:

POSICAO	VALOR	
0	50	GP x 10
1	100	t/GI - Parte fracionaria (0000.0100)
2	0	Parte inteira
3	6666	GD/(3.t) - Parte fracionaria (0001.6666)
4	1	Parte inteira
5	500	DES
6	000	MIN
7	800	MAX
8	620	PA

Tabela 12 - Parametrização da Instrução PID

4.15. Instrução Especial 13 - Horário e Calendário

4.15.1. Simbologia no Diagrama de Relés



4.15.2. Descrição

Esta instrução permite o fácil acesso ao módulo Relógio de Tempo Real (AL-1420), carregando em uma tabela informações de horário e calendário.

As células da instrução são programadas da seguinte maneira:

- 1ª Célula TXX Especifica o número de tabela a ser utilizada para a transferência. A tabela deve ter 7 posições; caso contrário, a instrução não é executada.
- 2ª Célula RXXX0 Especifica o endereço do módulo Relógio de Tempo Real.
- 3ª Célula Não utilizada.

A tabela que contém os valores de leitura ou acerto do módulo está organizada da seguinte maneira:

POSICAO	CONTEUDO	FORMATO
0	Segundos	00XX
1	Minutos	00XX
2	Horas	00XX
3	Dia do mes	00XX
4	Mes	00XX
5	Ano	00XX
6	Dia da semana	00XX

Tabela 13 - Organização da tabela da instrução relógio

O conteúdo destas posições pode ser lido ou escrito a qualquer momento, mas as posições são atualizadas com a hora real apenas quando a instrução é executada.

São disponíveis três sinais de controle na instrução:

- 1ª Célula Caso energizada, a instrução indica na saída PULSO se houve uma mudança no contador de segundos do relógio. O pulso dura uma varredura e pode ser usado para sincronizar tempos gerados no programa de usuário com o Relógio de Tempo Real. Outra aplicação deste pulso é ler o relógio apenas quando houver uma mudança de segundos, economizando-se tempo de leitura, já que a leitura só do pulso é mais rápida.
- 2ª Célula Caso energizado, transfere os valores do módulo para a tabela.
- 3ª Célula Caso energizado, transfere os valores da tabela para o módulo.

Caso as duas últimas entradas estejam energizadas ao mesmo tempo, não será efetuada nenhuma das funções. A entrada LÊ PULSO pode ser energizada em qualquer situação.

A saída BATERIA COM DEFEITO é ligada caso o relógio tenha ficado sem alimentação da bateria em uma falta da energia principal.

Esta saída é desligada, caso a entrada ACERTA RELÓGIO seja energizada.



A saída BATERIA COM DEFEITO só é atualizada na 1ª varredura do programa. No modo leitura de pulso, o intervalo entre o 1º e o 2º pulsos pode não ter a duração exata de um segundo.

4.16. Tratamento de Variáveis BCD

4.16.1. Generalidades

Variáveis BCD são valores lidos de chaves THUMB-WHEEL ou de outros módulos que trabalhem neste formato (Exemplo: Contador Rápido) e armazenados em operandos Memória ou Tabela no formato BCD, ou seja, cada dígito é representado por 4 bits.

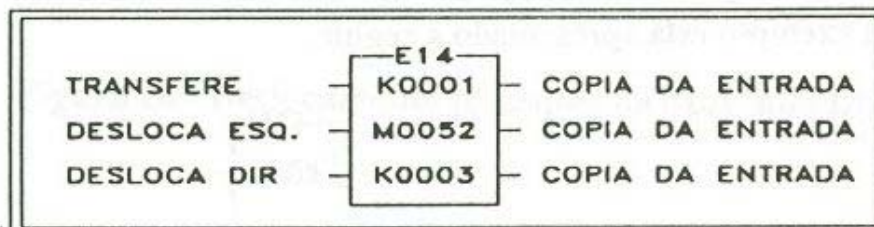
Um valor BCD possui o tamanho de até 7 dígitos mais o sinal, podendo variar de -9.999.999 até +9.999.999 e é armazenado na memória da seguinte forma:

- Os 4 dígitos menos significativos são armazenados na memória MXXXX;
- Os 3 dígitos mais significativos são armazenados na memória MXXXX+1;
- O sinal é armazenado como o dígito mais significativo da memória MXXXX + 1, devendo ser "0", se o valor for positivo, e qualquer outro dígito, se o valor for negativo.

Existem instruções para transferência de dígitos BCD, rotações, soma e subtração, além de instruções de conversão BIN/BCD e BCD/BIN.

4.17. Instrução Especial 14 -Transferência/ Deslocamento BCD

4.17.1. Simbologia no Diagrama de Relés



4.17.2. Descrição

Esta instrução realiza a transferência de dígitos BCD ou ainda o deslocamento para a direita/esquerda do valor BCD contido em uma memória, com inserção de 0 ou outro valor no dígito oposto ao do sentido da rotação. As células da instrução são programadas da seguinte maneira:

- 1ª Célula Contém o dígito da memória fonte (M001) que será transferido. Este valor varia de 0 a 3.
- 2ª Célula Contém a memória que possui o valor BCD para onde será transferido o dígito ou que sofrerá o deslocamento.
- 3ª Célula Contém o dígito da memória destino para onde será transferido o dígito lido da memória fonte. Este valor varia de 0 a 3.

Quando a entrada TRANSFERE é energizada, estando as outras entradas desenergizadas, a instrução move o dígito da memória M0001, indicado na 1ª célula, para o dígito indicado na 3ª célula da memória destino indicada na 2ª célula. Um exemplo é dado na figura a seguir.

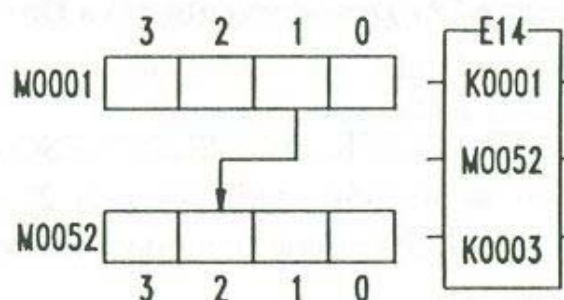


Figura 17 - Movimentação de Dígitos

Quando a entrada DESLOCA ESQ é energizada, estando as outras entradas desenergizadas, a instrução desloca o conteúdo da memória indicada na 2ª célula da instrução um dígito para a esquerda, sendo perdido o dígito superior (3) e inserido o valor "0" no dígito inferior (0).

Este algoritmo é executado a cada ciclo, devendo o usuário, portanto, providenciar que a instrução seja executada apenas durante um ciclo para cada deslocamento desejado. Um exemplo está apresentado a seguir.

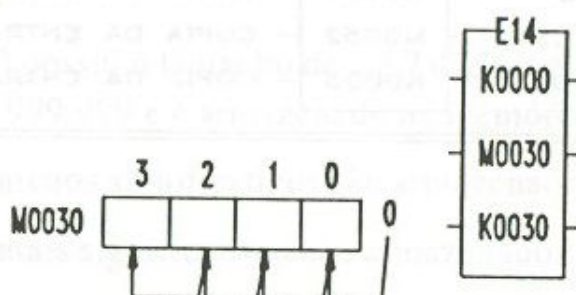


Figura 18 - Deslocamento para a Esquerda

O mesmo procedimento descrito acima é adotado quando a entrada DESLOCA DIR é energizada, invertendo-se apenas o sentido do deslocamento, conforme apresentado abaixo:

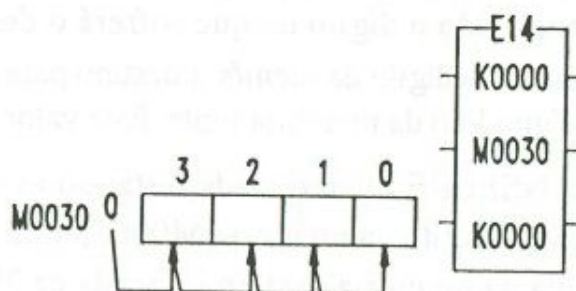


Figura 19 - Deslocamento para Direita

Ainda, se as entradas TRANSFERE e DESLOCA ESQ estiverem energizadas, a instrução desloca o valor da memória indicada pela 2ª célula um dígito para a esquerda, transferindo para o dígito inferior o dígito inferior (0) da memória M0001.

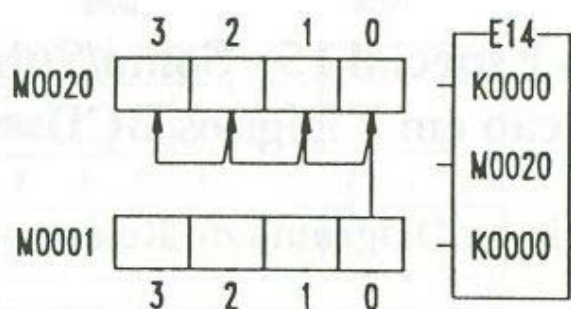


Figura 20 - Deslocamento p/ Esquerda Movendo Dígito

O mesmo procedimento descrito anteriormente é adotado quando as entradas TRANSFERE e DESLOCA DIR estiverem energizadas, sendo que neste caso, o dígito inferior (0) da memória M0001 é transferido para o dígito superior (3) da memória indicada pela 2ª célula.

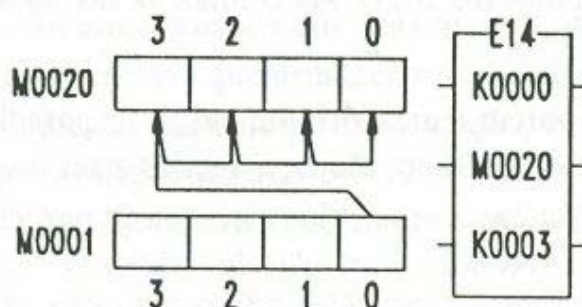


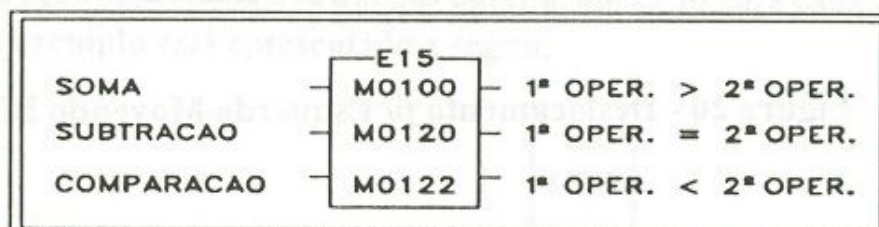
Figura 21 - Deslocamento p/ Direita Movendo Dígito



Estando as entradas DESLOCA ESQ e DESLOCA DIR energizadas, apenas DESLOCA ESQ é executada, sendo que as saídas continuam sendo a cópia das entradas. Se o valor das constantes que indicam o dígito fonte e destino for superior a 3, a instrução não é executada e as saídas são desenergizadas.

4.18. Instrução Especial 15 - Soma/Subtração/Comparação em 7 Dígitos BCD

4.18.1. Simbologia no Diagrama de Relés



4.18.2. Descrição

Esta instrução realiza a soma ou subtração algébrica em 7 dígitos BCD ou, ainda, a comparação entre dois números BCD. As células da instrução são programadas da seguinte maneira:

- 1ª Célula Contém a memória inicial da 1ª parcela da soma ou do minuendo ou, ainda, do valor a ser comparado.
- 2ª Célula Contém a memória inicial da 2ª parcela da soma ou do subtraendo ou, ainda, do valor a ser comparado.
- 3ª Célula Contém a memória inicial que conterá o resultado da operação de soma ou subtração

Quando a entrada SOMA é energizada, a instrução realiza a soma algébrica dos valores BCD contidos nas palavras de memória indicadas na 1ª e 2ª células, colocando o resultado na palavra de memória indicada na 3ª célula. Se o resultado da soma for maior que +9.999.999 ou menor que -9.999.999, todas as saídas são desenergizadas. Um exemplo de soma é apresentado a seguir.

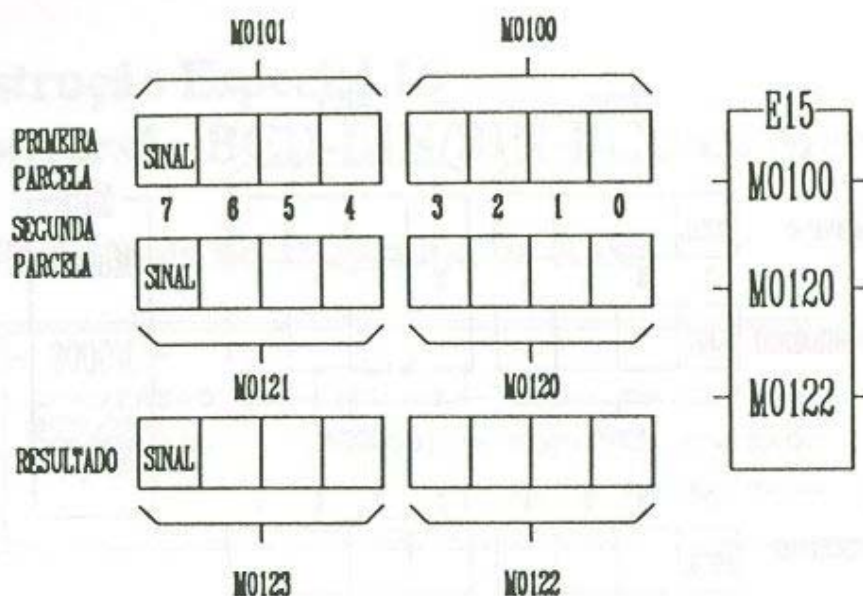


Figura 22 - Soma BCD com 7 Dígitos e Sinal

Os valores BCD são armazenados em duas memórias consecutivas (palavras de memória), sendo que a memória que armazena os 4 dígitos menos significativos é a de menor índice e é indicada nas células da instrução, devendo sempre ser um valor PAR. O sinal é sempre indicado no dígito 7. Se o valor do dígito 7 for 0, o número é considerado positivo. Qualquer outro valor será considerado negativo.

Quando a entrada SUBTRAÇÃO é energizada, a instrução realiza a subtração algébrica dos valores BCD contidos nas palavras de memória indicadas na 1ª e 2ª células, colocando o resultado na palavra de memória indicada na 3ª célula. Se o resultado da subtração for maior que +9.999.999 ou menor que -9.999.999, todas as saídas são desenergizadas.

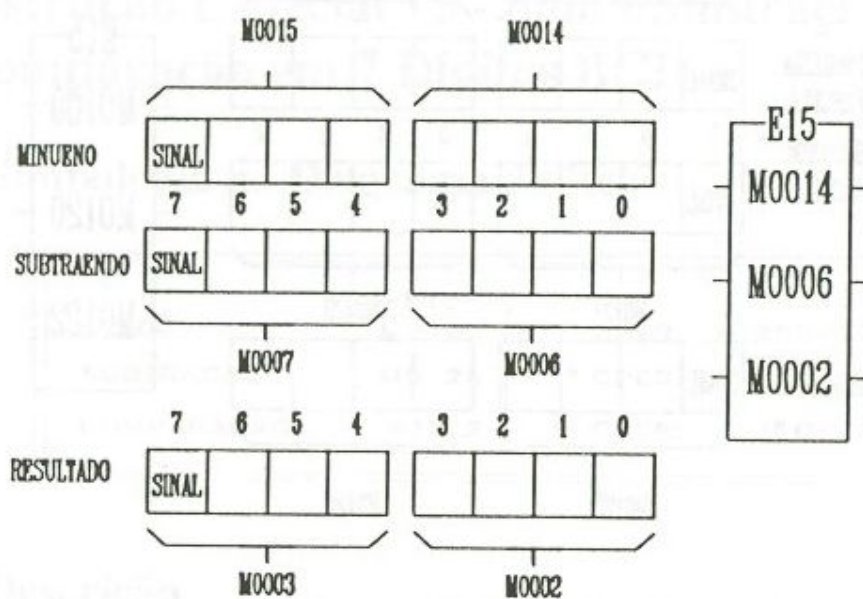


Figura 23 - Comparação BCD

Quando a entrada COMPARAÇÃO é energizada, a instrução compara os valores contidos nas palavras de memória da 1ª e 2ª células e atua nas saídas em função do resultado da comparação.

Assim, se o valor indicado pela 1ª célula for maior que o valor indicado pela 2ª célula, a 1ª saída é energizada. Se os valores indicados pela 1ª e 2ª células forem iguais, a 2ª saída é energizada. Se o valor indicado pela 1ª célula for menor que o valor indicado pela 2ª célula, a 3ª saída é energizada.

O conteúdo da memória indicado pela 3ª célula não é alterado. Isto permite que se compare valores sem necessidade de operá-los, com um tempo de execução menor.

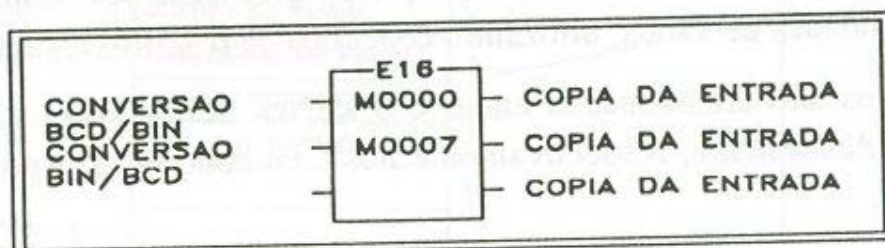
A comparação entre os operandos também é realizada quando as operações de SOMA ou SUBTRAÇÃO são habilitadas, sendo que o resultado da comparação é indicado nas saídas, se não houver um "overflow" na operação.

O "overflow" ou estouro ocorre quando o resultado da operação é maior que +9.999.999 ou menor que -9.999.999, sendo que as saídas são desligadas. Neste caso, o resultado colocado nas memórias representa apenas os 7 dígitos menos significativos do resultado final, com o sinal do limite ultrapassado.

4.19. Instrução Especial 16

Conversão BCD-BIN/BIN-BCD de Memórias

4.19.1. Simbologia no Diagrama de Relés



4.19.2. Descrição

Esta instrução permite a conversão de valores binários constantes ou armazenados em uma memória para valores BCD a serem armazenados em outra memória. Permite ainda a conversão de um valor armazenado em BCD em uma memória para a representação em binário deste mesmo valor a ser armazenado em outra memória. As células da instrução são programadas da seguinte maneira:

- 1ª Célula Contém o operando origem da conversão. No caso de um valor origem armazenado em BCD, só pode ser M. Para o caso de um valor origem armazenado em binário, pode ser M ou K.
- 2ª Célula Contém o operando destino da conversão. Deve ser sempre um operando M.
- 3ª Célula Não é interpretada.

As entradas indicam o sentido da conversão. Se a entrada CONVERSAO BCD/BIN estiver energizada, o valor em BCD contido na memória da 1ª célula é transformado para o seu correspondente em BINÁRIO e armazenado na memória indicada pela 2ª célula. Caso a entrada CONVERSAO BIN/BCD esteja energizada, o valor BINÁRIO armazenado na memória ou constante da 1ª célula é convertido para o correspondente em BCD e armazenado na memória indicada pela 2ª célula.

Se as duas entradas estiverem desenergizadas quando a instrução for encontrada, esta não é executada. Quando da conversão BCD/BIN, se a memória origem contiver um valor BCD inválido, 9999 em binário é movido para a memória destino. Quando da conversão BIN/BCD, se a memória origem contiver um valor binário superior a 9999, 9999 em BCD é movido para a memória destino.

4.20. Tratamento de Arquivos

4.20.1. Generalidades

O arquivo de dados é uma estrutura que possibilita ao CP a manipulação de grandes quantidades de dados, utilizando conceitos de registros e campos.

Os arquivos são armazenados em 6 e 8 kbytes de memória RAM (CMOS) endereçáveis diretamente, respectivamente nos CPs com até 256 pontos de E/S e AL-1000/512.

Cada arquivo pode possuir até 255 registros, numerados de 0 a 254, sendo que cada registro pode possuir até 255 campos, também numerados de 0 a 254. Note-se, entretanto, que a quantidade total de memória ocupada não pode exceder a memória disponível.

Cada campo é constituído de 2 bytes, podendo conter qualquer tipo de informação manipulada pelo CP. Assim, o número total de campos, distribuídos pelos registros e arquivos, não pode exceder a 3000 nos CPs AL-1000/128/256 e AL-500 e a 4000 no CP AL-1000/512.

A descrição de cada arquivo é feita através da tabela T01. Esta tabela contém a descrição dos arquivos e são utilizadas quatro posições para cada arquivo.

As duas primeiras posições indicam o endereço da memória onde inicia o arquivo. Considerando-se a memória dividida em 256 páginas de 256 bytes cada uma, a 1ª posição deve conter a página em que inicia o arquivo e a 2ª posição deve conter o endereço dentro da página.

A primeira página da área de arquivo é a de número 128, correspondendo ao endereço absoluto 32.768, e a última página disponível para o usuário é a de número 151, devendo os arquivos ficarem compreendidos dentro desta faixa.

A 3ª posição contém o número de registros do arquivo e a 4ª posição o número de campos que cada registro contém.

Os arquivos devem ser descritos sequencialmente, isto é, as primeiras quatro posições da tabela 01 referem-se ao arquivo número 0, as próximas quatro ao número 1, e assim por diante.

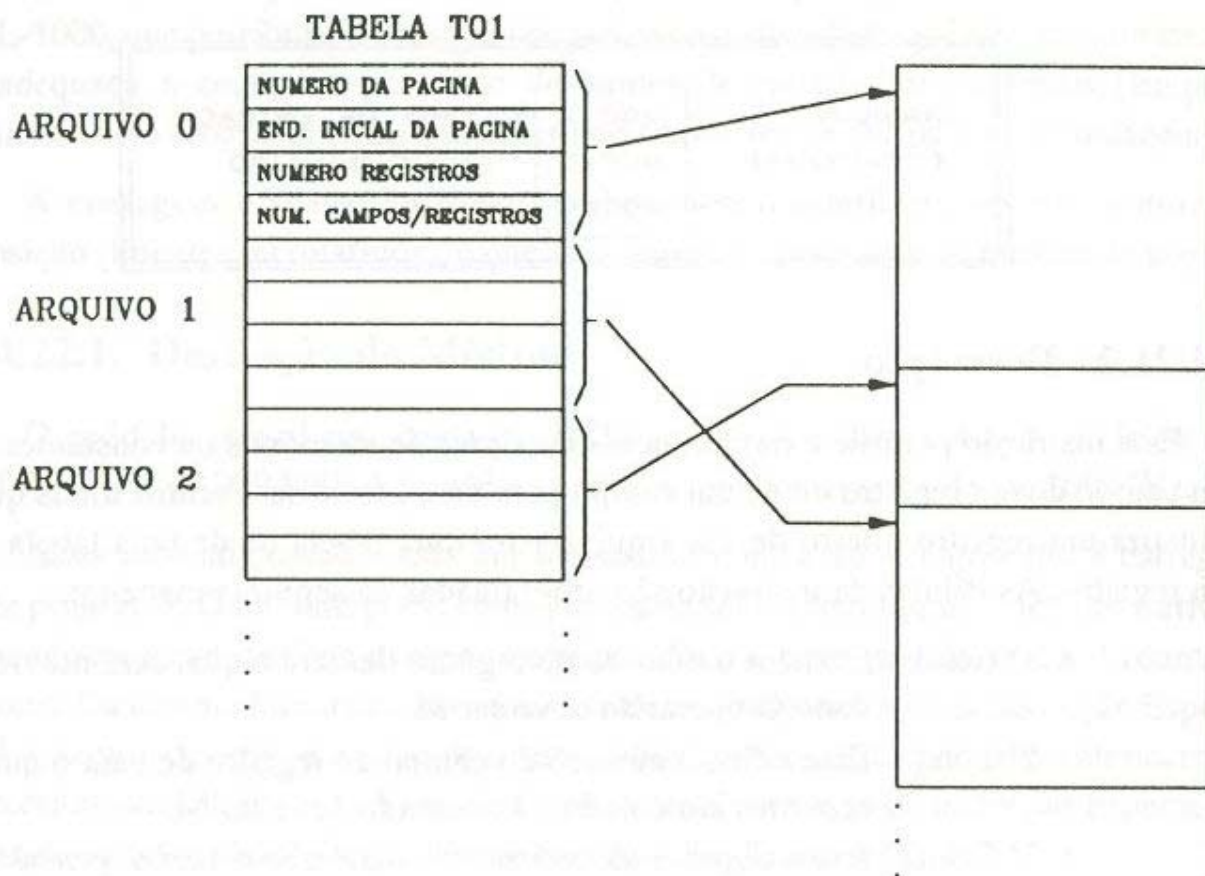
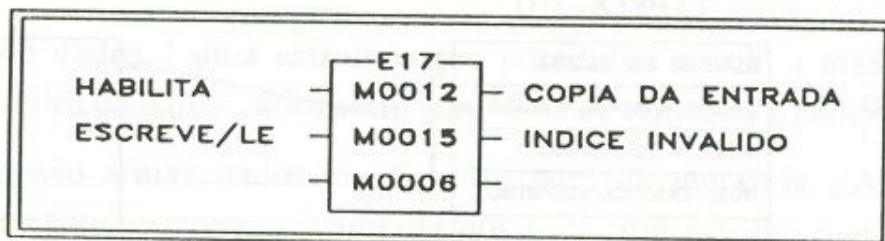


Figura 24 - Declaração de Arquivos

4.21. Instrução Especial 17 - Move Arquivo (*)

4.21.1. Simbologia no Diagrama de Relés



4.21.2. Descrição

Esta instrução permite a transferência de valores de memórias ou constantes para um campo de um registro ou de um campo para uma memória. Permite ainda que se transfira um registro inteiro de um arquivo para uma tabela ou de uma tabela para um registro. As células da instrução são programadas da seguinte maneira:

- 1ª Célula Contém o número do registro de/para o qual será movido o dado. O operando deve ser M.
- 2ª Célula Especifica o número do campo do registro de para o qual será movido o dado. O operando deve ser M.
- 3ª Célula Indica de/para onde será movido o dado. Se indicar o operando para onde será movido o dado, pode ser M ou T. Se indicar o operando fonte, isto é, de onde será movido o dado, pode ser M, K ou T.

A memória M0002 deve conter o número do arquivo ao qual refere-se a instrução. A tabela T01 deve possuir a descrição do arquivo indicado pela memória M0002.

A entrada ESCREVE/LÊ indica o sentido da movimentação do dado. Caso a entrada esteja energizada, a instrução escreve no arquivo o dado indicado. Se, no entanto, a entrada estiver desenergizada, o dado é lido do arquivo e transferido para o operando indicado.

O operando da 3ª célula indica o tipo de movimentação. Se o operando for uma TABELA, todo o REGISTRO indicado pela 1ª célula é movimentado, isto é, a transferência de dados faz-se entre o registro e a tabela indicada na 3ª célula, sendo que o conteúdo indicado na 2ª célula (o número do campo) não é examinado. Caso o operando seja uma MEMÓRIA ou CONSTANTE, a movimentação dá-se apenas entre o CAMPO especificado na 2ª célula e o operando da 3ª célula.

A entrada HABILITA permite que a instrução seja executada baseada nos parâmetros indicados pela outra entrada e pelos operandos da instrução.

4.22. Contador Rápido

O Contador Rápido é um módulo especial, compatível com a família dos CPs AL-1000, que possibilita a contagem de pulsos com elevada frequência, quando torna-se inadequada a contagem por meio de pontos de entrada convencionais (tempo de varredura do programa maior que o período de ocorrência dos pulsos de contagem).

A contagem é bidirecional e o módulo possui interface para transdutores de posição (lineares ou rotativos), o que lhe confere a capacidade de medida de posição.

4.22.1. Descrição do Módulo

O módulo possui um contador BCD com 6 décadas, cobrindo a faixa de -9.999.999 a +9.999.999. A frequência máxima de contagem é de até 100 KHz.

Associado ao contador está um registrador comparador, cujo valor é carregado por programa. O módulo possui dois pontos de saída (contatos de relé), um é ativado quando o contador atinge o valor do comparador e o outro quando o valor do contador é zero. Da mesma forma são ativadas as saídas correspondentes da Instrução Especial 18 a seguir descrita e dedicada a este módulo. As saídas por relé anteriormente descritas são habilitadas ou não de acordo com a execução da Instrução Especial 18, no que se refere à habilitação de comparação por "hardware".

Quando é ativada a entrada de leitura instantânea, o valor do contador naquele instante é memorizado até ser lido pelo CP. Porém, a contagem prossegue normalmente. Esta entrada pode ser desabilitada por ponte de ajuste na placa do módulo.

A ativação da entrada de zero/referência, quando habilitada por programa, provoca o zeramento do valor do contador.

O módulo de interface para transdutores de posição admite quatro modos de operação (dois canais de entrada: A e B):

a) Modo 1 : O sinal aplicado ao canal A produz uma contagem ascendente e o sinal aplicado ao canal B produz uma contagem descendente.

b) Modo 2 : O pulso de contagem é aplicado ao canal B e a direção é fornecida ao canal A. Nível lógico "1" no canal A implica contagem ascendente e nível lógico "0", em contagem descendente.

c) Modo 3 : Sinais em quadratura - a interface discrimina a direção do movimento a partir da relação de fase entre os sinais de entrada e fornece quatro pulsos de contagem por período destes sinais.

d) Modo 4 : Sinais em quadratura - a interface discrimina a direção do movimento a partir da relação de fase entre os sinais de entrada e fornece dois pulsos de contagem por período destes sinais.

A figura a seguir apresenta de forma simplificada o diagrama de blocos do módulo.

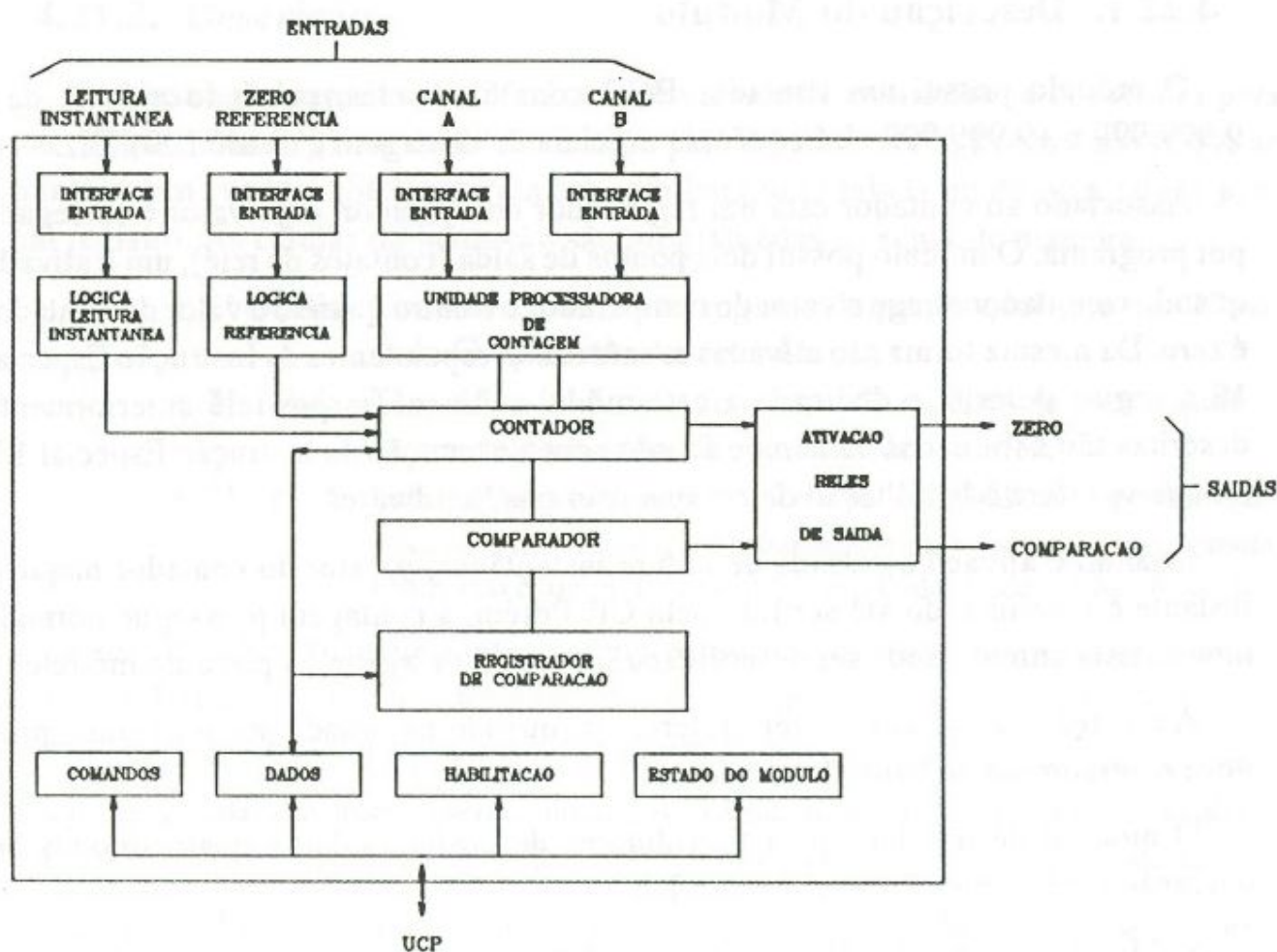
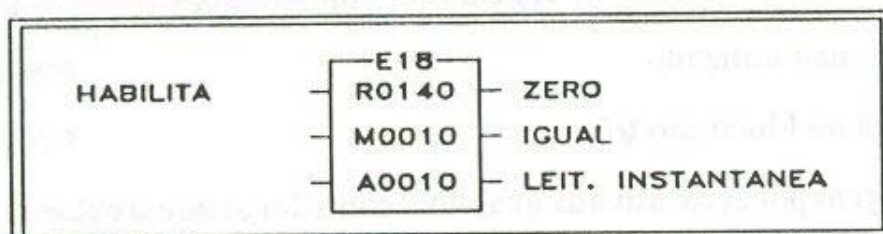


Figura 25 - Diagrama de Blocos do Módulo Contador Rápido

4.23. Instrução Especial 18 - Contador Rápido

4.23.1. Simbologia no Diagrama de Relés



4.23.2. Descrição

A Instrução Especial 18 efetiva o interfaceamento do programa do usuário com o módulo Contador Rápido, as células da instrução são programadas da seguinte maneira:

- 1ª Célula RXXX0 Indica o endereço do módulo.
- 2ª Célula MXXXX A memória indicada e a memória seguinte a ela contém o valor lido ou a ser escrito no Contador no formato BCD. As duas memórias seguintes a estas contém o valor a ser escrito no registrador comparador também no formato BCD. MXXXX deve ser uma memória par e não maior que 124.
- 3ª Célula AXXX0 Contém um conjunto de instruções para o módulo, conforme descrito a seguir:

Quando a entrada HABILITA está energizada, as instruções contidas no octeto de auxiliares AXXX0 são transferidas para o módulo Contador Rápido e as ordens de leitura/escrita são executadas.

Se a entrada HABILITA está desenergizada, o módulo mantém as instruções correntes, independente do estado dos auxiliares AXXXX, não realizando nenhuma operação de leitura/escrita; as saídas do bloco ficam desenergizadas.

AXXX0: inibe contagem.

AXXX1: zera contador.

AXXX2: habilita relés de saída (comparação por "hardware").

AXXX3: habilita entrada de referência.

AXXX4: executa leitura do valor do contador.

AXXX5: executa escrita no valor do contador.

AXXX6: executa escrita no registrador comparador.

AXXX7: não utilizado.

As saídas do bloco são três:

a) Passagem por zero: ativada quando o contador atinge o valor zero. Sua ativação é efetuada ao menos por uma varredura do programa, mesmo que o contador só permaneça neste estado por apenas 20 μ s.

b) Passagem por comparador: semelhante à saída anterior, é ativada quando o contador atinge um valor igual (em módulo e sinal) ao do registrador comparador.

c) Leitura instantânea: é ativada somente durante a varredura seguinte ao pedido de leitura instantânea, caso esta não esteja desabilitada por ponto de ajuste do módulo. Neste caso, a próxima leitura do contador fornecerá o valor deste no momento em que foi solicitada a leitura instantânea e não o valor atual.

Caso sejam solicitadas simultaneamente leitura e escrita no contador, é executada apenas a escrita. Independentemente desta situação, a escrita no registrador comparador é executada sempre que solicitada (se a instrução estiver habilitada).

O usuário deve levar em conta o fato de que cada operação de leitura/escrita dispensa cerca de 500 μ s. Desta forma, deve evitar operações desnecessárias, a fim de não comprometer o tempo de varredura do programa. A instrução ocupa quatro memórias: MXXXX e MXXXX + 1 para o contador, MXXXX + 2 e MXXXX + 3 para o registrador comparador. Ao ligar o CP, o valor do contador e do registrador é aleatório. Assim, o módulo deve ser inicializado convenientemente.

O dígito mais significativo do registrador indica o sinal do valor lido do comparador, ou seja:

- se o valor for negativo : MXXXX = 4XXX

MXXXX+1 = XXXX

- se o valor for positivo : MXXXX = 0XXX

MXXXX+1 = XXXX

Exemplo:

M0000 = 4099

M0001 = 9999

Registrador = -999.999

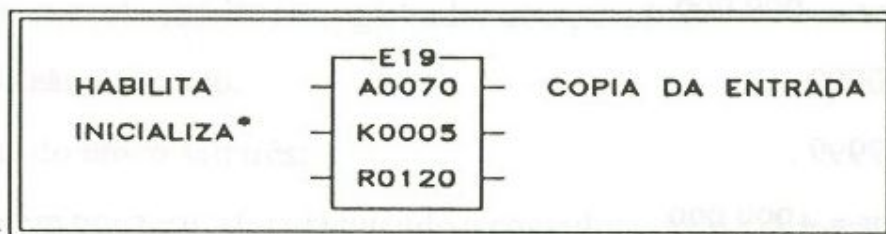
M0000 = 0099

M0001 = 9999

Registrador = +999.999

4.24. Instrução Especial 19 - Interface de LEDs

4.24.1. Simbologia no Diagrama de Relés



* Disponível apenas no AL-1000/512, a partir da versão P210H.

4.24.2. Descrição

A Instrução Especial 19 efetiva o interfaceamento do programa do usuário com o módulo Multiplexador de LEDs, AL-1460, permitindo enviar a este o estado de octetos de pontos digitais do sistema. As células da instrução são programadas da seguinte maneira:

- 1ª Célula Especifica o número do octeto inicial a partir do qual serão feitas as movimentações para o painel de LEDs. O operando pode ser A ou R.
- 2ª Célula Especifica o número do octeto inicial do painel de LEDs para onde serão feitas as movimentações. O operando deve ser K e estar compreendido no intervalo entre 0 e 31.
- 3ª Célula Especifica o endereço físico do módulo AL-1460.

Quando a entrada HABILITA está energizada, a instrução transfere o estado dos octetos, a partir do octeto indicado na 1ª célula, para o painel de LEDs, a partir do octeto de LEDs indicado na 2ª célula. O número de octetos a ser transferido é indicado pela memória M0003.

A transferência dá-se de modo seqüencial, isto é, a partir do octeto inicial; os octetos seguintes a serem transferidos correspondem à seqüência do espaço de endereçamento.

É importante notar que para o Controlador Programável, o espaço de endereçamento dos relés auxiliares encontra-se em seguida àquele dos relés de entrada e saída (RXXXX). Assim, se o octeto inicial a ser transferido for o R0360 e o número de

transferências for de 5, os octetos transferidos serão: R0360, R0370, A0000, A0010 e A0020. Se o número de transferências e o endereço inicial dos octetos fonte ou destino forem incompatíveis, a instrução não é realizada e as saídas são desenergizadas. Esta observação não é válida para o CP AL-1000/512 que, neste caso, não executará a transferência.

No AL-1000/512, a entrada INICIALIZA serve para retirar o módulo AL-1460 do modo autoteste. A energização desta entrada provoca, também, o zeramento em todos os LEDs do painel. Aconselha-se, portanto, que a reinicialização do AL-1460 seja feita apenas esporadicamente e não a cada ciclo de execução.

Capítulo 5

Informações Adicionais

5.1. Área de Memória Ocupada pelo Programa do Usuário

Para auxiliar o projetista no dimensionamento do tamanho do software de usuário, as tabelas 14, 15 e 16 apresentam o tamanho em bytes de cada instrução.

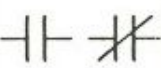
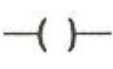
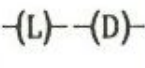
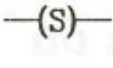
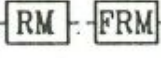


INSTRUCAO	TAMANHO (bytes)	OBSERVACAO
	4	Para operando reles com endereço acima de 127 ou operandos auxiliares, existe um acrescimo de 3 bites por instrucao
	4	Para operando reles com endereço acima de 127 ou operandos auxiliares, existe um acrescimo de 6 bites por instrucao
	6	Para operando reles com endereço acima de 127 ou operandos auxiliares, existe um acrescimo de 3 bites por instrucao
	8	
	4	
	4	
	2	

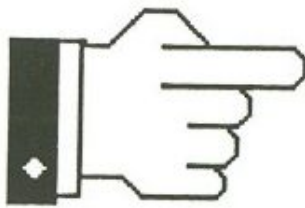
Tabela 14 - Instruções de Contatos

INSTRUCAO	TAMANHO (bytes)
CON	7
TEM	7
B/D D/B	6
MOV	8
MUL DIV SUB SOM	9
MOI	7

Tabela 15 - Instruções Básicas de Blocos

INSTRUCAO	TAMANHO (BYTES)	OBSERVACAO
COB	8	
MOT	8	
CT	6	Para cada posicao de tabela + 2 bytes
A/D D/A	6	
SEQ	6	
IMP	6	Para cada posicao de texto + 1 byte
ESP	13	

Tabela 16 - Instruções Estendidas e Especiais



A configuração ocupa 33 bytes.

A criação de uma lógica implica no gasto de 3 bytes que são referentes ao início desta lógica.

A simples ocupação de uma coluna em uma lógica gera o gasto de 6 bytes, menos para a última coluna.

5.2. Tempo de Execução das Instruções

Existem três grupos de UCP Altus.

Grupo 1: AL-500 (todos os modelos), AL-1000/256 versões S, N e P.

Grupo 2: AL-1000/128 e AL-1000/256 versões P2 e P3 e AL-1000/512L.

Grupo 3: AL-1000/512V.

As tabelas 17, 18 e 19 apresentam o tempo de execução (em μs) das instruções.

INSTRUÇÃO	GRUPO 1 (μs)		GRUPO 2 (μs)		GRUPO 3 (μs)	
	ATIVADA	DESATIVADA	ATIVADA	DESATIVADA	ATIVADA	DESATIVADA
CT00	136 + 70 P/ POSICAO	88	95 + 49 P/ POSICAO	61	66 + 34 P/ POSICAO	43
CT 01 A 63	119 + 75 P/ POSICAO	88	83 + 52 P/ POSICAO	61	58 + 36 P/ POSICAO	43
MOT LEIT.	247	94	173	66	121	46
MOT ESCR.	255	94	179	66	125	46
SEQ	240	85	168	59	118	41
COB	247	179	173	125	121	87
A/D	229	75	160	52	112	36
D/A	205		143		100	
IMP	200	105	140	73	98	51

Tabela 17 - Tempos de Execução das Instruções Estendidas



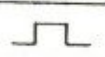
INSTRUCAO	GRUPO 1 (μs)		GRUPO 2 (μs)		GRUPO 3 (μs)	
INSTRUCAO DE COMANDO	ATIVADA	DESATIVADA	ATIVADA	DESATIVADA	ATIVADA	DESATIVADA
—(S)—	10.0		6,8		5	
—(S)—	77	65	54	45	38	31
 	51	34	35	23	24	16
	69		48		34	
TEM	179	116	125	81	87	57
CON	196	116	137	81	96	57
SOM	221	77	154	54	108	38
SUB	226	82	158	57	111	40
MUL	281	82	196	57	137	40
DIV	459	82	321	57	225	40
D/B	298	73	208	51	146	36
B/D	340	77	238	54	167	38
MOV R→M	204	77	143	54	100	38
M→R	213	77	149	54	104	38
M→M	170	77	119	54	83	38
K→M	145	77	101	54	71	38
K→R	175	77	122	54	85	38
R→R	240	77	168	54	118	38
MOI	145	85	101	59	71	41

Tabela 18 - Tempos de Execução das Instruções Básicas

INSTRUÇÃO		GRUPO 1 (μs)		GRUPO 2 (μs)		GRUPO 3 (μs)	
		ATIVADA	DESATIVADA	ATIVADA	DESATIVADA	ATIVADA	DESATIVADA
E00				73		51	
E01		280 + 60 P/MODULO	256 + 56 P/ MODULO	196 + 42 P/ MODULO	179 + 42 P/ MODULO	137 + 29 P/ MODULO	125 + 20 P/ MODULO
E02		230	115	161	80	113	56
E03 TESTE		195	110	136	77	96	54
E03 LIC/DESL.		204	110	143	77	100	54
E04		160	115	112	80	78	56
E05 C/ RESET N=0 P/ MEMOR.		297	65	189	41	132	29
E05 C/ RESET N=0 P/ TABELA		413	65	263	41	184	29
E05 C/ RESET N=1 P/ MEMOR.		933	65	593	41	415	29
E05 C/ RESET N=1 P/ TAB.		1039	65	661	41	463	29
E06				87		61	
E07 RESET N=0		438	64	297	43	208	30
E07 RESET N=1		856	64	571	43	400	30
E10	T->T	296 + 86 P/ POS.	107	207 + 60 P/ POS.	75	145 + 42 P/ POS.	52
	M->T	250 + 86 P/ POS.	107	175 + 60 P/ POS.	75	122 + 42 P/ POS.	52
	T->M	250 + 86 P/ POS.	107	175 + 60 P/ POS.	75	122 + 42 P/ POS.	52
E11		50	69	35	48	24	34
E13	LE PULSO	180	50	126	35	88	24
	LE RELOC.	125	50	910	35	637	24
	ACERT.PUL	1300	50	875	35	612	24

Tabela 19 - Tempos de Execução das Instruções Especiais

ALTUS SISTEMAS DE INFORMÁTICA S. A.

PORTO ALEGRE - RS : Av. São Paulo, 555 - CEP 90230-161 - Fone: (051) 337.3633 - Telex : (51)5010 - Fax : (051) 337.3632

SÃO PAULO - SP : Rua Dna. Maria Paula, 123 - 11º andar - Centro - CEP 01319-001 - Fone : (011) 239 2122 - Fax : (011) 606 9203

CAMPINAS - SP - Rua Visconde de Taunay, 421 Sala 1 - Guanabara - CEP 13023-200 - Fone : (0192) 32 7344 - Fax : (0192) 32 942

RIO DE JANEIRO - RJ : Av. Marechal Câmara 160 Cj. 305/306 - Castelo - CEP 20020-080 - Fone : (021) 240 4361 - Fax : (021) 240 4118

BELO HORIZONTE - MG : Av. Brasil, 248 Cj. 501 - CEP 30140-001 - Fone : (031) 241 3230 - Fax : (031) 241 4941